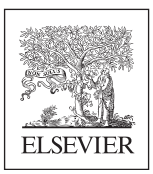
Understanding Automotive Electronics

An Engineering Perspective

Eighth edition

William B. Ribbens

Department of Electrical Engineering and Computer Science,
University of Michigan, Ann Arbor, MI, USA



Butterworth-Heinemann
An imprint of Elsevier

Butterworth-Heinemann is an imprint of Elsevier
The Boulevard, Langford Lane, Kidlington, Oxford OX5 1GB, United Kingdom
50 Hampshire Street, 5th Floor, Cambridge, MA 02139, United States

Copyright © 2017 Elsevier Inc. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or any information storage and retrieval system, without permission in writing from the publisher. Details on how to seek permission, further information about the Publisher's permissions policies and our arrangements with organizations such as the Copyright Clearance Center and the Copyright Licensing Agency, can be found at our website: www.elsevier.com/permissions.

This book and the individual contributions contained in it are protected under copyright by the Publisher (other than as may be noted herein).

Notices
Knowledge and best practice in this field are constantly changing. As new research and experience broaden our understanding, changes in research methods, professional practices, or medical treatment may become necessary.

Practitioners and researchers must always rely on their own experience and knowledge in evaluating and using any information, methods, compounds, or experiments described herein. In using such information or methods they should be mindful of their own safety and the safety of others, including parties for whom they have a professional responsibility.

To the fullest extent of the law, neither the Publisher nor the authors, contributors, or editors, assume any liability for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions, or ideas contained in the material herein.

Library of Congress Cataloging-in-Publication Data
A catalog record for this book is available from the Library of Congress

British Library Cataloguing-in-Publication Data
A catalogue record for this book is available from the British Library

ISBN: 978-0-12-810434-7

For information on all Butterworth-Heinemann publications
visit our website at https://www.elsevier.com/books-and-journals

Publisher: Jonathan Simpson
Acquisition Editor: Carrie Bolger
Editorial Project Manager: Carrie Bolger
Production Project Manager: Anusha Sambamoorthy
Cover Designer: Victoria Pearson

Typeset by SPi Global, India

I would like to thank my wife Katherine for her outstanding help in preparing this book. Without her dedication in editing/proofing and correcting errors, this book would not have been completed.
I dedicate this work to her.

CHAPTER

OVERVIEW 1

CHAPTER OUTLINE

Chapter 2 2
- Resistor 2
- Capacitor 3
- Inductor 4

Chapter 3 5
Chapter 4 7
Chapter 5 8
- Sensors 8
- Actuators 9

Chapter 6 10
Chapter 7 11
Chapter 8 14
Chapter 9 16
Chapter 10 18
Chapter 11 19
Chapter 12 20

This first chapter is intended to present an overview of the topics covered in the eighth edition of *Understanding Automotive Electronics*. The primary goal here is to explain the organization of the topics covered and to identify the chapter in which each of the subjects, including electronic systems, subsystems, and individual components, and their associated technically detailed theory are presented. For certain readers, there are some subjects that are redundant to their backgrounds. Therefore, it is one of the goals of this chapter to provide the book content in such a way as to permit each reader to select those chapters of individual interest and to be aware that much of the basic theory is presented in the appendices.

The first six editions of this book were written at a qualitative level in which automotive electronics was explained with minimal mathematics. However, beginning with the seventh edition, this book essentially was written from an engineering perspective that requires analytic models for automotive components and/or subsystems. This edition is an extension of the seventh edition. Topics have been updated with respect to automotive electronic technology, a field that has been evolving rapidly over the past few years. Although this is a major shift in perspective, qualitative discussions of the various topics are included for readers without the requisite mathematical background.

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-12-810434-7.00001-6
Copyright © 2017 Elsevier Inc. All rights reserved.

The seventh edition begins with two chapters that provide a review of the basics of system theory, with the first chapter devoted to continuous time and the second to discrete-time system theory. The discrete-time system theory is applicable to digital electronics that is widely used in present-day automotive electronics. Reader feedback indicated that only a subset of readers find the review of system theory necessary to follow the detailed analysis of modern automotive electronic subsystems/components. Therefore, the review of system theory found in Chapter 1 and Chapter 2 in the seventh edition have been moved to Appendices A and B in this edition, where they are available to any reader for whom this material is helpful. Whenever a topic involves modeling or analysis based on system theory, reference is made to the appropriate section of the relevant appendix. In addition to some reorganization of the material from the seventh edition, new material has been added to technically update the eighth edition.

CHAPTER 2

Chapter 2 deals with semiconductor-based electronic components. It begins with a discussion of the fundamentals of current flow in these materials and is followed by an explanation of the fundamental active components (e.g., diodes, bipolar transistors, and field-effect transistors (FETs)). The chapter includes a qualitative description of the operation of all active devices and the presentation of an analytic model for each device. Specific examples of circuit applications also are presented, including an exemplary circuit diagram and an analysis of circuit operation. A qualitative discussion of active element operation in each circuit example also is presented. This is followed by a discussion of circuits formed from numerous active electronic components (e.g., transistors) that function together to achieve a specific circuit operation. These groupings of components not only are commercially available as integrated circuits, but also form subsections of larger integrated circuits that include microprocessors and other important building blocks in modern automotive electronics.

It is assumed that engineering readers are familiar with the three ordinary circuit components that frequently are part of a larger circuit employing electronic (primarily solid-state) devices. These components include resistors, capacitors, and inductors. Readers who are familiar with these components can skip the next section, which develops simple models for these components.

RESISTOR

A resistor is a two-terminal circuit component having a linearly proportional relationship between the voltage v between the terminals and the current i passing through the resistor. Fig. 1.1 gives the circuit symbol for an idealized resistor.

For an ordinary resistor, the voltage/current model is given by

$$v = Ri$$

where R = resistance of the resistor

The two parameters that are key in selecting a resistor for a circuit application are its resistance R and its maximum power rating.

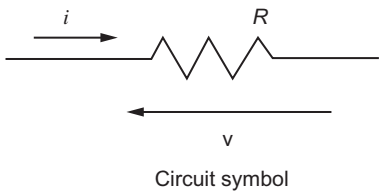


FIG. 1.1 Resister circuit symbol and model.

CAPACITOR

A capacitor is a two-terminal circuit component that consists of a pair of conducting electrodes (each one connected to one of the two terminal wire leads) that are separated by an electrically insulating material. Fig. 1.2 depicts a simple representative configuration of a capacitor and its circuit symbol.

Whenever a capacitor is incorporated in a circuit, any current *i(t)* flowing through it accumulates an electric charge Q on the electrodes positive on one and negative on the other as depicted in

$$Q(t) = \frac{1}{C}\int_{o}^{t} i(\tau)d\tau$$

The voltage between the two terminals v is related to the charge Q for a linear capacitor by the following

$$Q = C\mathrm{v}$$

where C = capacitance of the capacitor.

A related circuit model for the voltage and current can be found by differentiating the voltage charge equation with respect to time:

$$i = C\frac{d\mathrm{v}}{dt}$$

This model will be used throughout the book for any linear capacitor that is part of an electronic circuit.

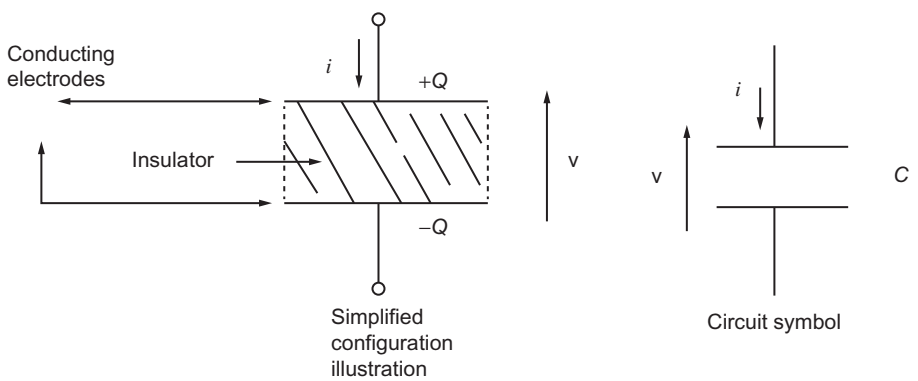


FIG. 1.2 Simplified capacitor configuration and circuit symbol.

INDUCTOR

An inductor is a two-terminal circuit component that is normally formed by a coil of wire wrapped around a magnetic material the nature of which is explained in Chapter 5 in association with the particular inductor being discussed. An example physical configuration of one type of inductor and its circuit symbol are depicted in Fig. 1.3.

In this example, the magnetic material is in the form of a closed-loop type structure that provides a path for a magnetic field $\bar{H}$ that is proportional to the current i. The motivation for having a closed-loop path is explained in Chapter 5 with respect to magnetic field theory in the discussion of certain vehicular electric components. The influence of magnetic field on the circuit properties is explained for many of the inductors described in the book. The circuit model for a linear inductor is given by

$$v = L\frac{di}{dt}$$

where L = inductance of the inductor. This model is used in this book multiple times in modeling electronic circuits that incorporate an inductor.

It is important to point out that the example circuits presented in the book are not those found in any production vehicle. The specific circuits employed in automotive electronic systems are very often proprietary to the OEM. To avoid any violation of OEM intellectual property, the exemplary circuits are taken from the public domain or are created to explain the operation of an automotive electronic system from design experience and other knowledge sources. Moreover, the detailed circuits employed in automotive electronic systems vary among the multiple automotive OEMs. Typically, the example circuits are a highly simplified version of a circuit that could potentially be found in an existing

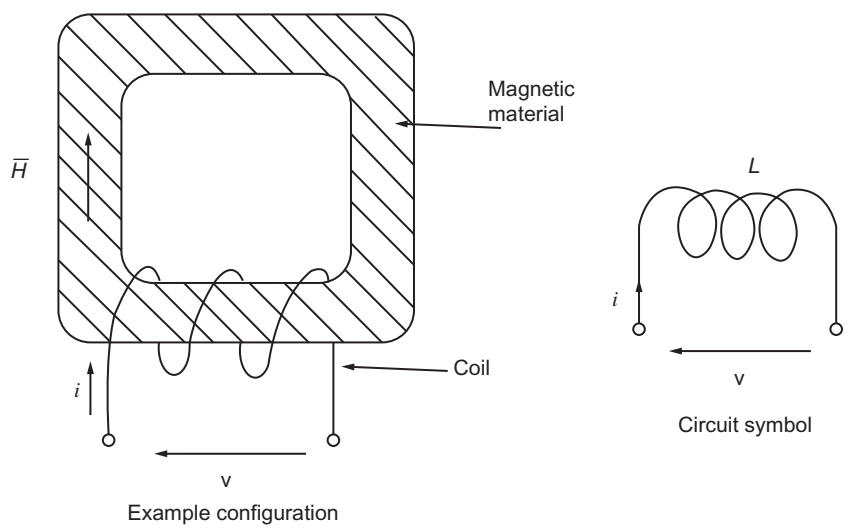


FIG. 1.3 Inductor configuration and circuit symbol.

vehicular system. The level of complexity of the example circuits is consistent with the level of complexity of systems being discussed in this book.

On the other hand, the exemplary circuits presented in association with the implementation of an automotive electronic system would perform the electronic task required by the circuit for the particular system to perform the intended task. The same issue of avoiding intellectual property rights for circuits applies to the systems or subsystems themselves. The examples of various automotive electronic systems are synthesized from design/analysis experience and provide support for explaining how a given system operates without taking the examples from an actual system used by any OEM.

The integrated circuits discussed in Chapter 2 include both analog and digital devices. Although the majority of automotive electronics are digital in nature, there are still a few analog circuits applied in relatively limited cases. The digital circuits discussed in Chapter 2 represent some fundamental building-block-type components that form the majority of the very large scale of integrated circuits such as discussed in Chapter 3.

CHAPTER 3

The electronic devices discussed in Chapter 3 are called by the traditional term "microprocessors." This chapter explains the configuration of the fundamental architecture of the early microprocessors. However, this architecture also applies broadly to devices that are more properly termed microcontrollers. The complexity of microprocessor/microcontroller integrated circuits has increased by many orders of magnitude over the earliest such devices that were available at the time of writing the first edition of this book. In addition to the discussion of the hardware aspects of these digital devices, there is an explanation given of the instructions/program or software that controls their operation.

The earliest microprocessors used in automotive electronics were programmed in the languages that were available at the time. Although various high-level computer languages were available, many of the early automotive microprocessor/microcontroller (MPC)-based systems were programmed in a language that was specific to each microprocessor and that was called an "assembly language." At the time of the writing of this eighth edition of the book, assembly language programming is no longer used. When microprocessors were introduced into automotive electronic systems, the high-level compiler-type languages typically resulted in far less computationally efficient programs for automotive control systems than assembly language programs.

Programming today is done with very efficient and powerful high-level languages and development systems (e.g., Autosar). However, for the purposes of explaining the operation of individual microprocessor components/subsystems in relationship to software, exemplary assembly language programming is presented in Chapter 3. This is done since each assembly language command controls the operation of the most basic microprocessor components. In addition, many assembly language commands can be represented in Boolean algebra statements. However, the high-level programming languages used to program contemporary automotive electronic systems also are discussed in Chapter 3 (primarily Autosar).

The development of a new vehicular electronics system involves several steps including system hardware architecture. The MPC incorporated must have the necessary computational capacity and speed for the maximum system performance requirements. All system input and output components must be either selected from existing devices or designed to function at the required performance

for the system. In addition, the interface electronics from the system controller to input/output also must be chosen from existing circuits or designed to meet system performance requirements.

Once the system architecture or hardware configuration is developed, the software to control it must be developed. Chapter 3 illustrates some of the basic MPC operations and presents assembly level commands for a simplified understanding of the advanced contemporary software development. However, before the actual software is written, the algorithms for controlling the system operation must be developed. Examples of such algorithm development are presented in Chapter 3 and other chapters that explain the design theory and performance evaluation for the associated electronic system. The fundamental analytic procedures for algorithm development are presented in Appendices A and B. One such approach to algorithm development involves analytic modeling of the vehicular system being controlled by the MPC-based electronics. The performance of the system then can be evaluated via computer simulation using advanced computational programs (e.g., MATLAB/SIMULINK).

The programs that are written to perform the algorithms are normally evaluated in a prototype system. One of the traditional procedures for testing the performance of the system is to build a prototype system using a portable computer that can emulate the intended MPC in a prototype. Software revisions are readily evaluated during the system development via experimental testing, typically on an automotive test track. Once the necessary software has been developed, it can be placed in a "read-only memory" (ROM) that is part of the MPC-based electronics.

The packaging of the electronic system to be used in production vehicles can take one of several forms. One such packaging form involves mounting individual integrated circuits and associated electronic components (e.g., resistors) on a printed circuit board. In an extreme case of packaging, the entire system can be fabricated in a single integrated circuit owing to the capability of modern-day integrated circuit fabrication with very high density of circuit elements.

The chapters from four to the end of the book discuss the application of electronics to the major physical/mechanical vehicular components. Essentially, all vehicles consist of basic components including body, suspension, steering, powertrain, braking, and lighting. In contemporary vehicles, electronics are incorporated in all of these major components. Typically, the first of the components to incorporate electronic controls was the engine portion of the powertrain. The powertrain itself consists of the engine, transmission, and wheel drive mechanism (e.g., differential).

In addition to explanation of the MPC devices, Chapter 3 explains a complete vehicular computer for performing the various control and measurement applications. In essence, a digital system used for control applications is a form of a special-purpose computer as distinguished from a general-purpose computer (e.g., a laptop) with primarily human inputs and providing human user-type outputs. The special-purpose computer discussed in Chapter 3 has inputs from electronic measuring devices (sensors) and/or switches that are activated by the vehicle itself. It also has output signals that operate electromechanical devices (actuators) or control applications or display devices for measurements that are to be read by the vehicle driver.

An MPC-based vehicular electronic control or instrumentation system is itself controlled by a stored program. The complete assembly of programs for control of the system is stored in a ROM. (Chapter 2 presents some exemplary traditional circuits and explained their operation for implementation of digital memory, e.g., ROM.) Chapter 3 presents exemplary block diagrams for MPC-based electronic systems. These block diagrams are simplified versions that are actually representative of earlier configurations such that the basic steps involved in applications of vehicular computers can more readily be explained than with reference to present-day systems.

Algorithms are presented for some of the operations to be performed by a vehicular computer. These include digital filters and representative digital control algorithms for closed-loop feedback control with proportional (P), proportional-integral (PI), and proportional-integral-differential (PID) control laws. Reference is made in Chapter 3 to the theory of operation of such control systems in either Appendix A or B. Another important control law discussed in Chapter 3 that has a vehicular application is a so-called limit-cycle control. As explained in Chapter 3, this control is a switched (on-off) control in which the actuator is switched on or off depending on the relationship between an input variable and switching level values. Other functions that are used in vehicular systems such as table-lookup interpolation of stored data from measurements of vehicular variables also are explained in this chapter.

The actual programming of the complete set of algorithms and all logic operations required to operate the electronic system is done in present-day development with advanced software development such as Autosar. Chapter 3 describes Autosar at the level appropriate for vehicular applications. A full description of such a computer language is a broad topic that is covered in other publications at the detailed level required by a programmer who is learning to use this programming capability.

CHAPTER 4

Chapter 4 presents the first vehicular system controlled by electronics—that of engine control. Although the engine is only a part of the vehicle powertrain, it is the prime mover. It is consistent with previous editions of this book to discuss this subject separately. This application of electronic control was the beginning of the adoption of electronic systems for controlling various vehicular systems. The motivation for the introduction of electronics for engine control was the governmental regulations of exhaust emissions of three exhaust gas constituents and the long-term requirement to meet the emission standards as explained in this chapter.

Qualitative explanations are given of the fundamentals of electronic engine control and analytic models and performance evaluation. However, this chapter is not intended to present the practical aspects of modern powertrain control; that is presented in Chapter 6.

This chapter also explains the influence of engine control variables and environmental and vehicular parameters on emissions of the regulated gases. The combination of catalytic converter in conjunction with electronic controls in meeting the regulation standards is explained in this chapter.

Chapter 4 is devoted solely to explaining the basic concepts involved in controlling emissions for various operating conditions. It presents a simplified version of an electronic engine control system with specific examples. It presents models of the engine performance and exhaust emissions and uses them to construct exemplary control laws. The engine analytic models presented in this chapter are dynamic models that can lead to an understanding of the exemplary control laws. For example, a hypothetical idle speed control (ISC) system is developed using the system theory concepts reviewed in Appendices A and B. In addition, however, a qualitative description of all topics covered is presented for readers who do not have the mathematical background.

The explanation of analytic modeling and analysis of an entire practical powertrain control system is given in Chapter 6 following the necessary discussion of the sensors and actuators that are involved in the electronic control of all vehicular systems. Reference is made to the electronic-engine-control-related

sensors and actuators. This organization of subject matter, in which the basics of electronic engine control are separate from the modeling and analysis of contemporary powertrain control, simplifies the explanation of the latter.

CHAPTER 5

Chapter 5 discusses sensors and actuators and provides a single location in the book for presenting the explanation modeling and performance analysis of these critically important components. Every electronic control system employed in vehicles fundamentally requires one or more sensor(s) and actuator(s). The subsequent chapters in this book that discuss electronic control systems for such systems (powertrain, braking, steering, and suspension) all make reference to the appropriate set of sensor(s) and actuator(s) presented here. This chapter is divided arbitrarily into two major sections: (1) sensors and (2) actuators.

SENSORS

The section on sensors begins with a discussion of engine-control-related sensors. A subset of these sensors has application only in engine control (e.g., exhaust gas oxygen and knock sensors). Other sensors that are presented with respect to the engine control application have applications in controlling other vehicular systems. These include, for example, sensors for measuring angular position and sensors for measuring pressure. It is simple to explain and model such sensors with respect to a single application. In Chapter 5, this application is engine-control-related.

The application in measuring the relevant variable in a system other than the exemplary system associated with sensor models and analysis given in Chapter 5 typically involves changing the materials used, the component parameters, the physical shape/geometry and the fabrication methods from the Chapter 5 examples. However, when such sensors are used in nonengine systems, the modeling and explanation of operation are sufficiently covered by the Chapter 5 descriptions. The detailed theory of the operation of the sensor in these nonengine-control-related applications is described in the associated chapter. The analytic model of the sensor also is presented in the relevant chapter based on its discussion in Chapter 5.

In developing an analytic model for a given sensor, the basic physics involved in its operation is discussed. In certain cases, the physics can only be modeled by reviewing the theory involved. For example, the modeling of a sensor that uses a magnetic field is explained by reviewing a few basic concepts and models from electromagnetic field theory. Many of the sensor applications are used in measuring time-varying variables at rates that require dynamic models. In some sensor models, the important output model involves an equivalent circuit including, for example, inductance or capacitance models. It is assumed that such devices are familiar for readers with an engineering background, and an equivalent circuit analysis is readily understandable. For nonengineering readers, the qualitative explanation should (hopefully) be sufficient for an understanding of the sensor operation.

Also included are a number of sensors that do not have engine-control-related applications. For example, there is discussion of a solid-state angular-rate sensor and an acceleration sensor, both of which have applications in vehicle motion control (e.g., enhanced vehicle stability, EVS). Another example is a vehicle-heading sensor (relative to true or magnetic north).

In addition, Chapter 5 presents sensors that actually are electronic subsystems. These include radar and lidar systems and optical image sensors (e.g., electronic camera). These subsystems have application in vehicle safety (e.g., blind spot detection and automatic braking systems that are part of advanced vehicle safety systems). The radar sensor has the capability of measuring range from an antenna to an object and the relative speed between the antenna and the object. An electronic camera consists of a lens system and an array of light sensors that can provide the data from which an image of an object within the field of view can be sensed and identified via image processing software. A detailed explanation and model for electronic cameras is given in this chapter.

ACTUATORS

The actuator section of Chapter 5 also begins with devices based on engine control applications. For example, the solenoid is an electromagnetic device having many engine control applications (e.g., supplying precise amounts of fuel to the engine in the form of a fuel injector). However, a solenoid can provide the basis for operating a valve that regulates fluid flow and/or pressure. Such a solenoid-operated valve has many applications in vehicle motion control (e.g., brakes). The models and explanations of solenoids presented in Chapter 5 pertain to all applications discussed in this book.

Another electromagnetic actuator with engine control applications is the ignition coil. The ignition coil generates the relatively high voltage required to create a spark at the electrodes of a spark plug for gasoline-fueled engines. As explained in Chapter 4, the spark ignites the fuel/air mixture in the engine combustion chamber at the optimum time in the engine cycle.

Another important electromagnetic actuator is an electric motor. Electric motors have a wide range of applications from side view mirror adjustment or seat position adjustment to providing the drive torque and power necessary to propel a hybrid/electric vehicle. This important application of motors is discussed in the chapter on powertrain control. Chapter 5 explains electric motor operation qualitatively, but the major discussion is on the theory of operation. Analytic models are developed for different motor types and are used for performance analysis of these motors. In order to develop these analytic models, there is a brief review of appropriate portions of electromagnetic field theory.

The analytic models for the motors include calculation of the motor torque and power for a given electric source. In addition, circuit models are presented that relate performance to the excitation that is created in the drive circuit, which in turn is operated by the motor controller. The latter subjects are explained in the chapter on powertrain control.

The motors discussed in Chapter 5 include single and polyphase induction motors, with models of the torque produced vs. speed for specific excitation currents. The means of controlling these motors are explained, and the model for the relationship between the motor rotational speed vs. applied current and torque load is developed.

The “brushless DC motor” also is examined. The rotor in this type of motor rotates synchronously with the frequency of excitation. Control of such a motor via generation of an excitation at a frequency corresponding to the desired motor rotation speed is accomplished by the digital motor control system in combination with power electronic circuit components. Both the qualitative explanation of brushless DC motors and analytic models and performance analysis are presented.

The type of motor known as a stepper motor also is presented. These motors advance angularly in single steps for pulses of excitation current, one angular step per excitation pulse. Their theory of operation and applications are examined.

CHAPTER 6

Chapter 6 is devoted to the entire vehicular powertrain including the traditional engine transmission drive axle coupling for a conventional vehicle. This chapter also presents a discussion of hybrid/electric vehicles. The chapter begins with a description of digital control electronics both qualitatively and quantitatively. This portion of the chapter is an extension of the basic concepts of electronic engine control introduced in Chapter 4. The discussion here concerns practical digital engine control electronics. In addition to the qualitative explanation, analytic models are developed for the control system with references to the basic discrete-time system theory of Appendix B.

Various control laws are presented for control of exhaust emissions and fuel economy. The goals of the engine control are to meet or exceed government regulations for emissions of the gases explained in Chapter 4 while optimizing important performance of the engine including fuel economy.

One of the benefits of digital control is its ability to compensate for various engine-operating modes including start-up, warm-up, acceleration, deceleration, and cruise as well as environmental parameters (e.g., ambient air pressure and temperature). The practical digital electronic engine control is capable of being adaptive to changes in vehicle parameters that can occur, for example, with vehicle age. As explained in Chapter 4, the vehicle must meet or exceed emission requirements for a specified number of miles driven. The digital engine control can assure engine emission performance for the specific period by being an adaptive control system and is explained here.

One of the design features of contemporary engines is variable valve timing (VVT) which also is called variable value phasing (VVP) and which can optimize a parameter called volumetric efficiency (see Chapter 4). The improvement in engine performance (while meeting emission requirements) through use of VVT/VVP is explained here, though the mechanism for implementing VVP is explained in Chapter 5 along with the associated actuator. The control subsystem for VVP is explained, and relevant analytic models are developed. The dynamic response characteristics of a VVP system are important for relatively rapid changes in RPM. The VVP models in this chapter are dynamic and are used in an analysis of the system dynamic performance.

Another subsystem of electronic engine control is idle speed control (ISC). There are vehicle-operating conditions under which ISC can maintain engine operation with minimum fuel consumption at idle (i.e., lowest operating) RPM. For example, if the vehicle is stopped by operator choice or traffic control, to avoid having to restart the engine, it is operated under control of the digital engine control system at a predetermined idle speed. In addition, a vehicle traveling downhill might require no engine power to maintain desired speed. In this case, the digital engine control maintains idle speed. The theory of operation of the ISC subsystem of the digital engine control is explained, and analytic models are developed for the described configuration. In addition, performance analysis of the ISC subsystem shows that the ISC is an adaptive control.

It is important to note that as of the time of this writing, there are vehicles for which the ISC is not alone in reducing fuel consumption for a stopped vehicle. Improvements in engine starting systems have permitted the engine to be shut off if the vehicle is stopped for a sufficiently long time. Reapplication of the throttle by the driver causes essentially an instantaneous engine start such that acceleration can occur relatively quickly. However, the ISC can maintain idle RPM for the short interval until the engine is shut off automatically. Vehicles with this feature can have significant reductions in overall fuel consumption, particularly those operated in heavy traffic urban environments. This automatic engine start/stop feature is commonly used in hybrid vehicles.

This chapter also explains electronic control of ignition that involves controlling the so-called ignition timing. Ignition timing refers to the angular position of the crankshaft relative to top dead center (TDC) that is the crankshaft angular position at which the piston is at the exact top of the compression stroke (also discussed in Chapter 4). Chapter 6 also gives a qualitative explanation and a partial analytic model for a closed-loop automatic ignition control system.

Explanation of the electronic control of the transmission (automatic) portion of the powertrain and the mechanical coupling from the transmission to the drive wheel axles (e.g., differential) are included in this chapter. There is a brief review of the mechanical components with illustrations. A qualitative explanation and analytic models of these components (including the torque converter) are presented. The gear ratio selection method, including the actuators involved for electronic control, is explained, as are the torque converter lockup methods mechanisms and actuators in the context of electronically controlled automatic transmissions.

A major portion of Chapter 6 is devoted to hybrid electric vehicles (HEVs). This section begins with a description of the physical configurations of two major categories of HEV that are known, respectively, as series or parallel HEVs. This explanation includes block diagrams of the two types of HEV and an explanation of their operation. Analytic models are developed for the electric portion of the HEV powertrain based on the discussion of electric motors in Chapter 5.

Performance analysis is derived from these analytic models. The performance analysis leads to an explanation of the control of an HEV. This control has many functions including the selection of the mechanical power source of the IC engine or the electric motor. The process by which energy is conserved during deceleration or braking involves converting the electric motor to a generator and storing the output electric power produced by the generator in a vehicle battery. In this section of Chapter 6, there is an explanation of the mechanisms by which the HEV achieves superior fuel economy compared to an IC engine only powered vehicle of comparable size and weight.

The performance analytic models relate the electric motor torque and power to this excitation. A representative HEV powered by an induction motor is explained via the analytic models and the electric excitation voltage. During electric motor propulsion operation of an HEV (with the engine off), the electric power comes from the vehicular storage batteries. The voltage level of these batteries is approximately constant and not compatible with the a-c voltages required to operate the drive electric motor. Chapter 6 explains the mechanism for generating the motor excitation voltages required for operating the motor at the power and speed required for any given vehicle-operating condition. Exemplary circuit diagrams and/or block diagrams for the voltage conversion in an HEV are presented here.

Chapter 6 concludes with a discussion of a purely electric vehicle (EV). Such a vehicle has some components found in an HEV, but it has no IC engine. Reference is made to the similar components found in an HEV.

CHAPTER 7

Chapter 7 discusses vehicle dynamic motion and the electronic control of this motion in terms of various subsystems. The chapter begins with a description of vehicle dynamic motion relative to a coordinate system that is fixed with respect to the earth. The subsystems involved in motion control include advanced cruise control, antilock braking systems, electronic suspension, electronic steering control, and traction control.

Some of the components of the subsystems discussed in Chapter 7 are used in subsystems described and explained in other chapters. For example, automatic braking control of individual wheels that is part of the antilock brake system is used in an enhanced stability system that is explained in Chapter 10 concerning vehicle/occupant safety. Although antilock braking is also vehicle safety-related and could potentially have been explained in Chapter 10, it is placed in Chapter 7 because it is primarily used for optimal braking.

The initial cruise control system discussed here is a traditional system that is designed to maintain a constant speed set for the vehicle by the driver. In cruise control, the engine throttle setting (for gasoline-fueled vehicles) is regulated by the cruise control system rather than by driver-controlled accelerator pedal.

Discussion of this subject begins with an analytic model of the vehicle forces that must be matched by the drive wheel torque/road force to maintain vehicle speed. These forces include tire rolling resistance, aerodynamic drag, and gravitational forces applied to the vehicle when it is traveling over a road that is not purely horizontal. The forces applied at the drive wheel are modeled in a simplified linear model for the purpose of presenting exemplary performance of a cruise control system. The basic concept of such a system is presented first in an analog (continuous-time) set of models. The performance of this simplified cruise control in terms of vehicle speed response to road slope discontinuities is determined from the models with various control laws.

Next, Chapter 7 explains the operation of a contemporary (essentially practical) representative digital control system. Discrete-time analytic models for the vehicle are developed from the continuous-time models using methods explained in Appendix B. Similarly, discrete-time models are developed for the closed-loop control system that, in practice, is implemented in a vehicle digital control system. References are made to discrete-time control theory in Appendix B. The performance analysis of the exemplary digital cruise control in response to a change in set point speed is given in this section.

One of the drawbacks of cruise control in which vehicle speed is regulated by throttle position occurs when the vehicle is traveling along a downward sloping road (e.g., on mountain roads). The minimum power from the engine is produced when the throttle is closed. In some instances of travel along a downward sloping road, the various vehicle friction forces are inadequate to maintain vehicle speed in the presence of an accelerating gravitational force.

This limitation is overcome in an advanced cruise control system that is explained in this chapter. An advanced cruise control system incorporates automatic braking for those operating conditions in which the vehicle would accelerate with the throttle closed. The configuration for such an advanced cruise control, an explanation of its operation, and analytic models all are presented in this chapter.

Chapter 7 also discusses components in the form of sensors and actuators used in cruise control. Analytic models are developed for these components that are included in the system models. Potential performance limitations of the system imposed by sensors and actuators are reviewed with methods of overcoming them. In addition, block diagrams are presented for representative cruise control systems that support the relevant system models.

Significant advances have been made in modern vehicle speed control systems that include automatic braking for collision avoidance. These advances are presented in Chapter 10, which is devoted to vehicle safety issues. It is possible to incorporate safety-motivated automatic braking as part of a vehicle motion control system that performs the cruise control function as one of its capabilities. The choice to discuss this level of speed control (or advanced cruise control) in Chapter 10 was based on the overall safety motivation for incorporating such systems in contemporary vehicles.

Another major vehicle subsystem discussed in this chapter is antilock braking (ABS). Significant improvement in vehicle braking in low tire/road friction (e.g., on ice) is achieved through the use of ABS. The discussion of ABS begins with a brief review of brake systems and their operation. The issues involved in reduced braking effectiveness (due to low road/tire friction) and how ABS can greatly improve braking with normal driver input while maintaining steering control are explained. The operation of ABS is explained in terms of automatic control of brake pressure for hydraulic brakes.

Analytic models are developed for the relationship between tire/road friction and variables related to vehicle motion. These models show the influence of road surface condition (i.e., dry vs. wet or ice covered) and the friction force component associated with vehicle deceleration. In addition, the friction model for lateral forces involved in steering and lateral stability of the vehicle also is explained in this chapter. This chapter explains the function of ABS in optimized braking and directional control for relatively low-friction conditions.

Various portions of the ABS are used in other vehicle applications. These include traction control and EVS. Traction control is discussed qualitatively in this chapter, but the EVS application fits better into Chapter 10 because the analytic models involved are more related to those of Chapter 10, which is concerned with safety-related vehicular electronics.

Chapter 7 also describes and explains the operation of an electronically controlled suspension system in technical detail with respect to the motion of the vehicle along a road (or off-road) surface. The suspension system has two major components: the sprung (car body) and the unsprung portion.

The two major performance issues for any suspension system are the so-called ride and vehicle handling characteristics. Dynamic analytic models are developed for the vehicle motion with respect to an earth-based inertial coordinate system as the vehicle moves over the earth surface. The models are continuous-time second-order differential equations that are linearized to simplify the quantitative suspension performance analyses. These equations are converted to transfer functions as explained in Appendix A.

The road surface induces motions of the relevant variables that are random processes. The suspension system response to these random processes is readily obtained from the statistical representation of the variables as inputs to the vehicle dynamic motion transfer functions. Using these models, the performance analyses of the suspension system yields a relationship between important properties and quantitative representation of both ride and handling and the parameters of the suspension system. The suspension system analysis leads to a table of optimum values for suspension parameters. However, improved ride performance often reduces vehicle handling characteristics and vice versa.

Included in this section is the presentation of an electronically controlled suspension in which the control system can vary suspension parameters. Actuators for varying these parameters and the configuration of a representative suspension control system are explained as are control strategies that can optimize the ride/handling qualities in driving circumstances in which handling is the dominant issue (e.g., cornering on rough roads). For such driving conditions, the representative control system changes suspension parameters in the sense of safe handling. In general, safety dominates over smooth ride. The control laws for such a suspension control system optimization choice are given. For example, when handling is not an issue (e.g., traveling over a straight smooth road), the ride can be optimized. This type of adaptive control law occurs in other electronically controlled vehicular subsystems.

Power steering also is presented in this chapter. Control for traditional power steering systems occurred via a control valve connected to the steering shaft. In contemporary vehicles, control is electronic. One of the earliest production vehicles to control steering electronically was a vehicle with

four-wheel steering (4WS). For this type of vehicle, the front steering wheels are under driver control (with some power steering boost). It is the rear wheels that also are steerable that are controlled electronically.

Relatively, simple linear analytic models for a 4WS vehicle are in the form of state variable equations. Two different levels of vehicle dynamic motion are presented: the first is a relatively simple one with minimal lateral dynamics that has a two-dimensional state vector; the second includes lateral dynamics and has a four-dimensional state vector. Analysis of vehicle motion with 4WS input is performed for both levels of complexity. An example of the dynamic response of a given vehicle to a lane change maneuver for 4WS in relation to conventional two-wheel steering is presented. This example is based on the physical parameters of a representative passenger car.

The components of the 4WS example vehicle can be used in automatic steering that is electronically controlled. Rather than introducing this topic in this chapter, it is presented in Chapter 12, which covers autonomous vehicles. The subject of automatic steering is introduced in the discussion of automatic parallel parking and lane tracking, which are commercially available at the time of this writing.

CHAPTER 8

Chapter 8 examines vehicular instrumentation. For many decades prior to the introduction of electronics in vehicles, instrumentation was devoted solely to provide drivers with measurements of important vehicular variables. These variables included vehicle speed, fuel quantity, engine oil pressure, and status of vehicle electric systems and, for some vehicles, engine RPM. The traditional, preelectronic instrumentation involved components that were purely mechanical, hydraulic, and incorporated simple electrical circuits. The variables were displayed on the instrument panel directly in front of the driver.

The fundamental components of an electronic instrumentation system/subsystem and the theory of their operation are explained in Appendices A and B. These appendices give quantitative explanations of instrumentation systems including algorithms (Appendix B) for accomplishing signal processing operations in a digital electronic instrumentation system. These components include a sensor that generates on electrical output signal that has a precisely known relationship to the variable being measured (ideally linear). Instrumentation of any form normally includes a display device capable of presenting the measured value of the variable to the driver. However, in contemporary vehicles, many display devices only provide a visual (and often audio) warning to the driver when the variable is out of limits, and the vehicle requires either repair or possible addition of a fluid (e.g., engine oil). Another component involved in electronic instrumentation is signal processing. The issues involved in signal processing and some implementation methods/devices for general electronic instrumentation also are explained in Appendix A. In contemporary vehicles, signal processing is implemented digitally as explained for discrete-time systems in Appendix B.

Chapter 8 deals with the specific signal processing for each example measurement discussed in vehicle instrumentation. The explanation of vehicular instrumentation begins with single-variable measurement systems. Later, the implementation of measurements of multiple variables via a single digital system (e.g., special-purpose computer) is presented. In contemporary vehicles, data for instrumentation along with other systems are passed along a dedicated vehicle network called “in-vehicle network” (IVN). The subject of an IVN is sufficiently important for various electronic systems that it is discussed in the chapter on vehicular communications (Chapter 9) rather than in Chapter 8. For

the purpose of explaining an instrumentation (IVN) network, references are made in Chapters 8 and to the relevant portion of Chapter 9.

The use of a common instrumentation computer for signal processing of the signals from multiple sensors to corresponding multiple displays is explained with respect to a block diagram. An example list of sensors and switches for vehicle monitoring status is presented. It can be seen from this list that many automotive instrumentation sensors are analog. The use of digital signal processing requires A/D conversion (explained in Chapter 3). The instrumentation computer processes each signal sequentially. The electronic mechanism for selecting one of N signals for processing (e.g., via multiplexing) also is explained in this chapter. In addition, exemplary signal processing algorithms for a number of typical signal processing operations are given quantitatively.

A major section of Chapter 8 is devoted to explaining vehicular display technology. It begins with an explanation and analytic model for a traditional electromechanical display in the form of a galvanometer. Contemporary vehicles sometimes incorporate this type of display as a part of the instrument panel.

Next, Chapter 8 discusses various electro-optic display devices based on a variety of materials and device configurations and explains the fabrication of arrays of electro-optic elements such that displays capable of depicting information in alphanumeric formats that can be fabricated. However, contemporary vehicles use arrays that also are capable of presenting pictorial formats much like the display of laptop computer or smartphone.

This modern picture capable display technology, however, follows a discussion of the various electro-optic basic principles. The physical principles and theory of operation are explained for each electro-optic technology. In addition, the analytic model relating the optical output for the electrical input is developed for each type of display technology discussion. These technologies include light-emitting diode (LED), liquid-crystal display (LCD), and vacuum fluorescent display (VFD). The electrical mechanism for varying the display optical intensity as a function of ambient light levels is explained in addition to the basic principles of the theory of the creation of display light levels. The relative advantages/disadvantages of each of the electro-optic technologies also are discussed.

As mentioned above, electro-optic display technology has the capability of displaying pictorial information in what often is referred to as a "flat panel display" (FPD). Chapter 8 explains that the configuration of an FPD is a two-dimensional array of individual electro-optic display elements (called pixels). Each pixel is sufficiently small that an image created on the FPD has a relatively high resolution. The creation of an image/picture superposed with alphanumeric data is accomplished by controlling the excitation of light from each electro-optic pixel. The method of controlling the display is explained in Chapter 8, along with a detailed explanation of its configuration. In addition, some analytic models are developed that yield a quantitative explanation of FPD technology. One important use of the FPD is the display of electronic maps that change under control of an instrumentation computer as the vehicle moves along its route. It is common place in contemporary vehicles to include a navigation system based on "global position satellites" (GPS). The FPD is capable of displaying the GPS calculated vehicle position on the associated displayed map.

In addition, it is possible with present-day technology to have more than a single FPD type of display. However, the format for any additional FPD type display should always be designed to minimize driver distraction. In aircraft applications, multiple FPD displays are termed "glass cockpits."

Another major vehicular technology introduced relatively recently is touch screen (TS) capability of an FPD which provides a user input to devices such as smartphones. Chapter 8 has a section that

explains TS technology and its use in vehicular applications. Analytic models are developed for the touch sensing technology and the way in which a user touching an FPD with TS capability provides an input to the instrumentation system. It is further explained that a TS capable FPD is functionally equivalent to a relatively large array of switches that would have to be built into the instrument panel to give the driver the ability to control various systems on the vehicle.

The user input to the TS comes from the location on the screen where it is touched (e.g., by one or more fingers). Chapter 8 explains the sensing mechanisms for detecting the contact point with the screen. This explanation includes analytic models for the sensing circuitry and gives exemplary circuit diagrams. The TS works in conjunction with the FPD that has symbols displayed that yield a specific input when the location on the screen where the symbol appears is touched. The details of the TS operation as instrumentation input are given here.

CHAPTER 9

Chapter 9 is concerned with vehicle communication systems both within the vehicle and with an external infrastructure (or potentially other vehicles). Communication systems in contemporary vehicles perform critically important functions for the operation of the vehicle, for navigation, and for information in addition to its more traditional role of entertainment (e.g., AM/FM radio). Communication within the vehicle exists in the form of a digital network that is termed "in-vehicle network" (IVN). There are several IVN systems available having different data rates, protocols, and costs. Chapter 9 presents four of the most commonly used IVNs incorporated (or in late stages of development) in contemporary vehicles. The communication media for these IVNs include wires, coaxial cable, and optical fibers. These IVNs are discussed in detail including a description of the physical layer and the protocol for the communication format. It is further explained how data are sent between the various electronic subsystems on board the vehicle. In-vehicle communication also is done using wireless medium. Both the theory and the applications are discussed in this chapter. The IVNs discussed include the "controller area network" (CAN), "flex ray," "local interconnect," (LIN) and "media-oriented systems transport" (MOST).

There are several issues in any IVN including data exchange rate, capacity for a given message, and system cost. Another issue is control of the network to assure access by any connected module, which includes assigning priority to a module whenever more than one is attempting access simultaneous (called "arbitration"). This chapter deals with this issue for each IVN and its protocol. Each IVN protocol in this chapter has a specific message format that is discussed.

The ability of IVNs to enhance the performance of a given vehicular system by incorporating data from other subsystems is another issue. The ability of any subsystem to obtain data from another system can expand the analytic model for the system, which often expands its operating envelope and can improve precision and accuracy of the variable(s) being controlled.

Each of the IVNs discussed in Chapter 9 requires circuitry to transmit and receive data along the network. Such circuitry is commonly referred to as a "transceiver." The representative circuitry for each IVN is presented and explained. Analytic models are developed for the operation of these circuits along with corresponding qualitative explanations. Wave forms are presented showing the individual signal models.

Chapter 9 also discusses communication between the vehicle and an external infrastructure. An example of this vehicle-to-infrastructure (V2I) communication is the global position system (GPS), which already has been mentioned with respect to navigation such as involved with the FPD. GPS operational theory is explained and detailed analytic models developed to assist this explanation. The operation and computation involved with GPS are illustrated with a somewhat simplified geometry for vehicle and satellites. However, the simplified geometry employed in this explanation of GPS allows for an analytic model that presents all of the important theoretical and practical aspects of GPS.

Communications V2I also includes cell phones with both voice and text capability that are the same as for normal cell phone use. The technology exists that provides a wireless connection from a cell phone to the vehicle electronic system that provides the capability of hands free cell phone use. The incoming vocal signal is sent to the vehicle loudspeaker such that the driver can listen to an incoming call without the necessity to hold the phone by hand. Depending on the system configuration, the outgoing audio can be picked up either by the cell phone microphone or by one installed in the vehicle audio electronics. Equally important to driving safety is the capability of the system to make a phone connection (i.e., to dial a number) verbally.

The capability of a driver to verbally dial another telephone comes from advanced voice recognition software. Again, depending on the system configuration, the voice recognition can be part either of the cell phone or the vehicle digital electronics. The software for voice recognition has advanced to the point that the driver can verbally compose and send a text message. The hardware for an in-vehicle wireless link between cell phone (as well as other digital devices) and the vehicle is explained in this chapter. However, the details of the software are beyond the scope of this book.

Chapter 9 does explain the theory of the operation of cell phone communication including its coding schemes (e.g., CDMA and TDMA). In this chapter, the code division multiple access (CDMA) is explained in detail with relevant analytic models. Detailed specific examples of coding are presented to illustrate this sometimes complex process. This chapter also explains the modulation techniques used in the cell phone radio link from the phone to the communication infrastructure. This explanation includes example circuitry for modulation and demodulation with accompanying analytic models. The modulation technique for cell phones minimizes the influence of variations in carrier signal strength due to multipath propagation and cell phone motion.

Another multiple user scheme available for cell phones is the so-called “time domain multiple access” (TDMA) technique. This technique involves assigning specific time slots within each cycle of cell phone operation to a specific pair of users. This chapter explains and models TDMA.

In addition, the cell phone infrastructure is described and explained. This infrastructure, which provides the link between each pair of users that are communicating, involves multiple fixed transceiver/antenna stations (cell towers). It also has controlling systems that maintain the connection when one or both users are moving. This is particularly important for users in vehicles that change from one cell tower to another. A somewhat complex control is required to maintain a given connection and is described here.

Chapter 9 also discusses a short-range wireless link for connection of a subsystem/device to the vehicle electronic systems. This wireless connection is called “Bluetooth” and involves a unique method of maintaining connection that is called “frequency hopping” (FH). In this FH technique, there are 79 possible carrier frequencies in the microwave portion of the electromagnetic spectrum. The carrier frequency linking a pair of devices changes in a pseudorandom fashion from one of these 79 to another, the details of which are discussed here. Once paired in Bluetooth, the pair of devices switches carrier frequencies synchronously to maintain the wireless connection.

Also presented are other short-range wireless vehicle communication applications for the Bluetooth system or equivalent. For example, it is possible to send vehicle maintenance stored data to a nearby vehicle diagnostic system. There are several other potential applications for this wireless communication system mentioned in this chapter.

Still another V2I communication system is the so-called digital audio broadcasting (DAB). In one DAB application, the broadcast is initialed at a satellite and is used primarily for entertainment. It is used both for fixed and mobile receivers.

DAB incorporates a relatively complex multiple carrier link to receivers. This system is called "orthogonal frequency-division multiplexing" (OFDM). Chapter 9 presents a detailed explanation of OFDM along with detailed analytic models. Also included in the explanation of OFDM are block diagrams and a few representative circuit diagrams. The OFDM uses discrete Fourier transforms and the corresponding inverse discrete Fourier transforms based on the theory presented in Appendix B to achieve the desired multiplexing of various channels being broadcast. Chapter 9 presents examples of the OFDM use to aid in the understanding of this relatively complex process.

CHAPTER 10

Chapter 10 is devoted to vehicular electronic safety-related systems. These systems include means for preventing or minimizing injuries to occupants in the event of a potentially damaging accident. In addition, this chapter discusses some relatively recent systems for preventing accidents.

Occupant protection systems include airbags that are an important supplement to seatbelts. This chapter brings the discussion of airbags up to date. It explains the theory of operation of an airbag system including a discussion of distinguishing an accident scenario requiring airbag deployment vs. other abrupt motion inputs to the vehicle (e.g., driving over a large pothole).

Representative physical configurations are described qualitatively with illustrative figures. Block diagrams for the associated electronic systems are presented along with analytic models for the various components. These models are combined to result in a performance analysis for exemplary airbag systems. The importance of sensors and signal processing for detailing a crash scenario such that airbag deployment is required is explained. Improper deployment of an airbag due to a noninjury-producing incident can actually lead to an accident, since a deployed airbag can block temporarily the driver's forward view.

Chapter 10 begins with a brief review of the earliest airbag configurations to provide a reference for the significant improvements in occupant safety that have occurred. Circuit diagrams and relatively straightforward analytic models are presented for these early configurations. This chapter then progresses through the technological advances in both sensing and signal processing for proper crash detection. Numerous exemplary algorithms are presented as well in this chapter.

The next safety-related topic covered in Chapter 10 is referred to as "blind spot detection" (BSD). This section explains the problems faced by a driver with observations of the space around the vehicle for all driving conditions. There are multiple technologies that assist the driver in detecting other vehicles or objects that are not within the field of view. These technologies include radar, lidar, and electronic cameras. In addition, signal processing is explained for detecting objects that could potentially cause an accident. Analytic models are derived for some of the calculations involved in BSD systems, including image recognition. Some of the prominent BSD algorithms also are discussed in this chapter.

Another safety-related topic discussed in Chapter 10 is "automatic collision avoidance systems" (ACAS). ACAS is an extension of the sensing systems employed in BSD. Electronic monitoring of the traffic/obstacle environment surrounding a CAS-equipped vehicle has signal processing that can predict potential collisions as explained in this section of the chapter. This signal processing has the capability to detect when a collision is imminent. If there is no driver action that can avoid the collision, automatic collision avoidance takes action. One action taken by ACAS (when a frontal or near-frontal collision is determined to be likely to occur by the signal processing) is automatic braking. This system uses components of ABS to operate vehicle brakes. For some ACAS equipped vehicles, seatbelt pretensioning is also automatically applied.

Chapter 10 develops analytic models to explain the mechanism by which the ACAS can determine that a collision is imminent. These models include representative calculations that are made on the sensor data. Additional models are given in this chapter for vehicle motion with automatic braking in action.

In addition to automatic braking, there are collision avoidance systems that employ automatic steering. However, this applies to certain levels of autonomous vehicles, and this mechanism of collision avoidance is discussed in Chapter 12. Only a brief reference to this topic is made in this chapter.

CHAPTER 11

Chapter 11 is devoted to the diagnosis of problems in vehicles via electronic systems or subsystems that assist technicians in repairing vehicles. Electronic diagnostic capability exists in the vehicle electronic systems themselves and in service-bay systems (e.g., at auto dealerships). From the earliest days of digital control systems, the controlling computer has had some self-diagnostic capability. For such control systems, the corresponding fault code is stored in memory once a failure or malfunction is detected by the system. The fault codes can be downloaded to a service-bay system for the purpose of diagnosing vehicle system problems. Some representative fault codes are presented in this chapter to give examples of the nature of the type of component malfunctions that can be detected. These codes are part of an SAE recommended practice that can standardize fault codes.

Chapter 11 illustrates representative diagnostic procedures that are followed by a service technician using a service-bay diagnostic tool. This tool, that has a display similar to a computer, presents a sequence of steps to be followed by the technician in the form of flow charts, several examples of which are presented in this chapter to illustrate the procedures for diagnosing a selected few component malfunctions.

In addition to assisting the servicing of vehicles, there are governmental regulations requiring vehicles to have certain self-diagnostic capabilities (called "on-board diagnostics" or OBD). OBD requirements pertain to malfunctions that affect vehicle exhaust emissions. One such requirement discussed in this chapter is called misfire detection. Misfire refers to improper combustion that can result, for example, from incorrect air/fuel (see Chapter 4) or possibly a failed spark plug. The EPA requires that misfires be detected and that a warning message be displayed to the driver to have the powertrain repaired when misfires exceed a specified threshold.

Chapter 11 presents a representative misfire detection method based on sensing and processing crankshaft instantaneous angular speed fluctuation that results from a misfire. An analytic model for crankshaft rotational dynamics upon which signal processing of the output of a sensor that measures

crankshaft angular speed yields an indication of misfire. An explanation of this model-based misfire detection includes, in addition to the model, the system block diagram, the signal processing algorithms, and a criterion for setting the misfire diagnostic code. In addition, actual experimental results of the performance of this system are presented to demonstrate its ability to meet the misfire detection regulatory requirements.

Chapter 11 also presents a brief discussion of the use of certain aspects of artificial intelligence in diagnosing vehicle system problems/malfunctions. One such aspect of artificial intelligence is the use of a so-called expert system. This chapter explains how diagnostic procedures followed by a service technician are based on the knowledge of recognized experts in the corresponding vehicle technology.

Finally, the chapter explains the procedures followed in developing an expert system for diagnosing problems/malfunctions in vehicles. Once it is developed, the inputs to the system can consist of symptoms entered by a technician in addition to the set of fault codes. Representative examples of the use of a vehicular expert system are presented illustrating how such a system would be used by technicians.

CHAPTER 12

Chapter 12 is devoted to autonomous vehicles that at the time of this writing are in a research and development phase. The end goal of this development will be to have a vehicle that does not need a driver. The autonomous vehicle will be controlled by a computer and various automatic subsystems. The chapter begins with a summary of the actions taken by a driver that the computer must be capable of undertaking.

Autonomous vehicles are classified in multiple levels depending on the amount of driver action required. At the lowest level, a driver is required to observe the environment and make normal driving decisions. At the highest level, no action is required by a driver, and the vehicle becomes entirely driverless.

All of the vehicle automatic systems required for an autonomous vehicle are already developed. Nearly all such systems are covered in previous chapters. An important automatic subsystem required for an autonomous vehicle that is not covered in previous chapters is automatic steering.

Automatic steering is available in very limited form in some production vehicles. This existing automatic steering is of the form of automatic parallel parking and automatic lane tracking. Analytic models for automatic steering (with parallel parking as an example) are developed, and hardware components are described, including physical configurations and models that are explained in detail. In addition, an example of the steering deflection is developed. A computer simulation of an exemplary automatic parallel parking also is developed with the performance presented graphically.

Given the explanation of automatic steering, Chapter 12 presents a representative block diagram of an autonomous vehicle. This block diagram depicts the automatic systems as blocks and presents a set of sensors required. In addition to the sensors required to successfully control each automatic system, a set of sensors is depicted that provide the data and information required for the complete system to evaluate the environment surrounding the vehicle with a full 360 degrees field of view. These sensors include the vehicular radar, lidar, and camera systems (with image identification software) that are discussed with respect to blind spot detection in Chapter 10. Any fully autonomous vehicle must have detailed information on its surrounding environment that includes lane tracking and other vehicle

and obstacle detection. Performance requirements of this sensor system for safe operation of an autonomous vehicle are presented.

In addition to sensing the environment, however, an autonomous vehicle must have software capable of evaluating this environment and making decisions similar to these performed by a human driver. The software must be capable of making relatively short-term predictions about changes in the environment that require action by the control system. This very important software is the primary area of research and development for the highest levels of autonomous vehicles. Essentially, the hardware elements already exist that provide the capabilities required for an autonomous vehicle. The development of the necessary software will require testing for any possible scenario that could require a decision or prediction to be made. This aspect of the development is challenging. It is generally agreed among the vehicle manufacturers and governmental regulatory agencies that successful and safe autonomous vehicle operation will require a communication infrastructure for V2I and V2V communications. Chapter 12 describes some of the features and technology associated with this communication infrastructure.

An element of autonomous vehicles that is necessary for safe operation is hardware redundancy. The failure or severe degradation in performance of a component involved in the automatic driving of a vehicle is potentially hazardous and requires some form of redundancy (e.g., a backup replacement component for the failed component). Chapter 12 explains the importance of reliably detecting and isolating the failed component. A specific example of an automatic steering actuator failure with an associated hardware redundancy is also given.

Following the completion of the necessary software and hardware redundancy and the completion of the required communication infrastructure, the routing operation of autonomous vehicles should be possible. This chapter discusses an example of the operation of an autonomous vehicle on a trip assuming these final components of the overall system are complete. This chapter explains the necessary navigation on any such trip, including the use of digital maps that are explained in Chapter 9. One method of navigation involves having the autonomous vehicle user select a destination. The navigation component of the autonomous vehicle will have the vehicle starting location from GPS. An optimal route between the starting point and intended destination is already a routine part of GPS/electronic map technology.

Chapter 12 explains that navigation along this route involves a tracking-type control. A detailed model of tracking of a given contour along a curve to illustrate this type of tracking control problem and its solution is presented. A block diagram of the tracking control portion of a hypothetical autonomous vehicle configuration is included. Analytic models are developed for the tracking control problem and a closed form solution is derived and presented to demonstrate an example of some of the detailed operation of autonomous vehicle navigation.

Given the demonstrated automatic navigation capability and the existence of all required vehicular automatic systems, Chapter 12 concludes that the only missing elements of routine autonomous vehicle operation at the time of this writing are the software, hardware redundancy, communication infrastructure, and governmental regulation.

CHAPTER

ELECTRONIC FUNDAMENTALS 2

CHAPTER OUTLINE

Semiconductor Devices 24
- Diodes 27
- Zener Diode 29

Electro Optics 30
- Photo Conductor 31
- Photo Diode 32
- Light Generating Diode 34
- Laser Diode 34

Rectifier Circuit 35
- Communications Applications of Diodes 37
- Transistors 37
- Field-Effect Transistors 45

FET Theory 47
- FET Amplifier 50

Integrated Circuits 52
- Operational Amplifiers 53

Use of Feedback in Op-Amps 54

Summing Mode Amplifier 56
- Comparator 57
- Zero-Crossing Detector 58
- Phase-Locked Loop 58
- Sample and Zero-Order Hold Circuits 60
- Zero-Order Hold Circuit 63
- Bidirectional Switch 64

Digital Circuits 66

Binary Number System 68

Logic Circuits (Combinatorial) 69
- AND Gate 70
- OR Gate 70
- NOT Gate 70
- Boolean Algebra 71

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-12-810434-7.00002-8
Copyright © 2017 Elsevier Inc. All rights reserved.

Exemplary Circuits for Logic Gates 71
Combination Logic Circuits 75
Logic Circuits with Memory (Sequential) 77
R-S Flip-Flop 77
JK Flip-Flop 78
D Flip-Flop 79
Timer Circuit 80
Synchronous Counter 83
Register Circuits 83
Shift Register 84
Digital Integrated Circuits 86
The MPU 87

This chapter is for the reader who has limited knowledge of electronics. It is intended to provide an overview of the subject so that discussions in later chapters about the operation and use of automotive electronics control systems will be easier to understand. The chapter discusses electronic devices and circuits having applications in electronic automotive instrumentation and control systems. Topics include semiconductor devices analog circuits, digital circuits, and fundamentals of integrated circuits.

SEMICONDUCTOR DEVICES

All of the active circuit devices (e.g., diodes and transistors) from which electronic circuits are built are fabricated from so-called semiconductor materials. A semiconductor material in pure form is neither a good conductor nor a good insulator. The ability of a material to conduct electric current is characterized by a property called conductivity. A model for current flow in semiconductor materials and an explanation for electric conductivity are developed later in this chapter. A metal such as copper, which is a good conductor, has a relatively high conductivity such that current flows in response to relatively low applied voltage. An insulator such as mica has a relatively low conductivity such that essentially zero current flows in response to an applied voltage. A semiconductor material has conductivity somewhere between that of a good conductor and that of a good insulator. Therefore, this material (also called semiconductor material) and devices made from it are semiconductor devices (also called solid-state devices).

There are many types of semiconductor devices, but transistors and diodes are two of the most important automotive electronics. Furthermore, these devices are the fundamental elements used to construct nearly all modern integrated circuits. Therefore, the discussion of semiconductor devices will be centered on these two. Semiconductor devices are made primarily from silicon or germanium (although other materials, e.g., gallium arsenide, are also in use) that is purposely infused with impurities that change the conductivity of the material.

The conductivity of a pure semiconductor can be varied in a predictable manner by diffusing precisely controlled amounts of very specific impurities into it. The process of adding impurities to silicon is called "doping." Boron and phosphorus are often used as impurity source materials to alter the conductivity of

silicon. When boron is used, the semiconductor material becomes a so-called p-type semiconductor. When phosphorus is used, the semiconductor material becomes an n-type semiconductor.

In order to understand the operation of these transistors and diodes, it is helpful to understand the basic physical mechanism of electric conductivity in both n-type and p-type semiconductor materials. The flow of an electric current through any material is due to the motion of electrons in the material in response to an applied electric field. This electric field results from the application of a voltage at the external terminals of the corresponding structure. The variable called an electric field in this chapter is a component of the general theory that is known as "electromagnetic field theory." This theory forms the basis of modeling all electric phenomena. This electric field is represented by a vector that is known as electric field intensity and denoted as $\bar{E}$ in this book. Although the advanced details of electromagnetic field theory are beyond the scope of this book, somewhat simplified theoretical models are presented in later chapters (e.g., Chapter 5). For the purpose of explaining electric properties of semiconductor materials, we present the simplest model of electric field intensity in which the magnitude varies in proportion to applied voltage and inversely with the distance between the electrodes to which the voltage is applied. The electrons that move in response to this electric field originate from the individual atoms that make up the material.

For a basic understanding of conductivity, it is helpful to refer to Fig. 2.1 that depicts a relatively long, thin slab of semiconductor material across which a voltage is applied.

In this figure, the electric field intensity is a vector denoted as $\bar{E}$ that is x-directed. In this book, vectors are indicated by a bar over the symbol for the vector as exemplified by the electric field intensity $\bar{E}$. A voltage v is applied to a pair of conducting (e.g., Cu) electrodes. For this relatively long, thin semiconductor material, the magnitude of the electric field intensity E is approximately constant over the semiconductor and is given approximately by

$$E = \frac{V}{L}$$

The vector $\bar{E}$ is given by

$$\bar{E} = E\hat{x}$$

where $\hat{x}$ = unit vector in the x direction.

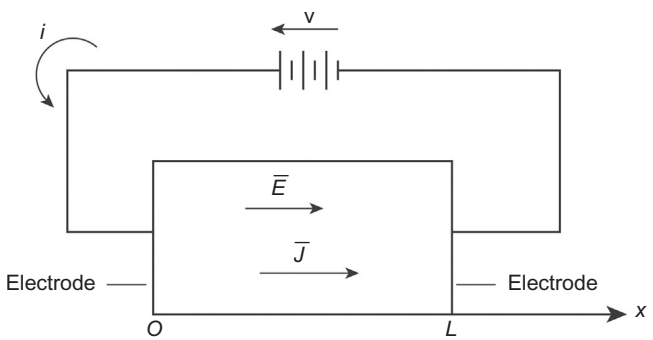


FIG. 2.1 Illustration of current conduction in semiconductor.

Also shown in Fig. 2.1 is the current density vector $\bar{J}$, which is also an x-directed vector. The current density vector is proportional to the electric field intensity:

$$\begin{aligned}\bar{J} &= \sigma\bar{E} \\ &= \sigma E\hat{x}\end{aligned} \tag{2.1}$$

where σ is the conductivity of the material. The magnitude of the current density J is the current per unit cross-sectional area (which by the assumption of essentially constant E is constant) and is given by

$$J = \frac{i}{A_c} \tag{2.2}$$

where A_c is the cross-sectional area of the slab in the y, z plane. The reciprocal of conductivity is known as the resistivity ρ of the material:

$$\rho = \frac{1}{\sigma} \tag{2.3}$$

The explanation of electron flow in any material is based upon the "band theory of electrons." This theory is a major component in modern atomic physics. According to this theory, the energy of the electrons associated with the atoms making up a material is constrained to certain ranges called bands. Any given electron will have an energy within one of these bands, and no electron can have energy outside these bands. Within each band, the electrons can have only discrete energy levels, and only one electron can "occupy" a given energy level. Consequently, the number of electrons within each band for any atom is constrained to the number of "allowed" energy levels. An electron can only move in response to an applied electric field and contribute to current flow if there is an unoccupied energy level to which it can move as its energy changes due to the electric field intensity force acting on it.

All of the energy levels of the lower energy bands of an atom are filled such that there is no energy level to which an electron can move in response to an applied electric field. Thus, these lower band electrons cannot contribute to current flow in response to an applied voltage. The electrons in the outermost band, known as the conduction band, are the least tightly bound, and for a material such as Si, they are few in number relative to the number of energy levels in that band. These outer band electrons can move to an adjacent energy level and effectively move freely in response to an applied electric field. These electrons are called "free electrons." Doping Si with phosphorus impurity results in an excess of free electrons relative to pure Si. The doped material is said to be an "n-type" semiconductor and has a conductivity that is greater than the undoped Si.

The next lowest energy band from the outermost is called the "valence band" since it is associated with the chemical valence of the material (in this case Si). The energy levels of this band are nearly (but not completely) filled. However, doping a semiconductor with a p-type impurity (e.g., doping Si with boron) yields a relative excess of energy levels in this valence band. The resulting doped material is called a p-type semiconductor. Electrons in this band can move to the available energy levels created by doping in response to an electric field, thereby contributing to current flow. However, functionally, this p-type material behaves as though it had excess of positively charged particles called "holes." The model for current flow in a semiconductor and the explanation of semiconductor devices use the fictitious holes and their response to an applied field as a basis for the contribution they make to current flow. The terminology used to describe these charge carriers is as follows: in n-type material electrons are called "majority carriers" and holes called "minority carriers"; the reverse is true in p-type material.

Doping a semiconductor material changes the relative densities of holes and electrons. However, there is a basic relationship between these densities, which is preserved regardless of the doping concentrations. If one starts with an intrinsic semiconductor such as Si that has an equal concentration of "free" electrons and holes (since each free electron leaves a "hole" in the valence band for an intrinsic semiconductor), we denote this concentration $n_i = 1.5 \times 10^{10}/\text{cm}^3$.

Doping Si with either a p-type or an n-type impurity changes the concentrations. Denoting electron density n, and hole density p, the following equation expresses the relationship between these concentrations under thermal equilibrium:

$$np = n_i^2 \tag{2.4}$$

There is another basic aspect of semiconductor physics that plays a role in the electric characteristics of semiconductor electronic components. Whenever a voltage V is applied to a slab of semiconductor material, it creates an electric field that is represented by the electric field intensity vector $\bar{E}$ as described above (in this text, the over bar for a variable is the notation indicating that the variable is a vector).

In a semiconductor material, any electric field due to an external potential causes the electrons and holes to move with mean velocity vectors $\bar{v}_e$ and $\bar{v}_h$, respectively. These velocities are given by

$$\bar{v}_e = \mu_e \bar{E}$$
$$\bar{v}_h = \mu_h \bar{E}$$

where μ_e is the electron drift mobility and μ_h is the hole drift mobility.

These mean velocities yield electron and hole current densities $\bar{J}_e$ and $\bar{J}_h$, respectively:

$$\bar{J}_e = nq\bar{v}_e$$
$$\bar{J}_h = pq\bar{v}_h$$

where q is the charge on an electron (1.6×10^{-19} coulomb). These relationships will appear in models for various components in this text.

Throughout this book, current flow is taken to be conventional current in which the direction of flow is from positive to negative, whereas in reality, current consists of electron motion from negative to positive. This choice of current is merely convenient for notational purposes and has no effect on the validity of any circuit analysis or design.

DIODES

The first electronic component to be considered is a device called a "diode." A diode is a two-terminal electric device having one electrode that is called the anode (a p-type semiconductor) and another that is called the cathode (an n-type semiconductor). A solid-state diode is formed by the junction between the anode and the cathode. In practice, a p-n junction is formed by diffusing p-type impurities on one side of the intended junction and n-type impurities on the other side of a region of an intrinsic semiconductor (e.g., Si).

The region in which the diode material changes from p-type to n-type material is called the p-n junction (or simply junction). The junction region is relatively short but plays a critical role in the diode operation. When the junction is formed, electrons in the vicinity of the junction migrate from the n-type to the p-type. Similarly, holes in the region migrate from p-type to n-type. This migration leaves behind

a positively charged dopant ion on the n-side and a negatively charged dopant ion on the p-side over a region known as the depletion region that creates a charge distribution that in turn creates a potential difference between the two regions. In equilibrium conditions, this potential inhibits further current flow. This potential is known as the junction barrier potential since it acts more or less like a barrier to the current flow. A detailed model for the relationship between the charge distribution in the depletion region and the potential (or equivalent voltage) is presented in the section of Chapter 5 on capacitor modeling. However, for the present discussion, it is only the existence of the barrier potential that is required for the operation of semiconductor device.

The current, which flows through the diode in response to an applied voltage, depends upon the polarity of the voltage and its magnitude. Fig. 2.2 illustrates the schematic symbol for a p-n diode showing the p-type (anode) and n-type (cathode) sides of the junction. If a voltage is applied with positive on the anode and negative on the cathode, it is said to be "forward biased." For the opposite polarity, the diode is said to be "reverse biased." Forward bias reduces the junction barrier potential, thereby increasing current flow. Reverse bias increases that potential, thereby inhibiting current flow.

The current through a forward-biased diode increases exponentially with applied voltage V, whereas the reverse-biased flow reaches a very low saturation current I_s. A model for this current flow is

$$I = I_s(\exp(V/nV_T) - 1) \tag{2.5}$$

where I_s and n are parameters that are specific to a particular diode. The parameter V_T is called the thermal voltage and is given by

$$V_T = kT/q$$

where k is the Boltzmann's constant, T is the junction absolute temperature, and q is the electron charge. At room temperature, $V_T \cong 26$ mv. The parameter n is normally between 1 and 2, and I_s is a few μ amp. Fig. 2.3 depicts this current flow vs. diode junction applied voltage V. The reverse-bias current is too small to be shown.

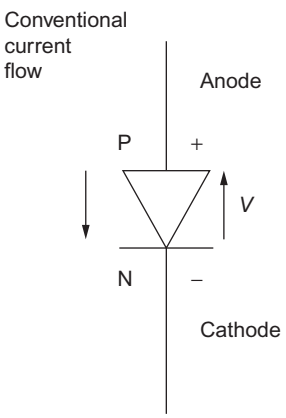


FIG. 2.2 Schematic symbol for p-n diode.

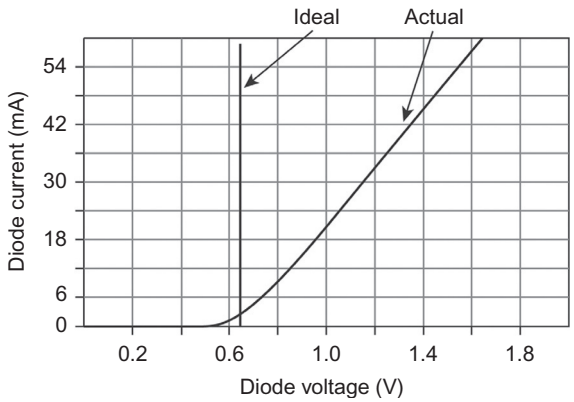


FIG. 2.3 Transfer characteristic. Diode transfer characteristics.

Although the model given above for diode voltage current characteristics is a very good representation for a practical diode (provided that the reverse-bias voltage is below its breakdown voltage), it is generally not necessary to represent the diode with this degree of accuracy for most circuit analysis or design purposes. Normally, it is sufficient for the voltage levels involved in automotive electronics to represent a p-n diode as a polarity-dependent switch as characteristic in associated figures. The switch can be modeled as being open for reverse bias and closed for forward bias. With this model, the diode current in the forward bias is limited by the external circuit components to which it is connected. The reverse-bias current is taken to be zero.

ZENER DIODE

A special p-n junction diode having a unique reverse-bias characteristic called a zener diode has many applications in electronic circuits. The transfer characteristics for a zener diode are depicted in Fig. 2.4A. The circuit symbol for a zener diode is depicted in Fig. 2.4B in which the cathode has a unique shape.

The forward-bias characteristics are similar to any p-n junction diode. The reverse-bias current is extremely low for voltages $-V_z < V_D \leq 0$. However, when the reverse voltage reaches $-V_z$, the current increases abruptly, while the voltage remains nearly constant at $V_D = -V_z$. The voltage V_z is called the zener voltage.

The operation of any p-n junction diode at specific reverse-bias voltages (called avalanche voltages for an ordinary diode) abruptly increases. At this voltage level, the energy of charge carriers is sufficient to cause further ionization due to collisions that create more carriers and increase the reverse current by a large amount. For an ordinary diode, the avalanche conduction causes sufficient heating to destroy the diode.

However, a zener diode is created with a special doping profile particularly in the vicinity of the electrodes. The zener diode can sustain relatively large reverse bias currents while maintaining $V_D \cong -V_Z$. There are many applications in electronic circuits that require a fixed voltage level over a wide range of currents. These applications are discussed at various places in example circuits in later chapters.

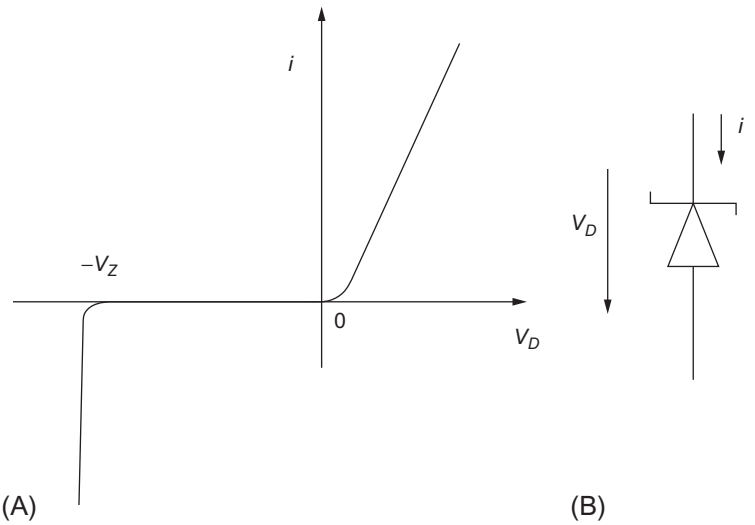


FIG. 2.4 Zener diode transfer characteristics (A) and circuit symbol for zener diodes (B).

ELECTRO OPTICS

Another important aspect of semiconductor materials and p-n diode junctions is the interaction between the material and optical power (light). Light is the propagation of energy in the form of quantum units called photons. Any given photon has a specific frequency of oscillation (f). The speed of propagation of light depends upon the medium in which propagation occurs and the frequency of oscillation. In a vacuum, this speed (denoted as c_o) is essentially 3×10^8 m/s. In any other medium, $c = c_o/n$ where $n =$ index of refraction for the material ($n \geq 1$). Light from an incandescent (high temperature) source contains a broad spectral distribution of frequencies that is characterized statistically by its power spectral density. Since light propagates as an electromagnetic wave, the power spectral density can also be expressed as a function of wave length λ

$$\lambda = \frac{c}{f}$$

where $c = c_o/n$.

Thermally generated light has a spectrum that is spread over a relatively broad range of wavelengths. On the other hand, light generated by a laser occupies a relatively narrow spectrum (ideally but not practically) at a single wavelength. Laser generated light is commonly used in vehicle electronics.

Light that is incident on the surface of a semiconductor material is partially reflected and partially absorbed by the material. The relationship between incident, reflected, and absorbed light is a function of the type of material and the spectral distribution of the incident light. Any incident photon crossing

the semiconductor material boundary can interact with various atoms that make up the material structure. For a photon to be absorbed by the semiconductor, the photon energy E_p must match the valence/conduction band-gap energy such that the photon energy ionizes the atom, thereby creating a hole-electron pair. This ionization process (called photoionization) causes the electric conductivity of the semiconductor material to increase in proportion to the absorbed light.

PHOTO CONDUCTOR

A specially designed semiconductor structure that absorbs incident light as represented by the so-called light intensity (which is the optical power per unit of cross-sectional area) is termed a photoconductor. A simplified model for a photoconductor is based on the relatively simple configuration that is depicted in Fig. 2.5.

The simplified configuration of the illustrative photoconductor consists of a slab of semiconductor material having a rectangular cross section of area A_c and of uniform thickness ℓ with a pair of conducting electrodes e_1 and e_2 attached at the ends as shown. The incident light is represented by optical intensity I. The electric conductivity σ of the semiconductor is assumed to be uniform over the semiconductor and is given by

$$\sigma(I) = K_p I$$

A voltage V is applied to the electrodes creating an electric field $\bar{E}$ within the material. In this simplified model, this electric field is assumed to be uniform over the material and given by

$$\bar{E} = -\frac{\mathrm{v}}{\ell}\hat{z}$$

where $\hat{z}$ = unit vector in + z direction.

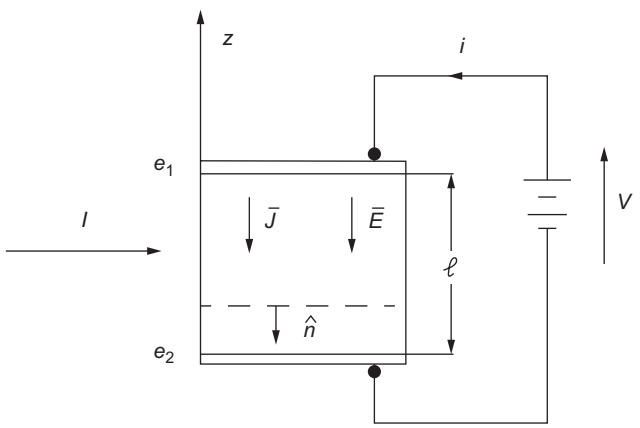


FIG. 2.5 Photoconductor configuration.

The current density vector $\bar{J}$ is given by

$$\bar{J} = \sigma(I)\bar{E}$$
$$= \frac{-Vk_pI}{\ell}\hat{z}$$

The total current flowing through the photoconductor i is given by

$$i = \int_S \bar{J} \cdot \hat{n} ds$$

where
S = cross section of the material

$$\hat{n} = \text{unit vector normal to } S$$
$$= -\hat{z}$$

Since the vector $\bar{E}$ is assumed to be uniform over S, the current is given by

$$i = \frac{VA_c}{\ell}\sigma(I)$$
$$= \frac{k_pVA_c}{\ell}I$$

where A_c = cross-sectional area of photo conductor.

The conductance of the photo conductor G that is the reciprocal of its resistance is given by

$$G = \frac{i}{\mathrm{v}}$$
$$G = k_p\frac{A_c}{\ell}I$$

That is, the conductance varies linearly (for the illustrative simplified example) with incident light intensity. In effect, the photoconductor is a sensor for incident light intensity.

PHOTO DIODE

In addition to photoconductive optical sensors, it is possible to fabricate a p-n junction optical sensor called a photodiode. As in the case of the photoconductor, the simplified physical configuration of a photodiode is somewhat similar to Fig. 2.5 except that the region near e_1 is doped such that it is an n region and the region near e_2 is doped to be p. In such a structure, the depletion region interacts with incident light in the same way as a photoconductor material such that photoionization creates hole-electron pairs. The photodiode must be fabricated such that the depletion region is exposed to the incident light by having a transparent cover.

In light detecting applications, the photodiode is reverse biased such as is depicted for the circuit of Fig. 2.5 in which e_1 is the cathode and e_2 is the anode of the doped semiconductor slab. In the absence of illumination of the p-n photodiode, the reverse-bias current is extremely small (ideally zero). Assuming that the incident light spectrum corresponds to the interband energy of the atoms in the depletion region and that a substantial portion of the incident light crosses the boundary of the semiconductor material,

the hole-electron pairs are created in sufficient numbers to substantially increase the reverse-biased current. In this case, the diode reverse-bias current $i(I)$ is proportional to the incident light intensity I and essentially independent of voltage and is given by

$$i = K_{pd} I$$

where K_{pd} = constant for the diode structure.

An idealized representative set of characteristic curves for an ideal photodiode is presented in Fig. 2.6. In Fig. 2.6A, the intensity increases with the index (i.e., $I_{n+1} > I_n$).

Fig. 2.6B depicts an idealized photodiode optical sensor circuit. In this circuit, the voltage across the load resistor R_L is given by

$$\begin{aligned} V_L &= R_L i \\ &= K_{pd} R_L I \quad V_L < V_{\text{sup}} \end{aligned}$$

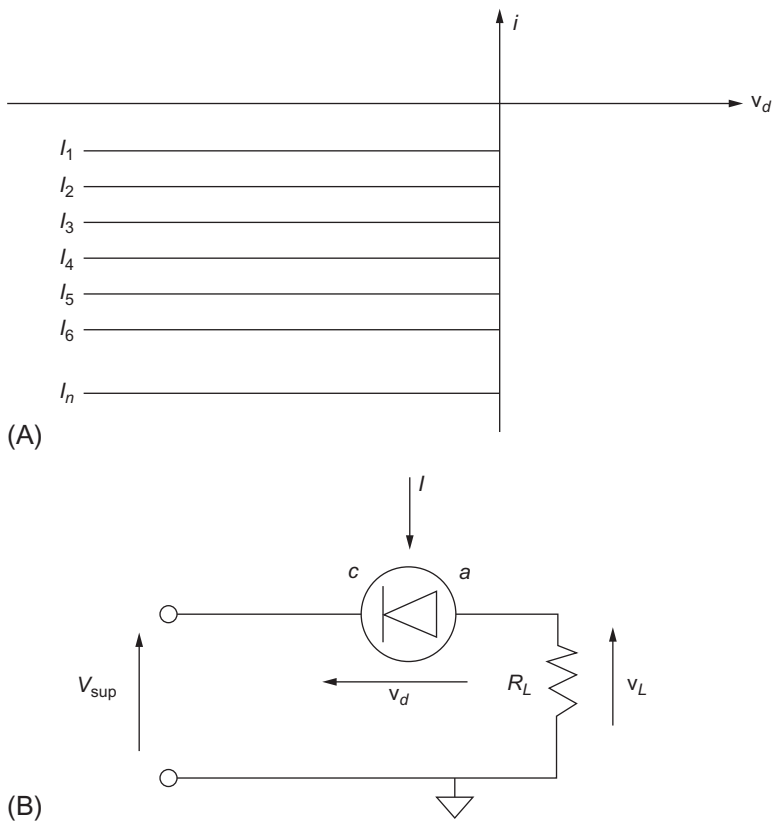


FIG. 2.6 (A) Photodiode characteristic curves. (B) Photodiode optical sensor circuit.

The supply voltage v_{sup} should be chosen such that the diode remains reverse biased for the maximum intended light intensity I_{max}.

For any practical photodiode, the actual characteristic curves have small curved sections near $v_d = 0$. Furthermore, the slope of $i(v_d)$ for constant intensity ($I = \text{const}$) is small but nonzero. However, in the normal use of a photodiode as an optical sensor, these nonideal characteristics have a negligible effect compared with the idealized model for photodiode operator as a light sensor.

LIGHT GENERATING DIODE

In addition to the light-sensing properties of p-n diodes, they can also be fabricated in such a way as to generate light. Such diodes are known as light-emitting diodes LEDs and are extremely important in vehicular electronic applications. However, since LEDs are used as display components, the theory of the operation is deferred to Chapter 8, which covers vehicular instrumentation. It is explained in that chapter that LEDs are sometimes used in the display portion of vehicle instrumentation, although they have many other applications as well that are covered elsewhere.

LASER DIODE

There is another semiconductor diode that can generate light, which is called a laser diode. The term laser is an acronym for "light amplification by stimulated electromagnetic radiation." A solid-state laser consists of a material with atoms that have conduction-band electrons in an unstable or metastable state. An incident photon having a frequency ν corresponding to the band-gap energy (E_g) interacts with this electron and triggers a transition to the valence band creating another photon at the same frequency, in phase with and propagating in the same direction as the incident photon. The emission of this second photon is termed "stimulated emission." The process continues as the light traverses the material effectively amplifying the light at frequency ν. The energy/frequency relationship is given by

$$E_g = h\nu$$

where $h = $ planks constant, $\nu =$ optical frequency, and $E_g = $ band-gap energy.

In order for the stimulated emission and resulting amplification to continue the optical path must consist of an optically resonant cavity having a resonant frequency at the frequency of the electron energy change ν. Such a cavity typically consists of a pair of parallel partially reflecting mirror surfaces at the boundary of the laser material.

A somewhat simplified illustrative structure for a laser diode is depicted in Fig. 2.7.

The laser diode is fabricated as a PIN diode in which the p and n regions are separated by an intrinsic depletion region. The semiconductor material from which the PIN diode is fabricated is of a type known as "direct-band-gap" material (e.g., GaAs). The electrons in one atom are adjacent to the holes in the next atom that occurs in such materials as GaAs. The parallel mirror surfaces also act as electrodes for connecting the PIN structure to the electric circuit as represented by voltage source v. The current i_d, which flows with this forward-biased diode, injects holes from the p region and electrons from the n region into the depletion (i) region, thereby maintaining a supply of metastable charge carriers. The mirror surfaces reflect some of the light incident on them from within the material, thereby creating standing waves within the structure. The mirror surface on the side from which the laser light is emitted is only partially reflecting such that a portion of the internal wave traveling toward that mirror leaves the structure.

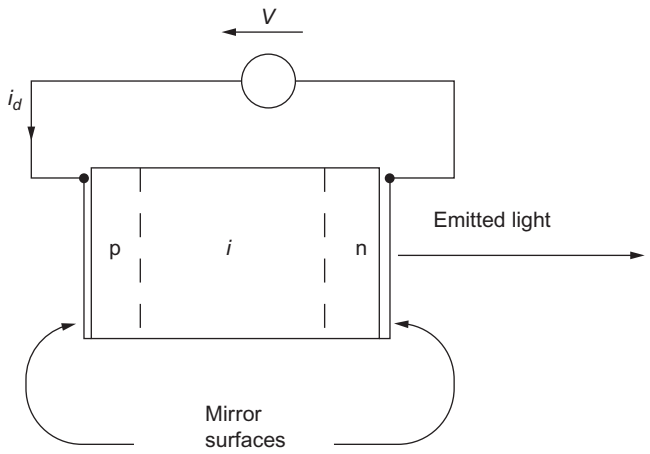


FIG. 2.7 Laser diode structure.

Laser light is unique in that it is both temporally and spatially coherent at essentially a single frequency. It propagates in a relatively narrow beam when leaving the laser diode. The significance of laser light temporal coherence is the ability to measure the phase difference between a pair of wave components. In the correct environment, this means that a laser can be phase modulated and that the Doppler shift in frequency of a reflected wave from a moving object can be used to measure the relative velocity between the reflecting object and the laser source.

There are many applications in vehicular safety-related systems that incorporate laser diodes. For example, single or multiple laser diodes can provide measurements in the region surrounding a vehicle that can detect its environment. Such systems are explained in Chapter 5, which covers vehicle sensors and actuators. The applications of laser diode-based sensors are described in Chapter 10, which discusses vehicular safety-related systems and in Chapter 12, which discusses various levels of autonomous vehicles. In addition, owing to the temporal coherence, lasers are incorporated in optical carrier frequency communication systems.

With the preceding background on p-n junction diodes, we consider next some common circuit applications. We begin with a so-called rectifier circuit.

RECTIFIER CIRCUIT

The circuit in Fig. 2.8, a very common diode circuit, is called a half-wave rectifier circuit because it effectively cuts the AC (alternating current) waveform in half in the sense that the diode passes the positive portion of the cycle and blocks the negative portion of the cycle.

Consider the circuit first without the dotted-in capacitor. The alternating current voltage source is assumed to be a sine wave with a peak-to-peak amplitude of 100 V (50 V positive swing and 50 V negative swing). Waveforms of the input voltage and output voltage plotted against time are shown as the solid lines in Fig. 2.9. Notice that the output never drops below 0 V. The diode is reverse biased

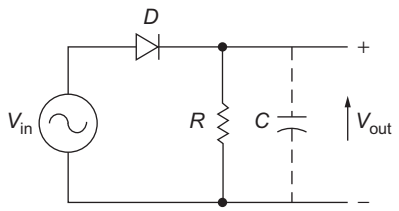


FIG. 2.8 Rectifier circuit.

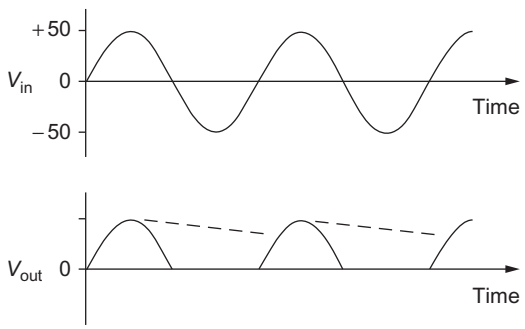


FIG. 2.9 Rectifier waveform.

and blocks current flow when the input voltage is negative, but when the input voltage is positive, the diode is forward biased and permits current flow. If the diode direction is reversed in the circuit, current flow will be permitted when the input voltage is negative and blocked when the input voltage is positive. Rectifier circuits are commonly used to convert the AC voltage into a DC voltage (e.g., for use with automotive alternators to provide DC current battery charging and to supply electric power to the vehicle). Using a capacitor to store charge and resist voltage changes smoothes the rippling or pulsating output of a half-wave rectifier.

The input voltage V_{in} of Fig. 2.8 is AC; the output voltage V_{out} has a DC component and a time-varying component, as shown in Fig. 2.9.

The output voltage of the half-wave rectifier can be smoothed by adding a capacitor, which is represented by the dashed lines in Fig. 2.9. The combination of the load resistance (R) and the capacitor (C) forms a low-pass filter (LPF), which acts to smooth the fluctuating output of the half-wave rectifier diode. Since the capacitor stores a charge and opposes voltage changes, it discharges (supplies current) to the load resistance R when V_{in} is going negative from its peak voltage. The capacitor is recharged when V_{in} comes back to its positive peak and current is supplied to the load by the V_{in}. The result is V_{out} that is more nearly a smooth, steady dc voltage, as shown by the dashed lines between the peaks of Fig. 2.9. The amplitude of the ripples in the output voltage can be made insignificant by choosing a capacitor having sufficiently large capacitance, which lowers the LPF corner frequency and attenuates the ripple components (see Appendix A).

COMMUNICATIONS APPLICATIONS OF DIODES

There are many diode applications including some in communication systems. Often, it is desirable to change the carrier frequency of an information-carrying signal. The highly nonlinear transfer characteristics of a diode make it an excellent component for a process known as "frequency mixing." Whenever two or more signals are passed through a diode, a signal is generated that includes components whose frequencies are the sum and the difference of the two original signal frequencies. Fig. 2.10 depicts a simplified embodiment of the frequency-mixing concept.

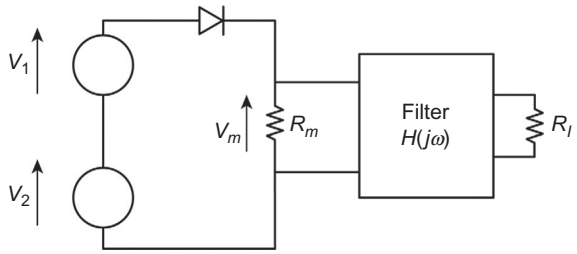


FIG.2.10 Frequecy "mixing" circuit.

Let the two voltage sources have terminal voltages v_1 and v_2, where

$$v_1(t) = V_1 \sin(\omega_1 t)$$
$$v_2(t) = V_2 \sin(\omega_2 t)$$

It can be shown that the voltage across R_m is given by

$$v_m = \sum_{n=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} C_{m,n}(V_1, V_2) \sin[(m\omega_1 + n\omega_2)t] \tag{2.6}$$

where the coefficients $C_{m,n}$ are functions of V_1,V_2 and R_m and m, n are integers. The amplitudes of these coefficients are largest for relatively small n and m and asymptotically approach 0 as $n \to \pm\infty$ and $m \to \pm\infty$. In most communications, application frequency-mixing circuits are used to select the desired frequency components. The filter pass band encloses the desired frequency component, and its stop bands reject the unwanted frequency components (see Appendix A).

TRANSISTORS

Diodes are static circuit elements; that is, they do not have gain or store energy. Transistors are active elements because they can amplify or transform a signal level. Transistors are three-terminal circuit elements that act like voltage- or current-controlled current amplifiers. Transistors come in two major categories that are termed "bipolar" or "field effect" depending upon whether they are current or voltage controlled, respectively. There are two common bipolar (i.e., consisting of n-type and p-type semiconductors) types denoted as (1) NPN and (2) PNP.

Physically, an NPN transistor structure consists of a thin p-type material (called *Base*) sandwiched between two n-type pieces, which are called the *Collector* and the *Emitter*. A PNP transistor has a thin n-type material as the *Base* between a p-type *Emitter* and p-type *Collector*.

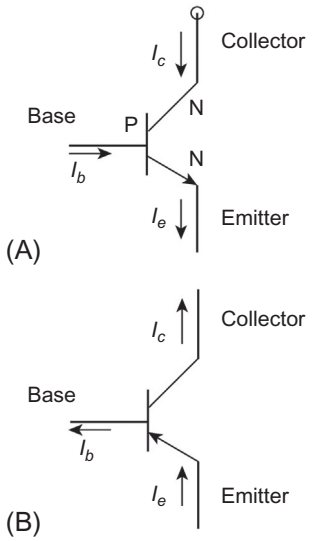


FIG. 2.11 Transistor schematic symbols. (A) NPN transistor schematic symbol, (B) PNP transistor schematic symbol.

Bipolar transistors can be made to amplify or switch in three different circuit configurations: (1) grounded emitter, (2) grounded base, and (3) grounded collector. For the present discussion, we will use Fig. 2.11A and B to depict the schematic symbols for both bipolar types. The direction of conventional current flow for each of the terminals for collector (I_c), emitter (I_e), and base (I_b) is shown in these schematic symbol drawings. Fig. 2.12 depicts a circuit configuration for a grounded-emitter NPN amplifier whose theory of operation is explained later in this chapter.

The signal being amplified is represented by source voltage v_s and source resistance R_s. The load resistance R_ℓ connects the collector to the positive DC power supply voltage V_{cc}. The output, amplified

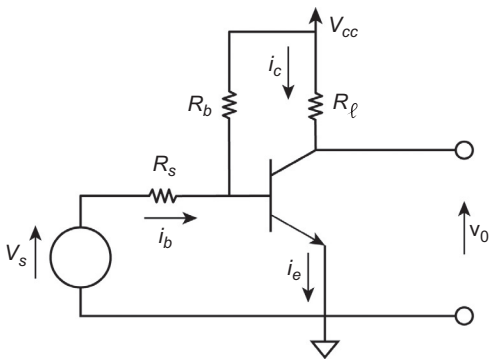


FIG. 2.12 NPN transistor amplifier circuit.

signal is taken at the collector-R_ℓ junction and is denoted as v_o. For linear amplification, a bias resistance R_b supplies a DC current to the base. The purpose of the bias current is explained later with respect to the operation of a transistor as an amplifier.

Transistors are useful as amplifying devices. During normal operation, current flows from the base to the emitter in an NPN transistor. The collector-base junction is reverse biased, so that only a very small amount of current flows between the collector and the base when there is no base current flow.

The base-emitter junction of a transistor acts like a diode. Under normal operation for an NPN transistor, current flows forward into the base and out the emitter, but does not flow in the reverse direction from emitter to base. The arrow on the emitter of the transistor schematic symbol indicates the forward direction of current flow. The collector-base junction also acts as a diode, but supply voltage is always applied to it in the reverse direction. This junction does have some reverse current flow, but it is so small (10^{-6}–10^{-12} amp) that it is ignored except when operated under extreme conditions, particularly temperature extremes. In some automotive applications, the extreme temperatures may significantly affect transistor operation. For such applications, the circuit may include components that automatically compensate for changes in transistor operation.

The operation of an NPN bipolar junction transistor (BJT) in grounded-emitter configuration can be understood with reference to Fig. 2.13. In this configuration, the individual component regions—collector, base, and emitter—are depicted along with the very important depletion regions in each at both junctions.

Voltages V_{ce} and V_{be} are the voltages of the collector (+) and the base (+) relative to emitter (ground), where $V_{ce} > V_{be}$. These voltages reverse bias the collector-base junction and forward bias the base-emitter junction. The electrically neutral portions of the collector and emitter are denoted as n, and the neutral portion of the base is denoted as p. The reverse-bias collector-base voltage is denoted as V_{cb} and is given by

$$V_{cb} = V_{ce} - V_{be}$$

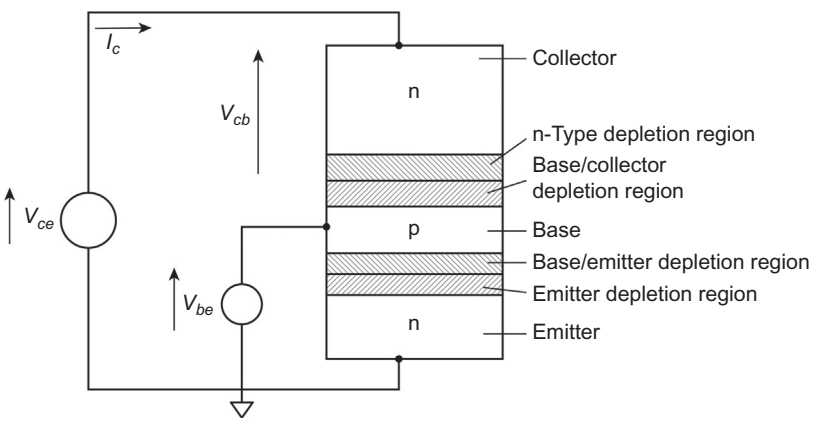


FIG. 2.13 NPN grounded-emitter configuration and voltages.

An increase in V_{ce} (without changing V_{be}) increases the reverse bias that increases the depletion region in both the collector and base. There is no change in the depletion region of the emitter-base junction (for fixed V_{bc}). Consequently, the neutral base region is decreased in size along the active path between collector and emitter.

The narrowing of the base reduces the probability of any recombination of a charge carrier with an impurity ion. In addition, the gradient of the charge density across the base is increased such that the current, due to minority carriers injected across the base-emitter junction, increases. The result of these effects is to increase the collector (i.e., output) current as V_{ce} is increased.

The operating characteristics for a bipolar transistor are given as a set of curves of $I_c(V_{ce}$ and $V_{be})$ known as characteristic curves. The characteristic curves for a typical small-signal NPN transistor (i.e., 2N4401) in the grounded-emitter configuration are given in Fig. 2.14.

These curves are parameterized in terms of base current (in μ amp). Note that each curve for a fixed V_{be} increases from $I_c = 0$ at $V_{ce} = 0$, reaching a saturation value and beyond this point increases roughly linearly, with V_{ce}. A large signal model for the grounded-emitter bipolar transistor is given by the so-called Early model:

$$I_c = I_s \exp\left(\frac{V_{be}}{V_T}\right)\left(1 + \frac{V_{ce}}{V_A}\right) \tag{2.7}$$

$$\beta_F = \left.\frac{\partial I_c}{\partial I_b}\right|_{V_{ce}}$$

$$\beta_F = \beta_{F_0}\left(1 + \frac{V_{ce}}{V_A}\right)$$

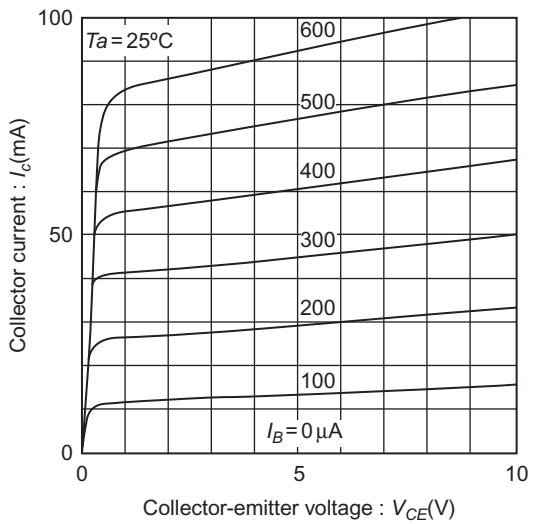


FIG. 2.14 Grounded emitter output characteristics.

where I_s is the saturation current for the reverse-biased collector-base junction, V_T is the thermal voltage $= kT/q$, V_A is the so-called Early voltage (in the approximate range 15–150 V), and β_{F_0} is the forward common-emitter current gain at zero bias:

$$\beta_{F_0} = \left.\frac{\partial I_c}{\partial I_b}\right|_{I_b=0}$$

A small-signal linear model for the transistor is given in Fig. 2.15. This model is the most useful for performance analysis/design of linear modes of operation. In this model, the collector current is modeled by a current-controlled current source ($h_{fe}I_b$) shunted by a resistance R (source impedance). The base current I_b is determined by the source being amplified along with external circuit impedances.

The base-emitter diode does not conduct (there is no transistor base current) until the voltage across it exceeds V_d volts in the forward direction. If the transistor is a silicon transistor, V_d equals 0.7 V just as with the silicon diode. The collector current I_c is zero until the base-emitter voltage V_{be} exceeds 0.7 V. This is called the cutoff condition or the off condition, when the transistor is used as a switch.

When V_{be} rises above 0.7 V, the diode conducts and allows some base current I_b to flow. Fig. 2.14 shows that the transistor voltage/current characteristics, though basically nonlinear, have a linear region of operation. A variation in base current about a point such as $I_o = 300$ μA at $V_{ce} = 5$ V produces a variation in collector current that is highly linear. The so-called common-emitter forward current gain, denoted as h_{fe}, is given by

$$h_{fe} = \left.\frac{\partial I_c}{\partial I_b}\right|_{V_{be}} \tag{2.8}$$

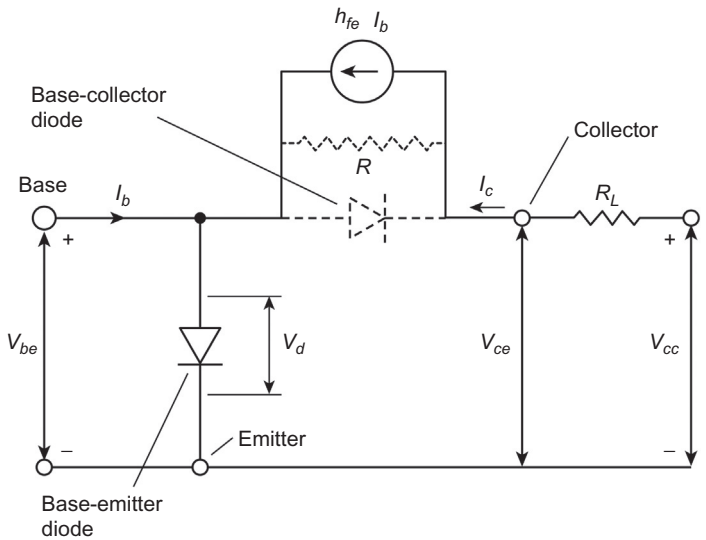


FIG. 2.15 Current and voltages for NPN transistor.

It is common practice in linear transistor circuit analysis/design to denote d-c components of voltage/current with uppercase letters and variations about these d-c values with lowercase letters. For example, collector current can be modeled as

$$\begin{aligned} I_c(t) &= I_{co} + \delta I_c \\ &= I_{co} + i_c(t) \end{aligned} \tag{2.9}$$

Similarly, the base current I_b can be modeled as given below:

$$\begin{aligned} I_b(t) &= I_o + \delta I_b \\ &= I_o + i_b(t) \end{aligned} \tag{2.10}$$

At any d-c base current (that is called the base bias current) and is denoted as I_o. In Fig. 2.15, the collector a-c current i_c is given by

$$i_c(t) = h_{fe}\, i_b(t) \tag{2.11}$$

The current gain (h_{fe}) can range from 10 to 200 depending on the transistor type but is nearly constant over a large range of V_{ce}, I_b, and I_c. The collector current is represented by a current generator in the collector circuit of the model in Fig. 2.15. This condition is called the active region because the transistor is conducting current and amplifying. It is also called the linear region because collector current is (approximately) linearly proportional to base current. The dotted resistance in parallel with the collector-base diode represents the leakage of the reverse-biased junction, which is normally neglected, as discussed previously.

A third condition, known as the saturation condition, exists under certain conditions of collector-emitter voltage and collector current. In the saturation condition, large increases in the transistor base current produce little increase in collector current. When saturated, the voltage drop across the collector-emitter is very small, usually less than 0.5 V. This is the "on" condition for a transistor switching circuit. This condition occurs in a switching circuit when the collector of the transistor is tied through a resistor R_L to a supply voltage V_{cc} as shown in Fig. 2.15. In this mode of operation, the source voltage is large enough that the base current drives the transistor into the saturated condition, in which the output voltage (voltage drop from collector to emitter) is very small and the collector-base diode may become forward biased.

Having briefly described the behavior of transistors, it is now possible to discuss circuit applications for them. As an example of the use of this small-signal model, consider the analysis of the simple amplifier circuit of Fig. 2.16A. In this figure, a signal is represented by the a-c voltage v_s and source resistance R_s, and capacitor C is amplified to an output signal v_o. The purpose of adding the capacitor is to block the d-c base current I_o through the bias resistor R_b from flowing through the source. This output voltage is produced by collector variation (due to source voltage variations) acting through load resistance R_L.

In a transistor amplifier, a small change in base current results in a corresponding larger change in collector current. In order to achieve linear amplification, the transistor is biased with a d-c current I_o via bias resistor R_b. The characteristic curves for this transistor are shown in Fig. 2.16B. The straight line (called the load line) connecting $V_{ce} = V_{cc}$ (10 V) with $I_c = I_{cs}$ represents the variation in voltage V_o and collector current I_c with variation in base current due to signal voltage V_s. The slope of the load line S_{LL} is given by

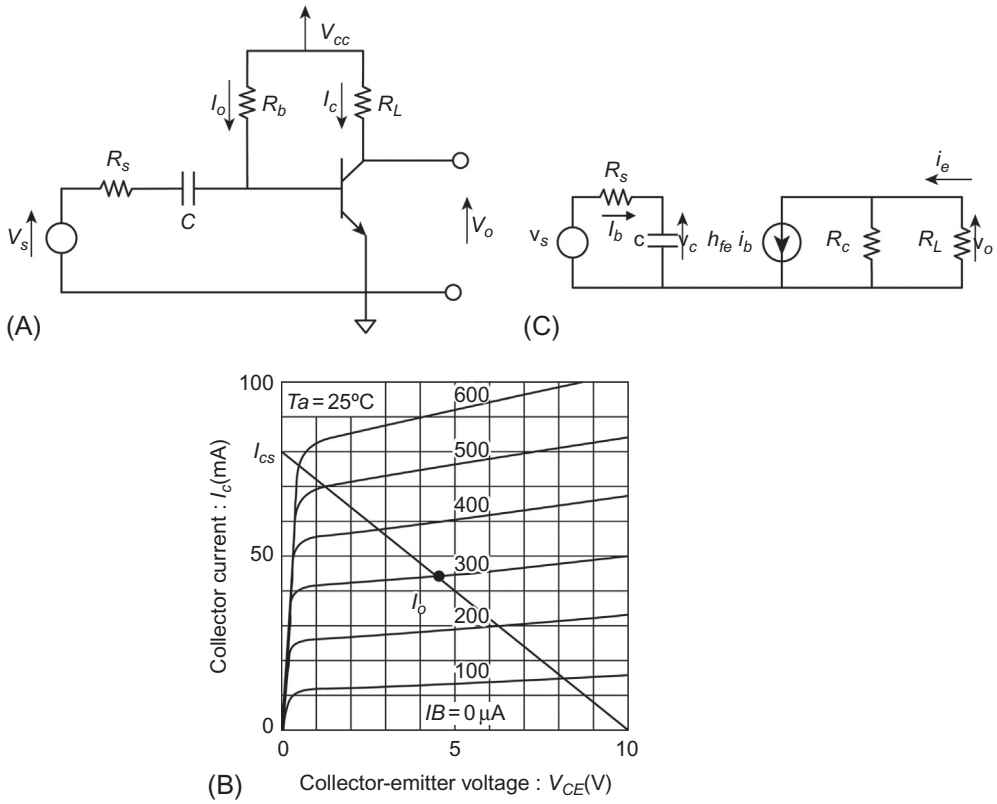


FIG. 2.16 Grounded emitter NPN transistor amplifier.

$$S_{LL} = -\frac{dI_c}{dV_{ce}} = \frac{I_{sc}}{V_{cc}} = \frac{1}{R_L} \tag{2.12}$$

With $v_s = 0$, the bias current is given by

$$I_0 = \frac{V_{cc} - V_d}{R_b} \cong \frac{V_{cc}}{R_b} \tag{2.13}$$

where V_d is the forward-bias voltage drop across the base-emitter junction, which is typically negligible in comparison with V_{cc}. The analysis of this circuit is done using the small-signal model of Fig. 2.16C in

which the load resistance at terminal V_{cc} is at a-c ground potential. The output voltage v_o of circuit in Fig. 2.16A is the a-c component of V_{ce}.

This model, which is often termed the "small-signal linear incremental transistor model," is actually an idealized (fictional) equivalent circuit that is only valid for linear amplification and represents only the time-varying (i.e., a-c) components of voltages and currents. In this model, the collector current i_c is represented by a current-controlled ideal current source shunted by a source resistance R_c. An ideal current source is an artificial circuit component that produces a current that is independent of any load impedance. The current generated is proportional to base current i_b and is given by

$$i_c = h_{fe} i_b \tag{2.14}$$

Assuming $R_c \gg R_L$ (which is the usual case), the output voltage v_o is given by

$$v_o(t) = -R_L i_c(t) \tag{2.15}$$

$$v_o = -h_{fe} R_L i_b(t) \tag{2.16}$$

Note that the collector voltage and base current are 180 degrees out of phase. This phase change occurs because load resistance end that is physically connected to the power supply (i.e., V_{cc}) is at a-c ground potential and the a-c collector current flows in the direction shown in Fig. 2.16C. The base circuit analysis is conducted by summing the voltage components around the base circuit loop:

$$v_s = i_b R_s + v_c \tag{2.17}$$

The capacitor voltage v_c is given by

$$v_c(t) = \frac{1}{C}\int_o^t i_b(\tau)d\tau \tag{2.18}$$

Eq. (2.17) can be solved for base current i_b by using the Laplace transform method of Appendix A yielding the following Equation for $i_b(s)$:

$$\begin{aligned} i_b(s) &= \frac{v_s(s)}{R_s + \dfrac{1}{sC}} \\ &= \frac{sCv_s(s)}{1 + R_s C s} \end{aligned} \tag{2.19}$$

Substituting $i_b(s)$ from Eq. (2.19) into Eq. (2.16) yields the transistor circuit voltage gain G:

$$G(s) = \frac{v_o(s)}{v_s(s)} \tag{2.20}$$

$$= -\frac{h_{fe} sCR_L}{(1 + sCR_s)}$$

$$= -h_{fe}\frac{R_L}{R_s}\left(\frac{s/\omega_o}{1 + s/\omega_o}\right) \tag{2.21}$$

where $\omega_o = \dfrac{1}{R_s C}$.

The dimension of the product $R_s C$ is time such that ω_o has the dimension of frequency (in rad/s).

The variation of gain with input frequency can be determined from the steady-state sinusoidal frequency response for the gain ($G(j\omega)$). In Appendix A, it was shown that the sinusoidal frequency response of any system is found by replacing s with $j\omega$ in the operational transfer function. Thus, the frequency dependence of amplifier gain is given by

$$G(j\omega) = -h_{fe}\frac{R_L}{R_s}\left(\frac{j\omega/\omega_o}{1+j\omega/\omega_o}\right) \tag{2.22}$$

For frequencies $\omega \gg \omega_o$, the amplifier gain approaches a constant value:

$$G(j\omega) \xrightarrow[\omega\to\infty]{} -h_{fe}\frac{R_L}{R_s} \tag{2.23}$$

That is, the circuit of Fig. 2.16A is a "high-pass" amplifier. The d-c blocking capacitor C is chosen during circuit design such that ω_o is smaller than the lowest component in v_s.

In practice, transistor amplifiers frequently consist of multiple stages of the form of Fig. 2.16A connected in cascade, each with a capacitor coupling its output to the input of the next stage. Of course, the time domain output can be found by taking the inverse Laplace transform of $v_o(s)$ as shown in Appendix A.

In addition to the linear region of operation, a transistor can be made to operate nonlinearly as a switch as explained above with respect to saturation and cutoff regions. For this application, the bias resistor is omitted. Circuit parameters and input voltages are chosen such that the transistor switches abruptly from cutoff in which $I_c \cong 0$ and saturation in which $I_c = I_{csat}$. This type of operation is used in digital circuits, which are discussed later in this chapter. As will be shown later, both input and output voltages are binary-valued.

FIELD-EFFECT TRANSISTORS

The types of transistors discussed above are known as bipolar transistors because they operate by conduction via both electrons and holes. As explained earlier, they amplify relatively weak base currents yielding relatively large collector-emitter output currents. They are in effect current-controlled current amplifiers. Another type of transistor operates as a voltage-controlled amplifier and is called a field-effect transistor (FET). There are many variations of FETs, as explained below.

Unlike the bipolar transistor, which is fabricated with two p-n junctions, the FET consists of a slab of either n-type or p-type semiconductor to which electrodes are bonded as depicted (in an introductory manner) in Fig. 2.17. An FET is known as either a p-channel or an n-channel FET depending upon whether the semiconductor substrate is n-type or p-type material, respectively, as explained later in this chapter.

An FET is a three-terminal active circuit element having a pair of electrodes connected at opposite ends of the slab of semiconductor and called source (denoted as S) and drain (denoted as D). A third electrode, called the gate (denoted as G), consists of a thin layer of conductor that is electrically insulated from the semiconductor slab.

There are many types of FETs characterized by fabrication technology, material doping (i.e., n-channel or p-channel), and whether the gate voltage tends to increase the number of charge carriers (called enhancement mode) or decrease the number of charge carriers (depletion mode). The FET

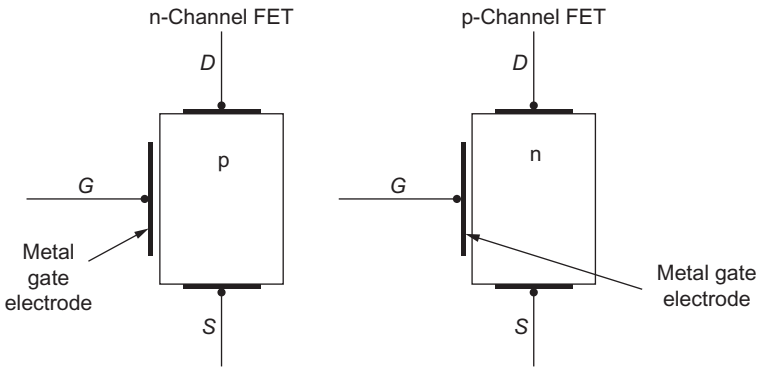


FIG. 2.17 Field-effect transistor configuration.

circuit schematic symbols for n-channel and p-channel enhancement modes are shown in Fig. 2.18B and for depletion mode in Fig. 2.18A.

The circuit symbols are drawn with a solid line from source to drain for a depletion mode FET and with three segments in a line from source to drain as depicted in Fig. 2.18 for an enhancement mode FET. The arrow is directed away from the middle SD line segment for p-channel FET and toward the line segment for n-channel FET. It is left as an exercise for the interested reader to draw the circuit symbols for an n-channel depletion FET and a p-channel enhancement FET.

The fabrication of the FET is accomplished by doping the substrate material near the S and D electrodes with charge carrier distributions as explained in the next section of this chapter on FET theory. The electrodes are formed such that the S and D electrodes are in ohmic contact, and the gate is insulated from the substrate.

Perhaps the most common gate fabrication involves a metal with an oxide layer placed against the semiconductor in the sequence metal-oxide semiconductor (MOS). The oxide layer insulates the metal electrode from the semiconductor so that no current flows through the gate electrode. Rather, the voltage applied to the gate creates an electric field that controls current flow from source to drain.

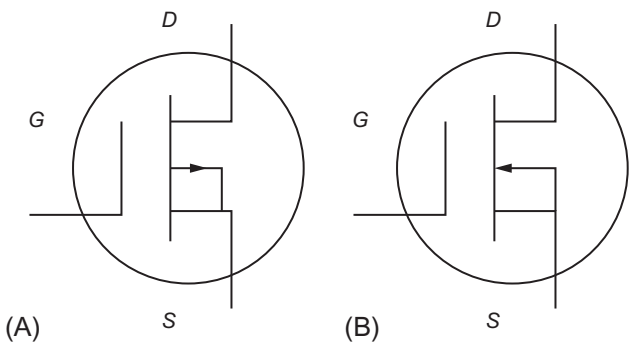


FIG. 2.18 Circuit symbol for FET transistors. (A) P-channel depletion and (B) N-channel enhancement.

The terminology for an FET having this type of gate structure is NMOS or PMOS, depending on whether the FET is n-channel or p-channel. Often, circuits are fabricated using both in a complementary manner, and the fabrication technology is known as complementary-metal-oxide semiconductor (CMOS)

FET THEORY

The theory of operation of an FET depends on its physical construction and impurity doping configurations. The conduction occurs within the so-called channel that, for an n-channel, is via negative charge carriers (electrons), and for a p-channel, conduction is via holes. For an enhancement-type FET, the size of the channel and thus the current flow is determined by the strength of an electric field intensity created by the voltage on the gate (hence the name field-effect transition). For the purposes of the present discussion, we take a very simplified explanation of electric field intensity that is a vector quantity and is denoted as $\bar{E}$ with the over bar used to denote a vector in field theory discussions. A more detailed and exact discussion of general electromagnetic field theory is presented in Chapter 5.

For the present explanation of FET theory, a relatively simple configuration of a structure in which $\bar{E}$ exists is depicted in Fig. 2.19.

This structure consists of a pair of parallel plate electrodes separated by an insulator to which a voltage V is applied. The electric properties of the insulator are characterized by a material properly known as the dielectric constant and denoted as $\in$. For a vacuum, the dielectric constant is denoted as $\in_o$, and for any homogeneous isotropic material, $\in$ is given by

$$\in = \in_r \in_o$$

where $\in_r$ = dimensionless relative dielectric constant.

For the structure of Fig. 2.19, we assume that $\in_r \gg 1$. In this case, the electric field intensity is nearly uniform in the space between the electrodes (i.e., $0 \leq y \leq w$) and is given by

$$\bar{E} = -\frac{\mathrm{v}}{d}\hat{x}$$

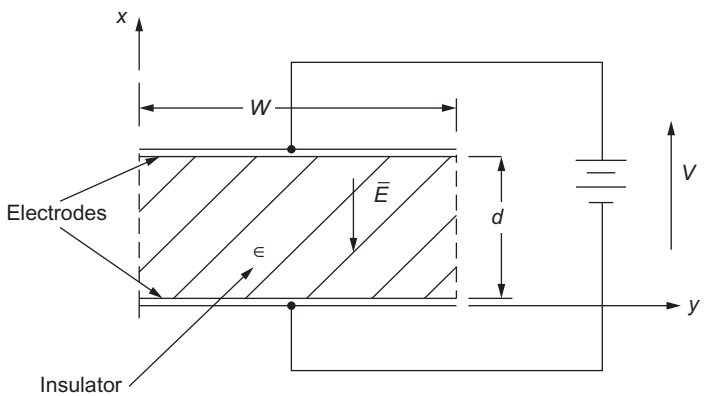


FIG. 2.19 Structure in which $\bar{E}$ exists.

where $\hat{x}$ = unit vector in the x direction.

The equation for $\bar{E}$ indicates that the direction of $\bar{E}$ is from the positive toward the negative electrode and has magnitude proportional to V and inversely proportional to the interelectrode spacing. This somewhat simplified model for $\bar{E}$ is sufficiently valid for our discussion of FETs.

Any electric charges exposed to $\bar{E}$ experience a force $\bar{F}$, which is given by

$$\bar{F} = q\bar{E}$$

where q = charge magnitude.

In a semiconductor, the force on electrons is toward the positive electrode, and on a hole, it is away from the + electrode.

The modeling of the electric conduction characteristics of an FET depends upon its configuration. For an n-channel enhancement mode FET, a somewhat simplified configuration is depicted in Fig. 2.20.

This type of FET is fabricated on a block of p-type semiconductor (called substrate) with of relatively heavily doped n-type (denoted n+) sections near the S and D electrodes. The gate electrode is electrically insulated from the semiconductor (e.g., with an oxide layer). The n-doped region and p substrate form p-n junctions.

The voltage between the gate and source creates the electric field intensity $\bar{E}_g$ depicted in Fig. 20, which has a magnitude ($\|\bar{E}_g\|$) proportional to V_{GS}. For $V_{GS} = 0$, there is no n-channel. For V_{GS} greater than a specific value called the threshold voltage that is denoted V_{th}, there is an n-channel created between S and D. This n-channel exists because $\bar{E}_g$ moves the p charge carriers away from the gate area. The width of the channel is proportional to $\bar{E}_g$ and thus to V_{GS}. The drain current i_D for an applied DS voltage denoted V_{DS} results from current flow through n-channel and is related to V_{GS} and is represented by the FET transistor characteristic curves (analogous to bipolar transistors). For the purposes

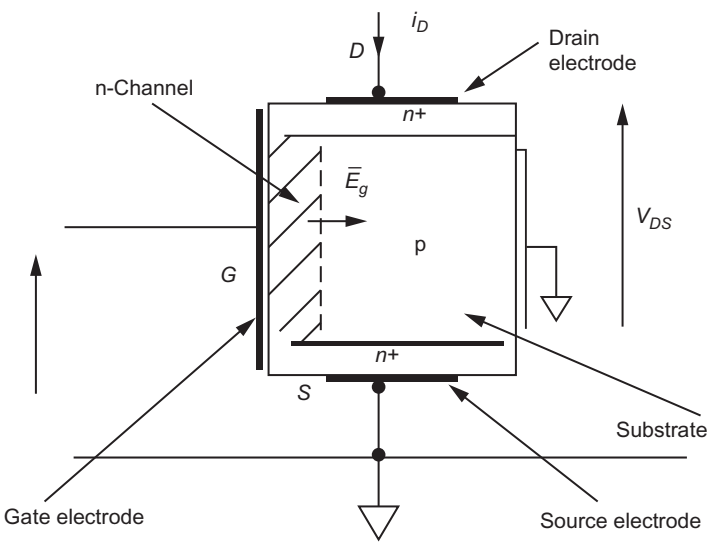


FIG. 2.20 N-channel enhancement FET.

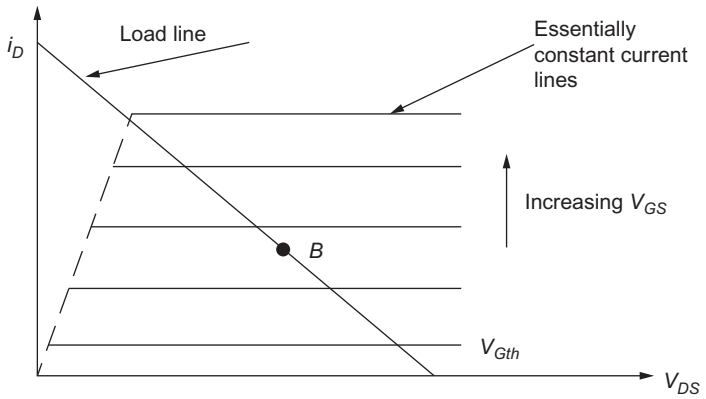


FIG. 2.21 Approximate (idealized) characteristic curves for n-channel enhancement FET.

of deriving a relatively straightforward model for the FET in a circuit, these characteristic curves are approximated by straight line segments as depicted in Fig. 2.21.

In this figure, each of the solid lines represents an essentially constant current. That is, for each line,

$$\left.\frac{\partial i_D}{\partial V_{DS}}\right| = \frac{1}{R_{DS}} \simeq 0$$

The dashed line represents $i_D(V_{DS})$ for very small V_{DS} that is a portion of the FET operating domain that is only used for switch-mode operation (as explained later in this chapter). The actual characteristic of $i_D(V_{DS})$ for a given V_{GS} in this region is a small curve connecting the dashed line with the constant current line for that specific V_{GS}. The parameter R_{DS} (which essentially is ∞) will appear in the equivalent circuit model for the FET in a way that normally makes it unnecessary in circuit modeling. Essentially, these characteristic curves show that the FET functions as a voltage-controlled current source for linear circuit models. As will be shown later in switching type circuits, the S to D path functions essentially as a voltage-controlled (i.e., by V_{GS}) resistance. The solid line drawn between the V_{DS} and i_D axes in Fig. 2.21 is a so-called load line that is associated with the amplifier circuit and is discussed in association with that circuit.

A p-channel enhancement FET has a structure similar to that shown in Fig. 2.20 except that all p regions become n regions and n regions become p. In addition, the voltage polarities of the p-channel enhancement mode are reversed from those of the n-channel. Although the characteristic curves for the p-channel enhancement FET are similar in shape to those of the n-channel, the actual drain current $i_D(V_{GS})$ is generally lower for a given magnitude of V_{DS} than for n-channel, because the charge carriers are holes that have less mobility than the electrons in an n-channel FET.

Depletion mode FETs are fabricated differently than enhancement types in that there is an existing channel between S and D in the absences of $\bar{E}_g$.This channel is created during fabrication by doping the substrate material near the gate surface. For this reason, the depletion FETs are conducting and have a nonzero i_D for $V_{GS}=0$. The characteristic curves for depletion type FETs are similar to those depicted in Fig. 2.21, except that the constant current line for $V_{GS}=0$ is well above that for $V_{GS}=V_{th}$ in an

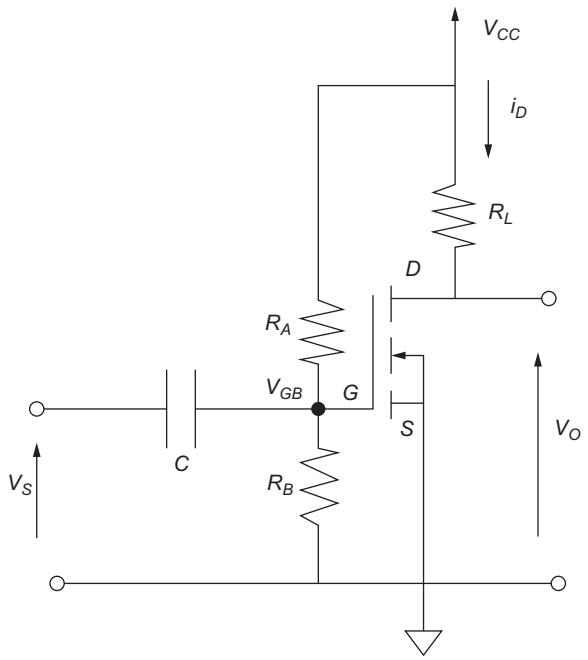


FIG. 2.22 Simple FET amplifier.

enhancement-type FET. In fact, the threshold voltage for a depletion-type FET is negative. An amplifier stage with a depletion-type FET can linearly amplify a-c voltages without requiring a bias voltage that, as will be shown next, is required for an enhancement-made FET linear amplifier.

The analysis procedure for all FET-type transistors is essentially the same as for a bipolar transistor. An example-amplifying circuit, shown in Fig. 2.22, depicts the current path from power supply V_{CC} through a load resistor R_L and through the transistor from D to S and then to ground using an n-channel enhancement mode FET. This example circuit configuration is termed a "grounded source amplifier." A signal voltage v_s applied at the gate electrode controls the current flow through the FET and thereby through the load resistance R_L. Functionally, the FET operates like a voltage-controlled current source. A relatively weak signal applied to the gate can yield a relatively large voltage v_o across the load resistance.

FET AMPLIFIER

The circuit of Fig. 2.22 operates along the solid line (called load line) connecting the V_{DS} and i_D axes in Fig. 2.21. The slope of this line S_L is given by

$$S_L = \left.\frac{\partial i_D}{\partial V_{DS}}\right|_{\text{loadline}}$$

$$= -\frac{1}{R_L}$$

For a linear amplifier, the V_{GS} voltage must remain between the threshold voltages that is denoted V_{Gth} in Fig. 2.21 and the point of intersection with the dashed line. If the voltage being amplified is a-c (i.e., its average voltage is 0), a bias voltage must be applied at the gate at a point (denoted B in Fig. 2.21) that will keep the V_{GS} within the linear range. The gate bias voltage (corresponding to point B of Fig. 2.21) is denoted V_{GB}. This bias voltage is accomplished with the voltage divider circuit consisting of resistors R_A and R_B. The capacitor that is denoted C in Fig. 2.22 provides an open circuit to the bias circuit. In addition, the resistance to ground from the gate (G) terminal is sufficiently large that its affect on V_{GB} is negligible. This voltage is given by

$$V_{GB} = \frac{V_{CC}R_B}{R_A + R_B}$$

The parameters R_A, R_B, and C can be chosen in relationship to the source impedance of the a-c signal being amplified that the bias circuit impedance has a negligible effect on the circuit amplification. For example, the series reactance X_C of the capacitor is given by

$$X_C = \frac{1}{\omega_{\min} C} \ll R_s$$

where $\omega_{\min}$ is the lowest frequency component in V_S and R_s = source resistance.

In addition, the combined load resistance presented to the source due to the bias resistors that is denoted R_P can be large compared with R_S. This combined load resistance on the source is given by

$$R_P = \frac{R_A R_B}{R_A + R_B}$$

By choosing R_A and R_B to be sufficiently large, the following inequality can readily be achieved:

$$R_P \gg R_S$$

It is assumed that the bias circuit satisfies these above inequalities such that the bias circuit has essentially no effect on the performance of the amplifier of Fig. 2.22.

The characteristic curves of Fig. 2.21 ideally should have zero slope. In other words, the drain current should be independent of V_{GS} and depend only on V_G. However, in practice, there is a small but nonzero slope S_{DS} to each constant V_G curve that is given by

$$S_{DS} = \left.\frac{\partial i_D}{\partial V_{DS}}\right|_{V_G}$$

The influence of this slope has an FET amplifier that can be represented in an equivalent circuit for the FET output circuit is by a resistance R_{DS}, where R_{DS} is given by

$$R_{DS} = \frac{1}{S_{DS}}$$

We illustrate such analysis with an example based upon Fig. 2.22. The small-signal (a-c) equivalent circuit model for this circuit is shown in Fig. 2.23.

The input impedance for an FET is sufficiently large, and the bias circuit (R_A, R_B, and C) is designed to be negligible such that the gate voltage is approximately the source voltage $v_G \cong v_S$. The equivalent

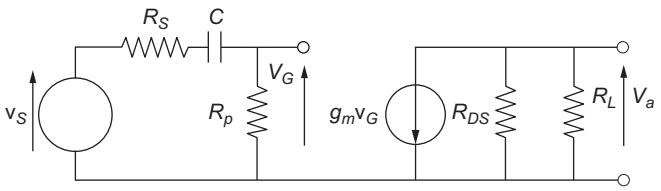


FIG. 2.23 Small-signal circuit model for FET amplifier.

circuit of Fig. 2.23 includes a voltage-controlled current source whose output current is controlled by gate voltage V_G. This current source is shunted by source resistance R_{DS}. In this case, the voltage-controlled current source with current is given by

$$\begin{aligned} i_D &= g_m v_G \\ &\cong g_m v_S \end{aligned}$$

where $g_m = \left.\dfrac{\partial i_D}{\partial v_G}\right|_{v_{GB}}$.

The parameter g_m is called the "transconductance" for the FET. The output voltage v_o is given by

$$\begin{aligned} v_o &= -\frac{R_{DS} R_L i_D}{R_{DS} + R_L} \\ &= -\frac{R_{DS} R_L g_m v_s}{R_{DS} + R_L} \end{aligned} \tag{2.24}$$

In a typical amplifier application, $R_L \ll R_{DS}$ so that v_o is given approximately by

$$V_O \cong -R_L g_m V_s$$

The amplifier gain G is given by

$$\begin{aligned} G &= \frac{v_o}{v_s} \\ &= -R_L g_m \end{aligned}$$

Note that this exemplary grounded source FET amplifier produces a 180 degrees phase shift from v_s to v_o similar to the bipolar grounded-emitter amplifier.

The above example illustrates the analysis procedures for any FET for the assumptions made for the bias circuit. Of course, any given FET amplifier circuit may incorporate frequency-dependent components (e.g., inductors and capacitors), which requires analysis via transform techniques as explained in Appendix A and similar to that used in the bipolar transistor analysis.

INTEGRATED CIRCUITS

In modern automotive electronic systems/subsystems, transistors only seldom appear as individual components (except for relatively high-power applications as drivers for fuel injection or spark generation). Rather, multiple transistors (numbering in the tens of thousands) are created on a single

semiconductor (e.g., Si) chip. This is particularly true of digital circuits that are discussed later in this chapter. These combined circuits are termed integrated circuits and are packaged with dozens or hundreds of leads configured such that they can be attached via soldered connections to a so-called printed circuit board. A printed circuit consists of a thin insulating board onto which conductors are formed that provides the interconnection between multiple integrated circuits to form an electronic system/subsystem.

Analog filtering or other signal processing has largely disappeared from contemporary automobiles. However, some older vehicles may still be on the road in which there is some analog signal processing. Moreover, analog signal processing is sometimes combined with an analog sensor. For the sake of completeness, a brief discussion of analog signal processing is included here. Analog filtering/signal processing is, perhaps, best illustrated with integrated circuits called "operational amplifiers."

OPERATIONAL AMPLIFIERS

An operational amplifier (op-amp) is an example of a standard building block of integrated circuits and has many applications in analog electronic systems. It is normally connected in a circuit with external circuit elements (e.g., resistors and capacitors) that determine its operation. An op-amp is a differential amplifier that typically has a very high-voltage gain of 10,000 or more and has two inputs and one output (with respect to ground), as shown in Fig. 2.24A. A signal applied to the inverting input (–) is amplified and inverted (i.e., has polarity reversal) at the output. A signal applied to the noninverting input (+) is amplified, but it is not inverted at the output.

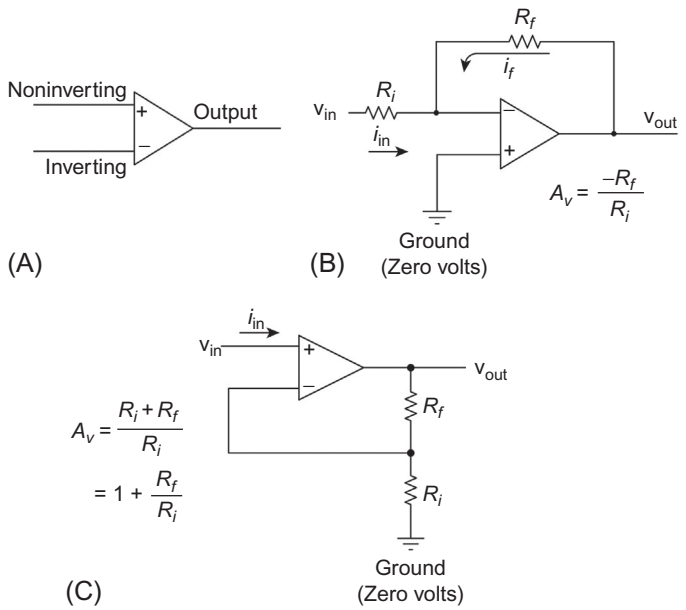


FIG. 2.24 Op-amp circuits. (A) Schematic symbol, (B) inverting amplifier, (C) noninverting amplifier.

USE OF FEEDBACK IN OP-AMPS

The op-amp is normally not operated open loop at maximum gain, but feedback techniques can be used to adjust the closed-loop gain and dynamic response to the value desired, as shown in Fig. 2.24B. The output is connected to the inverting input through circuit elements (resistors, capacitors, etc.) that determine the closed-loop gain.

The output voltage v_{out} for an op-amp having no feedback path (i.e., open loop) is given by

$$v_{out} = A(v_1 - v_2) \tag{2.25}$$

where v_1 is the noninverting input voltage and v_2 is the inverting input voltage.

Alternatively, this equation can be rewritten in a form from which the relationships between the inverting and noninverting inputs can be found for an ideal op-amp (having open-loop gain $A \to \infty$):

$$v_1 - v_2 = \frac{v_{out}}{A} \xrightarrow[A \to \infty]{} 0$$

Thus, the two input voltages will approach identity for a high-quality (i.e., high A) op-amp as represented by the following condition:

$$v_1 \cong v_2$$

The internal resistance between the inverting and noninverting inputs is denoted R_{in} and is relatively large compared with external resistances used in normal up-amp applications. In the example of Fig. 2.24B, the feedback path consists of resistor R_f. The gain is adjusted by the ratio of the two resistors and is calculated by the following analysis. For this circuit configuration, which is inverting mode with noninverting input at ground, the following relationship is valid:

$$v_1 = v_2 = 0$$

In order that $v_1 = 0$, the currents at the inverting input must sum to zero:

$$i_{in} = \frac{v_{in} - v_1}{R_i}$$

$$i_{in} = \frac{v_{in}}{R_i}$$

$$i_f = \frac{v_{out} - v_1}{R_f}$$

$$i_f = \frac{v_{out}}{R_f}$$

$$i_{in} + i_f = 0$$

$$\frac{v_{out}}{R_f} = -\frac{v_{in}}{R_i}$$

The closed-loop gain $A_{c\ell}$ is defined as

$$A_{c\ell} = \frac{v_{out}}{v_{in}} = -\frac{R_f}{R_i} \tag{2.26}$$

The phase change of 180 degrees between v_{in} and v_{out} is indicated by the negative $A_{c\ell}$.

The op-amp can readily be configured to implement a LPF using the circuit depicted in Fig. 2.25.

The components in the feedback path include the parallel combination of resistor R_f and capacitor C. In this circuit, as in any other inverting mode circuit with the noninverting mode grounded (through a low resistance R), the currents into the inverting input sum essentially to zero. Using the Laplace transform methods of Appendix A and the model for capacitor voltage/current relationships (i.e., $i_C = C\dfrac{dv_c}{dt}$), the model for the inverting mode currents is given below:

$$\frac{v_s(s)}{R_i} + v_o(s)\left[sC + \frac{1}{R_f}\right] = 0 \tag{2.27}$$

Solving for v_o/v_s gives the closed-loop gain $A_{c\ell}$ (as a transfer function):

$$\begin{aligned} A_{c\ell}(s) &= \frac{v_o(s)}{v_s(s)} \\ &= -\frac{R_f/R_i}{1 + sR_fC} \end{aligned} \tag{2.28}$$

With reference to the discussion in Appendix A on continuous time systems, it can be seen that steady-state sinusoidal frequency response $A_{c\ell}(j\omega)$ is a LPF having corner frequency $\omega_c = 1/R_fC$:

$$A_{c\ell}(j\omega) = -\frac{R_f/R_i}{1 + j\omega/\omega_C} \tag{2.29}$$

The minus sign in the equation means signal phase inversion from input to output. Moreover, the closed-loop gain is independent of the open-loop gain (as long as A is large). Furthermore, since both the inverting and noninverting inputs are held at ground potential, the input impedance of the op-amp circuit of Fig. 2.24B (as well as in Fig. 2.25) presented to input voltage v_{in} is the resistance of R_i.

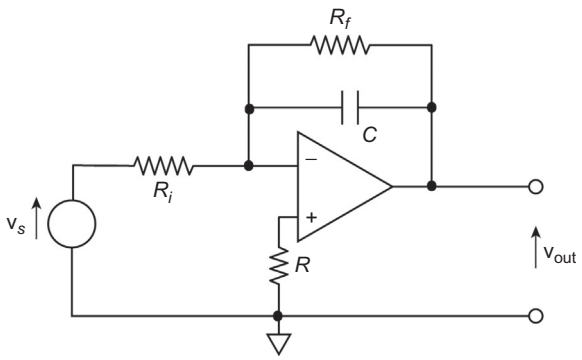


FIG. 2.25 Low-pass filter op-amp circuit.

A noninverting amplifier is also possible, as shown in Fig. 2.24C. The input signal is connected to the noninverting (+) terminal, and the output is connected through a series connection of resistors to the inverting (–) input terminal. The voltage gain, A_v, in this case is

$$A_v = \frac{v_{out}}{v_{in}} = 1 + \frac{R_f}{R_i} \tag{2.30}$$

Note that this noninverting circuit has no phase inversion from input to output voltage. The minimum closed-loop gain for this noninverting amplifier configuration (with $R_f = 0$) is unity. Besides adjusting gain (via the choice of R_f and R_i), negative feedback also can help to correct for the amplifier's nonlinear operation and distortion. The input impedance presented to the input voltage v_{in} by the noninverting op-amp configuration of Fig. 2.24C is very large (ideally infinite). This high input impedance is one of the primary features of the noninverting op-amp configuration.

SUMMING MODE AMPLIFIER

One of the important op-amp applications is summing of voltages. Fig. 2.26 is a schematic drawing of a summing mode op-amp circuit.

In this circuit, a pair of voltages v_a and v_b (relative to ground) is connected through identical resistances R to the inverting input. Using the property of inverting mode op-amps that the currents into the inverting input sum to 0, it can be shown that the output voltage v_o is proportional to the sum of the input voltages:

$$v_o = -\frac{R_f(v_a + v_b)}{R} \tag{2.31}$$

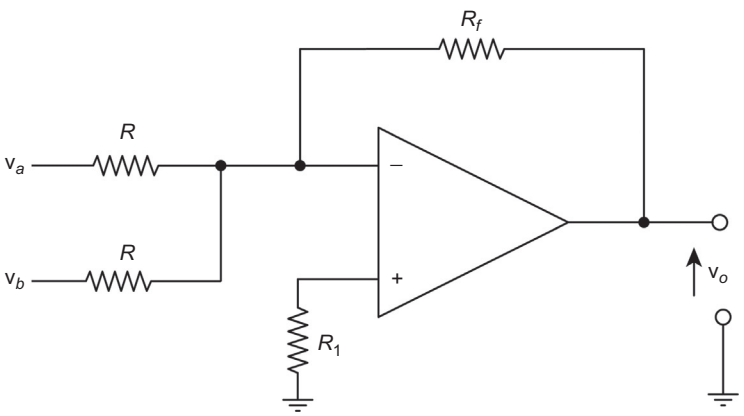


FIG. 2.26 Schematic drawing of a summing mode op-amp circuit.

COMPARATOR

One of the important circuits involving op-amps is the so-called analog comparator. The basic operation of an analog comparator is the generation of binary-valued voltages that switch between the two binary levels when an analog input crosses a threshold voltage (V_{th}). Fig. 2.27C depicts a comparator circuit that is implemented with an op-amp and a zener diode.

The open-loop output voltage without the zener diode is given by

$$V_o = A(V_{in} - V_{Th})$$

where V_{in} is the analog signal that is being compared with the threshold voltage V_{th}. The circuit is only linear over the very small range of voltage differences:

$$\frac{V_\gamma}{A} \leq \text{v}_{\text{in}} - \text{v}_{th} < \frac{\text{v}_z}{A}$$

where V_z = zener voltage of the diode and V_γ = forward diode voltage.

The open-loop gain A is sufficiently large that this linear range extends over a negligible voltage span in terms of its operation. A practical approximate model for the comparator is given below:

$$\begin{aligned} V_O &= V_z \qquad V_{\text{in}} > V_{th} \\ &\simeq 0 \qquad V_{\text{in}} < V_{th} \end{aligned}$$

In fact, there is a small hysteresis in the transition from one output voltage to the other. The majority of applications of an analog comparator are the generation of binary voltages that are input to a digital system and correspond to logic 1 (V_z) and logic 0.

Internally, the output circuit of the op-amp has a source resistance R_o that limits the output current to the zener diode and any additional load impedance. For a typical commercially available operational amplifier, it is of the order of 50–100 Ω. Whenever the comparator switches states, the open circuit output voltage is $\pm V_{\text{sat}}$ where V_{sat} is the saturation voltage of the amplifier. The zener diode must be capable of handling a current i_{zmax} given by

$$\begin{aligned} |i_z\text{max}| &= \frac{V_{\text{sat}} - V_z}{R_o} \qquad \text{v}_{\text{in}} > V_{\text{th}} \\ &= \frac{V_{\text{sat}}}{R_o} \qquad \text{v}_{\text{in}} < V_{\text{th}} \end{aligned}$$

If the zener diode chosen for a given comparator circuit cannot safely pass currents of $|i_{\text{zmax}}|$, it is possible to add some additional resistance between the op-amp output and the zener diode to safely limit the diode forward and reverse polarity currents.

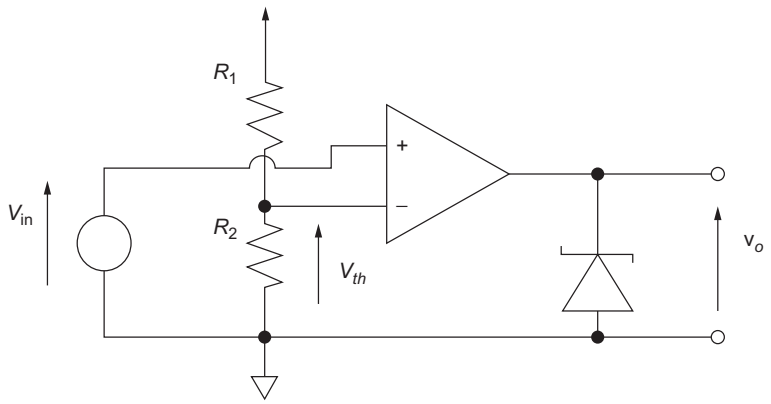


FIG. 2.27 Example analog comparator circuit.

ZERO-CROSSING DETECTOR

The circuit of Fig. 2.27 can be a so-called zero-crossing detector (ZCD) by setting $V_t = 0$ (i.e., at ground potential). The transition from high-to-low voltage levels in v_o correspond to the crossing of zero volts by V_{in}. This transition can readily be detailed by a digital control system. Chapter 5 makes reference to a ZCD in a crankshaft angular position sensor application. There are other applications of a ZCD in vehicular electronics.

PHASE-LOCKED LOOP

Another example of analog integrated circuit signal processing having automotive application is a device known as a "phase-locked loop" (PLL). This circuit can be used with certain analog (continuous time) sensors to provide an analog signal that can be further processed by a digital electronic system after it is sampled. The PLL finds application in the demodulation of phase- or frequency-modulated signals. At least one automotive application is the measurement of instantaneous crankshaft angular speed as explained later in this book.

A block diagram for the PLL circuit is shown in Fig. 2.28.

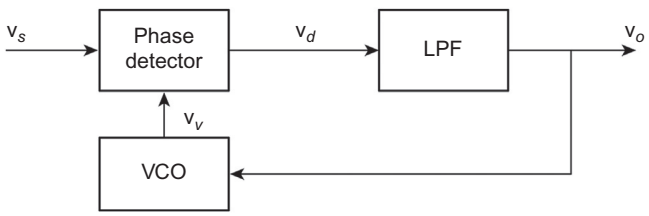


FIG. 2.28 Block diagram for PLL.

In this figure, the input signal $v_s(t)$ is assumed to be phase (ϕ) modulated and is given by

$$v_s(t) = V\cos(\omega_s t + \phi(t)) \tag{2.32}$$

where $\phi(t)$ is the instantaneous phase modulation.

A corresponding frequency-modulated signal has instantaneous frequency given by

$$\omega_s(t) = \Omega_s + \delta\omega_s(t)$$

where Ω_s is the time average frequency and $\delta\omega_s$ is the frequency deviation from mean (i.e., modulation). With respect to the model of Eq. (2.32), $\delta\omega_s$ is given by

$$\delta\omega_s = \dot{\phi}$$

In any practical application of the PLL for automotive systems, the modulation deviation is a small fraction of the carrier frequency (i.e., $|\delta\omega_s| \ll \Omega_s$).

The other components in Fig. 2.28 include a phase detector, a low pass filter (LPF), and a voltage-controlled oscillator (VCO). The phase detector is functionally an electronic multiplier that generates an output voltage v_d given by

$$v_d = K_d v_s v_v \tag{2.33}$$

where K_d is the constant for the device.

The VCO is an oscillator having an output voltage $v_v(t)$ whose instantaneous frequency ($\omega_v(t)$) is controlled by voltage v_o (from the LPF) such that

$$v_v(t) = V_v\cos(\phi_v(t)) \tag{2.34}$$

where

$$\phi_v(t) = \int_o^t \omega_v(\tau)d\tau \tag{2.35}$$

$$\omega_v(t) = \omega[v_o(0)] + K_v v_o(t) \tag{2.36}$$

where K_v is the constant for the VCO circuit.

The PLL circuit is an electronic closed-loop control system (see Appendix A). After a brief transient period during which the VCO frequency is controlled, its frequency is "locked" to ω_s (i.e., $\omega_v(t) = \omega_s(t)$) provided Ω_s is within the so-called capture range for the VCO. That is, PLL lock occurs provided that

$$|\Omega_s - \omega_v(0)| \le \Omega_c \tag{2.37}$$

where

$$\Omega_c = \text{PLL capture range} \tag{2.38}$$

For frequency modulation cases, under lock conditions, the VCO voltage is given by

$$v_v(t) = V_v\cos[\omega_v t + \delta\phi] \tag{2.39}$$

where $\omega_v = \omega_s$.

The instantaneous phase difference $\delta\phi(t)$ is linearity proportional to the frequency deviation $\delta\omega_s$ from the mean frequency Ω_s:

$$\delta\phi(t) = K_\phi \delta\omega_s \tag{2.40}$$

where K_ϕ is a constant for the VCO.

The phase detector output voltage is given by

$$v_d = K_d V_s V_v \cos(\omega_s t)\cos(\omega_s t + \delta\phi) \tag{2.41}$$

$$= \frac{K_d V_s V_v}{2}[\sin(2\omega_s t + \delta\phi) + \sin(\delta\phi)] \tag{2.42}$$

The LPF suppresses the term at frequency $2\omega_s$ and for small modulation $\sin\phi \approx \phi$ such that the output voltage is given by

$$v_o(t) = \frac{K_d V_s V_v K_\phi}{2}\delta\omega_s(t) \tag{2.43}$$

That is, the LPF output signal is proportional to the frequency modulation. Thus, this circuit is an FM demodulator. The filter pass band must be sufficiently large to accommodate the spectrum of $\delta\omega_s$.

SAMPLE AND ZERO-ORDER HOLD CIRCUITS

Chapter 3 and Appendix B illustrate the use of practical sample and zero-order hold circuits, which are used in discrete time digital systems. In order to understand the implementation of digital electronics in automotive systems, it is, perhaps, worthwhile to discuss, briefly, some actual circuit configurations for practical realizations of these important system components. The input (x_k) to a digital system is essentially a numerical representation in binary or binary-coded format of a sample of a continuous time voltage variable $V(t)$ at sample time t_k:

$$x_k = V(t_k, NB) \tag{2.44}$$

where $t_k = kT \quad k = 0, 1, 2\ldots$ and $T =$ sample period.

Where NB signifies an N-bit binary representation of $V(t_k)$ (i.e., see Chapter 3). This sampling process involves two steps: (1) obtaining a voltage sample at t_k and (2) converting this sample to the N-bit numerical format. The first step can be accomplished, in theory, via a switch that connects the continuous time voltage to a low-loss capacitor for a sufficient duration to charge the capacitor to the voltage value $V(t_k)$.

Fig. 2.29 depicts a sampler for a digital system that works in conjunction with an A/D converter as described above.

The A/D converter is explained in detail with example circuits in Chapter 3. This figure depicts the equivalent circuit of the source being sampled including its source impedance (assumed to be resistive) R_s and a very low-leakage capacitor C. The capacitor maintains voltage $V(t_k)$ sufficiently long to permit the A/D to complete its conversion. In Fig. 2.29A, the switch S model is given by

$$\begin{aligned} &\text{closed switch} \rightarrow S = 1 \quad && t_k \le t_k < +\tau_s \\ &\text{open switch} \rightarrow S = 0 \quad && t_k + \tau_s \le t < t_{k+1} \end{aligned} \tag{2.45}$$

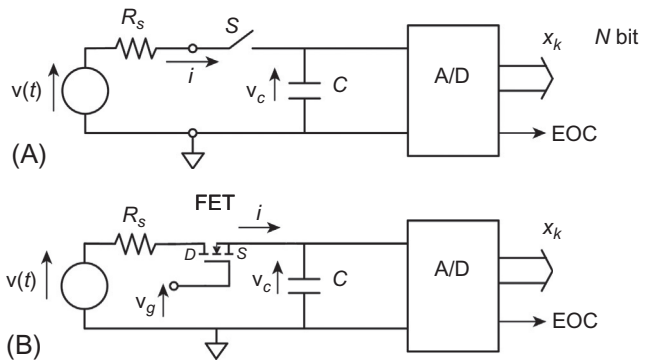


FIG. 2.29 Sample circuit. (A) Sampler with ideal switch *S*. (B) Sample with FET switch.

The duration of the switch closure τ_s must be long enough for the A/D conversion to be completed at which time an output EOC = 1 (logical) indicating "end of conversion" to the digital system. It is assumed for convenience that the input impedance of the A/D converter is sufficiently large that its input current is negligible.

During the period in which the switch is closed (at sample time t_k), the source voltage supplies current to the capacitor to change its voltage from $v_c = v(t_{k-1})$ to $v_c = v(t_k)$. The model for this circuit is given by

$$\begin{aligned} R_s i + v_c &= v(t_k) \quad t_k \le t < t_k + \tau_s \\ i &= 0 \qquad t_k + \tau_s < t \le t_{k+1} \end{aligned} \tag{2.46}$$

where $i = \dfrac{dq}{dt}$; q is the charge on capacitor; $v_c = \dfrac{q}{C}$; and C is the capacitance of capacitor.

The voltage that is held from the end of sample interval until the beginning of the next sample is denoted v_{ck} for the kth sample. For example, the held voltage at the end of the $k-1$ sample is given by

$$v_{c(k-1)} = v_c(t_{k-1} + \tau_s)$$

Eq. (2.46) can readily be rewritten in terms of v_c:

$$\begin{aligned} R_s C \dot{v}_c + v_c &= v(t_k) \quad t_k \le t < t_k + \tau_s \\ v_c(t) &= v_c(t_k + \tau_s) \quad t_k + \tau_s < t \le t_{k+1} \\ &= v_{ck} \end{aligned} \tag{2.47}$$

where v_{ck} is ideally held constant by the capacitor until time t_{k+1}. Because the output voltage of the above circuit "holds" the sample of source voltage v_{ck} for the indicated period, it is usually called a "sample-and-hold" circuit. The solution to the first-order differential Eq. (2.47) is readily obtained using the Laplace transform method given in Appendix A:

$$\begin{aligned} v_c(t) &= \left[v(t_k) - v_{c(k-1)}\right]\left(1 - e^{-(t-t_k)/\tau}\right) + v_{c(k-1)} \quad t_k \le t < t_k + \tau_s \\ &= v_{ck} \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\quad t_k + \tau_s < t + t_{k+1} \end{aligned} \tag{2.48}$$

where $\tau = R_s C$.

Ideally, the capacitor voltage v_{ck} should equal the sampled value of the source voltage v at $t = t_k$. This ideal voltage is approximated by the actual voltage v_{ck} provided $\tau \ll \tau_s$. Furthermore, τ_s should be small compared with the time that the source changes such that

$$v(t_k + \tau_s) \simeq v(t_k)$$

In order for this latter condition to be achieved, the sample duration period τ_s and the system sample period T must both be small. It is shown in Chapter 3 and Appendix B that the sample frequency $F_s = 1/T$ must be greater than twice the highest frequency in $v(t)$ to avoid aliasing errors (i.e., Nyquist sample rate). Furthermore, the sample duration must be larger than τ (i.e., $\tau_s \gg \tau$) in order for the sampler to approximate the ideal sampler performance. This latter objective requires that the capacitance C satisfies the following inequality:

$$C \ll \frac{\tau_s}{R_s}$$

In the practical sample circuit of Fig. 2.29B, the switch function is implemented via an FET whose source to drain resistance (R_{SD}) is a function of a control voltage v_g applied to the gate of the transistor. The switching operation can be achieved a control voltage by a periodic pulse train form as given by

$$\begin{aligned} v_g(t) &= V_H \quad t_k \le t < t_k + \tau_s \\ &= V_L \quad t_k \le \tau_s \le t < t_{k+1} \end{aligned} \tag{2.49}$$

It is assumed that V_H is sufficiently large to drive the FET into saturation and that V_L is sufficiently low (ideally 0) such that the FET is in cutoff. With $v_g(t)$ above applied to the FET gate, the source/drain resistance is denoted $R_{SD}(v_g)$ and is given by

$$\begin{aligned} R_{SD}(V_H) &= R_{\text{on}} \quad \text{(transistor in saturation)} \\ R_{SD}(V_L) &= R_{\text{off}} \quad \text{(transistor in cutoff)} \end{aligned}$$

Ideally, these resistances should be

$$\begin{aligned} R_{\text{on}} &= 0 \\ R_{\text{off}} &\to \infty \end{aligned}$$

However, in practice, R_{off} is finite but large and R_{on} is small but nonzero. An equivalent circuit for the FET in switch mode is given in Fig. 2.30.

Provided R_{off} is sufficiently large, the model for the circuit of Fig. 2.29 is given by

$$\begin{aligned} (R_S + R_{\text{on}})C\dot{v}_c + v_c &= v(t_k) \quad t_k \le t < t_k + \tau_s \\ &\cong v(t_k) \quad t_k + \tau_s < t < t_{k+1} \end{aligned} \tag{2.50}$$

The circuit in which the switch is implemented by the FET has the same dynamic response as that of the ideal switch model except that the time constant τ is given by

$$\tau = (R_s + R_{\text{on}})C$$

The performance of the practical sample circuit can approach that of the ideal sample by proper choice of circuit and system parameters.

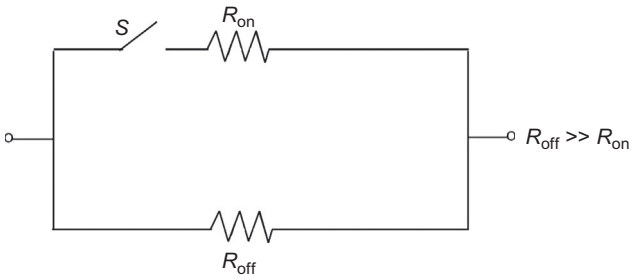


FIG. 2.30 Equivalent circuit for FET in switch mode.

ZERO-ORDER HOLD CIRCUIT

In addition to the ideal sampler component introduced above, in Chapter 3 and Appendix B, a circuit called a "zero-order hold." ZOH is shown to be required whenever a digital system output must be converted to an analog electric signal (e.g., to operate an analog actuator). The ZOH circuit is similar in certain respects to the "sample-and-hold" circuit introduced above; in that often, it is synchronous with the sampler at the system input and that it must hold a voltage between successive sample times. In addition, it incorporates a low-leakage capacitor to "hold" the voltage.

Fig. 2.31 depicts a ZOH circuit in which the system input y_k is the kth digital system output. In the figure, the digital system (not shown) generates an output sequence $\{y_k\}$ in the form of an N-bit binary "word" on a set of N-leads that are connected to the D/A converter.

The D/A converter is explained in detail (along with the schematic diagrams) in Chapter 3. The digital control system also generates a signal that controls the D/A operation such that, at the end of the conversion (EOC), the D/A output analog voltage $\bar{u}_k$ corresponds to the numerical value of y_k. The EOC output triggers a pulse generator having output voltage $v_g(t)$ given by

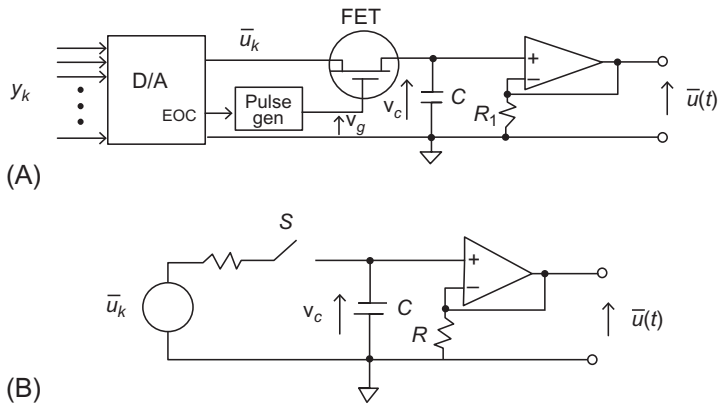


FIG. 2.31 ZOH circuit. (A) ZOH circuit configuration. (B) ZOH equivalent circuit.

$$\begin{aligned} \mathrm{v}_g &= V_H \quad t_k \le t \le t_k + \tau_s \\ &= V_L \quad t_k + \tau_s < t < t_{k+1} \end{aligned} \tag{2.51}$$

The pulse duration τ_s must be sufficiently long that the capacitor voltage is approximately (ideally) $\bar{u}_k$. Voltage v_g is applied to the FET gate, which functions as a voltage-controlled switch (as explained above with respect to the sample circuit). The source-drain resistance is given by $R_{SD}(V_g)$:

$$R_{SD}(V_g) = R_{\text{on}}$$

$$R_{SD}(V_L) = R_{\text{off}}$$

The model for the capacitor voltage $\mathrm{v}_c(t)$ is similar to that given for the sample circuit:

$$\begin{aligned} (R_{\text{on}} + R_s)C\dot{\mathrm{v}}_c + \mathrm{v}_c &= \bar{u}_k \\ &= \bar{u}_{ck} \quad t_k + \tau_s < t < t_{k+1} \end{aligned} \tag{2.52}$$

It is left as an exercise for the reader to find the capacitor voltage $\mathrm{v}_c(t)$ and show that it is a piecewise continuous function of time that approximates the output of the ideal ZOH of Appendix B. The primary differences between $\mathrm{v}_c(t)$ and $\bar{u}_t$ for an ideal ZOH are the (ideally) short intervals from $t = t_k$ to $t = t_k + \tau_s$, during which periods the capacitor voltage is changing. Except for the short intervals in which the capacitor voltage is changing, $\bar{u}_t$ is a stepwise continuous function of time t as depicted in Appendix B.

The circuit of Fig. 2.31 also incorporates an operational amplifier connected as a noninverting voltage follower having output voltage $\bar{u}_t$ where

$$\bar{u}_t = \mathrm{v}_c(t) \tag{2.53}$$

This op-amp provides isolation of the capacitor such that any circuit, which is connected to the ZOH output, will not place a load on v_c that would otherwise cause "loading" (with a drop in v_c from its desired value).

BIDIRECTIONAL SWITCH

Another practical circuit incorporated in digital systems that have analog signals at certain inputs or outputs is an electronically controlled switch that can function with input at either end of the switch. This circuit can be implemented with MOS fabrication involving both n-channel and p-channel FETs, also known as CMOS technology. A CMOS circuit that is termed a bidirectional gate and the circuit symbol for it are depicted in Fig. 2.32.

This circuit can function as a voltage (V_c)-controlled switch that can pass signals in direction either to/from – in/out or to/from – in/out b. The control voltage and its logical inverse are applied to the gates of the switching FETs. The input voltage range is limited to

$$o \le \mathrm{v}_{\text{in}} < V_{DD}$$

Typically, these voltage-controlled bidirectional switches are implemented as part of an IC for a specific function as will be described elsewhere in this book but will be represented by the circuit symbol of Fig. 2.32. In this symbol, the control is represented by the logical variable C_c. The circuit control voltage v_c is binary-valued in which the high level (i.e., $\mathrm{v}_c = V_H$) corresponds to $C_c = 1$, and the low voltage level (i.e., $\mathrm{v} = V_L$) corresponds to $C_c = 0$. In the circuit, the voltage denoted V_{EE} is a voltage that

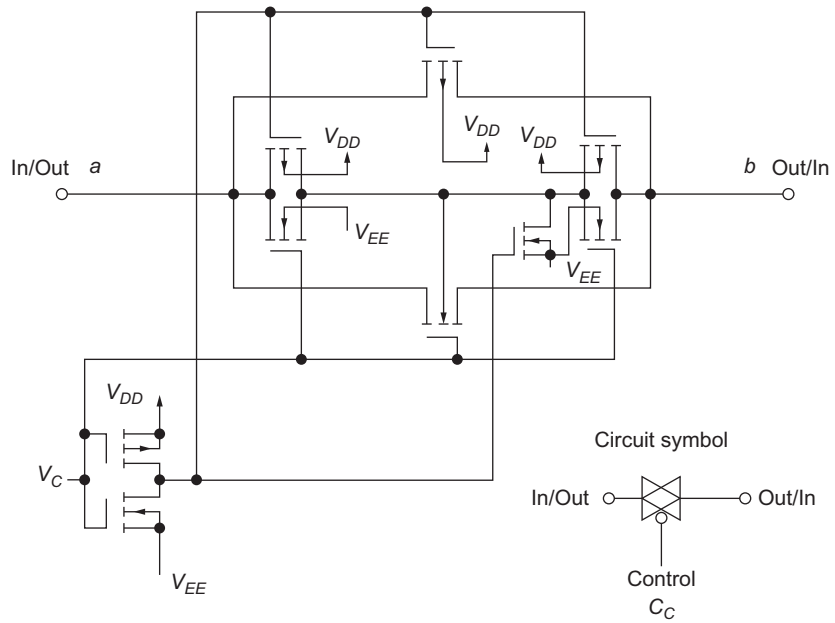


FIG. 2.32 Bidirectional gate schematic and circuit symbol.

must be less than or equal to the source voltage (V_{SS}) for the other FETs in the larger portion of the IC. The control voltage is relative to V_{SS}, and the analog voltages are referenced to V_{EE}.

One of the important circuits that incorporate bidirectional gates is the so-called analog multiplexer (MUX) or demultiplexer (DEMUX). These devices have multiple potential applications in vehicular digital electronics. An analog MUX is used to selectively pass one of N analog signals to a digital computer that performs various computations on the multiple analog signals one at a time. Often, the analog signals are sent to the computer sequentially in time via a process known as time domain multiplexing (TDM). For example, in Chapter 8, which is devoted to vehicular instrumentation, it is explained that a single computer performs analytic transformations on several analog signals coming from devices known as sensors (many of which are explained in Chapter 5).

A simplified exemplary diagram of a four-analog signal TDM is presented in Fig. 2.33.

For illustrative purposes, this figure assumes that only four-analog input signals (v_1, v_2, v_3, and v_4) must be sent sequentially to a single output v_{out}. It is further assumed that the computer that is performing signal processing operations generates an output clock signal (C_k) that is fed to a 4-bit binary counter (BC). The BC output are the logical variables A and B, which selectively activate one of the four bidirectional gates (i.e., BD_1, BD_2, BD_3, and BD_4).

For the purposes of this example, it is assumed that a bidirectional gate (BD_n) is closed (i.e., R_{on}) whenever the corresponding control (C_{cn}) is logical 1 (for $n = 1, 2, 3, 4$). The output voltage v_{out} is given by

$$V_{out} = v_n \text{ if } C_{cn} = 1 \quad n = 1,2,3,4$$

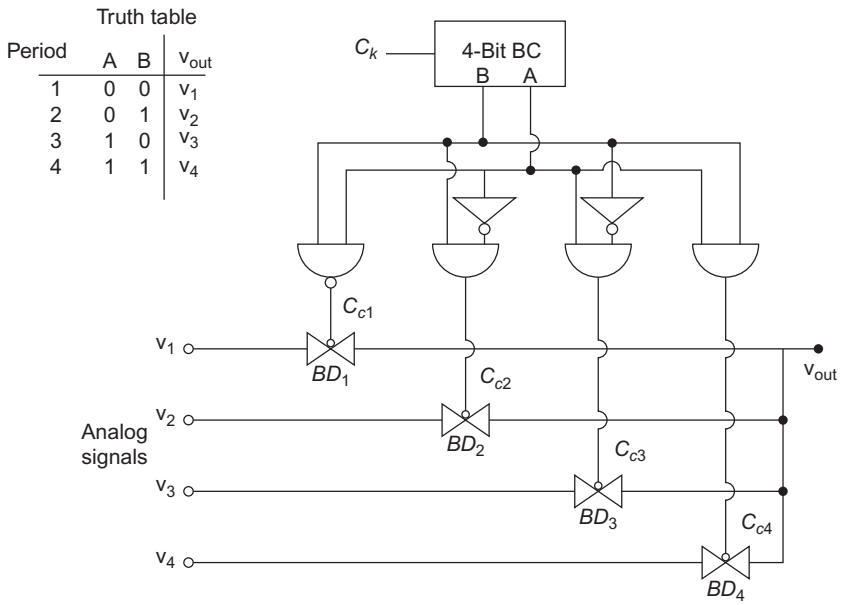


FIG. 2.33 Example four-input analog multiplexer.

By comparing the truth table for the AND, NAND gates of this figure, it can be shown that the output switches sequentially to one of the four inputs depending on the state of A and B. For each C_k pulse, the BC increments by 1 and its outputs sequentially from 00 to 11 and then repeats continuously to count modulo 4.

In a practical application, the MUX output voltage would be sent to a sampling ZOH and A/D converter that are also controlled by the given computer. In this way, the computer sequentially receives appropriate digital inputs for processing. This application of analog MUX is explained in Chapter 8.

The bidirectional nature of the BD switches makes it possible to supply multiple analog outputs from a computer. In this DEMUX application of the block diagram/circuit of Fig. 2.33, the computer sequentially applies a digital output to a D/A converter that would then be connected to the terminal labeled v_{out}. In this case, it would be the input to the DEMUX and the analog voltages v_1, v_2, v_3, and v_4 are the analog outputs. In one application of DEMUX, the computer could be supplying inputs to analog type displays as explained in Chapter 8.

It should be noted that although Fig. 2.33 is in a block diagram format, example circuits for each component except the BC have been presented. The components and operation of a BC are explained later in this chapter. Thus, Fig. 2.33 is effectively an example circuit for a MUX/DEMUX IC.

DIGITAL CIRCUITS

Digital circuits, including digital computers, are formed from binary circuits. Binary digital circuits are electronic circuits whose output can be only one of the two different states. Each state is indicated by a particular voltage or current level. Binary circuits can operate in only one of the two states (on or off) corresponding to logic 1 or 0, respectively. Digital circuits also can use transistors.

In a digital circuit, a transistor is in either one of the two modes of operation: on, conducting (at saturation), or off (in the cutoff state).

The corresponding binary voltage levels in digital circuits have two states: a high-voltage state denoted V_H corresponding to logical 1 and a low voltage state denoted V_L corresponding to logical 0.

In electronic digital systems, a transistor is used as a switch. As explained above, a transistor (either bipolar or FET) has three operating regions: cutoff, active, and saturation. If only the saturation or cutoff regions are used, the transistor acts like a switch. When in saturation, the transistor is on and has very low resistance; when in cutoff, it is off and has very high resistance. In digital circuits, the input voltage to the transistor switch must be capable of either saturating the transistor or putting it into a cutoff condition without allowing operation in the active region. The on condition is indicated by a very low output voltage, and the off condition is by an output voltage equal to or slightly below power supply voltage.

Fig. 2.34 depicts an NPN transistor circuit configuration for use in a digital circuit.

In this figure, it can be seen that no bias resistor is present since this transistor is not operated in the linear (active) mode. Rather, the source voltage is binary-valued having only two voltage levels:

$$\begin{aligned} v_s &= V_H(\text{high voltage}) \\ &= V_L(\text{low voltage}) \end{aligned}$$

The operation of this type of transistor circuit can be illustrated assuming that it is a 2N4401 transistor having characteristic curves as depicted in Fig. 2.16B.

In the present example, it is assumed that the low voltage $V_L < V_d$ where V_d is the base-emitter voltage threshold (discussed above) above which base current flows. Whenever $v_s = V_L$, the base current and collector current are essentially zero. The output voltage v_o is given by

$$v_o = V_{cc} - i_c R_L \simeq V_{cc} \tag{2.54}$$

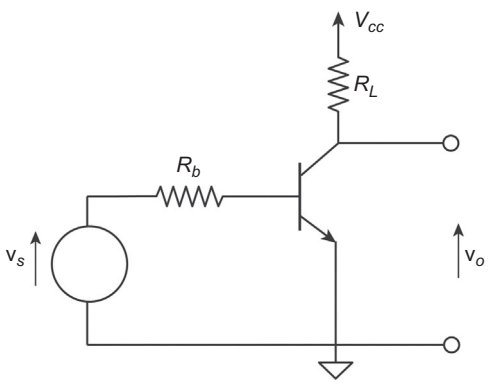


FIG. 2.34 NPN transistor digital circuit.

It is assumed that the high voltage for this example is sufficient that the base current i_b is given by $i_b = \frac{v_H}{R_s} > 600\mu\,\text{amp}$

In this case, the output voltage is <0.5 Volts.

The above example is presented simply as an illustration of a transistor operating in a binary state. Actual binary voltage levels for transistor digital circuits depend upon the type of transistor used and the voltage conventions for representing logical 1 or 0.

BINARY NUMBER SYSTEM

Digital circuits function by representing various quantities numerically using a binary number system or some other coded form of binary such as octal or hexadecimal numbers. In a binary number system, all numbers are represented using only the symbols 1 (one) and 0 (zero) arranged in the form of a place position number system. Electronically, these symbols can be represented by transistors in either saturation or cutoff or circuits having voltage level V_H or V_L. Before proceeding with a discussion of digital circuits, it is instructive to review the binary number system briefly.

An M-bit binary number (which we denote here as N_2) is represented by a set of binary digits called bits $\{A_n = 0,1\}$ arranged in a place position number system as shown below (with A_M the most significant):

$$N_2 = A_M A_{M-1} \ldots A_m \ldots A_1 \tag{2.55}$$

Each bit in this M-bit binary number is a multiple of a power of 2 in a decimal equivalent. The decimal equivalent of N_2 is denoted N_{10} and is given by

$$N_{10} = \sum_{m=1}^{M} A_m 2^{m-1} \tag{2.56}$$

For example, the decimal equivalent of the binary number 1010 (i.e., $M = 4$) is given by

$$\begin{aligned} N_{10} &= 1 \times 2^3 + 0 \times 2^2 + 1 \times 2 + 0 \times 1 \\ &= 8 + 2 = 10 \end{aligned}$$

As mentioned above, another coded number system can be formed from binary by grouping bits to form a new base (as long as it is an integer power of 2). For example, an octal number system is base 8 that uses octal digits such that $A_m = 0, 1, \ldots 6, 7 \ldots$. In such a system that can be implemented by groups of three transistor switches to yield eight possible combinations, an octal number (denoted N_8, with A_M being the most significant digit) is given by

$$N_8 = A_M \cdots A_1$$

The decimal equivalent of N_8, which we denote $N_{10}(M)$, is given for an M-digit octal number by

$$N_{10}(M) = \sum_{m=1}^{M} A_m 8^{m-1} \tag{2.57}$$

LOGIC CIRCUITS (COMBINATORIAL)

Essentially, all electronic systems in contemporary vehicles are digital in nature and incorporate integrated circuits formed with very large numbers of transistors that are connected together to implement a number of basic logic circuits euphemistically called "gates" that perform operations on logical variables.

There are two major categories of logic circuits that perform the basic operations in any digital electronic system: (1) combinatorial and (2) sequential. We begin with the combinatorial logic circuits that perform operations that depend on the state of the logical input variables at any given time. The sequential logic circuits (which are explained later in this chapter) perform operations that are dependent upon previous inputs and the present state of input variables.

Fig. 2.35 presents a summary of important representations for the five basic logic gates with which essentially all combinatorial logic circuits are built. This figure includes schematic symbols for each that are often conveniently used for preparing a schematic for a digital circuit. In addition, each section

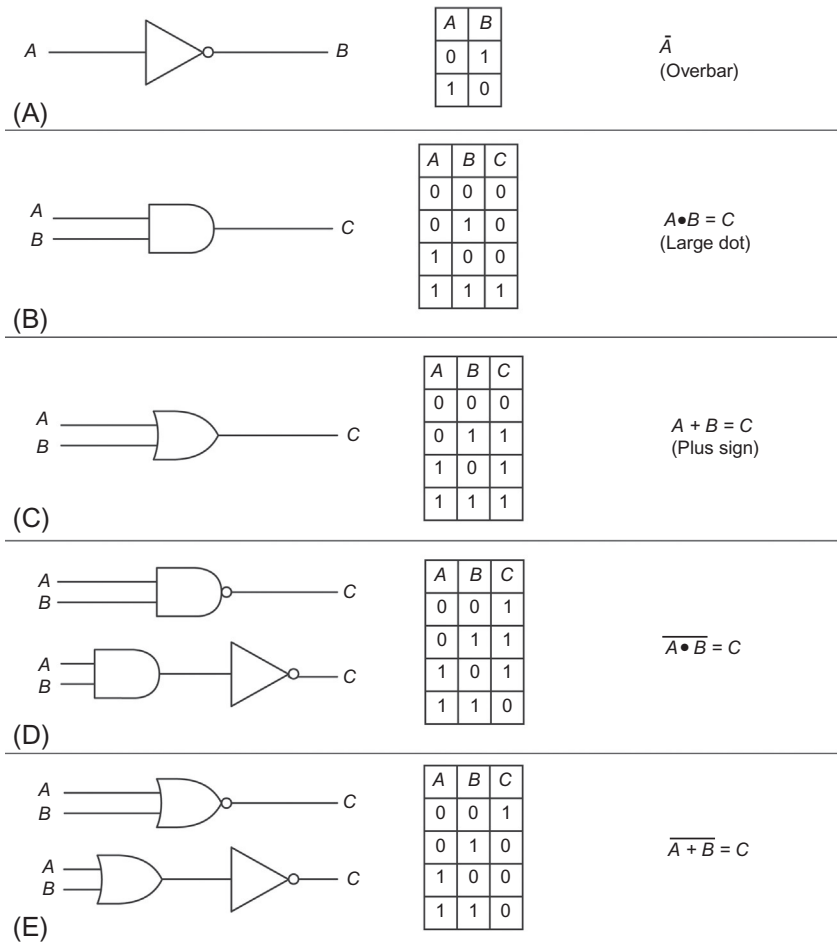


FIG. 2.35 Logic "gates." (A) NOT, (B) AND, (C) OR, (D) NAND, and (E) NOR.

of Fig. 2.35 presents a "truth table" for the corresponding gate. A truth table presents the logical result (i.e., output of the gate) for all possible combinations of input variables. Fig. 2.35 involves only three logical variables, A and B inputs and C, the output for the associated logical operation.

The analysis of the operation of logic gates is formulated with a special purpose algebra that is known as "Boolean" algebra. The variables in Boolean algebra (e.g., A, B, and C in Fig. 2.35) have only two values: 1 or 0. The symbols representing the logical operation in Boolean algebra are explained with respect to the five gates presented in Fig. 2.35. Following the discussion of these five gates, in which the algebraic symbols are introduced, there is a discussion of the rules of Boolean algebra. These rules provide a basis for the analysis of the operation of arbitrarily complex combinations of logic gates such as occur in vehicular digital electronics.

AND GATE

The *AND* circuit effectively performs the logical conjunction operation on binary numbers or logical conditions. The *AND* gate has at least two inputs and one output. The one shown in Fig. 2.35B has two inputs. The output is high (1) only when both (all) inputs are high (1). If either or both inputs (or any) are low (0), the output is low (0). Fig. 2.35B shows the truth table, schematic symbol, and logic symbol for this gate. The two inputs are labeled A and B. Notice that for two inputs, there are four combinations of A and B, but only one results in a high output. In general, for N inputs, there are 2^N combinations with only one having a high (logic 1) output. The Boolean algebra notation for the AND logical operation between any two variables is center dot as depicted in Fig. 2.35B.

$$A \cdot B \rightarrow A\,AND\,B$$

Alternatively, by analogy to ordinary algebra, where the product of two variables x and y is often written without the center dot, the logical *AND* is sometimes written without the center dot in the form $AB \rightarrow A\,AND\,B$

OR GATE

The *OR* gate, like the *AND* gate, has at least two inputs and one output. The one shown in Fig. 2.35C has two inputs. The output is high (1) whenever one or both (any) inputs are high (1). The output is low (0) only when both inputs are low (0). Fig. 2.35C shows the schematic symbol, logic symbol, and truth table or OR gate. The Boolean algebra notation for the logical OR operation for any part of logical variables is the + sign. That is, A OR $B \rightarrow A+B$ as depicted in Fig. 2.35C.

NOT GATE

The NOT gate is a logic inverter. If the input is a logical 1, the output is a logical 0. If the input is a logical 0, the output is a logical 1. It changes zeros to ones and ones to zeros. The simple bipolar transistor circuit of Fig. 2.34 performs the same function if operated from cutoff to saturation. A high base voltage (logical 1) produces a low collector voltage (logical 0) and vice versa. Fig. 2.35A shows the schematic symbol for a NOT gate. Next to the schematic symbol is called a truth table. The truth table lists all of the possible combinations of input A and output B for the circuit. The Boolean algebra logic symbol is shown also, which is read as "NOT A." The bar over a logical variable indicates the logical inverse of the variable and is the Boolean algebra symbolic notation for this operation; that is, if $A = 1$, then $\bar{A} = 0$ or if $A = 0$, then $\bar{A} = 1$.

In addition to the *AND*, *OR*, and *NOT* logical circuits, there are combinations of *AND* and *NOT* yielding the so-called NAND gate. Similarly, the combination of *OR* with *NOT* yields the so-called NOR gate. Combining these two pairs of functions in a single circuit is often advantageous for building up larger digital circuit subsystems or systems on a single IC. Fig. 2.35D and E depict the schematic symbols for the *NAND* and *NOR*, respectively, along with truth tables and logical symbols.

BOOLEAN ALGEBRA

Analysis of circuits formed by combinations of the basic logic gates can be done using Boolean algebra. Boolean algebra is formulated based on a set of logical rules involving multiple logical variables. Table 2.1 is a summary of these rules applied to logical variables A, B, and C.

The rules of Boolean algebra are supplemented with two important theorems called DeMorgan's theorems, which can simplify a Boolean algebra expression. The two DeMorgan's theorems written in Boolean algebra notation are the following:

1. $\overline{A+B} = \bar{A} \cdot \bar{B}$
2. $\overline{A \cdot B} = \bar{A} + \bar{B}$

EXEMPLARY CIRCUITS FOR LOGIC GATES

As mentioned earlier and illustrated with a bipolar example, digital circuits operate with transistors in one of the two possible states: saturation or cutoff. Since these two states can be used to represent multiple-digit binary numbers. The input and output voltages for such digital circuits will be either

Table 2.1 Rules of Boolean Algebra

1.	$0+A=A$
2.	$1+A=1$
3.	$A+A=A$
4.	$A+\bar{A}=1$
5.	$0 \cdot A=0$
6.	$1 \cdot A=A$
7.	$A \cdot A=A$
8.	$A \cdot \bar{A}=0$
9.	$A+B=B+A$
10.	$A \cdot B=B \cdot A$
11.	$A+(B+C)=(A+B)+C$
12.	$A \cdot (B \cdot C)=(A \cdot B) \cdot C$
13.	$A \cdot (B+C)=A \cdot B+A \cdot C$
14.	$A+A \cdot C=A$
15.	$A \cdot (A+B)=A$
16.	$(A+B) \cdot (A+C)=A \cdot (A+C)+B \cdot (A+C)=A+B \cdot C$
17.	$A+\bar{A} \cdot B = A + B$
18.	$A \cdot B + B \cdot C + \bar{A} \cdot C = A \cdot B + \bar{A} \cdot C$

"high" or "low," corresponding to 1 or 0. High voltage means that the voltage exceeds a high threshold value that is denoted V_H. If the voltage at the input or output of a digital circuit is denoted V, then symbolically, the high-voltage condition corresponding to logical 1 is written as

$$V \geq V_H \tag{2.58}$$

Similarly, low voltage (corresponding to logical 0) means that voltage V is given by

$$V \leq V_L \tag{2.59}$$

where V_L denotes the low threshold value. The actual values for V_H and V_L depend on the technology for implementing the circuit. Representative values are $V_H = 2.4$ and $V_L = 0.8$ V for bipolar transistors.

Although it is not necessary to understand the circuit details of logic gates since they are implemented in large numbers in digital ICs, for the interested reader, a few exemplary circuits are presented for certain gates that are based on FET transistors. In digital circuits, FETs are operated in a switching mode (rather than as a linear device) as explained with respect to the sampling circuit of Fig. 2.29. In the following example circuits, the voltages are assumed to be binary-valued such that the high-voltage state corresponds to logic 1 and the low voltage state to logic 0.

We begin with an example circuit for a logic inverter that has a logic function explained above. In principle, a single FET connected as shown in Fig. 2.22 could function as a logic inverter provided the input voltage levels caused the FET to be in saturation or cutoff. However, an inverter circuit must function by supplying the output to another circuit having a specific input impedance (normally resistive). A typical practical FET inverter circuit is depicted in Fig. 2.36.

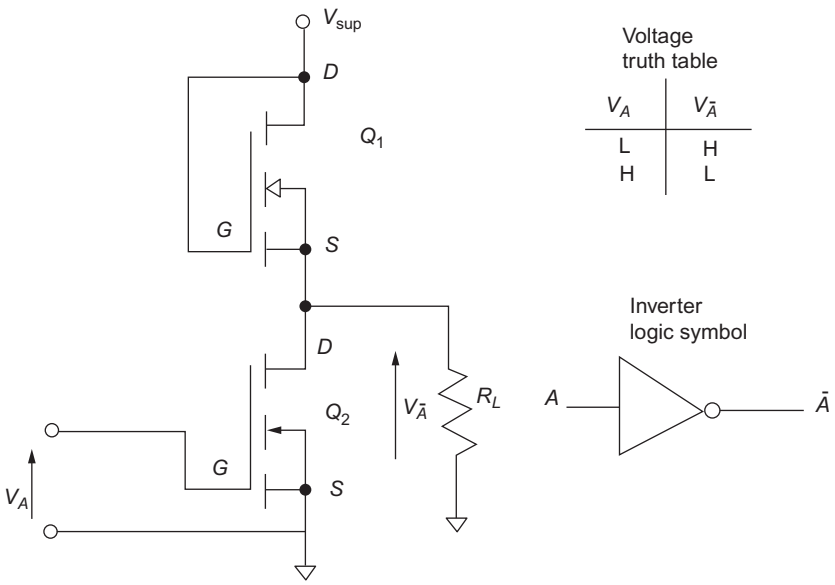


FIG. 2.36 Example FET inverter circuit.

In this circuit, FET Q_1 is a load transistor on the switching transistor Q_2. This load transistor is always in the on state and limits the current available to the load R_L, which represents the input to the circuit that receives the logical inverse of the input V_A to the inverter. When the input V_A is in the high state (corresponding to logic A = 1), the switching transistor Q_2 is in saturation state, and the output voltage $V_{\bar{A}}$ is in the low state corresponding to logic 0. When the input is in the low state, the switching transistor is in cutoff, and the output voltage $V_{\bar{A}}$ is in the high state corresponding to logic 1.

Another example circuit diagram for a logic gate implemented with FETs is the *NAND* gate seen in Fig. 2.37.

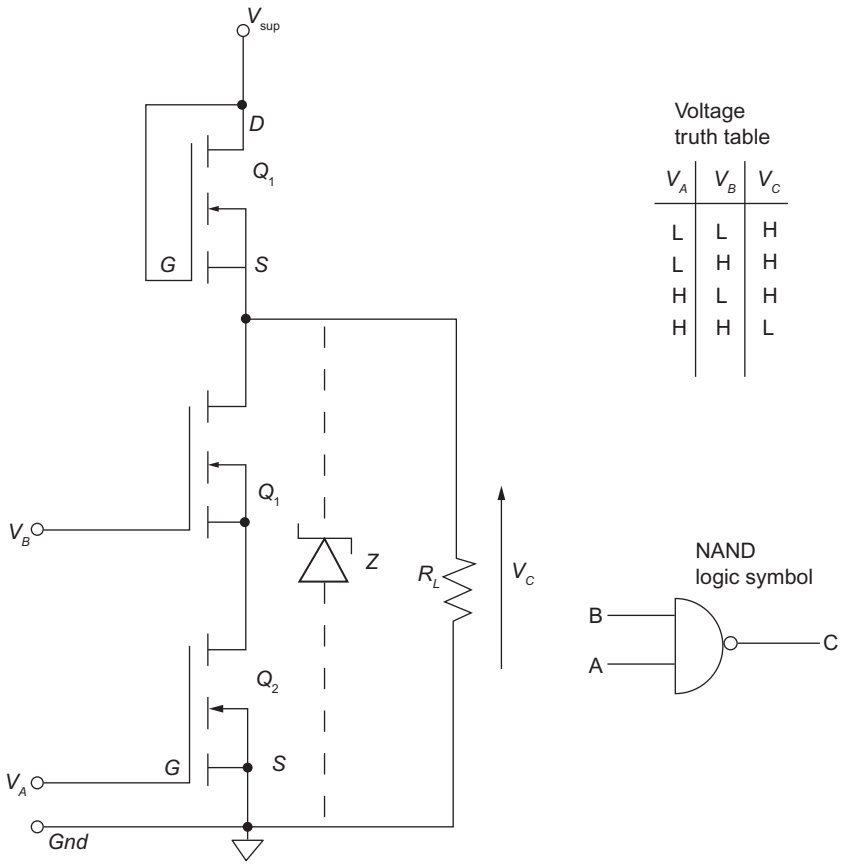


FIG. 2.37 Example NAND gate FET circuit.

This figure also depicts a truth table based on the two input and output voltages where $H \rightarrow$ high-voltage state and $L \rightarrow$ low-voltage state. This circuit incorporates a load transistor Q_L and two switching transistors Q_1 and Q_2. In general, it is not required, but for some versions of this circuit, a zener diode (denoted z) clamps the high-voltage state to a fixed value. In the circuit

of Fig. 2.37, the output state voltage is only low if both inputs are high and both Q_1 and Q_2 are in saturation. If either input is low, the corresponding transistor is in cutoff with a very high resistance (ideally open circuit). The possible combinations of the input voltages and the corresponding output voltage are given in the truth table of Fig. 2.37. The voltages *L*, H correspond to logic 0, 1 for the corresponding logical variable (i.e., *A*, *B*, or *C*). Also shown in Fig. 2.37 is the logic symbol presented in Fig. 2.35D.

Another example of a logic gate circuit is for a *NOR* gate and is depicted in Fig. 2.38.

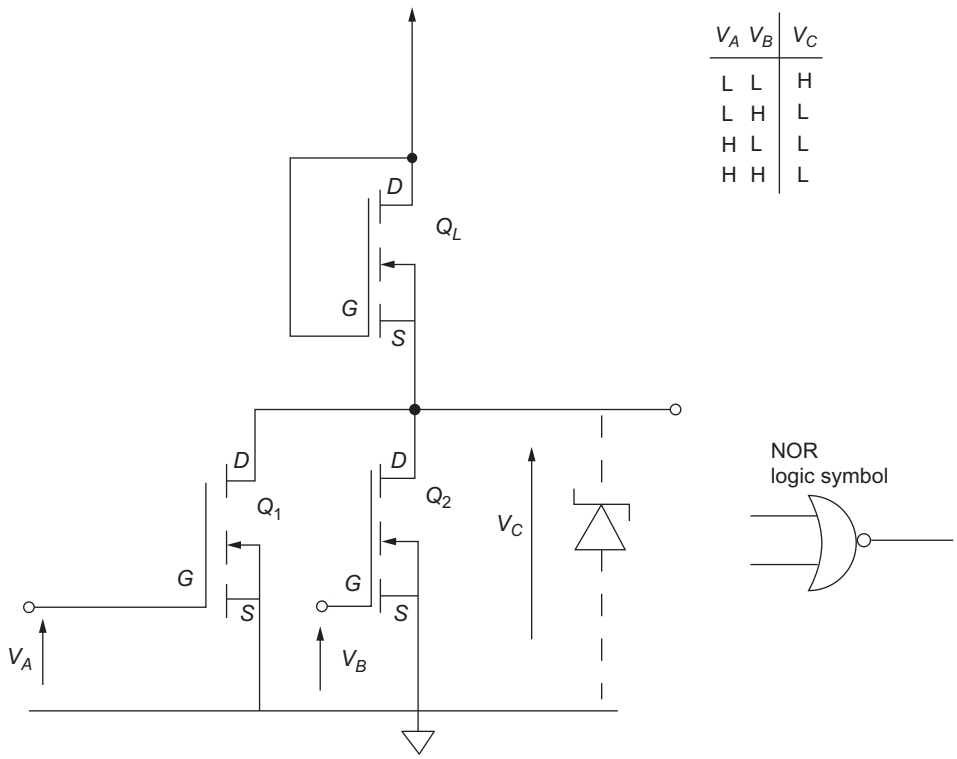


FIG. 2.38 Example circuit for NOR logic.

This circuit employs a load transistor Q_L, the same as the *NOT* and *NAND* gates explained above. The switching transistors Q_1 and Q_2 are able to switch the output low if either V_A or V_B is high by saturating Q_1 or Q_2, respectively. The output is only high when both inputs (V_A and V_B) are low. For all three voltages depicted in this circuit, a high voltage *H* is equivalent to logic 1, and a low voltage *L* is equivalent to logic 0. By comparing the voltage truth table with the logic truth table of Fig. 2.35E, it can be seen that the circuit of Fig. 2.38 is a *NOR* gate circuit.

The circuits depicted above can be combined to yield positive logic circuits. That is, a *NOT* gate (inverter) circuit of Fig. 2.36 connected to the output of the *NAND* gate of Fig. 2.37 will yield an *AND* gate circuit. That is, the inverter will take the C output for Fig. 2.37, which is $C = \overline{A \cdot B}$ and invert it as given by

$$\bar{C} = \overline{\overline{A \cdot B}} = A \cdot B(AND)$$

Similarly, an inverter circuit connected to the output of the NOR circuit of Fig. 2.37 yielding an *OR* gate circuit. There are other circuits for implementing the logic gates of Fig. 2.35, but the examples given above illustrate the basic building blocks of any digital circuit.

COMBINATION LOGIC CIRCUITS

Still, another important logical building block that can be built up with the three basic logic gates is the exclusive OR denoted XOR. This circuit has logical 1 output if and only one if its inputs is nonzero. A two-input example is shown in Fig. 2.39A. The schematic symbol for this device is depicted on the upper left of Fig. 2.39A. Its implementation using the three basic gates is shown at the lower left. The XOR truth table and logic symbol $\oplus$ are also given in Fig. 2.39A. In addition, the Boolean algebra notation for A, XOR, B is given in Fig. 2.39A.

All of these gates can be used to build digital circuits that perform all of the arithmetic functions of a calculator or computer. Table 2.2 shows the addition of two binary bits in all the combinations that can occur. Note that in the case of adding a 1 to a 1, the sum is 0, and a 1, called a carry, is placed in the next place value (in a place position number sequence) so that it is added with any bits in that place value. A digital circuit designed to perform the addition of two binary bits is called a half adder and is shown in Fig. 2.39B. Note that it incorporates an *XOR* gate. It produces the sum and any necessary carry, as shown in the truth table.

A half-adder circuit does not have an input to accept a carry from a previous place value. A circuit that does accept a carry input is called a full adder (Fig. 2.39C). A series of full-adder circuits can be combined to add binary numbers with as many digits as desired. Any digital computing system from a simple electronic calculator to the largest digital computer performs all arithmetic operations using full-adder circuits (or some equivalents) and a few additional logic circuits. In such circuits, subtraction is performed as a modified form of addition by using some of additional logic circuits as explained in Chapter 3. Multiplication of two 1-bit numbers is characterized by elementary rules of multiplication:

$$0 \times 0 = 0$$

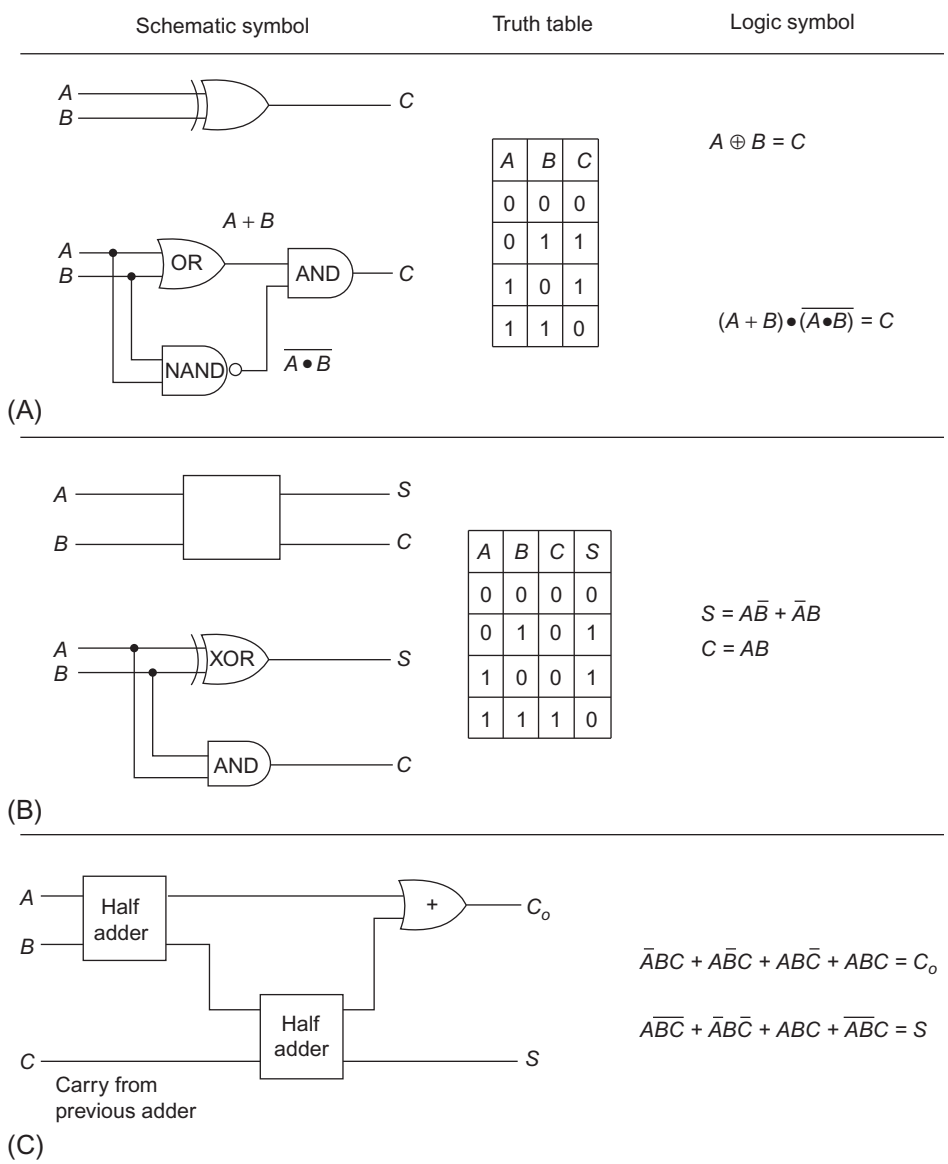
$$0 \times 1 = 0$$

$$1 \times 0 = 0$$

$$1 \times 1 = 1$$

Multiplication of an N-bit number by an M-bit number is implemented under program control in some form of stored program computer as explained in Chapter 3.

Of course, the addition of pairs of 1-bit numbers has no major application in digital computers. On the other hand, the addition of multiple-bit numbers is of crucial importance in digital computers and,



A	B	C
0	0	0
0	1	1
1	0	1
1	1	0

A	B	C	S
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

FIG. 2.39 Combination logic circuit. (A) XOR, (B) half-adder, and (C) full-adder.

Table 2.2 Addition of Binary Bits				
Bit *A*	0	0	1	1
Bit *B*	0	1	0	1
Sum	0	1	1	10

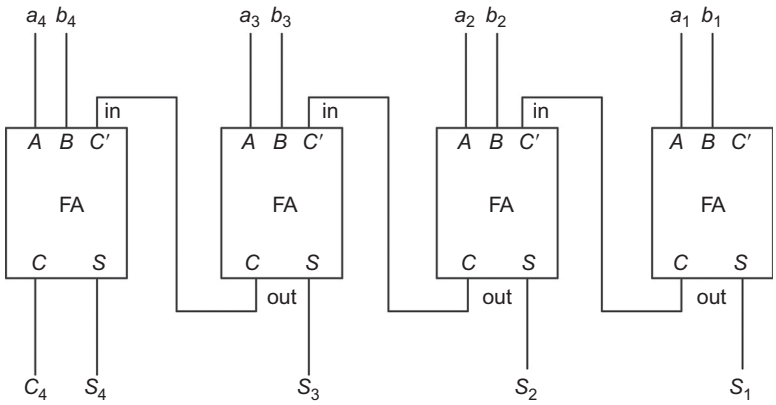


FIG. 2.40 4-Bit adder circuit.

of course, in automotive digital control systems. The 1-bit full-adder circuit can be expanded to form a multiple-bit-adder circuit. By way of illustration, a 4-bit adder is shown in Fig. 2.40. Here, the 4-bit numbers in place position notation are given by

$$A = a_4a_3a_2a_1$$

$$B = b_4b_3b_2b_1$$

where each bit is either 1 or 0. The sum of two 4-bit numbers has a 5-bit result, where the fifth bit is the carry from the sum of the most significant bits. Each block labeled FA is a full adder. The carry out (C) from a given FA is the carry in (C') of the next-highest full adder. The sum S is denoted (in place position binary notation) by

$$S = C_4S_4S_3S_2S_1$$

LOGIC CIRCUITS WITH MEMORY (SEQUENTIAL)

The logic circuits discussed so far have been simple interconnections of the three basic gates *NOT*, *AND*, and *OR*. The output of each system is determined only by the inputs present at that time. As explained in the introduction to digital circuits, these circuits are called combinatorial logic circuits. There is another type of logic circuit that has a memory of previous inputs or past logic states. This type of logic circuit is called a sequential logic circuit because the sequence of past input values and the logic states at those times determines the present output state. Because sequential logic circuits hold or store information even after inputs are removed, they are the basis of semiconductor computer memories.

R-S FLIP-FLOP

A very simple memory circuit can be made by interconnecting two *NAND* gates, as depicted in Fig. 2.41A.

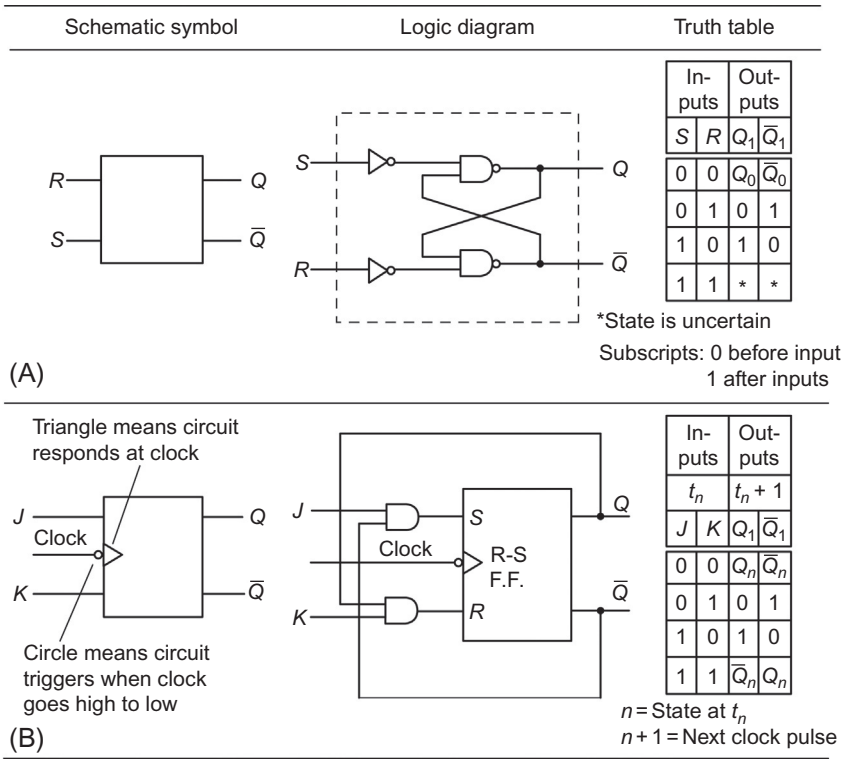


FIG. 2.41 Sequential logic circuits. (A) R-S flip-flop, (B) JK flip-flop.

A careful study of the circuit reveals that when *S* is high (1) and *R* is low (0), the output *Q* is set high and remains high regardless of whether *S* is high or low at any later time. The high state of *S* is said to be latched into the state of *Q*. The only way *Q* can be unlatched to go low is to let *R* go high and *S* go low. This resets the latch. This type of memory device is called a reset-set (R-S) flip-flop and is the basic building block of sequential logic circuits. The term "flip-flop" describes the action of the logic level changes at *Q*. Notice from the truth table that R and *S* must not be 1 at the same time. Under this condition, the two gates are logically indeterminate, and the final state of the flip-flop output is uncertain.

JK FLIP-FLOP

A flip-flop where the uncertain state of simultaneous inputs on *R* and *S* is solved is shown in Fig. 2.41B. It is called a JK flip-flop and can be obtained from an RS flip-flop by adding additional logic gating, as shown in the logic diagram. When both J and K inputs are 1, the flip-flop changes to a state other than

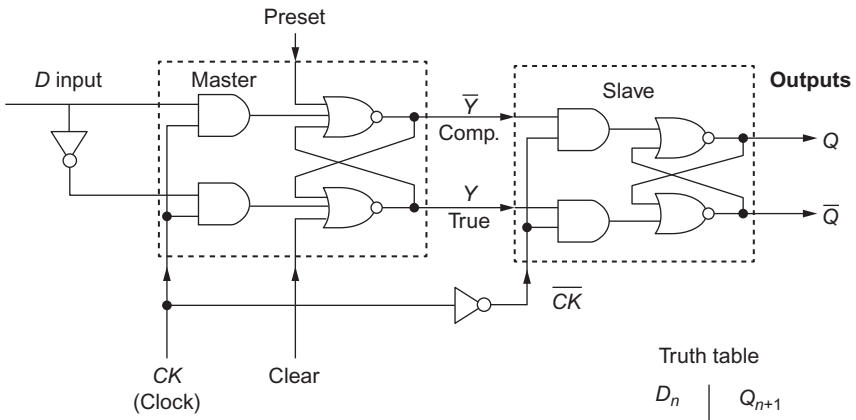
the one it was in. The flip-flop shown in this case is a synchronized one. That means it changes state at a particular time determined by a timing pulse, called the clock, being applied to the circuit at the terminal marked by a triangle. The little circle at the clock terminal means the circuit responds when the clock goes from a high level to a low level. If the circle is not present, the circuit responds when the clock goes from a low level to a high level. As is shown in the section of this chapter called "Synchronous Counter," there are many uses of JK-type circuits for their equivalent implementation in computers that operate with a clock as explained later.

D FLIP-FLOP

Another type of flip-flop, which is commonly used in both data storage (memory) and shift register applications, is the so-called D flip-flop (DFF). The D in the label represents delay or data. A DFF is implemented with a pair of cascade connected R-SFFs with some additional logic circuits for control and with timing provided by a periodic pulse train called a clock (*Ck*). Fig. 2.42 is a schematic of a DFF implemented with some of the logic gates explained in the section on combinatorial logic circuits.

The DFF is made up of two sections labeled "master" and "slave," respectfully, as denoted in Fig. 2.42. Each section incorporates an R-SFF as implemented with pairs of NOR circuits. In the master section, the NOR gates have three inputs, and in the slave section, the NOR gates have two inputs.

The DFF has the two sections because in its normal operation; for example, to store or shift data, it is necessary to store data before changing it with the next data bit. This process is controlled by the *Ck* pulse. The data are transferred from master to slave as the clock switches from high to low (i.e., from 1 to 0). Then, the next data bit is entered to the master when the *Ck* switches to high state.



Truth table

D_n	Q_{n+1}
0	0
1	1

FIG. 2.42 D-type FF circuit (A) and truth table (B).

This transition to the high state enables data to enter from either the D input or the preset input. During the high Ck time interval, the slave is inhibited from receiving input and changing its output.

When a shift register is made from DFFs, the individual DFFs are connected in a cascade such that the Q output of the first is the D input to the second and similarly for all of the other DFFs in an M-bit register. The data entry into a register formed in this way can be (1) synchronous parallel input with serial output (from the last stage), (2) asynchronous input with serial output, or (3) synchronous serial input with serial output. Parallel output can be achieved by inverting the $\bar{Q}$ output from each stage and connecting the inverted outputs to an M-bit bus.

Fig. 2.42 includes a truth table for the DFF circuit. In this truth table, the Q_{n+1} is the Q output after the nth clock pulse has switched high to low. The symbol D_n represents the nth input on the D lead during the nth clock transition to the high state.

Parallel inputs to a DFF register are presented to the preset input during the period in which the Ck is in the low state (i.e., 0). During this interval, the D input to the master is inhibited.

TIMER CIRCUIT

An example circuit that incorporates an R-SFF is one of the relatively highly used ICs that is implemented in multiple example circuits explained in Chapter 5 and is known as the 555 timer circuit. It has many applications involving its three modes of operation: bistable, monostable, and astable multivibrator circuits. In digital vehicular electronic systems, it is capable of generating pulses to identify the time of occurrence of certain events in electrical waveforms (e.g., threshold crossing of voltages). In the astable mode, it is connected as an oscillator to a capacitor in a way that yields a measurement of the capacitance of the capacitor.

Fig. 2.43 is a block diagram of a 555 timer IC that incorporates circuits that have already been explained in this chapter.

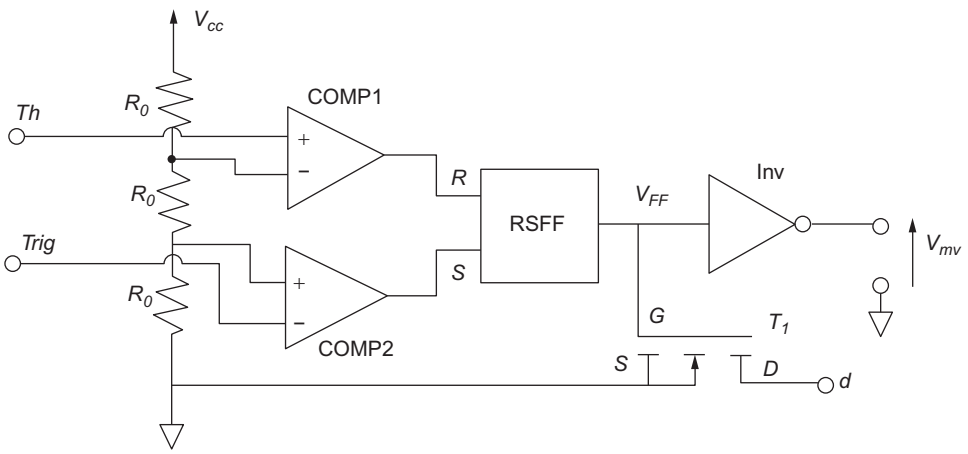


FIG. 2.43 555 Timer IC block diagram.

The circuit components depicted in this block diagram include the following:

COMP1 and COMP2 analog comparators
R-SFF, RS flip-flop
Inv, logical inverter
T_1, *n*-channel enhancement-type FET

The signals within input and output are the following:

V_{CC}, supply voltage
V_{FF}, R-SFF output voltage
V_{mV}, timer output voltage
Th, threshold input
Trig, trigger input
d, discharge lead
CTRL, control input

The circuit includes three identical resistors having resistance R_o such that the voltage at the input to COMP1 is 2/3 V_{CC} and the voltage at the + input of COMP2 is 1/3 V_{CC}. The commercial versions of the 555 are fabricated with either bipolar or CMOS technology as explained earlier in this chapter.

The particular modes of operation and the output waveforms are determined by the connection of the 555 to external circuitry. We illustrate this point with a specific application and the associated external circuitry. This exemplary application involves the generation of an output pulse with a specific duration (T_{ZCD}) in response to the zero voltage crossing of an a-c voltage $V(t)$. This circuit that is called a ZCD is depicted in Fig. 2.44.

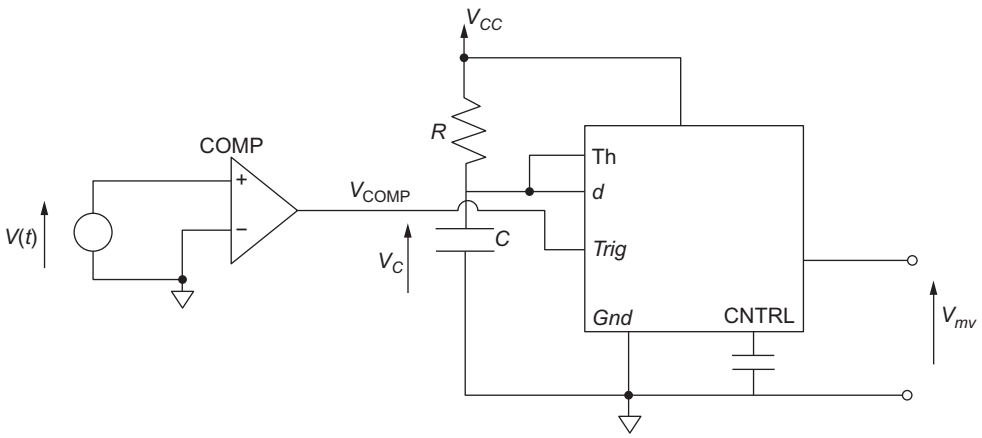


FIG. 2.44 ZCD example circuit.

The comparator is designed such that the binary-valued output voltage V_{comp} is high whenever $V(t) > 0$ and low whenever $V(t) < 0$. The timer circuit is activated for each high-to-low transition provided that the capacitor has been discharged and $V_C = 0$ at this transition time. The capacitor voltage $V_C = 0$ because V_{FF} is high causing T_1 to be in saturation with the capacitor discharged to 0 by the R_{on} of T_1. The zero crossing of $V(t)$ triggers the timer and resets V_{FF} low such that T_1 is essentially an open circuit and the capacitor begins charging through R. The capacitor voltage satisfies the following equation from the trigger event until the $V_C = 2/3\ V_{CC}$:

$$RC\frac{dV_C}{dt} + V_C = V_{CC}$$

The solution to this differential equation can be found using the Laplace transform methods of Appendix A. It can be shown that $V_C(t)$ is given by

$$\begin{aligned} V_C(t) &= V_{CC}\left(1 - e^{-(t-t_k)/\tau}\right) \quad t_k \le t < t_k + T_{ZCD} \\ &= 0 \qquad\qquad\qquad\qquad\quad t_k + T_{ZCD} < t \le T_{k+1} \end{aligned}$$

where t_k = time of kth zero crossing, T_{ZCD} = time from t_k until $V_C = 2/3\ V_{CC}$, and $\tau = RC$.

The value for T_{ZCD} can be found from the condition:

$$\begin{aligned} V_C(t_k + T_{ZCD}) &= \frac{2}{3}V_{CC} \\ &= V_{CC}\left(1 - e^{\frac{-T_{ZCD}}{\tau}}\right) \end{aligned}$$

This equation can be rewritten to find T_{ZCD}:

$$e^{\frac{-T_{ZCD}}{\tau}} = \frac{1}{3}$$

for which $T_{ZCD} = \ell n(3)RC$.

When $V_C = 2V_{CC}/3$, the R-SFF is set high such that T_1 is in saturation that discharges V_C (almost instantly) and the timer is ready for the next event.

The output voltage V_{mv} is a rectangular pulse of duration T_{ZCD} for each zero crossing:

$$\begin{aligned} V_{mV} &= V_H \quad t_k \le t \le t_k + T_{ZCD} \quad k = 1, 2, 3, \cdots \\ &= V_L \quad t_k + T_{ZCD} \le t \le t_{k+1} \end{aligned}$$

where V_H = high voltage level of the pulse, V_L = low voltage level of the pulse, and $V_L \simeq 0$.

Other applications of the 555 timer circuit are presented in other chapters of this book. The measurement of the capacitance of a capacitive sensor is obtained using the timer in an astable oscillator mode. The frequency of oscillation and/or the high state of the output yields the data in an appropriate form for the particular digital vehicular electronic system as explained in detail for the relevant application (e.g., flex fuel sensor section of Chapter 5).

SYNCHRONOUS COUNTER

Fig. 2.45 shows a four-stage synchronous counter that incorporates 4 J-K FFs. It is synchronous because all stages are triggered at the same time by the same clock pulse (Clk).

It has four stages; therefore, it counts 2^4 or 16 clock pulses before it returns to a starting state. The timed waveforms appearing at each Q output are also shown in Fig. 2.45B. The waveforms of Fig. 2.45B indicate how such circuitry can be used for counting, for generating other timing pulses, and for determining timed sequences. A synchronous counter can also be implemented with D-flip-flop circuits.

REGISTER CIRCUITS

One of the most important circuits for building a computer is formed using multiple **JK**-type circuits (or DFFs) as depicted in Fig. 2.46.

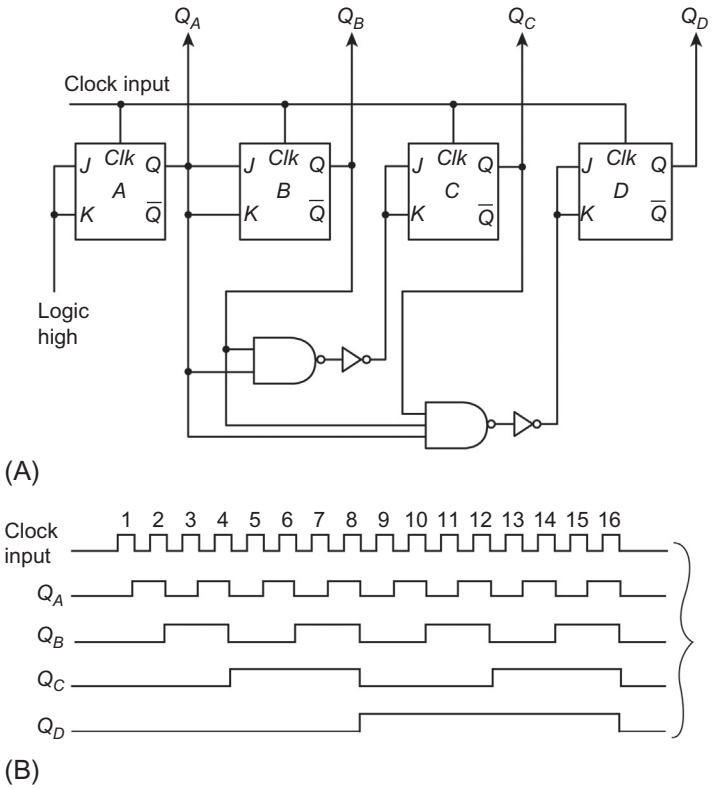


FIG. 2.45 Synchronous counter circuit.

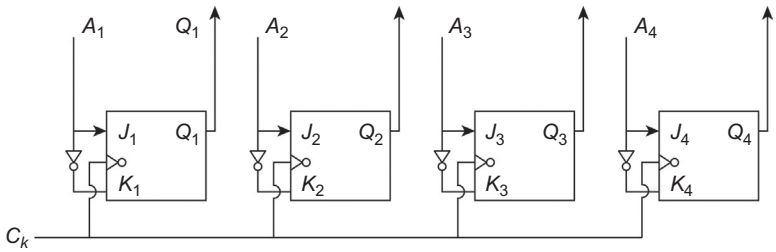


FIG. 2.46 Storage register circuit example.

Such circuits are known as registers. Fig. 2.46 is shown as a 4-bit device simply to explain the operation of the register. Practical register circuits used in computers normally have many more **JK**-type circuits (or DFFs) than depicted here. There are many classes of register depending upon usage in the computer and the operation performed. These operations include

1. storage of data,
2. shift right or left, and
3. synchronous counting.

When used as a storage register or memory, the data to be shared (which, e.g., might be digital input data from a sensor and (A to D) converter), the output of a full adder, or the contents of another register are provided to the J inputs. The data $A_4A_3A_2A_1$ are transferred to the corresponding Q output at the clock time. It will remain there until new data and a clock pulse are provided to the computer.

SHIFT REGISTER

Another very important circuit that is implemented with a **JK** type of circuit (or its functional equivalent, e.g., DFF) is a so-called shift register. Fig. 2.47A depicts a simple 4-bit shift register having the capability of so-called parallel load in which all data bits are transferred simultaneously into the corresponding flip-flop with control C high at the clock pulse.

This latter register has two modes of operation: (1) transfer of data ($A_4A_3A_2A_1$) into the register with control C high and (2) shift right or left with control C low. The shift operation refers to changing the position of a bit in a digital number to a higher (shift left) or lower (shift right) position. Consider an M-bit binary number N,

$$N = A_M \ldots A_m \ldots A_1$$

where A_M is the most significant bit. Shift right or left is a synchronous operation, in that it is associated with clock time t_n. A shift left at time t_{n+1} means that A_m becomes A_{m+1}. Similarly, a shift-right operation means that the A_m bit at t_n becomes A_{m-1} at t_{n+1}. For the example circuit of Fig. 2.47A, the operation is a shift left. The same circuit could be made into a shift-right operation by reversing the data order; that is, the most significant bit (A_M) is entered into **JK**1, then in a decreasing order until the least significant is loaded into **JK**4.

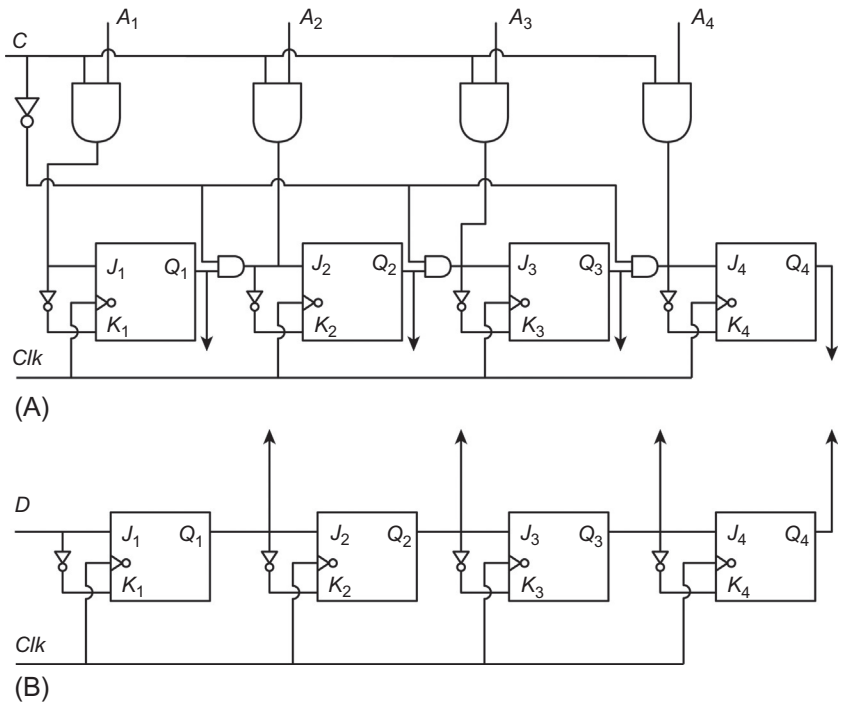


FIG. 2.47 4-Bit shift register circuit example.

The circuit of Fig. 2.47 depicts a data load operation that is said to be parallel in which all bits are loaded synchronously. It is also possible to load the data serially in which the data bits are entered in sequence, beginning with either the most or the least significant bit as depicted by the circuit of Fig. 2.47B. At each clock cycle, the bits are shifted one position right or left depending upon circuit configuration and/or program control.

The data to be entered into this register consist of a sequence of bits A_m that are synchronous with the clock. At each clock pulse, the corresponding data bit (i.e., either a 0 or a 1) is presented along the data line (D in Fig. 2.47B). Prior to clock time t_n, the previous data bit (i.e., at clock time t_{n-1}) is stored in Q_1 and the data at time t_{n-2} in Q_2, etc. Each of these bits is presented to the J input of the next flip-flop. The result of the circuit operation is a shift to the next bit position for each clock pulse.

There are additional categories of shift register that perform various logical operations on binary numbers or data depending upon how the circuit treats the least and most significant bits. A shift-right operation drops the A_1 bit and shifts a 0 into the A_M bit location. The reverse is true for a shift-left operation.

It is also possible to connect the least and most significant bit locations; such an operation is termed "rotate right or left." For a rotate-right operation, the least significant bit at t_n becomes the most significant bit at t_{n+1}. Similarly, the reverse is true for a rotate-left operation. Such operations can be implemented either with dedicated (hard-wired) circuits or more commonly through program-controlled switching.

Digital electronic systems send and receive signals made up of ones and zeros in the form of codes. The digital codes represent the information that is moved through the digital systems by the digital circuits. Digital systems are made up of many identical logic gates and flip-flops interconnected to do the function required of the system. As a result, digital circuits are ideal for implementation in integrated circuits (ICs) because all components can be made at the same time on a small silicon area.

DIGITAL INTEGRATED CIRCUITS

One of the important consequences of integrated circuit (IC) technology progress is that digital circuits have become available (in IC form) as electronic systems or subsystems; that is, the functional capability of digital circuits in single IC packages, or chips, has spectacularly increased in the past decades. One of the important digital systems that were available in the early days of IC was the arithmetic and logic unit (ALU), which is no longer used as a stand-alone device in a digital system. In contemporary ICs, the greatly expanded functional equivalent of the ALU is a part of larger ICs (e.g., microprocessor (MPU))

Fig. 2.48 is a sketch of a typical ALU showing the various connections.

For illustrative purposes of digital circuit that incorporates, many of the logic circuits presented above a (now obsolete) 4-bit ALU are presented to illustrate the wide range of arithmetic or logical operations that are possible with only four control lines. This 4-bit ALU has the capability of performing 16 possible logical or arithmetic operations on two 4-bit inputs, *A* and *B*. Table 2.3 summarizes these various operations using the logical notation of Boolean algebra explained earlier in this chapter. The functional equivalent of this ALU (implemented with many more than 4-bits) that is now termed a

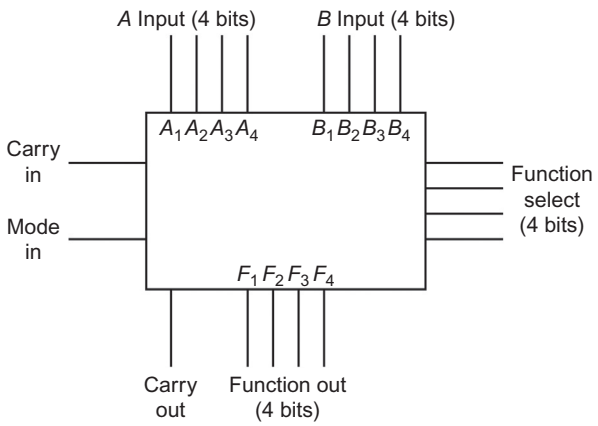


FIG. 2.48 ALU symbol.

Table 2.3 Arithmetic Logic Functions

Select S input	Logic Function, $M=1$	Arithmetic Function, $M=0$	
		$C_n=1$	$C_n=0$
0000	$F=\bar{A}$(NOT)	$F=A$	$F=A$ Plus 1
0001	$F=\overline{A+B}$(NOR)	$F=A+$B	$F=(A+B)$ Plus 1
0010	$F=\bar{A}\cdot B$	$F=A+\bar{B}$	$F=(A+\bar{B})$ Plus 1
0011	$F=0$	$F=$Minus 1 (2's Complement)	$F=0$
0100	$F=\overline{AB}$(NAND)	$F=A+A\cdot\bar{B}$	$F=(A+A\cdot\bar{B})$ Plus 1
0101	$F=\bar{B}$(NOT)	$F=(A+B)$ Plus $A\cdot\bar{B}$	$F=(A=B)$ Plus $A\cdot\bar{B}+1$
0110	$F=AB+B\cdot\bar{A}$ (Exclusive OR)	$F=(A-B)$ Minus 1	$F=A$ Minus B
0111	$F=A\cdot\bar{B}$	$F=A\cdot\bar{B}$ Minus 1	$F=\overline{AB}$
1000	$F=\bar{A}+B$ (Implication)	$F=A+AB$	$F=(A+B)$ Plus 1
1001	$F=\overline{AB}+AB$(NOT exclusive OR)	$F=A+B$	$F=(A+B)$ Plus 1
1010	$F=B$	$F=(A+\bar{B})$ Plus AB	$F=(A+\bar{B})$ Plus $A+1$
1011	$F=AB$ (AND)	$F=AB$ Minus 1	$F=AB$
1100	$F=1$	$F=A$ Plus A[a]	$F=(A+A)$ Plus 1
1101	$F=A+\bar{B}$	$F=(A+B)$ Plus A	$F=(A+B)$ Plus $A+1$
1110	$F=A+B$ (OR)	$F=(A+\bar{B})$ Plus A	$F=(A+\bar{B})$ Plus $A+1$
1111	$F=A$	$F=A$ Minus 1	$F=A$

[a] *Each bit is shifted to the next more significant position.*

"central processing" unit (CPU) is a component of the most important single-chip IC for all digital electronic systems: the MPU. Table 2.3 is included in this chapter to illustrate, in an exemplary manner, the combination of logical and arithmetic operations, which can be performed with a binary signal on the function select and MODE inputs.

THE MPU

Perhaps, the single most important digital IC to evolve has been the MPU. This important device, incorporating hundreds of thousands of transistors in an area of about ¼ in. square or less, has truly revolutionized digital electronic system development. A MPU is the operational core of a microcomputer and digital control system and has broad applications in automotive electronic systems.

The MPU incorporates a relatively complicated combination of digital circuits including an ALU or CPU, registers, and decoding logic. A representative (though simplified) MPU block diagram is shown in Fig. 2.49. The double lines labeled "bus" are actually sets of conductors for carrying digital data throughout the MPU. A block diagram of a MPU/microcontroller with more detail is given in Chapter 3. Common IC MPUs use 8, 16, or 32 (or higher multiples of 8) conductor buses.

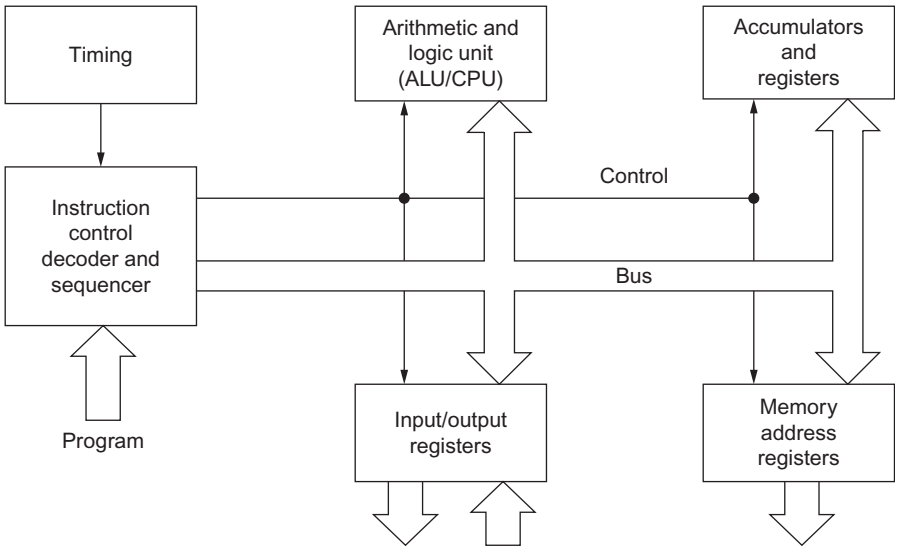


FIG. 2.49 Simplified microprocessor block diagram.

Early 21st century automobiles incorporate dozens of MPUs that are applied for a variety of uses from advanced power train control to simple tasks such as automatic seat and side-view mirror positioning. A MPU by itself can accomplish nothing. It requires a program (typically stored in a memory circuit to control its operation). Typically, it also requires additional external digital circuitry as explained in the next chapter. One of the tasks performed by the external circuitry is to provide the program consisting of instructions in the form of digitally encoded electric signals. By way of illustration, an 8-bit MPU operates with 8-bit instructions. There are 2^8 (or 256) possible logical combinations of 8 bits, corresponding to 256 possible MPU instructions, each causing a specific operation. A complete summary of these operations and the corresponding instructions (called microinstructions) is beyond the scope of this book. A few of the more important instructions are explained in the next chapter, which further expands the discussion of this important device.

CHAPTER

MICROCOMPUTER INSTRUMENTATION AND CONTROL

3

CHAPTER OUTLINE

Microcomputer Fundamentals 91
- Digital Computer 91
- Parts of a Computer 91
- Microcomputers Versus Mainframe Computers 92
- Programs 93

Microcomputer Tasks 93

Microcomputer Operations 94
- Buses 94
- Memory-Read/Write 94
- Timing 96
- Addressing Peripherals 96

CPU Registers 97
- Accumulator Register 98
- Condition Code Register 98
- Microprocessor Architecture 100

Reading Instructions 100
- Initialization 102
- Operation Codes 102
- Program Counter 102
- Branch Instruction 104
- Jump Instruction 105
- Jump-to-Subroutine Instruction 105

Example Use of a Microcomputer 107
- Buffer 107

Programming Languages 108
- Assembly Language 109
- Logic Functions 109
- Shift 110
- Programming the AND Function in Assembly Language 111
- Masking 112

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-12-810434-7.00003-X

Copyright © 2017 Elsevier Inc. All rights reserved.

Shift and AND 113
Use of Subroutines 113
Microcomputer Hardware 114
Central Processing Unit 114
Memory: ROM 115
Memory: RAM 115
I/O Parallel Interface 115
Digital-to-Analog Converter 116
Analog-to-Digital Converter 118
Sampling 120
Polling 121
Interrupts 121
Vectored Interrupts 122
Microcomputer Applications in Automotive Systems 122
Instrumentation Applications of Microcomputers 124
Digital Filters 126
Microcomputers in Control Systems 128
Closed-Loop Control System 128
Limit-Cycle Controller 128
Feedback Control Systems 128
Table Lookup 130
Multivariable and Multiple Task Systems 132
AUTOSAR 133

The technological advances in all levels of digital computers, including those found in vehicular applications, have been and continue to be rapid. A detailed description of the present state of this technology is beyond the scope of this book. The goal of this chapter is to present models for the basic structure and operation of devices that are euphemistically termed "microcomputers." The level of description is intended to provide a model for such devices that will assist in the explanation of their widespread vehicular electronic applications in later chapters. Each vehicular digital electronic system presented throughout the remainder of the book is illustrated by a block diagram in which the digital processing operation is represented by a "microcomputer" based subsystem.

This chapter describes microcomputers and explains how they are used in instrumentation and control systems. Topics include microcomputer fundamentals, microcomputer equipment, microcomputer inputs and outputs, computerized instrumentation, and computerized control systems. The specific automotive applications of microcomputers are explained in later chapters.

Individual vehicular digital electronic systems vary in configuration and operation. However, the term "microcomputer" is used here to have a single word for characterizing the major component in each system. The justification for using this terminology is based on the essential similarity in the mechanisms by which the various arithmetic and logical operations are performed in a typical vehicular digital system to those of a computer in general. This will be evident in each of the electronic systems discussed throughout the remainder of this book.

Much of the material in this chapter is based on what was traditionally the model of a microcomputer used by the person(s) writing the code or software that controlled its operation. Although this

level of programming is presently almost completely obsolete, it is given here simply as a way of explaining the detailed steps taken by the microcomputer (under program control) in performing the various arithmetic and logical operations.

MICROCOMPUTER FUNDAMENTALS

DIGITAL COMPUTER

In digital computer-based systems, physical variables are represented by a numerical equivalent using a form of the binary (base 2) number system. In the previous chapter, it was shown that transistor circuits can be constructed to have one of two stable states: saturation and cutoff. These two states can be used to represent a 0 (zero) or a 1 (one) in a binary number system. To be practically useful, there must be groups of such circuits that are arranged in the form of a place position, binary number system.

As will be shown in subsequent chapters, digital automotive electronic systems are implemented with microprocessors in combination with other components to form a type of special-purpose digital computer (as opposed to a general-purpose computer, e.g., a laptop) or a form of digital controller having a structure very much like a computer. The later discussion of automotive digital systems can perhaps be best understood following a brief discussion of digital computer technology. In any application, including automotive, a computer performs various operations on the data. To explain the operation of a digital computer, it is helpful first to explain the operation of its various components.

PARTS OF A COMPUTER

A few of the parts of a general-purpose digital computer are shown in Fig. 3.1.

This figure is presented only as an illustration of a representative digital computer. The actual configuration of any given computer such as might be used in an automotive application is determined by the specific tasks it is to perform. For example, an engine control computer (as described in Chapter 6) would not include disk drive, keyboard, printer, or monitor.

The *central processing unit* (CPU) is the processor that is the principle operating part of the system. When made separately in an integrated circuit, it is called a microprocessor. This is where all of the arithmetic and logical decisions are made and is the calculator part of the computer. Automotive digital computers are implemented with one or more microprocessors. A more detailed description of a microprocessor is given later in this chapter.

The *memory* holds the program and data. The computer can change the information in memory by writing new information into memory, or it can obtain information contained in memory by reading the information from memory. Each memory location has a unique address that the CPU uses to find the information it needs.

Information (or data) must be put into the computer in a form that the computer can read, and the computer must present an output in a form that can be read by humans or used by other computers or digital systems. The input and output devices, called I/O, perform these conversions. In a general-purpose computer, peripherals are devices such as keyboards, monitors, magnetic disk units, modems, and printers. The arrows on the interconnection lines in Fig. 3.1 indicate the flow of data.

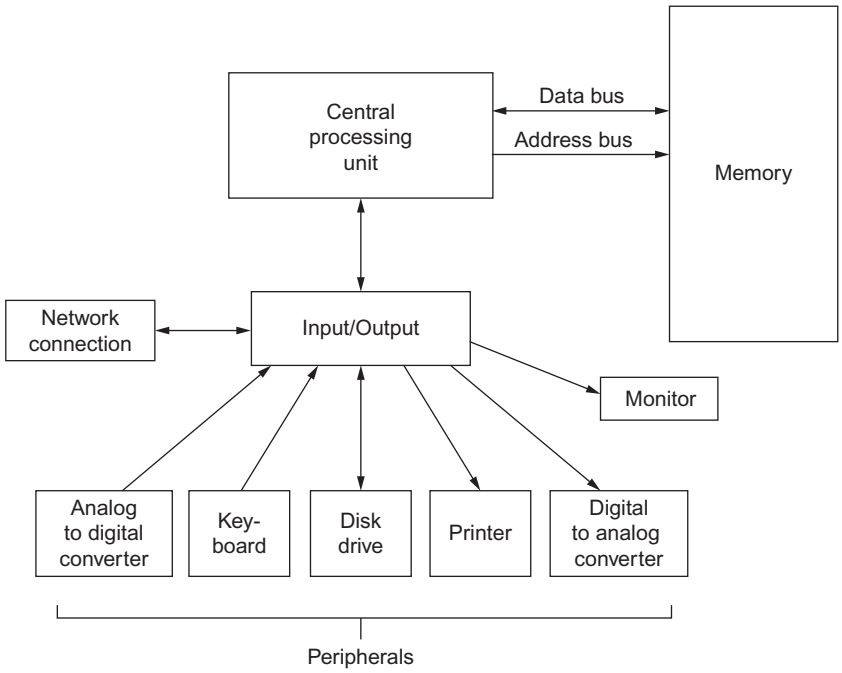


FIG. 3.1 General-purpose computer block diagram.

MICROCOMPUTERS VERSUS MAINFRAME COMPUTERS

With this general idea of what a computer is, it is instructive to compare a general-purpose mainframe computer with a microcomputer. A microcomputer is just a small computer, typically thousands of times smaller than the large, general-purpose mainframe computers used by banks and large corporations or military or government labs. At the upper end of the spectrum of computers are the very large scientific computers, many of which are made up of large numbers of smaller computers operating in parallel. These large computers perform floating-point operations (FLOPS) at something of the order of a 10^{12} FLOPS. On the other hand, a typical automotive digital system is of the scale of a microcomputer. Depending on feature content, a typical automobile will incorporate dozens of microcomputers. A typical mainframe computer is capable of multiple billions of arithmetic operations per second (additions, subtractions, multiplications, and divisions). A microcomputer can perform multiple millions operations per second. As important for mathematical calculations as the speed of the operation is the accuracy of the operation.

The precision and accuracy of calculations performed by a digital computer are functions of the number of bits used to represent numerical values. In order to give a numerical frame of reference, recall that an 8-bit binary number can represent 256 decimal numbers. If (as in the case of many automotive digital systems) the number is to have a sign (i.e., + or –), then only numbers from −127 to 128 can be so represented.

PROGRAMS

A *program* is a set of instructions organized into a particular sequence to do a particular task. The first computers were little more than fancy calculators. They did only simple arithmetic and made logical decisions. They were programmed (given instructions) by punching special codes into a paper tape that was then read by the machine and interpreted as instructions. A program containing thousands of instructions running on an early model machine might require yards of paper tape. The computer would process the program by reading an instruction from the tape, performing the instruction, reading another instruction from the tape, performing the instruction, and so on until the end of the program. Reading paper tape was a slow process compared with the speed with which a modern computer can perform necessary operations and functions. In addition, the tape had to be fed through the computer each time the program was run, which was cumbersome and allowed for the possibility of the tape wearing and breaking.

To increase computational efficiency, a method was invented to store programs inside the computer. The program is read into a large electronic memory made out of thousands of data latches (flip-flops), one for each bit, that provide locations in which to store program instructions and data. Each instruction is converted to binary numbers with a definite number of bits and stored in a memory location. Each memory location has an address number associated with it. The computer reads the binary number (instruction or data) stored in each memory location by going to the address of the information called for in the instruction being processed. When the address for a particular memory location is generated, a *copy* of its information is transferred to the computer. (Depending on the instruction, the original information might stay in its location in memory, while the memory is being read.)

The computer can use some of its memory for storing programs (instructions) and other memory for storing data. The program or data can be easily changed simply by loading in a different program or different data. The stored program concept is fundamental to all modern electronic computers.

Computers have memory components of two major types: (1) read-only memory (ROM) and (2) random-access memory (RAM) that could also be called read/write since data/program steps can be written (i.e., placed in memory for temporary storage) or read (obtained electronically from the memory). In automotive computers or digital subsystems, the program is typically stored in a ROM. Both types of memory are discussed in detail later in this chapter.

MICROCOMPUTER TASKS

A suitably configured microcomputer can potentially perform any automotive control or instrumentation task. For example, it will be shown in a later chapter that a microcomputer can be configured to control fuel metering and ignition for an engine along with many other tasks. The microcomputer-based engine control system has much greater flexibility than the earliest electronic engine control systems, which, typically, used elementary logic circuits and analog circuits. For these early systems, changes in the performance of the control system required changes in the circuitry (hardware). With a microcomputer performing the logical functions, most changes can be made simply by altering certain parameters used in the associated algorithms; that is, the software data are changed rather than the hardware (logic circuits). This makes the microcomputer a very attractive building block in any digital system.

Microcomputers can also be used to replace analog circuitry. Special interface circuits can be used to enable a digital computer to input and output analog signals (this will be discussed later). The important point here is that microcomputers are excellent alternatives to hardwired (dedicated) logic and analog circuitry that is interconnected to satisfy a particular design.

In the subsequent portions of this chapter, both the computer hardware configuration and programs (software) are discussed. Because these two aspects of computers are so strongly interrelated, it is necessary for the following discussion to switch back and forth between the two.

In a modern personal computer, the program instructions and data are stored electronically in register-type circuits as described in Chapter 2. Recall that a register circuit consists of a sequence of flip-flop (or similar) binary circuits. Modern register-type circuits are extremely fast circuits having the capability of storing data that can be inserted or retrieved in a small fraction of a microsecond during the execution of a program. In addition, programs and data can be stored indefinitely via magnetic or optical disk media.

MICROCOMPUTER OPERATIONS

Recall the basic computer block diagram of Fig. 3.1. The CPU obtains data from memory (or from an input device) by generating the address for the data in memory. The address with all its bits is stored in the CPU as a binary number in a temporary data latch-type memory called a *register* (see Chapter 2). The outputs of the register are sent at the same time over multiple wires to the computer memory and peripherals.

BUSES

As shown in Fig. 3.2, the group of wires that carries the address is called the address bus (AB). (The word *bus* refers to groups of wires that form a common path to and from various components in the computer.)

For example, consider a computer having an address register 32 bits; these bits enable the CPU to access 2^{32} memory locations. In a microcomputer, each memory location usually contains multiples of 8 bits of data. A group of 8 bits is called a *byte*, and a group of 16 bits is sometimes called a *word*.

Data are sent to the CPU over a data bus (DB) (Fig. 3.2). The DB is slightly different from the AB in that the CPU uses it to obtain (*read*) data from memory or peripherals and to send (*write*) data to memory or peripherals. Signals on the AB originate only at the CPU and are sent to devices attached to the bus. Signals on the DB can be either data to/from memory or other registers or inputs to or outputs from the CPU that are sent or received at the CPU by the data register. In other words, the DB is a two-way communication bus, while the AB is a one-way communication bus. In addition to the address and DB, there are sets of or wires that are called the control bus (CB). It is this CB that sends the binary signals to the components involved in any operation at the appropriate time to cause them to perform the specific operation.

MEMORY-READ/WRITE

The CPU always controls the direction of data flow on the DB because, although it is bidirectional, data can move in only one direction at a time. The CPU provides a special read/write control (R/W) signal (Fig. 3.2) that activates circuits in the memory, which determine the direction of the data flow. For example, when the read/write (R/W) line is high, the CPU transfers information from a memory location to the CPU.

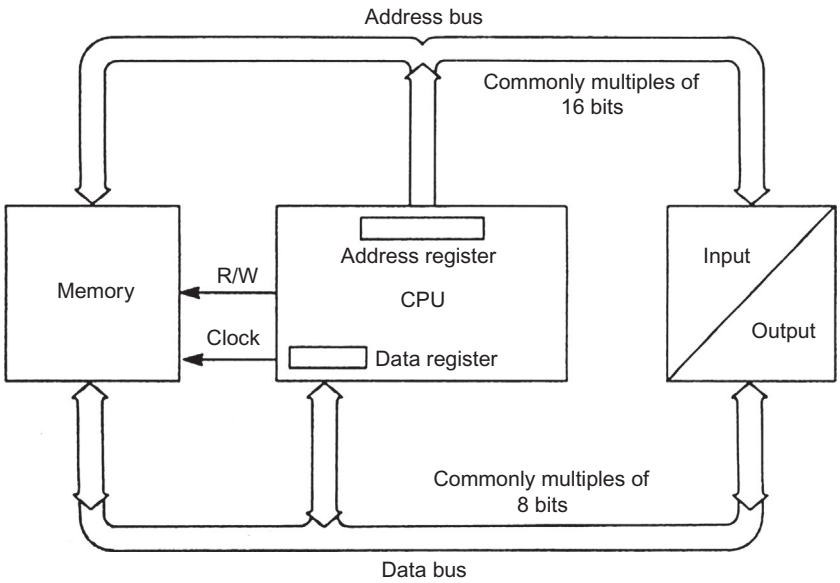


FIG. 3.2 Computer buses.

The timing diagram for a memory-read operation is shown in Fig. 3.3.

Suppose the computer has been given the instruction to read data from memory location number 10. To perform the read operation, the CPU raises the R/W line to the high-level to activate memory circuitry in preparation for a read operation. Almost simultaneously, the address for location 10 is placed on the AB ("address valid" in Fig. 3.3). The number 10 in 16-bit binary (0000 0000 0000 1010) is sent

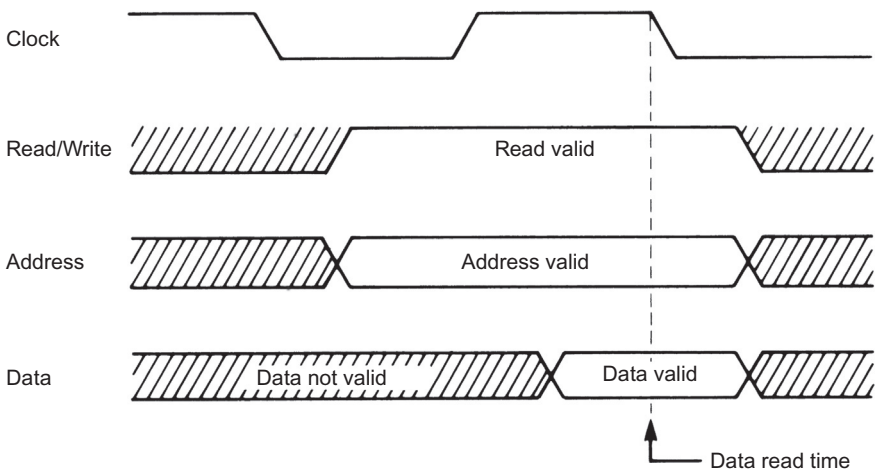


FIG. 3.3 Read/write timing.

to the memory in the AB. The binary electric signals corresponding to 10 operate the specific circuits in the memory to cause the binary data at that location to be placed on the DB. The CPU has an internal register that is activated during this read operation to receive and store the data. The data are then processed by the CPU during the next cycle of operation according to the relevant instruction.

A similar operation is performed whenever the CPU is to send data from one of its internal registers to memory, which is a "write" operation. In this case, the R/W line will be set at the logical level opposite to the read operation (i.e., low in this example). During the write operation, the data to be sent are placed on the DB at the same time the destination address is placed on the AB. This operation will transfer data from the CPU source location to the destination, which could be a memory location in RAM or could be an external device (as will be explained later).

TIMING

A certain amount of time is required for the memory's address decoder to decode which memory location is called for by the address and also for the selected memory location to transfer its information to the DB. To allow time for this decoding, the processor institutes an appropriate time delay before receiving the information requested from the DB. Then, at the proper time, the CPU opens the logic gating circuitry between the DB and the CPU data register so that the information on the bus from memory location (e.g., 10) is latched into the CPU. During the memory-read operation, the memory has temporary control of the DB. Control must be returned to the CPU, but not before the processor has read in the data. The CPU provides a timing control signal, called the *clock*, which regulates the memory internal timing to take and release control of the DB.

Refer again to Fig. 3.3. Notice that the read cycle is terminated when the clock goes from high to low during the time that the read signal is valid. This is the signal generated by the CPU at the end of the read cycle at which time the DB can be released for other operations.

The bus timing signals are very important for the reliable operation of the computer. However, they are built into the design of the machine and, therefore, are under machine control. As long as the machine performs the read and write operations correctly, the programmer can completely ignore the details of the bus timing signals and concentrate on the logic of the program. In any microprocessor-based electronic system, there is an oscillator running at frequency f_{osc} that establishes all timing operations. Typically, submultiples of the master oscillator f_n are obtained via frequency division where $f = f_{osc}/n$ (n = integer). For automotive control operations, real-time calculations are required (as explained later). The calculation speed for these operations is an increasing function of f_{osc}. Frequency division such as is required for many computer operations is readily accomplished using sequences of J-K (or equivalent) flip-flop circuits (see Chapter 2). Frequency division of this type can be accomplished with a binary counter circuit, for example, as explained in Chapter 2. Division of the oscillator frequency by an integer n is achieved by taking the output of the nth bit of a binary counter where $n = 1$ is the least significant bit.

ADDRESSING PERIPHERALS

The reason for distinguishing between memory locations and peripherals is that they perform different functions. Memory is a data storage device, while peripherals are input/output devices. However, many microcomputers address memory and peripherals in the same way because they use a design called memory-mapped I/O (input/output). With this design, peripherals, such as data terminals, are

equivalent to memory to the CPU so that sending data to a peripheral is as simple as writing data to a memory location. In systems where this type of microcomputer has replaced some digital logic, the digital inputs enter the computer through a designated memory slot. If outputs are required, they exit the computer through another designated memory slot.

The relatively efficient means of input/output of data via memory map facilitates operations such as digital filtering of sampled analog signals or for control operations as explained in Appendix B. In such cases, the A/D converter providing the sampled input would have a specific memory location. Similarly, any D/A or ZOH (see Chapter 2) would also have a designated memory location. Such memory-mapped I/O is particularly helpful for speeding computer operations in discrete time control for automotive systems.

CPU REGISTERS

The programmer uses a different model (a programming model) of the microprocessor used in a system compared with the hardware designer. Traditionally, this model would show the programmer which registers in the CPU are available for program use and what function the registers perform. Although in contemporary design this level of programming is no longer performed, it is presented here to illustrate basic operations in a microcomputer. Fig. 3.4 shows a programming model microprocessor for an 8-bit microcomputer (that is, of course, now obsolete).

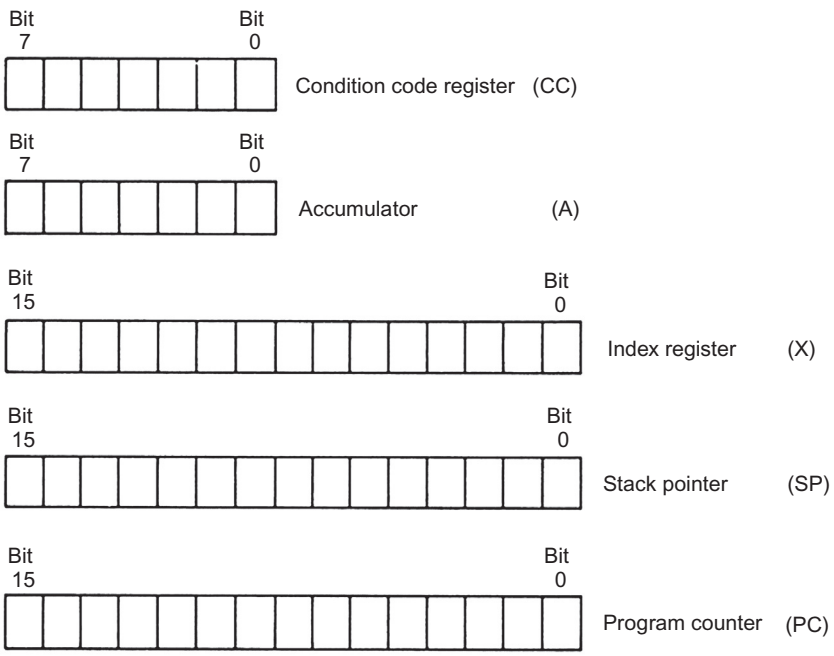


FIG. 3.4 Selected CPU registers.

The 8-bit example is presented solely to simplify the explanation of the operation of these registers. In any practical computer system, these registers have many more bits. In the example, the computer has two 8-bit registers and three 16-bit registers. The 16-bit registers are discussed later; the 8-bit registers are discussed here. Circuit implementation of computer registers is explained in Chapter 2.

ACCUMULATOR REGISTER

One of the 8-bit registers in the exemplary CPU is an *accumulator*, which is a general-purpose register that is used for arithmetic and logical operations. The accumulator can be loaded with data from a memory location, or its data can be stored in a memory location. The number held in the accumulator can be added to, subtracted from, or compared with another number from memory. The accumulator has traditionally been the basic work register of the exemplary computer. Traditionally, it has been called the *A register*.

CONDITION CODE REGISTER

The other 8-bit register, the *condition code* (CC) *register* (also called *status register*), indicates or flags certain conditions that occur during accumulator operations. Rules are established in the design of the microprocessor so that a 1 or 0 in the bit position of the CC register represents specific conditions that have occurred in the last operation of the accumulator. The bit positions and rules are shown in Fig. 3.5A. One bit of the CC register indicates that the A register is all zeros. Another bit, the carry

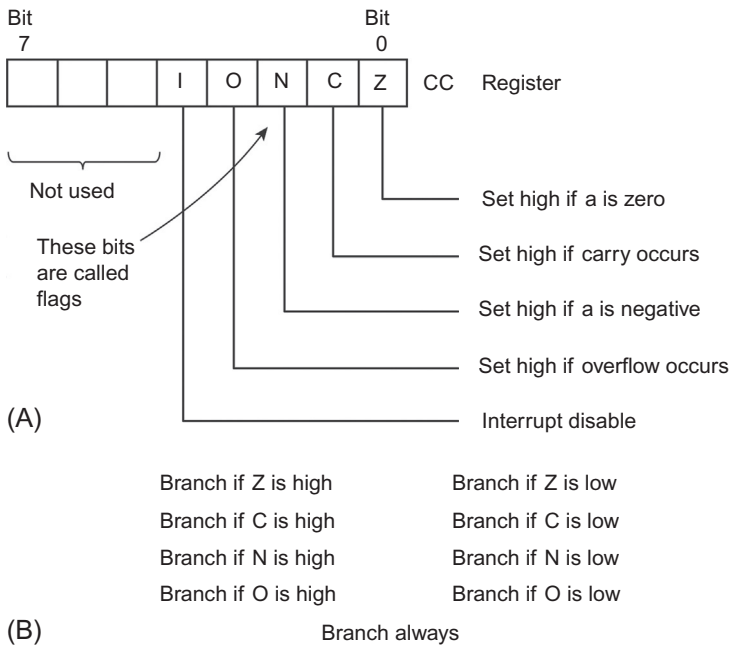


FIG. 3.5 Condition code register bits. (A) Bit functions and (B) branch instructions.

bit, indicates that the last operation performed on the accumulator caused a carry to occur. The carry bit acts like the ninth bit of the accumulator (in this 8-bit example). As an illustration of carry from the most significant bit in an accumulator register and the effect on the CC register, the result of the addition of 1 to decimal 255 is given below:

Decimal	Binary
255	input 11111111
+1	add +1
256	sum 00000000
	carry 1

The 8 bits in the accumulator are all zeros, but the carry bit being set to a 1 (high) indicates that the result is actually not 0, but 256. Such a condition can be checked by examining the CC register carry bit for a 1. For the following discussion, we remain with the simplified 8-bit example registers. In general, the microprocessor will operate with N bits.

The CC register also provides a flag that, when set to a 1, indicates that the number in the accumulator is negative. Most microcomputers use a binary format called *two's complement notation* for doing arithmetic. In two's complement notation, the leftmost bit indicates the sign of the number. Since one of the 8 bits (of the example) is used for the sign, 7 bits (or $N-1$ in general) remain to represent the magnitude of the number. The largest positive number that can be represented in two's complement with 8 bits is +127 (or $2^{(N-1)}-1$ for N bits, in general); the largest negative number is -128 (or $2^{(N-1)}$ in general). Since the example accumulator is only 8 bits wide, it can handle only 1 byte at a time. However, by combining bytes and operating on them one after another in time sequence (as is done for 16-bit arithmetic), the computer can handle very large numbers or can obtain increased accuracy in calculations. Handling bits or bytes one after another in time sequence is called *serial operation.*

Branching

The CC register traditionally provided programmers with status indicators (the flags) that enabled them to monitor what happened to the data as the program executed the instructions. The microcomputer has special instructions that allow it to go to a different part of the program. Bits of the CC register are labeled in Fig. 3.5A. Typical branch-type instructions are shown in Fig. 3.5B.

Program branches are either conditional or unconditional. Eight of the nine branch instructions listed in Fig. 3.5B are conditional branches; that is to say, the branch is taken only if certain conditions are met. These conditions are indicated by the CC register bit as shown. The branch-always instruction is the only unconditional branch. Such a branch is used to branch around the next instruction to a later instruction or to return to an earlier instruction. Another type of branch instruction that takes the computer out of its normal program sequence is indicated for the *I*-bit of the CC register. It is associated with an interrupt. An interrupt is a request, usually from an input or output (I/O) peripheral, that the CPU stops its present operation and accepts or takes care of (service) the special request. There will be more about interrupts later in this chapter.

MICROPROCESSOR ARCHITECTURE

The central component that controls and performs all operations in any microcomputer is the microprocessor, which is made up of many electronic subsystems all implemented in a single integrated circuit. As described in Chapter 2, a microprocessor consists of hundreds of thousands of transistors (on a single silicon chip) that are grouped together and interconnected to form the various subsystems, all of which are interconnected with internal address, data, and control buses.

Fig. 3.6 is a block diagram of a representative (simplified to 8 bits) commercial microprocessor such as has been used in legacy automotive digital electronic system.

The silicon chip on which the microprocessor is fabricated is mounted in a housing (usually a plastic structure) and connected to external pins that enable the microprocessor to be connected to the microcomputer. The connections to the external circuitry are depicted and labeled in Fig. 3.6 and include address and data buses. In addition, external connections are made to an oscillator (clock) and inputs and outputs: interrupt, ready (rdy), R/W, and data bus enable (DBE), the operation of which is explained later in this chapter.

This block diagram is divided into two main portions—a register section and a control section. The actual operations performed by the microprocessor are accomplished in the register section. The specific operations performed during the execution of a given step in the program are controlled by electric signals from the instruction decoder.

During each program step, an instruction in the form of an 8 or more bit binary number is transferred from memory to the instruction register. This instruction is decoded using logic circuits similar to those presented in Chapter 2. The result of this decoding process is a set of electric control signals that are sent to the specific components of the register section that are involved in the instruction being executed.

The data upon which the operation is performed are similarly transferred from memory to the DB buffer. From this buffer, the data are then transferred to the desired component in the register section for execution of the operation.

Note that a CPU or an arithmetic logic unit (ALU) is included in the register section of a simplified, representative microprocessor as shown in Fig. 3.6. The CPU/ALU is a complex circuit capable of performing the arithmetic and logical operations, as explained in Chapter 2 with respect to a simplified version of an (now obsolete) ALU. Also included in the register section is the accumulator, which is the register used most frequently to receive the results of arithmetic or logical operation. In addition, the example microprocessor register section has an index register, stack pointer (SP) register, and program counter register. The program counter register holds the contents of the program counter and is connected through the internal AB to the address buffer register. The AB for the example microprocessor has 16 lines and thereby can directly address 65,536 locations of memory. It should be emphasized that the microprocessor components and organization presented above are merely representative of this class of devices. There are many potential variations on the architecture shown and the number of bits associated with instructions and data. However, at the highest level of abstraction, the above description and architecture serve to illustrate any microprocessor that might be used in automotive applications.

READING INSTRUCTIONS

In the following sections of this chapter, the operation of a computer or microcomputer such as is found in modern automobiles is explained. The operation of any digital computer is fully controlled by the program. For automotive control applications, the controlling program must be efficient in order to

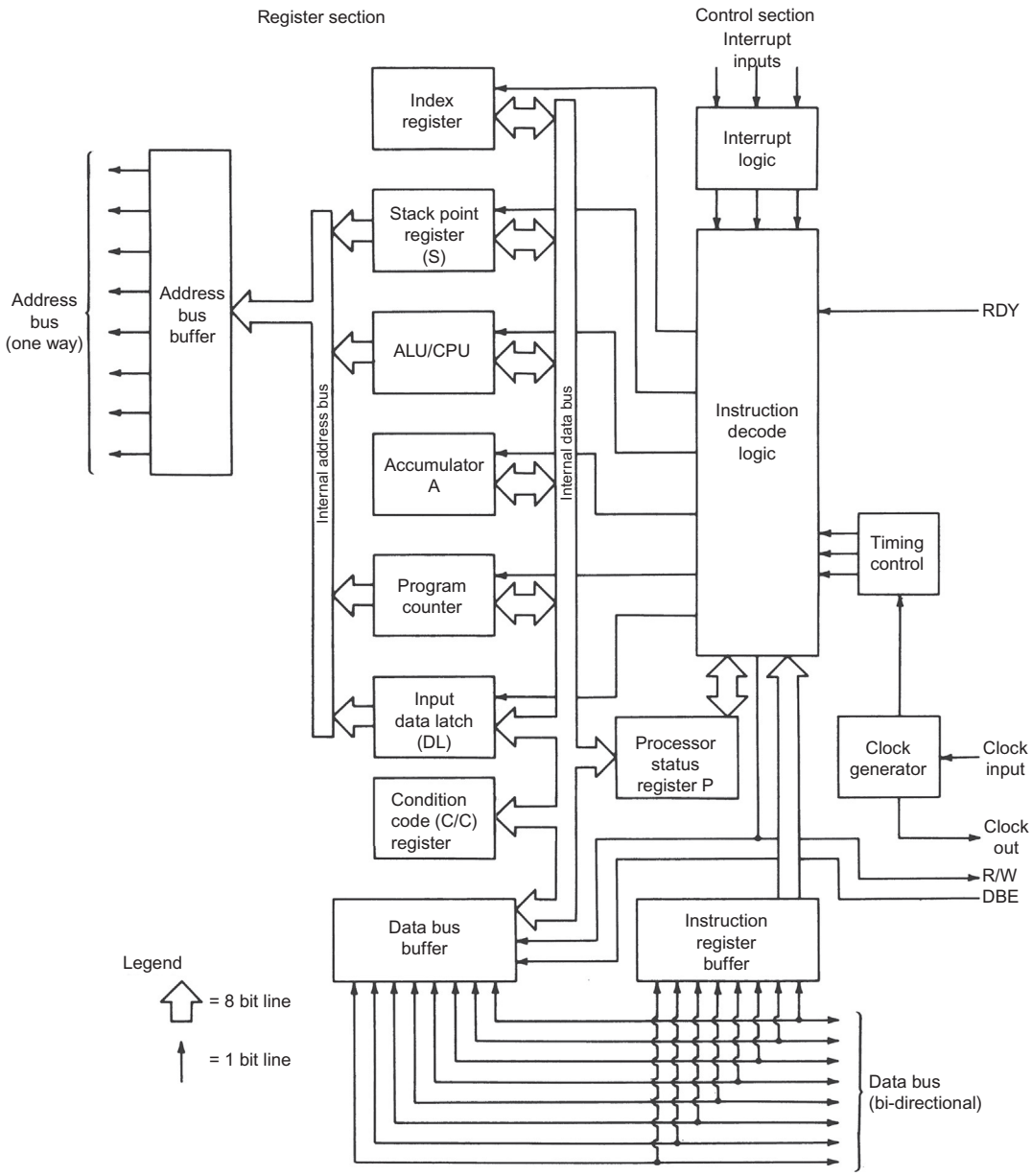


FIG. 3.6 Simplified microprocessor block diagram.

perform its various tasks at a rate that is compatible with the speed of operations of the external sensors, actuators, and switches. As explained below, each instruction is in the form of a string of binary characters similar to data. For illustrative purposes only, we assume all data and instructions are 8 bits wide.

To understand how the computer performs its functions, one must first understand how the computer obtains program instructions from memory. Recall that program instructions are stored sequentially (step by step) in memory as binary numbers, starting at a certain binary address and ending at some higher address. The computer uses a register called the program counter (Fig. 3.4) to keep track of where it is in the program.

INITIALIZATION

To start the computer, a small startup (boot) program that is permanently stored in the computer (in a ROM) is run. This program sets all of the CPU registers with the correct values and clears all information in the computer memory to zeros before the operations program is loaded. This is called *initializing* the system. Then, the operations program is loaded into memory, at which point the address of the first program instruction is loaded into the program counter. The first instruction is read from the memory location whose address is contained in the program counter register; that is, the 16 bits in the exemplary program counter are used as the address for a memory-read operation. Each instruction is read from memory in sequence and placed on the DB into the instruction register, where it is decoded. The instruction register is another temporary storage register inside the CPU (or microprocessor). It is connected to the DB when the information on the bus is an instruction.

OPERATION CODES

Numeric codes called *operation codes* (or *opcodes* for short) contain the instructions that represent the actual operation to be performed by the CPU. The block diagram of Fig. 3.7, which illustrates part of the CPU hardware organization, should help clarify the flow of instructions through the CPU.

The instruction register has a part that contains the numeric op codes. A decoder determines from the op codes the operation to be executed, and a data register controls the flow of data inside the CPU as a result of the opcode instructions.

One important function of the opcode decoder is to determine how many bytes must be read to execute each instruction. Many instructions require two or three bytes. Fig. 3.8 shows the arrangement of the bytes in an instruction. The first byte contains the opcode. The second byte contains address information, usually the lowest or least significant byte of the address.

PROGRAM COUNTER

The program counter is used by the CPU to address memory locations that contain instructions. Every time an opcode is read (this is often called *fetched*) from memory, the program counter is incremented (advanced by one) so that it points to (i.e., contains the address of) the next byte following the opcode. If the operation code requires another byte, the program counter supplies the address, the second byte is fetched from memory and the program counter is incremented. Each time the CPU performs a fetch operation, the program counter is incremented; thus, the program counter always contains the address of the next byte in the program. Therefore, after all bytes required for one complete instruction have been read, the program counter contains the address for the beginning of the next instruction to be executed.

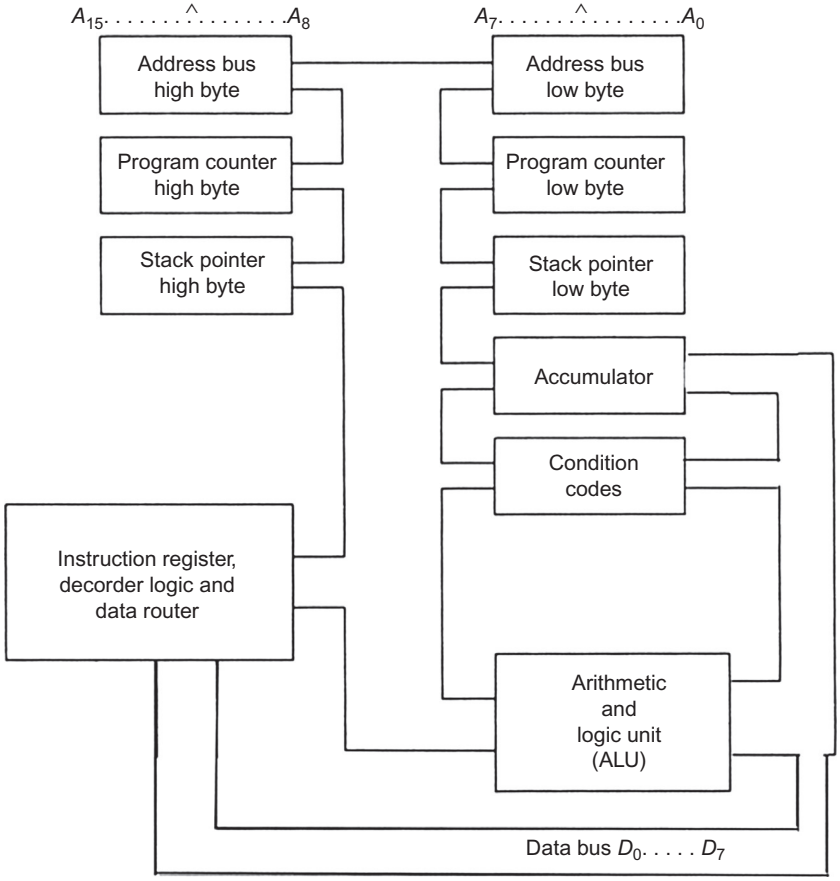


FIG. 3.7 Instruction decode subsystem.

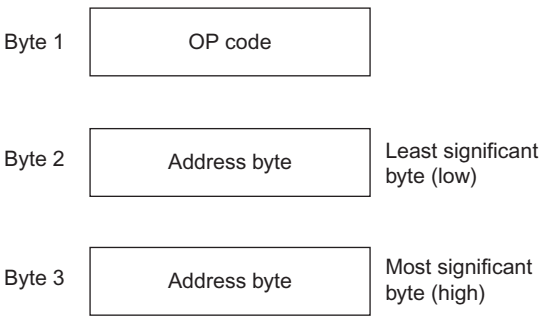


FIG. 3.8 Instruction decode bytes.

BRANCH INSTRUCTION

In any practical computer program, there is normally the need to change the sequence of instructions as a result of some logical condition being met. An instruction of this type is called a branch instruction. In the simplified example program, all of the branch instructions require two bytes. The first byte holds the operation code, and the second byte holds the location to which the processor is to branch.

Now, if the address information associated with a branch instruction is only 8 bits long and totally contained in the second byte, it cannot be the actual branch address. In this case, the code contained in the second byte is actually a two's complement number that the CPU adds to the lower byte of the program counter to determine the actual new address. This number in the second byte of the branch instruction is called an *address offset* or just *offset*. Recall that in two's complement notation, the 8-bit number can be either positive or negative; therefore, the branch address offset can be positive or negative. A positive branch offset causes a branch forward to a higher memory location. A negative branch offset causes a branch to a lower memory location. Since 8 bits are used in the present example, the largest forward branch is 127 memory locations and the largest backward branch is 128 memory locations.

Offset example

By way of exemplary illustration, suppose the program counter is at address 5122 and the instruction at this location is a branch instruction. The instruction to which the branch is to be made is located at memory address 5218. Since the second byte of the branch instruction is only 8 bits wide, the actual address 5218 cannot be contained therein. Therefore, the difference or offset (96) between the current program counter value (5122) and the desired new address (5218) is contained in the second byte of the branch instruction. The offset value (96) is added to the address in the program counter (5122) to obtain the new address (5218), which is then placed on the AB. The binary computation of the final address from the program counter value and second byte of the branch instruction is shown in Fig. 3.9.

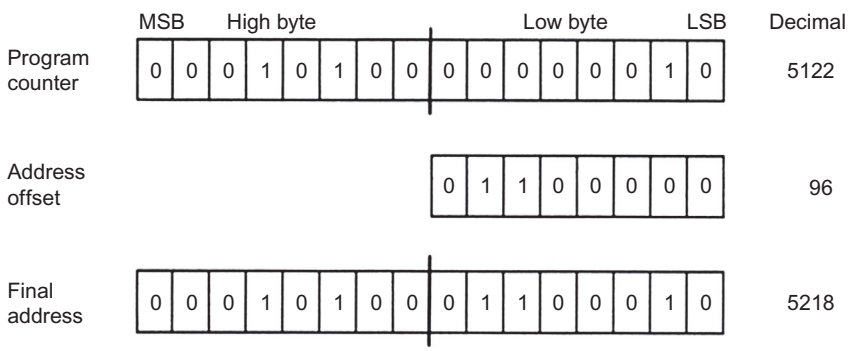


FIG. 3.9 Program counter offset.

JUMP INSTRUCTION

Branch instructions have a range of +127 to −128 (in the present 8-bit example). If the branch needs to go beyond this range, a jump instruction must be used. The jump instruction is a 3-byte instruction. The first byte is the jump opcode, and the next two bytes are the actual jump address. The CPU loads the jump address directly into the program counter, and the program counter is effectively restarted at the new jump location. The CPU continues to fetch and execute instructions in exactly the same way it did before the jump was made.

The jump instruction causes the CPU to jump out of one section of the program into another. The CPU cannot automatically return to the first section because no record was kept of the previous location. However, another instruction, the jump-to-subroutine, does leave a record of the previous instruction address.

JUMP-TO-SUBROUTINE INSTRUCTION

A *subroutine* is a short program that is used by the main program to perform a specific function. Its use in programming is explained in a later section of this chapter. It is located in sequential memory locations separated from the main program sequence. If the main program requires some function such as multiplication several times at widely separated places within the program, the program can contain one subroutine to perform the multiplication and then have the main program jump to the memory locations containing the subroutine each time it is needed. This saves the task of having to introduce the multiplication program repeatedly in the program. To perform the multiplication, the program simply includes instructions in the main program that first load the numbers to be multiplied into the data memory locations used by the subroutine and then jump to the subroutine.

Refer to Fig. 3.10 to follow the sequence. The sequence of steps performed in the jump-to-subroutine operation is depicted by circled numbers from 1 to 10 in Fig. 3.10. It begins with the program counter holding to address location 100, where it gets the jump-to-subroutine instruction (step 1). Each jump-to-subroutine instruction (step 2) requires that the next two bytes must also be read to obtain the jump address (step 2a). Therefore, the program counter is incremented once for each byte (steps 3 and 4), and the jump address is loaded into the address register. The program counter is then incremented once more so that it points to the opcode byte of the next instruction (step 5).

Saving the program counter

The contents of the program counter are saved by storing them in a special memory location before the jump address is loaded into the program counter. This program counter address is saved so that it can be returned to in the main program when the subroutine is finished. This is the record that was mentioned before.

Now, refer back to Fig. 3.4. There is a register in Fig. 3.4 called the SP. The address of the special memory location used to store the program counter content is kept in this 16-bit SP register. When a jump-to-subroutine opcode is encountered, the CPU uses the number code contained in the SP as a memory address to store the program counter to memory (step 2b of Fig. 3.10). The program counter is a 2-byte register, so it must be stored in two memory locations (in the present example). The current SP is used as an address to store the lower byte of the program counter to memory (step 6). Then, the SP is decremented (decreased by one), and the high byte of the program counter is stored in the next lower

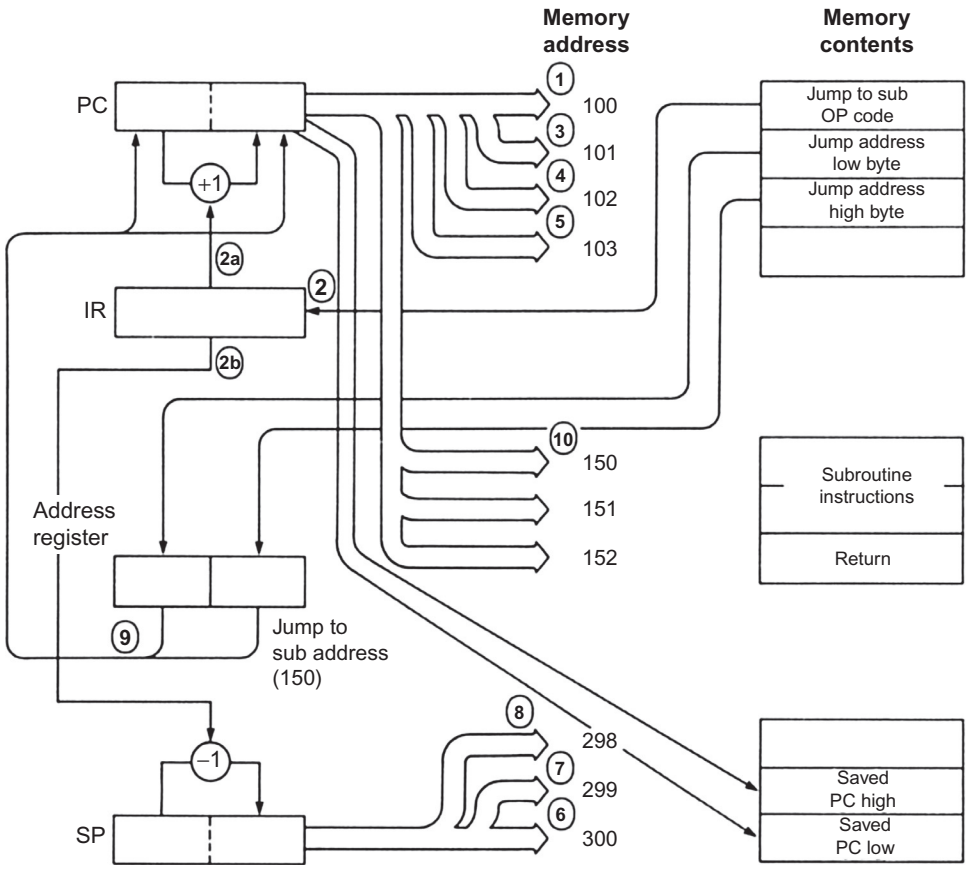


FIG. 3.10 Jump-to-subroutine sequence.

memory location (step 7). The SP is then decremented again to point to the next unused byte in the stack to prepare for storing the program counter again when required (step 8). The special memory locations pointed to by the SP are called stacks.

After the program counter has been incremented and saved, the jump address is loaded into the program counter (step 9). The jump to the subroutine is made, and the CPU starts running the subroutine (step 10). The only thing that distinguishes the subroutine from another part of the program is the way in which it ends. When a subroutine has run to completion, it must allow the CPU to return to the point in the main program from which the jump occurred. In this way, the main program can continue without missing a step. The return-from-subroutine (RTS) instruction is used to accomplish this. It is decoded by the instruction register, and increments the SP as shown in Fig. 3.11 (step 1). It uses the SP to address the stack memory to retrieve the old program counter value from the stack (steps 2 and 4). The old program counter value is loaded into the program counter register (steps 3 and 5), and execution resumes in the main program (step 6). The RTS instruction works like the jump-to-subroutine instruction, except in reverse.

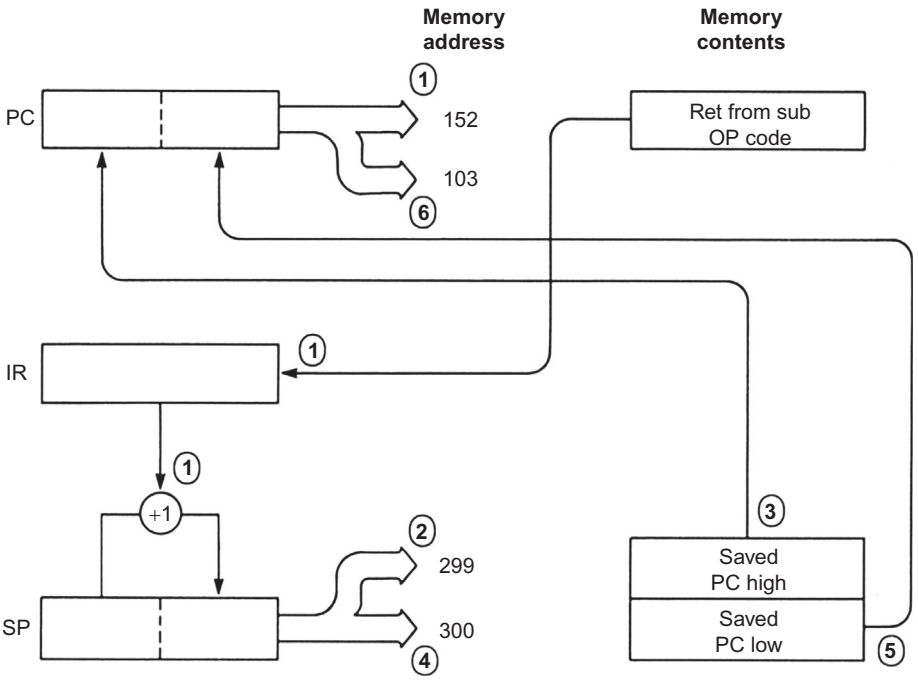


FIG. 3.11 Return-from-subroutine steps.

EXAMPLE USE OF A MICROCOMPUTER

Let us look at a trivial example of how a microcomputer might be used to replace some digital logic, and along the way learn about some more microcomputer instructions. The digital logic to be replaced in this example is a simple AND gate circuit. Now, no one would use a microcomputer to replace only an AND gate, because an AND gate costs a fraction of what a microcomputer costs. However, if the system already has a microcomputer in it, the cost of the AND gate could be eliminated by performing the logical AND function in the computer rather than with the gate.

Suppose there are two signals that must be ANDed together to produce a third signal. One of the input signals comes from a pressure switch located under the driver's seat of an automobile; its purpose is to indicate whether someone is occupying the seat. This signal will be called A, and it is at logical high when someone is sitting in the seat. Signal B is developed within a circuit contained in the seat belt and is at logical high when the driver's seat belt is fastened. The output of the AND gate is signal C. It will be at logical high when someone is sitting in the driver's seat *and* has the seat belt fastened.

BUFFER

In order to use a microcomputer to replace the AND gate, the computer must be able to detect the status of each signal. The microcomputer used here has the so-called memory-mapped I/O (as explained earlier in this chapter), in which peripherals are treated exactly like memory locations. The task is to

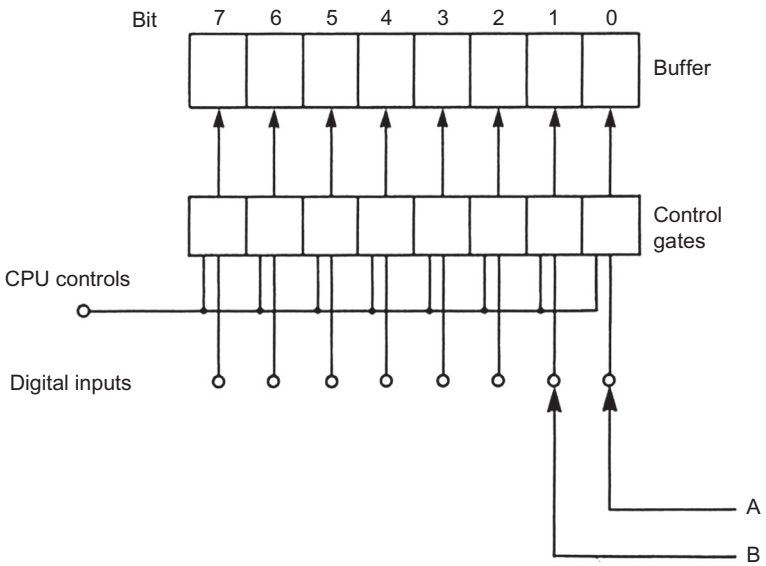


FIG. 3.12 Buffer configuration.

provide a peripheral that allows the computer to look at the switch signals as if they were bits in a memory location. This can be done easily by using a device called a *buffer* (Fig. 3.12).

To the microcomputer, a buffer looks just like an 8-bit memory slot at a selected memory location. The 8 bits in the memory slot correspond to 8 digital signal inputs to the buffer. Each digital input controls the state of a single bit in the memory slot. The digital inputs are gated into the buffer under control of the CPU. The microcomputer can detect the state of the digital inputs by examining the bits in the buffer any time after the inputs are gated into the buffer.

In this application, signal A will be assigned to the rightmost bit (bit 0) and signal B to the next bit (bit 1). It does not matter that the other 6 bits are left unconnected. The computer will gate in and read the state of those lines, but the program will be written to purposely ignore them. With the logical signals interfaced to the microcomputer, a program can be written that will perform the required logical function.

PROGRAMMING LANGUAGES

The trend in contemporary vehicular electronic system programming is via Automotive Open System Architecture (AUTOSAR), which is explained near the end of this chapter. Traditionally, before writing a program, the programmer had to know the code or language in which the program is to be written. Computer languages come in various levels, including high-level language such as C. A program written in a high-level language such as AUTOSAR is essentially independent of the individual hardware on which it is to be run. However, to be useful on any given computer, it must be converted to a language that is specific to that hardware. For traditional computer programming, the high-level language

program converted the code to machine language via a software called a compiler. When microcomputers were first used in vehicular electronics, the programs were written in a so-called assembly language. An assembly language is designed for a specific microprocessor. A typical assembly language program consists of a sequence of instructions as explained below that are highly mnemonic to an English word. Machine language is the actual language in which a program is stored in memory in a binary or binary-coded format. For the present example, we choose the intermediate-level language (assembly language) to illustrate specific CPU operations.

ASSEMBLY LANGUAGE

Although programming assembly language is practically obsolete, it is, perhaps, informative to review some elements to illustrate specific steps performed by a microprocessor. Assembly language is a special type of abbreviated language, each symbol of which pertains to a specific microprocessor operation. Some assembly language instructions, such as branch, jump, jump-to-subroutine, and RTS, have already been discussed. Others will be discussed as they are needed to execute an example program. Assembly language instructions have the form of initials or shortened (so-called mnemonics) words that represent microcomputer functions. These abbreviations are only for the convenience of the programmer because the program that the microcomputer eventually runs must be in the form of binary numbers. When each instruction is converted to the binary code that the microcomputer recognizes, it is called a machine language program (or executable code).

Table 3.1 shows the assembly language equivalents for typical traditional microprocessor instructions, along with a detailed description of the operation called for by the instruction. When writing a microcomputer program, it is easier and faster to use the abbreviated name rather than the complete function name. Assembly language simplifies programming tasks for the computer programmer because the abbreviations are easier to remember and write than the binary numbers the computer uses. However, the program eventually must be converted to the executable binary codes that the microcomputer recognizes as instructions, which is done by a special program called an *assembler*. The assembler program is run on the computer to convert assembly language to binary codes. Traditionally, this enabled the programmer to write the program using words that had meaning to the programmer and also to produce machine codes that the computer can use.

LOGIC FUNCTIONS

Microprocessors are capable of performing all of the basic logical functions such as AND, OR, NOT, and combinations of these. For instance, the NOT operation can affect the accumulator by changing all ones to zeros and zeros to ones. Other logical functions are performed by using the contents of the accumulator and some memory location. All 8 bits of the accumulator are affected, and all are changed at the same time. As shown in Fig. 3.13, the AND operation requires two inputs.

One input is the contents of the accumulator, and the other input is the contents of a memory location; thus, the eight accumulator bits are ANDed with the eight memory bits. The AND operation is performed on a bit-by-bit basis. For instance, bit 0 of the accumulator (the rightmost bit) is ANDed with bit 0 of the memory location, bit 1 with bit 1, bit 2 with bit 2, and so on. In other words, the AND operation is performed as if eight AND gates were connected with one input to a bit in the accumulator and with the other input to a bit (in the same position) in the memory location. The resulting AND

Table 3.1 Assembly Language Mnemonics

Mnemonic	Operand	Comment
a. *Program Transfer Instructions*		
JMP	(Address)	Jump to new program location
JSR	(Address)	Jump to a subroutine
BRA	(Offset)	Branch using the offset
BEQ	(Offset)	Branch if accumulator is zero
BNE	(Offset)	Branch if accumulator is nonzero
BCC	(Offset)	Branch if carry bit is zero
BCS	(Offset)	Branch if carry bit is nonzero
BPL	(Offset)	Branch if minus bit is zero
BMI	(Offset)	Branch if minus bit is nonzero
RTS		Return from a subroutine
b. *Data Transfer Instructions*		
LDA	(Address)	Load accumulator from memory
STA	(Address)	Store accumulator to memory
LDA	# (Constant)	Load accumulator with constant
LDS	# (Constant)	Load stack pointer with constant
STS	(Address)	Store stack pointer to memory
c. *Arithmetic and Logical Operations*		
COM		Complement accumulator (NOT)
AND	(Address)	AND accumulator with memory
OR	(Address)	OR accumulator with memory
ADD	(Address)	Add accumulator with memory
SUB	(Address)	Subtract accumulator with memory
AND	# (Constant)	AND accumulator with constant
OR	# (Constant)	OR accumulator with constant
SLL		Shift accumulator left logical
SRL		Shift accumulator right logical
ROL		Rotate accumulator left
ROR		Rotate accumulator right

outputs are stored back into the accumulator in the corresponding bit positions. The OR logical function is performed in exactly the same way as the AND except that a 1 would be produced at the output if signal A or signal B were a 1 or if both were a 1 (i.e., using OR logic).

SHIFT

Instead of the AND gate inputs being switched to each bit position as shown in Fig. 3.13, the microcomputer uses a special type of sequential logical operation, the shift, to move the bits to the AND gate inputs. A type of register that is capable of such shift operations was discussed in

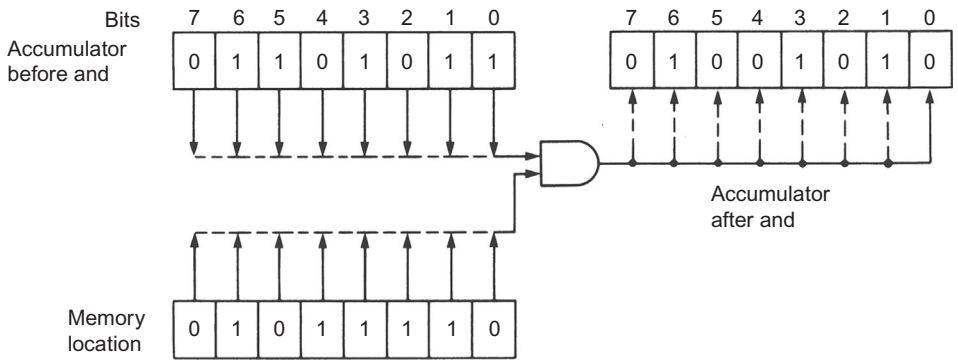


FIG. 3.13 AND logic illustration.

Chapter 2 and is called a shift register. A shift operation causes every bit in the accumulator to be shifted one bit position either to the right or to the left. It can be what is called a logical shift or it can be a circulating shift. Fig. 3.14 shows the four types of shifts (logical, circulating, right, and left) and their effects on the accumulator.

In a left shift, 7 bit (the leftmost bit) is shifted into the carry bit of the CC register, 6 bit is shifted into 7 bit, and so on until each bit has been shifted once to the left. Bit 0 (the rightmost bit) can be replaced either by the carry bit or by a zero, depending on the type of shift performed. Depending on the microprocessor, it is possible to shift other registers and the accumulator.

PROGRAMMING THE AND FUNCTION IN ASSEMBLY LANGUAGE

When preparing a program in assembly language, it was always the task of the programmer to choose instructions and organize them in such a way that the computer performs the desired tasks. To program the AND function, one of the instructions will be the AND, which stands for "AND accumulator with contents of a specific memory location," as shown in Table 3.1c. Since the AND affects the accumulator and memory, values must be put into the accumulator to be ANDed. This requires the load accumulator instruction LDA.

The assembly language program of Fig. 3.15 performs the required AND function. The programmer must first know which memory location the digital buffer interface (Fig. 3.12) occupies. This location is identified, and the programmer writes instructions in the assembler program so that the buffer memory location will be referred to by the label or name SEAT. The mnemonic SEAT is easier for the programmer to remember and write than the address of the buffer.

The operation of the program is as follows. The accumulator is loaded with the contents of the memory location SEAT. Note in Fig. 3.12 that the two digital logic input signals, A and B, have been gated into bits 0 and 1, respectively, of the buffer that occupies the memory location labeled SEAT. Bit 0 is high when someone is sitting in the driver's seat. Bit 1 is high when the driver's seat belt is fastened. Only these two bits are to be ANDed together; the other six are to be ignored. But there is a problem because both bits are in the same 8-bit byte and there is no single instruction to AND bits in the same byte. However, the two bits can be effectively separated by using a mask.

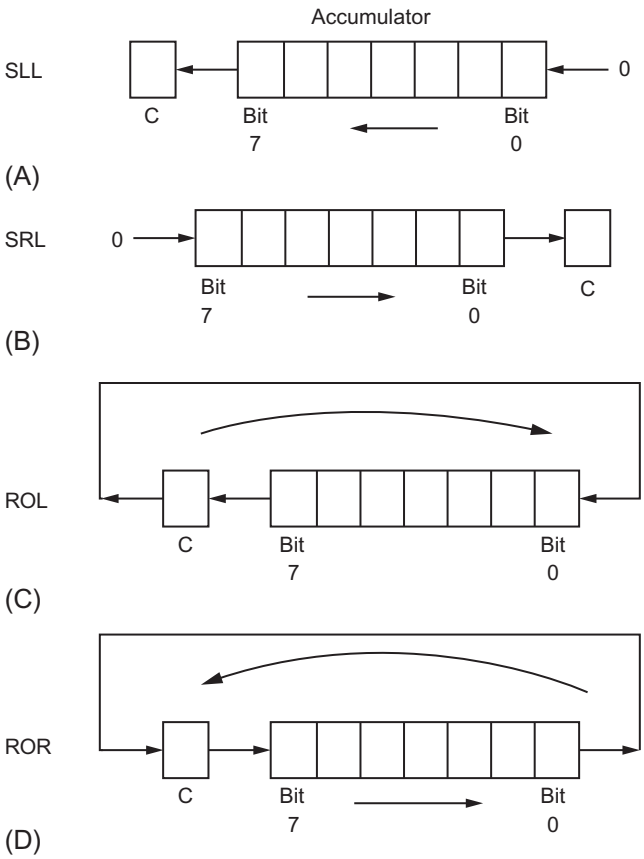


FIG 3.14 Shift register operations. (A) Shift left logically, (B) shift right logically, (C) rotate left (circulate with carry), and (D) rotate right (circulate with carry).

MASKING

Masking is a technique used to allow only selected bits to be involved in a desired operation. Since the buffer contents have been loaded into the accumulator, only bits 0 and 1 have meaning, and these two bits are the only ones of importance that are to be kept in the accumulator. To do this, the accumulator is ANDed with a constant that has a zero in every bit location except the one that is to be saved. The binary constant in line 2 of Fig. 3.15A (00000001) is chosen to select bit 0 and set all others to zero as the AND instruction is executed.

The ANDing procedure is called *masking* because a mask has been placed over the accumulator that allows only bit 0 to come through unchanged. If bit 0 was a logical 1, it is still a logical 1 after masking. If bit 0 was a logical 0, it is still a logical 0. All other bits in the accumulator now contain the correct bit information about bit 0.

Program Label	Mnemonic	Operand
1 CHECK	LDA	SEAT
2	AND	#00000001 B
3	SLL	
4	AND	SEAT
5	RTS	

(A)

Program Label	Mnemonic	Operand
1 WAIT	JRS	CHECK
2	BEQ	WAIT
3	RTS	

(B)

FIG. 3.15 Assembly language AND subroutine. (A) Subroutine CHECK and (B) subroutine WAIT.

SHIFT AND AND

In our example program, the accumulator is still not ready to perform the final AND operation. Remember that SEAT contains the contents of the buffer and the conditions of signal A and signal B. The contents of the accumulator must be ANDed with SEAT so that signals A and B are ANDed together. A copy of signal A is held in the accumulator in bit 0, but it is in the wrong bit position to be ANDed with signal B in SEAT in the bit 1 position. Therefore, signal A must be shifted into the bit 1 position. To do this, the shift left logical instruction is used (Fig. 3.14A). With signal A in bit 1 of the accumulator and signal B in bit 1 of SEAT, the AND operation can be performed on the two bits. If both A and B are high, the AND operation will leave bit 1 of the accumulator high (1). If either is low, bit 1 of the accumulator will be low (0). This trivial example of the AND operation could be used to issue a warning that the sent occupant has not fastened the seat belt if the result of the AND is a zero.

USE OF SUBROUTINES

The previous example program has been written as a subroutine named CHECK so that it can be used at many different places in a larger program. For instance, if the computer is controlling the speed of the automobile, it might be desirable to be able to detect whether a driver is properly fastened in the seat before it sets the speed at 55 miles per hour.

Since the driver's seat information is very important, the main program must wait until the driver is ready before allowing anything else to happen. A program such as that shown in Fig. 3.15B can be used to do this. The main program calls the subroutine WAIT, which in turn immediately calls the subroutine CHECK. CHECK returns to WAIT with the CCs set as they were after the last AND instruction. The Z bit (see Fig. 3.5A) is set if A and B are not both high (the accumulator is zero). The BEQ instruction (see Table 3.1) in line 2 of WAIT branches back because the accumulator is zero and causes the computer to

reexecute the JSR instruction in line 1 of WAIT. This effectively holds the computer in a loop, rechecking signals A and B until the accumulator has a nonzero value (A and B are high).

In automotive electronic systems for control or instrumentation, there are many subroutines that are called repeatedly. Among those is the routine for multiplication (and for division). The algorithm on which the subroutine is based is derived from the fundamental multiplication of a pair of bits:

$$\begin{aligned} 0 \times 0 &= 0 \\ 1 \times 0 &= 0 \\ 0 \times 1 &= 0 \\ 1 \times 1 &= 1 \end{aligned}$$

The product of a binary multiplicand A by a binary multiplier B yields binary result C. It is perhaps instructive to illustrate with an example in which A = 13 (decimal) and B = 2 (decimal):

A = 1101		(13 decimal)
B = 10		(2 decimal)
A	1101	
B	×10	
	0000	
	+1101	
C	11010	(26 decimal)

In obtaining this result for each bit in the multiplier, the multiplicand is either copied (i.e., multiplied by 1) if the multiplier bit is a 1 or replaced by all 0's (i.e., multiplication by 0) and shifted to the left by the position of the bit in the multiplier. After performing this operation for each multiplier bit, the results are summed according to the rules of binary addition.

MICROCOMPUTER HARDWARE

The microcomputer system electronic components are known as computer hardware. (The programs that the computer runs are called software.) The basic microcomputer parts are the CPU, memory, and I/O (input and output peripherals). We next expand upon this discussion of important components and their associated operations.

CENTRAL PROCESSING UNIT

The CPU is a microprocessor which is an integrated circuit. It contains hundreds of thousands of transistors and diodes on a chip of silicon small enough to fit on the tip of a finger. It includes some form of CPU or ALU, as well as registers for data and instruction storage and a control section. Contemporary versions of a microprocessor are packed in a flat package that has pins all the way around the periphery such that the IC can be attached to the surface of a printed circuit board

(known as surface-mounted ICs). The CPU gets program instructions from a memory device. Near the end of this chapter is a discussion of multicore processors that allow for parallel processing for multiple vehicle electronic systems.

MEMORY: ROM

There are several types of memory devices available, and each has its own special features. Systems such as those found in the automobile that must permanently store their programs use a type of permanent memory called ROM. This type of memory can be programmed only one time and the program is stored permanently, even when the microcomputer power is turned off. The programs stored in ROM are sometimes called *firmware* rather than software since they are unchangeable. This type of memory enables the microcomputer to immediately begin running its program as soon as it is turned on.

Several types of ROM can be used in any microcomputer, including those found in automotive digital systems. For program storage, a ROM is used that is not alterable. The program and data storage are determined by physical configuration during manufacturing. In certain cases, it may be desirable to modify certain parameters. For example, in automotive applications it may be desirable to permit authorized persons to modify a control system parameter of a vehicle after it has been in operation for some time to improve system performance. In this case, it must be possible to modify data (parameters) stored in ROM. Such modification is possible in a ROM that can be electrically erased and reprogrammed. This type of ROM is termed electrically erasable programmable read-only memory (EEPROM). In principle, of course, it is theoretically possible to have the ROM or a portion of it stored on a removable chip. New parameters can be installed by simply replacing this chip.

MEMORY: RAM

Another type of memory, one that can be written to as well as read from, is required for the program stack, data storage, and program variables. This type of memory is called RAM. This is really not a good name to distinguish this type of memory from ROM because ROM is also a random-access type of memory. Random-access means the memory locations can be accessed in any order rather than in a particular sequence. A better name for the data storage memory would be read/write memory (RWM). However, the term RAM is commonly used to indicate a RWM, so that is what will be used here. A typical microcomputer contains both ROM- and RAM-type memory.

It is beyond the scope of this book to discuss the detailed circuitry of all types of memory circuits. However, one example of a type of circuit that can be used for memory is the register circuit, which is implemented with flip-flop circuits as described in Chapter 2.

I/O PARALLEL INTERFACE

Microcomputers require interface devices that enable them to communicate with other systems. The digital buffer interface used in the driver's seat application discussed earlier is one such device. The digital buffer interface is an example of a parallel interface because the eight buffer lines are all sampled at one time, that is, in parallel. The parallel buffer interface in the driver's seat application is an input or readable interface. Output, or writable, interfaces allow the microcomputer to affect external logical systems. An output buffer must be implemented using a data latch so that the binary output is

retained after the microcomputer has finished writing data into it. This permits the CPU to go on to other tasks, while the external system reads and uses the output data. This is different from the parallel input, in which the states could change between samples.

DIGITAL-TO-ANALOG CONVERTER

The parallel input and output interfaces are used to monitor and control external digital signals. As explained in Appendix B, the microcomputer can also be used to measure and control analog signals through the use of special interfaces. The microcomputer can produce an analog voltage by using a digital-to-analog converter (D/A converter). A D/A converter accepts inputs from the digital system of a certain number of binary bits and produces an output voltage level that is proportional to the input number and may incorporate a zero-order hold (ZOH; see Chapter 2). D/A converters come in many different versions with different numbers of input bits and output ranges. A representative example microcomputer D/A converter has 8-bit inputs and a 0–5 V output range.

A simple ideal 8-bit D/A converter is shown in Fig. 3.16. This type of D/A converter uses a parallel input interface and two operational amplifiers.

The 8 bits are written into the parallel interface and stored in data latches (e.g., J-K flip-flop as explained in Chapter 2). For the purposes of explaining the operation of this simplified example D/A converter, it is assumed that the parallel interface includes output circuitry associated with each data bit latch such that the voltages corresponding to the two logical levels are given by

$$\begin{aligned} D_n &= 5\text{V if } A_n = 1 \quad n = 1,2,\ldots,N \\ &= 0\text{V if } A_n = 0 \end{aligned} \tag{3.1}$$

where $A_n = n$th bit of the 8-bit input digital data where A_8 is the most significant bit (MSB). In this example, the output of each latch is a digital signal that is ideally 0 if the bit is low and 5 V if the bit is high. As explained in Chapter 2, fixed voltage levels for D_n can be achieved using zener diodes. The first op-amp is an inverting mode summing amplifier for which the gain for input n is given by $(R/R_s(n))$.

The source resistance for the nth data bit is given by

$$R_S(n) = 2^{N-n+1}R \quad n = 1,2,\ldots,N \tag{3.2}$$

where, for the present 8-bit example, $N=8$. The output voltage of the first op-amp circuit V_1 (in accordance with the discussion of summing op-amp circuits of Chapter 2) is given by

$$V_1 = -\sum_{n=1}^{N} \frac{D_n}{2^{N-n+1}} \tag{3.3}$$

$$= -\frac{5}{2^N}\sum_{n=1}^{N} A_n 2^{n-1} \tag{3.4}$$

$$= -\frac{5}{2^N} N_{10} \tag{3.5}$$

where N_{10} is the decimal numerical value of the input digital data.

The second op-amp has a closed-loop gain of

$$A_{c\ell} = -R_f/R \tag{3.6}$$

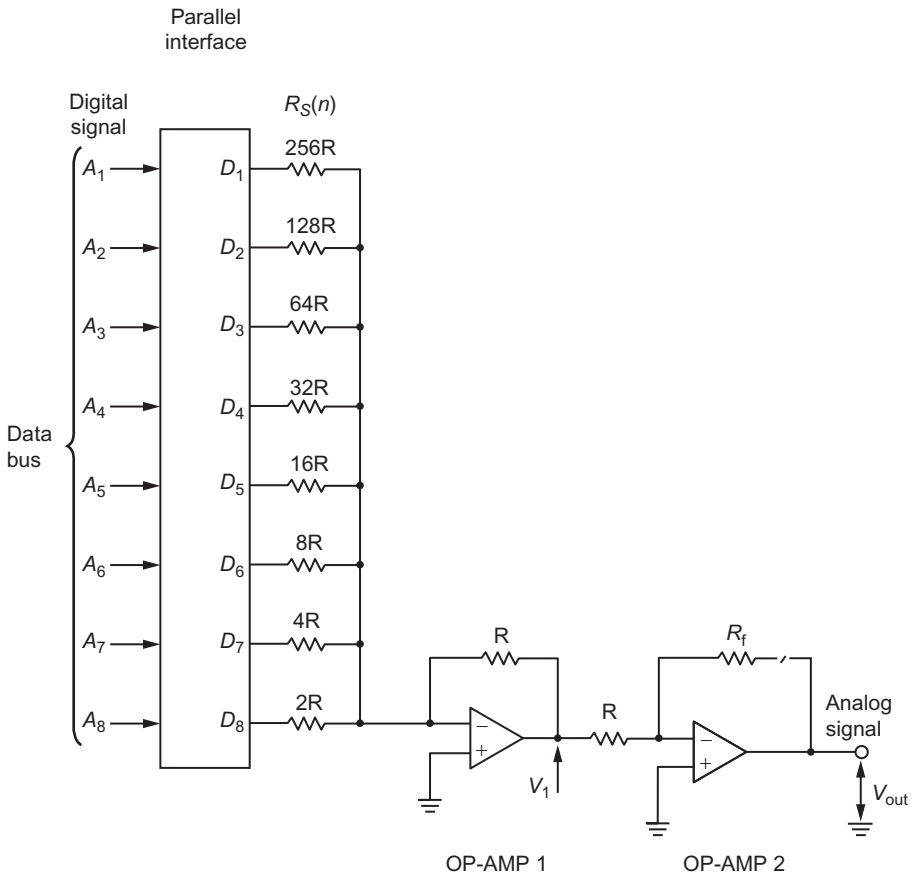


FIG. 3.16 Simplified D/A configuration.

The output voltage of this second op-amp is given by

$$V_{\text{out}} = \frac{5R_f}{2^N R} N_{10} \tag{3.7}$$

$$= K_{\text{DA}} N_{10} \tag{3.8}$$

where K_{DA} is the scale factor for the D/A converter. The effect of the two amplifiers is to scale each bit of the parallel interface by a specially chosen factor and add the resultant voltages together such that the D/A converter output voltage (V_{out}) is proportional to the decimal equivalent of the input binary data. The scale factor is chosen by the system designer to be compatible with the voltage requirements of the component (e.g., actuator) to which the D/A is converted. Typically, in control applications, the D/A converter output is connected to a ZOH before the converted voltage is sent to the destination component (e.g., actuator) (see Appendix B and Chapter 2).

The D/A converter output voltage can change only when the computer writes a new number into the D/A converter data latches. As explained in Appendix B, in control applications, the D/A converter ZOH combination is synchronized to the sampler, which samples the input to the control system. The computer must generate each new output often enough to ensure an accurate representation of the changes in the digital signal. The analog output of the D/A converter can take only a specific number of different values and can change only at specific times determined by the sampling rate. The output of the converter will always have small discrete step changes (resolution). The resolution of the representation of the A/D output varies in proportion to the number N of bits. The associated vehicle system designer must decide how small the steps must be to produce the desired shape and smoothness in the analog signal so that it is a reasonable duplication of the variations in the digital levels. The smoothness of the D/A output voltage can be improved by filtering, although care must be exercised in the filter design to prevent waveform distortion and phase delay.

ANALOG-TO-DIGITAL CONVERTER

In addition, microcomputers can measure analog voltages by using a special interface component called an analog-to-digital A/D converter. Analog-to-digital converters convert an analog voltage input into a digital number output that the microcomputer can read. Fig. 3.17 shows a conceptually simple hypothetical, but not necessarily optimal, way of making an A/D converter by using a D/A converter and a voltage comparator. Control of the A/D circuitry is exercised by the computer via output logical variables "reset" and convert C and input logical variable end-of-conversion (EOC) as shown in Fig. 3.17.

At the sample time (t_k), the analog-to-digital conversion process begins under computer control with several operations. The computer sends a sample trigger signal to the sample and hold circuit causing it to sample v_{in} at the sample time t_k. The sample and hold output voltage v_k is given by:

$$v_k = v_{in}(t_k) \quad t_k \le t \le t_{k+1}$$

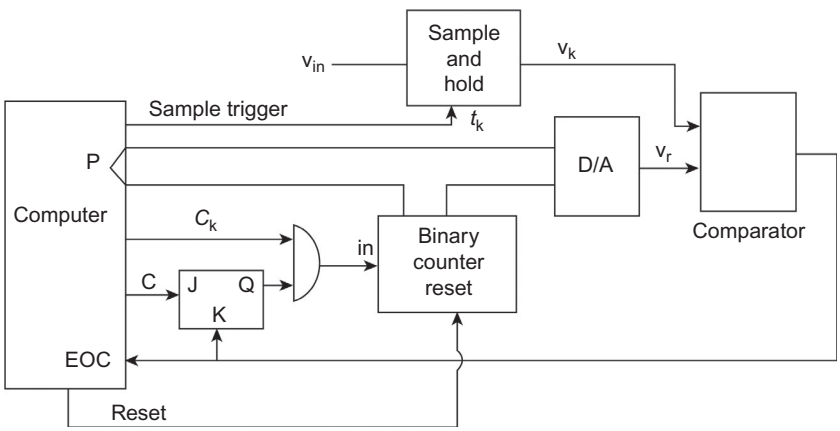


FIG. 3.17 Example A/D converter.

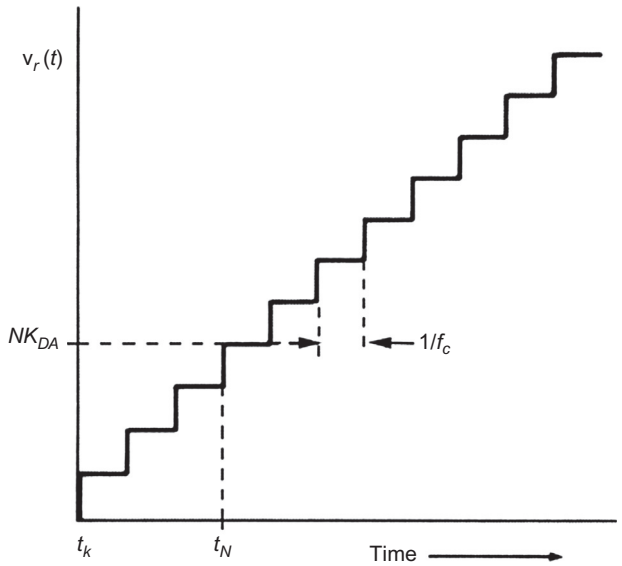


FIG. 3.18 Digital ramp waveform.

The computer also generates a signal that resets the counter to zero and then sends a signal C to the J-K flip-flop circuit that enables the AND gate such that the counter begins counting clock (C_k) pulses (generated by the computer timing circuitry). The D/A converter output voltage changes in discrete steps at each clock pulse. This causes the analog output voltage to have a staircase appearance as the binary number at the input is increased one bit at a time from minimum value to maximum value, as shown in Fig. 3.18.

This example 8-bit D/A converter can have any one of 256 different voltage levels. For many applications, this is a close enough approximation to a continuous analog ramp signal. The counter contents at time t_N are the binary equivalent of N where N is the largest integer in the following:

$$N = \{\lfloor f_c(t_N - t_k)\rfloor\} \tag{3.9}$$

The ramp voltage $v_r(t)$ for the time interval specified in Eq. (3.9) is given by

$$v_r(t) = K_{DA}N$$

as explained above for a D/A converter and shown in Fig. 3.18. The counting of clock pulses continues until the ramp reaches a condition (called coincidence) at which point the comparator changes state. The comparator output (v_{comp}) is a binary-valued voltage, which is given by

$$v_{comp} = v_L \quad v_r < v_k \tag{3.10}$$

$$= v_H \quad v_r \geq v_k \tag{3.11}$$

where v_L and v_H are voltages corresponding to logic low and high, respectively. At coincidence, the ramp voltage is essentially given by

$$v_r(t_c) = v_k \tag{3.12}$$

where t_c is the time of coincidence. When the comparator voltage switches from low to high, the count is inhibited via the K input to the J-K input. The contents of the counter are the binary equivalent N_2 of $N(t_c)$ and remain at this value until the counter is reset. At this point (i.e., $t = t_c$), the computer receives an EOC signal (as v_{comp} switches from V_L to V_H) via an interrupt input.

The computer responds under program control to read the counter contents (N_c), which are the binary equivalent of the number of clock pulses N_c counted from t_k to $t_k + t_c$:

$$N_c = N(t_c)$$
$$N_c = \frac{v_k}{K_{DA}} \tag{3.13}$$

The binary equivalent of N_c is denoted N_{c2}.

As shown in Fig. 3.17, the counter output lines are connected to a computer parallel input (P) such that the counter contents are available to the computer DB. The computer can be configured to read the counter contents via a special memory operation called memory-mapped I/O data read as explained earlier in this chapter. Thus, the computer reads the binary equivalent of a number, which is proportional to v_k. Conversion to v_k is accomplished by multiplying N_{c2} by the D/A converter constant K_{DA}. It is important that the conversion time for the largest value of v_k be small compared to sample times (i.e., $\max(t_c) \ll T$). This condition can be met with sufficiently high clock frequency (f_c).

SAMPLING

As explained in detail in Appendix B, a discrete time digital system operating on continuous-time variables requires sampling the input signal at the Nyquist or higher rate. Fig. 3.19 shows a sine wave analog signal and some digital approximations with various sampling rates. Notice that Fig. 3.19A with

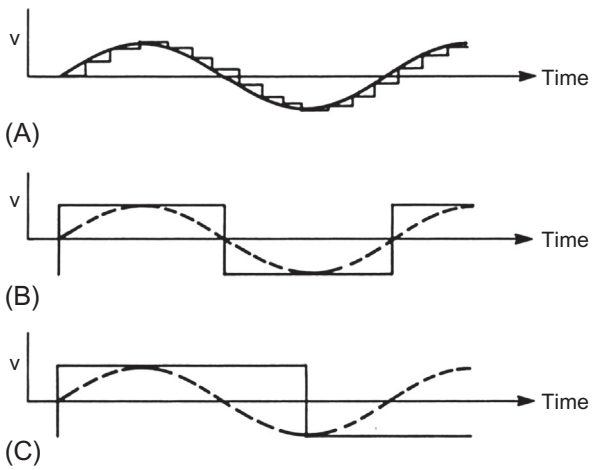


FIG. 3.19 Sampling rate illustration. (A) 13 samples per cycle, (B) 2 samples per cycle, and (C) less than 2 samples per cycle.

13 samples per sine wave cycle follows the sinusoid much closer than Fig. 3.19B, which only samples twice in a cycle. When the sampling rate is <2, as in Fig. 3.19C, aliasing errors occur as explained in Appendix B.

POLLING

The so-called stand-alone analog-to-digital converters are available that perform conversions independent of the direct involvement of the computer in the conversion process. The microcomputer outputs a control signal to cause the conversion to be initiated. At the EOC, the A/D outputs a signal when the conversion is done.

During the A/D or D/A process, the computer can run other operations. However, at the end of either conversion, the computer must be capable of obtaining the A/D data or outputting to a D/A converter. One-way of doing this is for the microcomputer to periodically check the interface while it is running another part of the program. This method is called *polling*. A subroutine is included in the main program and is called up whenever an A/D converter interface is being used. In a traditional assembly language program, this usually consists of a few lines of assembly language code that check to see if the interface is done and collect the result when it is finished. When the polling subroutine determines that the A/D converter is finished, the main program continues without using the polling subroutine until the A/D converter interface is called up again. The problem with such a scheme is that the polling routine may be called many times before the interface is finished. This is an inefficient use of computer capabilities and can degrade computer throughput. Therefore, an evaluation must be made in certain systems to determine if polling is worthwhile.

INTERRUPTS

An efficient alternative to polling uses control circuitry, called an *interrupt*. An interrupt is an electric signal that is generated outside of the CPU and is connected to an input on the CPU. The interrupt causes the CPU to temporarily discontinue the program execution and to perform some operation on data coming from an external device. A relatively slow A/D converter, for instance, could use an interrupt line to signal the processor when it has finished converting. When an interrupt occurs, the processor automatically jumps to a designated program location and executes the interrupt service subroutine. For the A/D converter, this would be a subroutine to read in the conversion result. When the interrupt subroutine is done, the computer returns to the point in the program before the interrupt occurred. Interrupts reduce the amount of time the computer spends dealing with the various peripheral devices relative to continuously monitoring them.

Another important use for interrupts is in timekeeping. Suppose that a system is being used that requires actions to be taken at particular absolute times; for instance, sampling an analog signal is a timed process. A special component called a timer could be used. A timer is a device that maintains absolute time. A square-wave clock signal is counted in counter registers like the one discussed in Chapter 2. The timer can be programmed to turn on the interrupt line when it reaches a certain count and then reset itself (start over). It may be inside the CPU itself, or it may be contained in peripheral devices in the microcomputer system. Timers have many automotive applications (as shown later).

Such a technique is sometimes used to trigger the output of a new number to a D/A converter at regular intervals such as at sample times. The microcomputer program includes routines to control

the timer for the desired amount of time by presetting the counter to some starting value other than zero. Each time the timer counts out the programmed number of its clock pulses, it interrupts the computer. The interrupt service subroutine then gets the new binary number that has been put into memory by the microcomputer and transfers this number to the D/A converter data latches.

VECTORED INTERRUPTS

All of the interrupt activity is completely invisible to the program that gets interrupted. In other words, the interrupted program does not contain data to indicate that it was interrupted because its execution continues without program modification with minimum delay. Interrupts allow the computer to handle two or more operations almost simultaneously. In some systems, one interrupt line may be used by more than one device. For instance, two or more A/D converters may use the same interrupt line to indicate when any of them are ready. In this case, the computer cannot identify which device caused the interrupt. The computer could poll all the devices each time an interrupt occurs to see which one needs service, but as discussed, polling may waste time. A better way is to use vectored interrupts.

In computer parlance, a *vector* is a memory location that contains another address that locates data or an instruction. It may be a specific memory location that contains the address of the first instruction of a subroutine to service an interrupt or it may be a register that contains the same type address. In this specific case, an interrupt vector is a register that peripherals use to identify which device caused the interrupt. When a peripheral causes an interrupt, it writes a code into the interrupt vector register so that the processor can determine which device interrupted it by reading the code. The decoder for an interrupt vector usually includes circuitry that allows each device to be assigned a different interrupt priority. If two devices interrupt at the same time, the processor will service the most important one first.

The vectored interrupt enables the microcomputer to efficiently handle the peripheral devices connected to it and to service the interrupts rapidly. Interrupts allow the processor to respond to operations in peripheral devices without having to constantly monitor the interfaces. They enable the microcomputer to handle many different tasks and to keep track of all of them. A microcomputer system designed to use interrupts is called a *real-time computing system* because it rapidly responds to peripherals as soon as requests occur. Such real-time systems are used in digital instrumentation and control systems in automotive applications.

MICROCOMPUTER APPLICATIONS IN AUTOMOTIVE SYSTEMS

There is a great variety of applications of microprocessors in automobiles. As will be explained in later chapters of this book, microprocessors find applications in engine and driveline control, instrumentation, ride control, antilock braking and other safety devices, entertainment, heating/air-conditioning control, automatic seat position control, and many other systems. In each of these applications, the microprocessor serves as the functional core of what can properly be called a special-purpose microcomputer.

Although these applications are widely varied in operation, the essential configuration (or *architecture*) has much in common for all applications. Fig. 3.20 is a simplified block diagram depicting the various components of each of the automotive systems having the applications listed previously.

In this block diagram, the microprocessor is denoted microprocessor/microcontroller units (MPU). It is connected to the other components by means of three buses: AB, DB, and CB. As explained above,

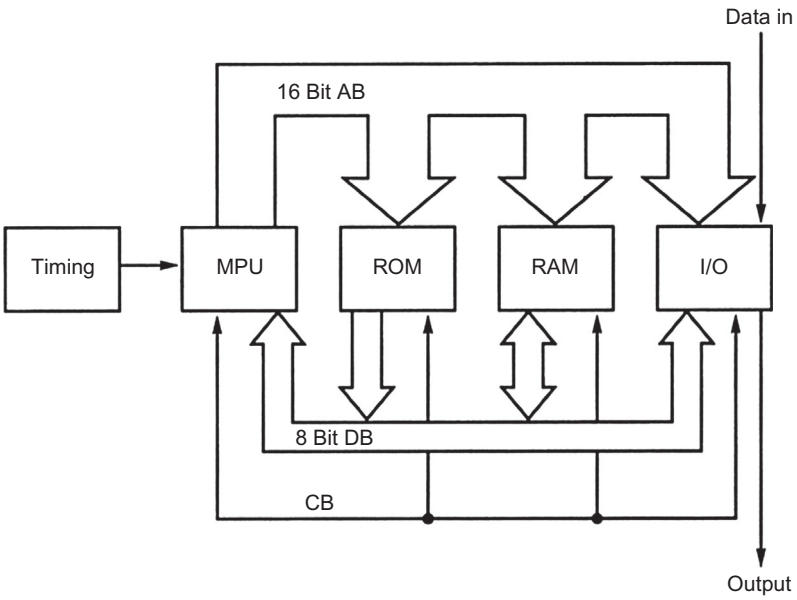


FIG. 3.20 Representative vehicular microcomputer block diagram.

each bus consists of a set of wires over which binary-valued electric signals are transmitted. By way of illustration, in early automotive applications, the DB consists of 8 wires. Although the size of the DB is much larger in contemporary vehicles, the AB is typically larger than the DB and the CB also is a set of wires, the number of which is determined by the complexity of the microprocessor.

The operation of each special-purpose microcomputer system is controlled by a program stored in ROM. As explained earlier in this chapter, the MPU generates addresses for the ROM in sequence to obtain each instruction in corresponding sequence. The operation of each microprocessor-based automotive subsystem has a specific program that is permanently stored (electronically) in the ROM. Changes in the system operation can be achieved by replacing the ROM chip(s) with new chip(s) that contain the appropriate program for the desired operation. This feature is advantageous during the engineering development phase for any microprocessor-based system. While the hardware remains fixed, the system modifications and improvements are achieved by substituting ROM chips. Rules from the EPA prohibit a vehicle user from making such ROM changes in any system that affects exhaust emissions. Only authorized repair personnel can legally and safely make such changes.

A typical automotive microprocessor-based system also incorporates some amount of RAM. This memory is used for a variety of purposes, including storing temporary results, storing the stack, and storing all of the variables, not to mention all of the other activities discussed earlier in this chapter.

The input/output (I/O) device for any given automotive microcomputer system serves as the interface connection of the microcomputer with the particular automotive system. Standard commercial I/O devices are available from the manufacturers of each microprocessor that are specifically configured to

work with that processor. These I/O devices are implemented as an IC chip and are very versatile in application. Such a typical I/O device has multiple data ports for connecting to peripheral devices and a port that is connected to the DB of the computer.

Fig. 3.21 is a block diagram of a typical commercial I/O device. In this device, there are two ports labeled A and B (which service BUS A or BUS B), respectively. These ports can be configured to act as either I/O, depending on the data in the data direction register. Normally, the correct code for determining direction is transferred to the I/O device from the microprocessor via the system DB.

Whenever the microprocessor is either to transfer data to the I/O device or to receive data from it, a specific address is generated by the processor. This address is decoded, using standard logic, to form an electric signal that activates the chip select inputs to the I/O. In addition, the read/write (R/W) output of the microprocessor is activated, causing data to be received (read) from a peripheral device or transmitted (write) to a peripheral device.

Recall from earlier in this chapter that this use of address lines to activate the I/O is known as *memory-mapped I/O*. In memory-mapped I/O, I/O of data is selected by reading from the I/O input address or writing to the I/O output address.

INSTRUMENTATION APPLICATIONS OF MICROCOMPUTERS

In instrumentation applications of microcomputers, the signal processing operations are performed numerically under program control. The details of the instrumentation application of microcomputers in-vehicle electronics are explained in Chapter 8. However, this section of the present chapter illustrates the generic configuration for microcomputers in vehicle instrumentation. The block diagram of a typical computer-based instrument is depicted in Fig. 3.22. In this example instrument, an analog sensor provides a continuous-time voltage, v_o, that is proportional to the quantity (x) being measured. The continuous-time voltage is sampled at times (t_k) determined by the computer. The sampled analog voltage is then converted to digital format using an A/D converter as explained above. The digital data are connected to port A of the I/O device of the computer to be read into memory.

Microcomputers can convert the nonlinear output voltage of some sensors into a linear voltage representation. The sensor output voltage is used to look up the corresponding linear value stored in a table. The A/D converter generates an EOC signal when the conversion from analog-to-digital is completed. Typically, the EOC signal provides an interrupt signaling the computer that data are ready as explained above.

The signal processing to be performed is expressed as a set of operations that is to be performed by the microprocessor on the data. These operations are written in an *algorithm* for the signal processing operation by the system designer. The algorithm is converted to a set of specific computer operations that becomes the program for the signal processing. After the signal processing is completed, the result is ready to be sent to the display device. The digital data are sent through I/O to port B to the D/A converter. There it is converted back to sampled analog as explained earlier in this chapter to drive the display. The sampled data often are "smoothed" to a suitable continuous-time voltage by means of a special filter known as a reconstruction filter. The continuous-time output of this filter drives the continuous-time display.

In a great many applications, the display is digital (e.g., automotive-fuel quantity measurement). In this case, the conversion from digital-to-analog is not required, and the computer output data can directly activate the digital display in the correct format.

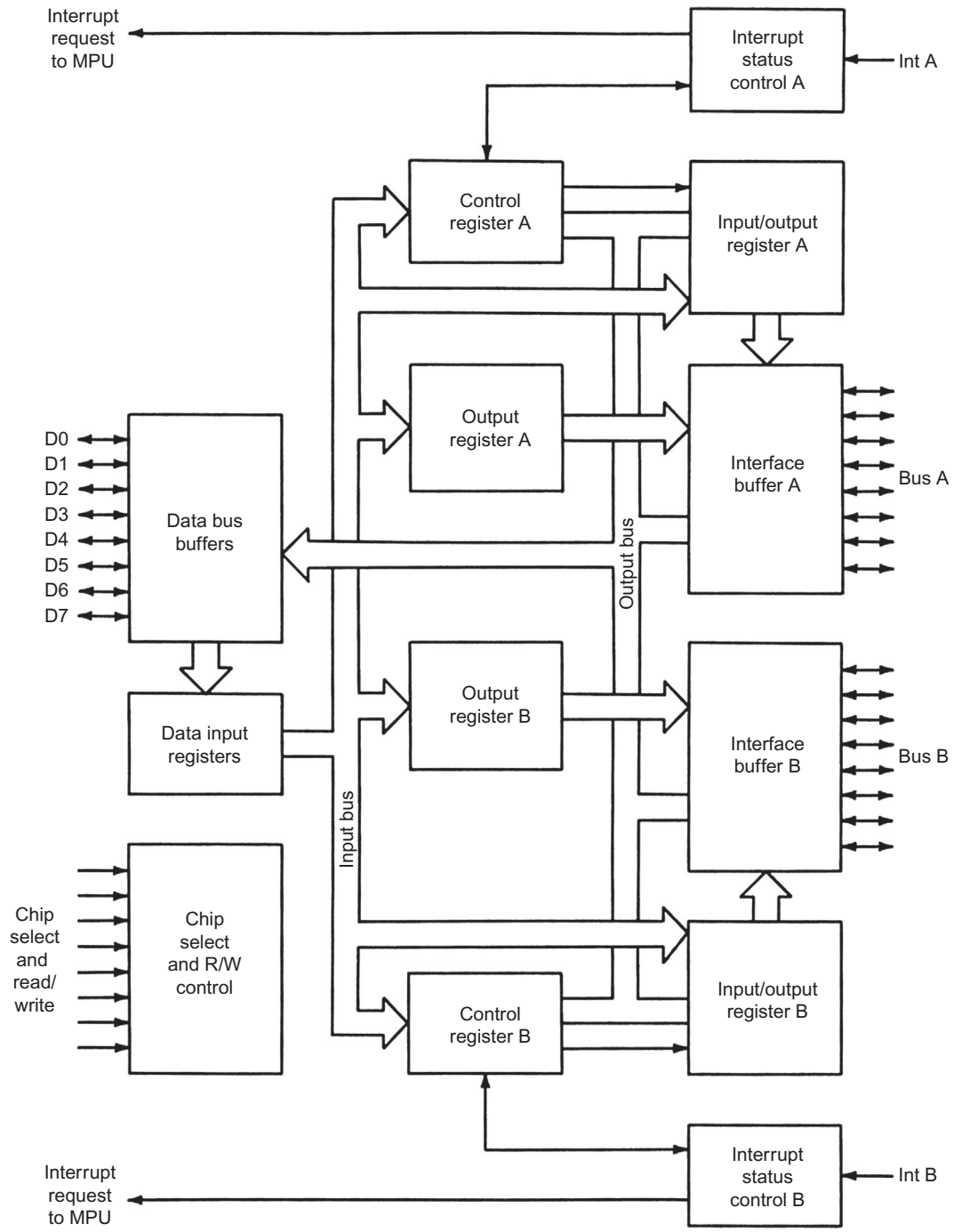


FIG. 3.21 Illustration of I/O ports.

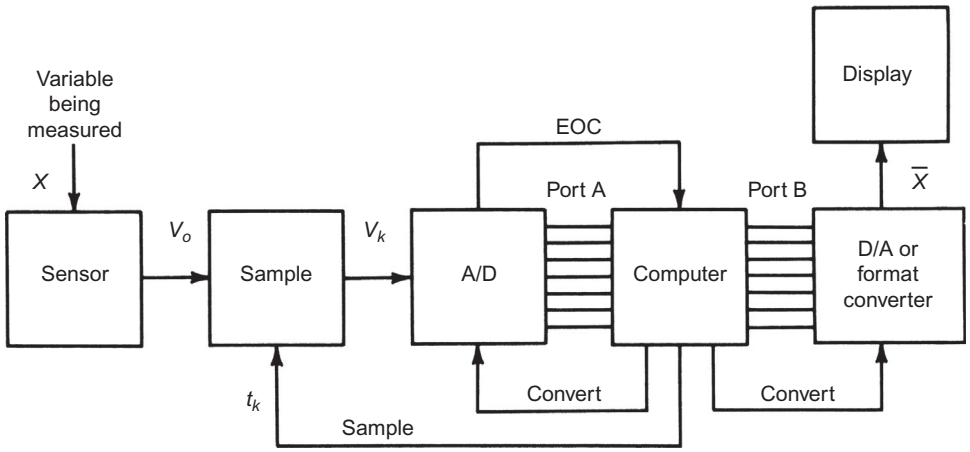


FIG. 3.22 Automotive digital instrumentation block diagram.

DIGITAL FILTERS

In Appendix B, the analysis/design of digital (discrete time) filters was explained. Here, some implementation issues are discussed with respect to automotive digital electronic systems, nearly all of which are accomplished using a microprocessor, either as a stand-alone system or as an operation embodied within a larger, multifunction digital system.

As an example of computer-based instrumentation signal processing applications, consider the relatively straightforward task of filtering the output of a sensor. As described in the Appendix A discussion of filters, low-pass filters pass low-frequency signals but reject high-frequency signals. High-pass filters do just the reverse: They pass high-frequency signals and reject low-frequency signals. Band-pass filters pass midrange frequencies but reject both low and high frequencies.

Appendix B discusses the concept of digital (discrete time) filters that perform filtering operations on samples x_k of the signal ($x(t)$) that are to be filtered. One relatively commonly used algorithm for a digital filter described in Appendix B is the recursive algorithm for calculating the output y_n, which is repeated here for convenience:

$$y_n = \sum_{k=1}^{K} a_k x_{n-k} - \sum_{j=1}^{J} b_j y_{n-j} \tag{3.14}$$

It is further shown that the so-called z-operational transfer function $H(z)$ is given by

$$H(z) = \frac{\sum_{k=1}^{K} a_k z^{-k}}{1 + \sum_{j=1}^{J} b_j z^{-j}} \tag{3.15}$$

Filtering the input signal sequence $\{x_k\}$ to calculate the output at sample time t_n (i.e., y_n) is done by the digital system under program control. The filter coefficients, a_k and b_j, are stored in ROM and read at

the appropriate time. The K previous input values from x_{n-1} to x_{n-K} must be stored in RAM along with previously calculated values of y_{n-j}. After input x_n has been read into the digital system, a program (subroutine) is called by the main program, which implements the recursive filter algorithm. Multiplication can be performed by repeated use, under program control, of the basic multiply subroutine described above. Once all products have been computed, the filter output y_n is obtained by addition or subtraction as indicated in the recursive filter algorithm. There are many automotive filter applications, each of which is implemented as described above.

A digital low-pass filter could be used, for instance, to smooth the output of an automotive-fuel-level sensor. The fuel-level sensor produces an electric signal that is proportional to the height of the fuel in the center of the tank as described in Chapter 5. The level at that point will change as fuel is consumed, and it also will change as the car slows, accelerates, turns corners, and hits bumps. The sensor's output voltage varies widely because of fuel slosh even though the amount of fuel in the tank changes slowly. If the sensor output voltage is sent directly to the fuel gauge, the resulting variable indication will fluctuate too rapidly to be read.

The measurement can be made readable and meaningful by using a low-pass filter to smooth out the signal fluctuations to reduce the effects of sloshing. The low-pass filter can be implemented in a microcomputer by programming the computer to average the sensor signal over several seconds before sending it to the display. For instance, if the fuel-level sensor signal is sampled once every second and it is desirable to average the signal over a period of K samples, the computer saves only the latest K samples, averages them, and displays the average. When a new sample is taken, the oldest sample is discarded so that only the K latest samples are kept. A new average can be computed and displayed each time a new sample is taken.

The algorithm for calculating the average $\bar{x}_N$ at time t_n of K previous samples of data x_k: $k = 1,2,\cdots,K$ is given by

$$\bar{x}_n = \frac{1}{K}\sum_{k=1}^{K} x_{n-k} \tag{3.16}$$

This algorithm is of the same structure as that used in a recursive filter (e.g., Eq. 3.14) in which all coefficients b_j are 0 and all coefficients a_k are $1/K$; that is to say, averaging a sequence of data samples $\{x_{n-k}\}$ is a form of filtering the data. Programming to compute this average involves some of the same steps of retrieval of data and forming an arithmetic average. The division by K can be performed quickly by multiplication of the sum of samples by the reciprocal of K (i.e., $1/K$), which value can be stored in ROM.

Digital filtering (e.g., averaging) can be performed by a computer under the control of the software. Sometimes the section of code that performs any such task is simply called "the filter." Digital signal processing is very attractive because the same computer can be used to process several different signals. During the engineering development of an automotive digital system, the desired filter characteristics often evolve. Such evolution can be readily implemented via changes in the stored filter coefficients. For any evolution of analog filters or signal processing, the hardware itself must be changed. For the digital filter, once the filter coefficients have been determined and filter performance is acceptable, the numerical values (i.e., a_k and b_j) are ready for storage in production vehicle ROM.

There are limitations to the use of digital filters however. The frequency range of digital filters is determined by the speed of the processor. The microcomputer must be able to sample each signal at or above the rate required by the Nyquist sampling theorem (see Appendix B). It must also be fast enough

to perform all of the averaging and linearization for each signal before the next sample is taken. This is an important limitation, and the system designer must be certain that the computer is not overloaded by trying to make it perform too many tasks too quickly.

MICROCOMPUTERS IN CONTROL SYSTEMS

Microcomputers are able to handle inputs and outputs that are either digital or converted analog signals. With the proper software, they are capable of making decisions about those signals and can react to them quickly and precisely. These features make microcomputers ideal for controlling other digital or analog systems, as discussed in the following sections.

CLOSED-LOOP CONTROL SYSTEM

The detailed theory of closed-loop control system is presented in Appendix A (continuous-time) and Appendix B (discrete time). A continuous-time control system performs the control law operations on the error signal to generate a continuous-time control signal (u) which is sent to the actuator via hardware. Appendix B explains that the discrete time system performs the control law calculation by performing operations on the sampled error between the desired and actual numerical values of the plant variable being controlled. The calculations to be performed to obtain the control variable (i.e., $\bar{u}_k$ of Appendix B) can readily be done in a digital computer under program control. The computer can compare command input and plant output and perform the computation required to generate a control signal.

LIMIT-CYCLE CONTROLLER

The limit-cycle controller, discussed in Appendix A, can be readily implemented with a microcomputer. Appendix A explains in detail that the limit-cycle controller controls the plant output so that it falls somewhere between an upper and lower limit, preferably so that its average value is equal to the command input. The controller must read in the command input and the plant output and determine via appropriate logic the value of the control signal to be sent to the plant based on those signals alone.

Using a microcomputer, the upper and lower limits can be determined from the command input by using a lookup table similar to that discussed later in this chapter. The plant output is compared with these two limits. If the plant output is above the higher limit or below the lower limit, the microcomputer outputs the appropriate on/off signal to the plant to bring the output back between the two limits.

FEEDBACK CONTROL SYSTEMS

In Appendices A and B, the concept of a feedback control system is introduced. Those appendices deal with the basic analytical models and control algorithms on an abstract and detailed level. In this chapter, the specific configuration incorporating a microcomputer as the control system implementation is considered.

A feedback control system can also be implemented using a digital computer and the limit-cycle controller. Fig. 3.23 shows the physical configuration of a control system employing a computer.

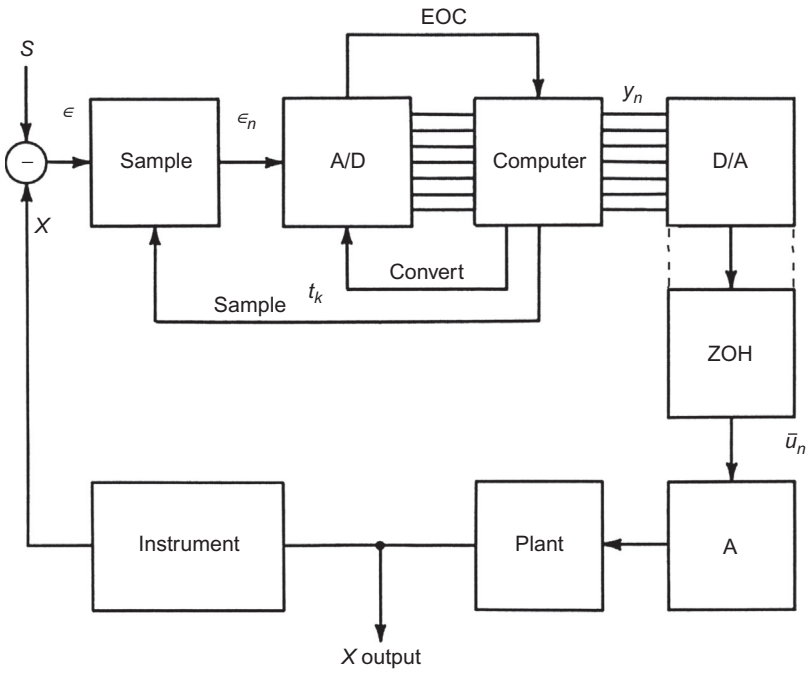


FIG. 3.23 Digital feedback control system block diagram.

In this figure, there is a physical system or plant that is to be controlled. The specific variable being controlled is denoted x. For example, in an automobile, the plant might be the engine, and the controlled variable might be engine speed. Examples of feedback control are presented in later chapters of this book.

The desired value for x is the set point s. An error signal $\in$ is obtained:

$$\in = s - x$$

The error signal is sampled, yielding samples $\in_n$ (where n represents sample number; i.e., $n = 1, 2, \cdots$). As explained in Appendix B, a representative value of a control algorithm is the PID control law by which an output y_n for each input sample is calculated by the computer:

$$y_n = K_p \in_n + \frac{K_I T}{2} \sum_{k=1}^{K} (\in_{n-k} + \in_{n-k-1}) + \frac{K_D (\in_n - \in_{n-1})}{T} \tag{3.17}$$

where K_p is the proportional gain, K_I is the integral gain, K_D is the differential gain, and T is the sample period.

In this PID controller, K is the number of samples from which the integral term is calculated. The program for implementing this exemplary PID control law involves temporary storage of K previous error samples for retrieval and computation of y_n. The same type of program steps for implementing

this control is used as those used for digital filter applications; that is, retrieval of variables, multiplication by the appropriate coefficients, and forming the algebraic sum of the various terms in the control law.

After computing y_n for each input sample, a digital version of y_n is transmitted through the I/O to the D/A converter and ZOH as explained in Appendix B. The dashed lines between the D/A and ZOH (see Chapter 2) blocks indicate that the ZOH may be implemented as part of the D/A. There it is converted to analog format, providing a control signal $\bar{u}_k$ to the actuator (A), which is presumed here to be analog. The actuator controls the plant in such a way as to cause the error to be reduced toward zero. Many examples of the application of computer-based electronic control systems in automobiles are presented in later chapters of this book.

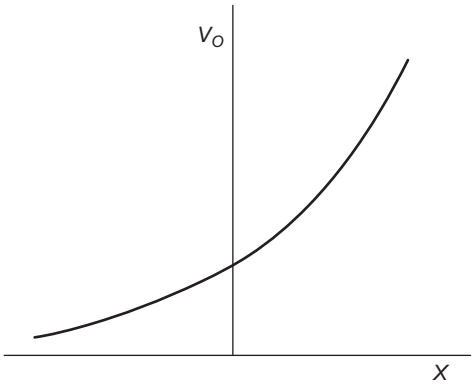
TABLE LOOKUP

One of the important functions of a microcomputer in automotive applications is table lookup. These applications include

1. linearization of sensor data,
2. multiplication,
3. calibration conversion.

The concept of table lookup is illustrated in Fig. 3.24, in which a pair of variables, V_o and X, are related by the graph depicted therein.

Also shown in Fig. 3.24 is a table listing certain specific values for the relationship. The functional relationship between V_o and X might, for example, be the output voltage of a nonlinear sensor V_o for measuring a quantity X. If the value for V_o is known, then the corresponding value for X can theoretically be found using the graph or the tabulated values. In the latter case, the nearest two tabulated values for V_o are located, and the corresponding values for X are read from the table. Denoting V_1



Table

V_O	X
0	–5
1	0
2	4
.	.
.	.
.	.

FIG. 3.24 Illustration of table lookup.

and V_2 as the nearest values for V_0 and X_1, X_2 as the corresponding tabulated values, assuming the tabulated values are sufficiently close, the value for X corresponding to V_o typically is found by linear interpolation:

$$X = X_1 + (X_2 - X_1)(V_0 - V_1)/(V_2 - V_1) \tag{3.18}$$

A microcomputer can perform the interpolation operation given above using tabulated values for the relationship between V_o and X (i.e., V_o (X)). This method is illustrated using a specific example of the measurement of a variable X using a sensor output voltage, and variable X is assumed to be that which is illustrated in Fig. 3.24. The table lookup operation can also be programmed to use a nonlinear interpolation algorithm or regression polynomial fit.

The portion of the microcomputer that is involved in the table lookup process is illustrated in Fig. 3.25. The relationship V_o (X) is stored in ROM for representative points along the curve. These data are stored using V_o values as addresses and corresponding values of X as data. For example, consider a point (V_1, X_1). The data X_1 are stored at memory location V_1 in binary format.

The operation of the table lookup is as follows. The sensor has output voltage V_o. The computer reads the values of V_o (using an A/D converter to convert to digital format) and reads the digital result through the I/O device. Then the MPU under program control calculates the addresses for the two

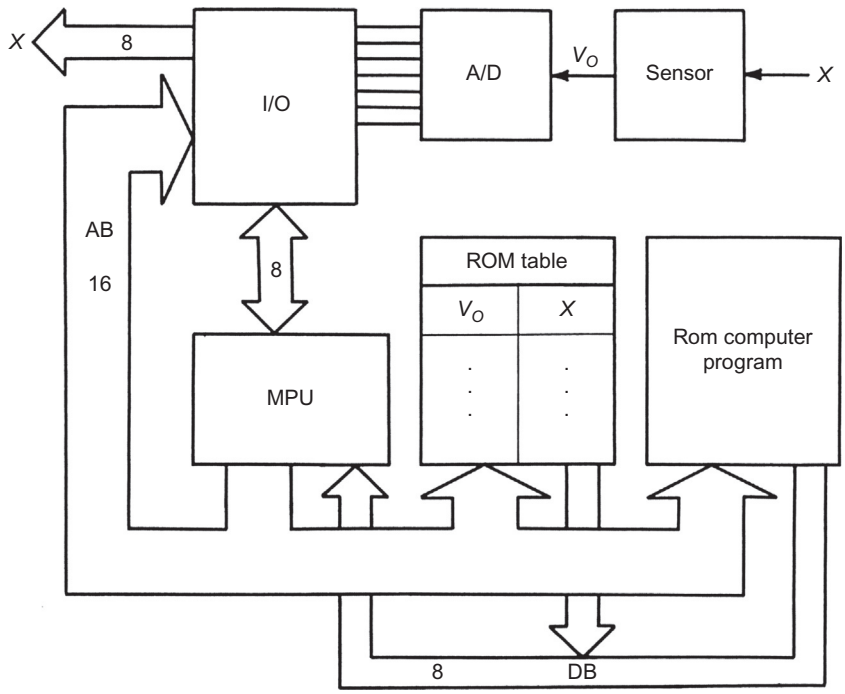


FIG. 3.25 Table lookup block diagram.

nearest tabulated values to V_o, which are V_1 and V_2 ($V_1 < V_o < V_2$). The computer, under program control, reads values X_1 and X_2 and then calculates X using Eq. (3.18) (or higher-order polynomial fit algorithm).

Repeated reference will be made to the table lookup function in later chapters. In particular, Chapter 6 will discuss how a typical digital engine control system frequently obtains data using table lookup.

MULTIVARIABLE AND MULTIPLE TASK SYSTEMS

A very important feature of microcomputer control logic is the ability to control multiple systems independently and to control systems with multiple inputs and outputs. The automotive applications for microcomputer control involve both of these types of *multivariable* systems. For instance, the automobile engine controller has several inputs (such as mass airflow rate, throttle angle, and camshaft and/or crankshaft angular position) and several outputs. All of the outputs must be controlled as close to simultaneously as is feasible within hardware capability and computation time limits because some inputs affect more than one output. These types of controllers can be very complicated and are difficult to implement in analog fashion. The increased complexity (and cost) of a multivariable microcomputer system is not much higher than for a single-variable microcomputer system, presuming the microcomputer has the capacity to do the task. It only affects the task of programming the appropriate control scheme into the microcomputer. In applications requiring a relatively high microcomputer throughput, it is possible to implement the microcomputer with multiple microprocessors often termed "multicore" processors. In such a multicore system, control is typically based in one of the core segments. This type of control is discussed in a later chapter.

The organization of the program for any computer performing multiple tasks simultaneously is extremely complex. One such organizational scheme involves having a so-called "main program loop." This main loop calls up appropriate subroutines for each of the tasks to be performed in sequence. The main loop continuously cycles at a rate that is determined by the computation time required for each task (subroutine). However, not all of the tasks need to be performed for each cycle through the main loop. Certain tasks such as fuel and spark control are required to be performed for every main loop cycle. Other, less time-critical tasks, such as filtering fuel quantity measurements, need to be performed at a much lower rate than the fuel and spark control tasks.

Other tasks such as diagnosis of problems with the vehicle subsystems are required to be performed only when a problem is detected (e.g., see Chapter 11). The main loop must be programmed to respond to signals that are generated when a problem is detected. These applications are explained in detail in later chapters where appropriate.

The development of a program for any automotive electronic system is normally very time consuming and requires the efforts of some very talented and capable computer programmers. Typically, a full program for the very complex power train control system (see Chapter 6) involves many thousands of individual lines of code. Some assistance in the form of automatic code generation is available from certain software (e.g., AUTOSAR), although a complete discussion of this subject is beyond the scope of this book.

After a chapter on basics of automotive engine control and a chapter on sensors and actuators, this book will deal more specifically with particular microcomputer automotive instrumentation and control systems to show how these systems are used in the automobile to control the engine and drivetrain

and many auxiliary functions. In addition, specific algorithms, along with dynamic performance calculations/simulations, are presented for selected applications. The programming of the subroutine for their implementation follows procedures discussed and explained in this chapter.

AUTOSAR

The remaining chapters of this book contain multiple examples of the application of MPU. Each electronically controlled subsystem or system in a contemporary vehicle has a module or control device that, for convenience, is euphemistically referred to as an electronic control unit (ECU). Each ECU has one or more MPUs that have a fundamental hardware structure that has been explained earlier in this chapter. Moreover, it has been emphasized that an MPU performs its intended operation within the ECU under control of a stored program. The stored program consists of sequences of instructions in a binary format that can perform the operation in the MPU. A program in a format capable of causing the MPU to perform the necessary operations is termed "executable code" or "machine language program." This executable code is stored in ROM-type memory.

During the development of a vehicular ECU, there are multiple steps taken by the individual or team that is (are) responsible for designing the given ECU. This development team that is responsible for designing the hardware also generates the algorithms related to the operation of the ECU. The algorithms must be used to create the executable code that constitutes the controlling program. In the early days of the application of MPUs to the equivalent of an ECU, the programming of the algorithms was accomplished by writing the program in the assembly language of the MPU. The executable code was created using a program (which ran on a development system) called an assembler. This assembler would create one or more lines of executable code for each assembly language instruction (as illustrated earlier in this chapter).

In addition to performing algorithms, ECU software provides a variety of other operations. For example, the ECU software must control the input of data as well as the output signals that operate the components that the ECU is controlling. In addition, the stored program must contain parameters that are specific to the overall system performance and that, for example, are part of the associated algorithms. As explained in Chapter 11, the programming any ECU can assist in the diagnosis of system and even overall vehicle diagnosis of problems.

It was not long in the evolution of vehicular electronics before the program to run an ECU was created with a high-level language (e.g., C). However, the rapid increase in the complexity of vehicular electronics inspired a significant jump in the efficiency of programming via the creation of Automotive Open System Architecture (AUTOSAR). AUTOSAR is essentially a consortium between automotive Original Equipment Manufacturers (OEMs) and various suppliers of electronic systems/components that began in 2003 that has permitted collusion free cooperation between the various members on software generation. It has the potential to improve the efficiency of ECU code generation by having standardized modules that can be adapted and employed in the software generation for a new ECU or for improvements in an existing ECU.

Although a given ECU for a vehicular electronically controlled subsystem can, in principle, be fabricated by a division of the OEM, it is typical for the production ready ECU to be provided by one or more top tier supplier(s). In the latter case, the OEM provides a sufficient description of the ECU

functionality (which can include operating algorithms) and system configuration that programming by the supplier can be done via AUTOSAR.

The complete program for running a given ECU along with its interaction with other systems consists of software modules that, together, form the AUTOSAR architecture. Each module contains standardized basic software modules (BSW in AUTOSAR) that perform functions commonly found in digital systems. Any digital system working with analog inputs and/or outputs requires the function of analog/digital and vice versa conversion, as well a sampling and, often, multiplexing, bus communication, memory management, etc.

One of the elements of AUTOSAR architecture is called software components with AUTOSAR abbreviation SWCs. Each SWC has interfaces to the required BSW that are specified formally within the SWC. The control of connections between various SWCs and to/from BSW modules is an AUTOSAR element call the runtime environment (RTE). These connections are handled via an AUTOSAR element called the virtual function bus (VFB).

The description of software component is done via an XML file (AUTOSAR XML) that contains sufficient configuration information and data to create the programs for any given ECU (or for multiple ECUs and their interaction during vehicle operation). Among the necessary files are the BSW module descriptions. Normally, the necessary XML files come from the OEM.

In the contemporary versions of AUTOSAR, the functional software for the entire electronic systems of a given vehicle is described. This entire system is partitioned into individual SWCs. The connection/communication between SWCs (even in separate ECUs) is accomplished by the VFB. The VFB is implemented by the RTE, which is specific to any given ECU. The executable code for a given function can be implemented by the AUTOSAR as a C function that is ultimately done by the RTE.

In contemporary vehicles, the addition of a new ECU or the modification of a new ECU to a given or new version of an existing vehicle model, the AUTOSAR software developer requires description/modeling of the complete vehicle electronic system. As explained in Chapter 9 on vehicular communication, a given vehicle model will have one or (usually) more in-vehicle networks (IVNs) that support all digital communications between the various individual ECUs or subsystems. The interconnections between electronic systems via IVNs with all associated protocols (see Chapter 9), for example, bus data rates and data structure, must be specified in the documents provided to the AUTOSAR software developer. A description/model for the hardware and system topology is also necessary including sensors actuators (as explained in Chapter 5) and the individual microprocessors/microcontrollers. Once the vehicular electronic system description is adequately specified, for any ECU the BSW and RTE for that ECU is assembled and is specific to that particular ECU. From this assembly, the executable software code is produced by AUTOSAR. Eventually, the code required to run any given ECU can be stored in ROM as part of the final ECU configuration.

There are multiple benefits to programming vehicular ECUs or subsystems via AUTOSAR. The reuse of individual software or BSW elements increases the efficiency of programming any new digital electronic system. The reuse of portions of code and the automatic code generation can significantly reduce development time and costs compared with the traditional methods of programming for digital electronic systems. The full details of AUTOSAR are beyond the scope of this book. However, AUTOSAR provides documentation that is available online, for example, Autosar_ppt and AUTOSAR_EXP_LayeredSoftwareArchitecture.

CHAPTER

THE BASICS OF ELECTRONIC ENGINE CONTROL

4

CHAPTER OUTLINE

Motivation for Electronic Engine Control 136
Exhaust Emissions 136
Fuel Economy 137
Federal Government Test Procedures 137
Fuel Economy Requirements 140
Meeting the Requirements 141
The Role of Electronics 141
Concept of an Electronic Engine Control System 142
Inputs to Controller 144
Output from Controller 145
Basic Principle of Four-Stroke Engine Operation 146
Definition of Engine Performance Terms 150
Torque 150
Power 153
Fuel Consumption 154
Engine Overall Efficiency 156
Calibration 156
Engine Mapping 157
Effect of Air/Fuel Ratio on Performance 157
Effect of Spark Timing on Performance 158
Effect of EGR on Performance 159
Exhaust Catalytic Converters 161
Oxidizing Catalytic Converter 161
The Three-Way Catalyst 162
Electronic Fuel Control System 164
Engine Control Sequence 166
OL Control 167
CL Control 167
CL Operation 169

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-12-810434-7.00004-1
Copyright © 2017 Elsevier Inc. All rights reserved.

Analysis of Intake Manifold Pressure 172
Measuring Air Mass 173
Influence of Valve System on Volumetric Efficiency 175
Idle Speed Control 176
Electronic Ignition 181

Engine control in the vast majority of engines means regulating fuel and air intake and spark timing to achieve desired performance in the form of power output. Until the 1960s, control of the engine output torque and RPM was accomplished through some combination of mechanical, pneumatic, or hydraulic systems. Then, in the 1970s, electronic control systems were introduced.

This chapter is intended to explain, in general terms, the theory of electronic control of a gasoline-fueled, spark-ignited automotive engine. Chapter 6 explains practical digital control methods and systems. The examples used to explain the major developments and principles of electronic control have been culled from the techniques of various manufacturers and do not necessarily represent any single automobile manufacturer at the highest level of detail. Moreover, Chapter 6 presents major improvement in electronic control of the entire power train. However, the most basic aspects of engine control are presented in this chapter in preparation for the detailed explanation of contemporary engines.

MOTIVATION FOR ELECTRONIC ENGINE CONTROL

The initial motivation for electronic engine control came, in part, from two government requirements. The first came about as a result of legislation to regulate automobile exhaust emissions under the authority of the Environmental Protection Agency (EPA). The second was a thrust to improve the national average fuel economy by government regulation. The issues involved in these regulations along with normal market forces continue to motivate improvements in reduction of regulated gases and fuel economy. Electronic engine control is only one of the automotive design factors involved in fuel economy improvements. However, this book is only concerned with the electronic systems.

EXHAUST EMISSIONS

Although diesel engines are in common use in heavy trucks, railroads, and some pickup trucks, the gasoline-fueled engine is the most commonly used engine for passenger cars and light trucks in the United States. This engine is more precisely termed the gasoline-fueled, spark-ignited, four-stroke/cycle, normally aspirated, liquid-cooled internal combustion engine. It is this engine, which is denoted the SI engine, that is discussed in this book. The following discussion of exhaust emission regulations applies to the SI engine.

The engine exhaust consists of the products of combustion of air and gasoline mixture. Gasoline is a mixture of chemical compounds that are called *hydrocarbons*. This name is derived from the chemical formation of the various gasoline compounds, each of which is a chemical union of hydrogen (H) and carbon (C) in various proportions. Gasoline also contains natural impurities and chemicals added by the refiner. All of these can produce undesirable exhaust elements. The combustion of gasoline in an engine results in exhaust gases, including CO_2, H_2O, CO, oxides of nitrogen, and various hydrocarbons.

During the combustion process, the carbon and hydrogen combine with oxygen from the air, releasing heat energy and forming various chemical compounds. If the combustion were perfect, the exhaust gases would consist only of carbon dioxide (CO_2) and water (H_2O), neither of which is considered harmful to human health in the atmosphere. In fact, both are present in an animal's breath.

Unfortunately, the combustion of the SI engine is not perfect. In addition to the CO_2 and H_2O, the exhaust contains amounts of carbon monoxide (CO), oxides of nitrogen (chemical unions of nitrogen and oxygen that are denoted NO*x*), unburned hydrocarbons (HC), oxides of sulfur, and other compounds. Some of the exhaust constituents are considered harmful and are now under the control of the federal government. The exhaust emissions controlled by government standards are CO, HC, and NO*x*.

Automotive exhaust emission control requirements began in the United States in 1966 when the California state regulations became effective. Since then, the federal government has imposed emission control limits for all states, and the standards became progressively tighter throughout the remainder of the twentieth century and will continue to tighten in the 21st century. Auto manufacturers found that the traditional engine controls could not control the engine sufficiently to meet these emission limits and maintain adequate engine performance at the same time, so they turned to electronic controls.

FUEL ECONOMY

Everyone has some idea of what fuel economy means. It is related to the number of miles that can be driven for each gallon of gasoline consumed. It is referred to as miles per gallon (MPG) or simply *mileage*. In addition to improving emission control, another important feature of electronic engine control is its ability to improve fuel economy.

It is well recognized by layman and experts alike that the mileage of a vehicle is not unique. Mileage depends on the size, shape, and weight of the car and how the car is driven. The best mileage is achieved under steady cruise conditions. City driving, with many starts and stops, yields worse mileage than steady highway driving. In order to establish a regulatory framework for fuel economy standards, the federal government has established hypothetical driving cycles that are intended to represent how cars are operated on a sort of average basis.

The government fuel economy standards are not based on one car but are stated in terms of the average rated MPG fuel mileage for the production of all models by a manufacturer for any year. This latter requirement is known in the automotive industry by the acronym CAFE or corporate average fuel economy. It is a somewhat complex requirement and is based on measurements of the fuel used during a prescribed simulated standard driving cycle.

FEDERAL GOVERNMENT TEST PROCEDURES

For an understanding of both emission and CAFE requirements, it is helpful to review the standard cycle and how the emission and fuel economy measurements were made in the earliest days of emission control. The US federal government published test procedures that included several steps. The first step was to place the automobile on a chassis dynamometer, like the one shown in Fig. 4.1.

In many states, the government requires a yearly measurement of exhaust emissions with the vehicle placed on a *chassis dynamometer* and operated with a specific set of load and speed conditions

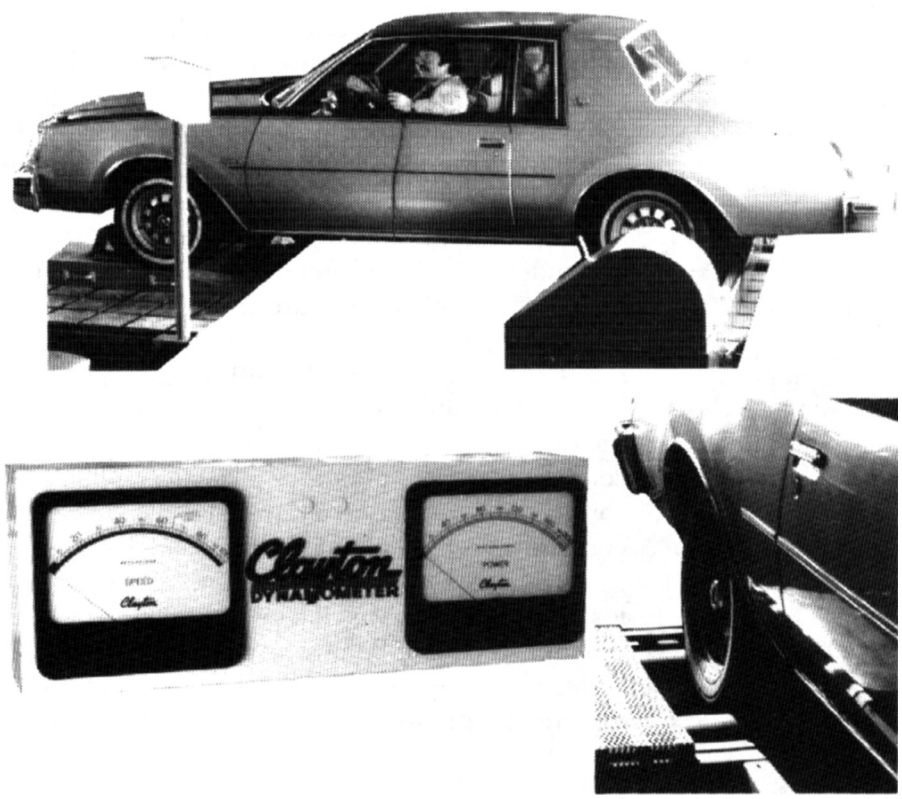


FIG. 4.1 Chassis dynamometer.

simulating normal vehicle operation. A chassis dynamometer is a test stand that holds a vehicle such as a car or truck. It is equipped with instruments capable of measuring the power that is delivered at the drive wheels of the vehicle under various conditions. The vehicle is held on the dynamometer so that it cannot move when power is applied to the drive wheels. The drive wheels are in contact with two large rollers. One roller is mechanically coupled to an electric generator that can vary the load on its electric output. The other roller has instruments to measure and record the vehicle speed. The generator absorbs and provides a measurement of all mechanical power that is delivered at the drive wheels to the dynamometer. The power is calculated from the electric output in the correct units of kW or hp (horsepower where 1 hp = 0.746 kW). The controls of the dynamometer can be set to simulate the correct load (including the effects of tire rolling resistance and aerodynamic drag) and inertia of the vehicle moving along a road under various conditions. The conditions are the same as if the vehicle actually was being driven except for wind loads.

The vehicle is operated according to a prescribed schedule of speed and load to simulate the specified trip. One driving cycle simulates an urban trip, and another simulates a highway trip. Over the years, the hypothetical driving cycles for urban and rural trips have evolved. Fig. 4.2 illustrates sample

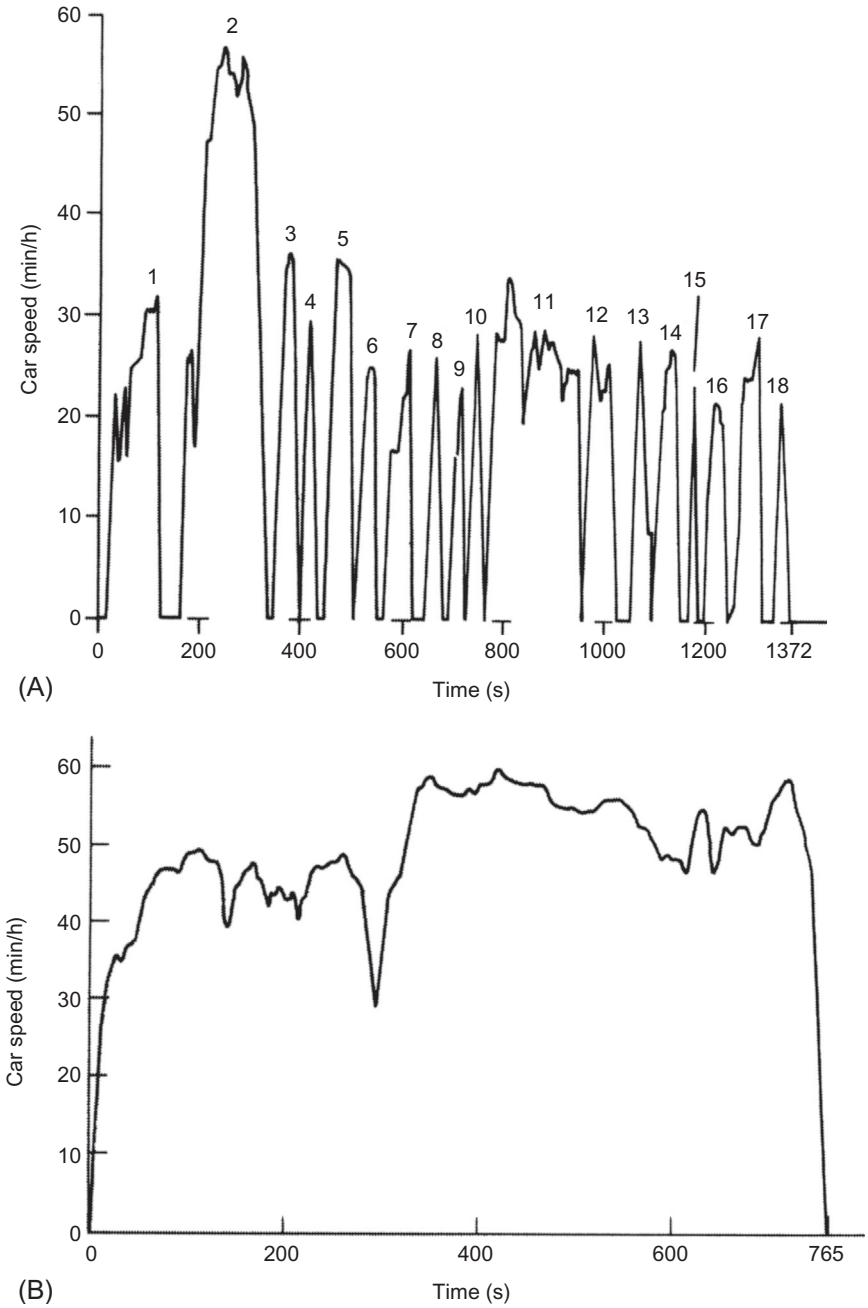


FIG. 4.2 Federal driving schedules (Title 40 US Code of Federal Regulations). (A) Urban and (B) Highway.

driving cycle trips (one for each) that demonstrate the differences in those hypothetical test trips. It can be seen that the urban cycle trip involves acceleration, deceleration, stops, starts, and steady cruise such as would be encountered in a "typical" city automobile trip of 7.45 mi (12 km). The highway schedule takes 765 s and simulates 10.24 mi (16.5 km) of highway driving.

During the operation of the vehicle in the tests, the exhaust is continuously collected and sampled. At the end of the test, the absolute mass of each of the regulated exhaust gases is determined. The regulations are stated in terms of the total mass of each exhaust gas divided by the total distance of the simulated trip.

FUEL ECONOMY REQUIREMENTS

In addition to emission measurement, each manufacturer must determine the fuel consumption in MPG for each type of vehicle and must compute the CAFE for all vehicles of all types produced in a year. Fuel consumption is measured during both an urban and a highway test, and the composite fuel economy is calculated.

Table 4.1 is a summary of the exhaust emission requirements and CAFE standards for a few representative years. It shows the emission requirements and increased fuel economy required, demonstrating that these regulations have become and will continue to become more stringent with passing time. Not shown in Table 4.1 is a separate regulation on nonmethane hydrocarbon (NMHC). Because of these requirements, each manufacturer has a strong incentive to minimize exhaust emissions and maximize fuel economy for each vehicle produced.

New regulations for emissions have continued to evolve and encompass more and more vehicle classes. Present-day regulations affect not only passenger cars but also light utility vehicles and both heavy- and light-duty trucks. Furthermore, regulations apply to a variety of fuels, including gasoline, diesel, natural gas, and alcohol-based fuels involving mixtures of gasoline with methanol or ethanol.

As an example, we present below the standards that were written for the vehicle half-life (5 years or 50,000 mi—whichever comes first) and full-life cycle (10 years or 100,000 mi) as of 1990. The standards were as follows:

HC	0.31 g/mi
CO	4.20 g/mi
NO*x*	0.60 g/mi (nondiesel) 1.25 g/mi (diesel)

Table 4.1 Emission and MPG Requirements

Year	Federal HC/CO/NO_x	California HC/CO/NO_x	CAFE MPG
1968	3.22/33.0/–	–	–
1971	2.20/23.0/–	–	–
1978	1.50/15.0/2.0	0.41/9.0/1.5	18.0
1979	1.50/15.0/2.0	0.41/9.0/1.5	19.0
1980	0.41/7.0/2.0	0 41/9 0/1 5	20.0
1989	0.31/4.1/1.0	0.31/4.1/1.0	27.5

These regulations were phased in according to the following schedules:

Model year 1994, 40%
Model year 1995, 80%
Model year 1996, 100%

There are many details to these regulations that are not relevant to the present discussion. However, the regulations themselves are important in that they provided motivation for expanded electronic controls.

MEETING THE REQUIREMENTS

Unfortunately, as seen later in this chapter, meeting the government regulations causes some sacrifice in performance. Moreover, attempts to meet the standards exemplified by Table 4.1 using mechanical, electromechanical, hydraulic, or pneumatic controls like those used in pre-emission control vehicles have not been cost-effective. In addition, such controls cannot operate with sufficient accuracy across a range of production vehicles, overall operating conditions, and over the life of the vehicle to stay within the tolerance required by the EPA regulations. Each automaker has had to verify that each model produced will still meet emission requirements after traveling 100,000 mi. As in any physical system, the parameters of automotive engines and associated peripheral control devices can change with time. An electronic control system has the ability to automatically compensate for such changes and to adapt to any new set of operating conditions and made electronic controls a desirable option in the early stages of emission control.

THE ROLE OF ELECTRONICS

The use of digital electronic control has enabled automakers to meet the government regulations by controlling the system accurately with excellent tolerance. In addition, the system has long-term calibration stability. As an added advantage, this type of system is very flexible. Because it uses microcomputers, it can be modified through programming changes to meet a variety of different vehicle/engine combinations. Critical quantities that describe an engine can be changed relatively easily by changing data stored in the system's computer memory.

Additional cost incentive

Besides providing control accuracy and stability, there is a cost incentive to use digital electronic control. The system components—the multifunction digital integrated circuits—are decreasing in cost, thus decreasing the system cost. From about 1970 on, considerable investment was made by the semiconductor industry for the development of low cost, multifunction integrated circuits. In particular, the microprocessor and microcomputer have reached an advanced state of capability at relatively low cost. This has made the electronic digital control system for the engine and other onboard automobile electronic systems commercially feasible. As pointed out in Chapter 2, as multifunction digital integrated circuits continue to be designed with more and more functional capability through very large-scale integrated circuits (VLSI), the costs continue to decrease. At the same time, these circuits offer improved electronic system performance in the automobile.

In summary, the electronic engine control system duplicates the function of conventional legacy fluidic control systems but with greater precision and long-term stability via adaptive control processes.

It can optimize engine performance while meeting the exhaust emission and fuel economy regulations and can adapt to changes in the plant.

CONCEPT OF AN ELECTRONIC ENGINE CONTROL SYSTEM

In order to understand electronic engine control, it is necessary to understand some fundamentals of how the power produced by the engine is controlled. Any driver understands intuitively that the throttle directly regulates the power produced by the engine at any operating condition. It does this by controlling the airflow into the engine.

In essence, the engine is an air pump such that at any rotational speed RPM, the mass flow rate of air into the engine varies directly with throttle plate angular position (see Fig. 4.3).

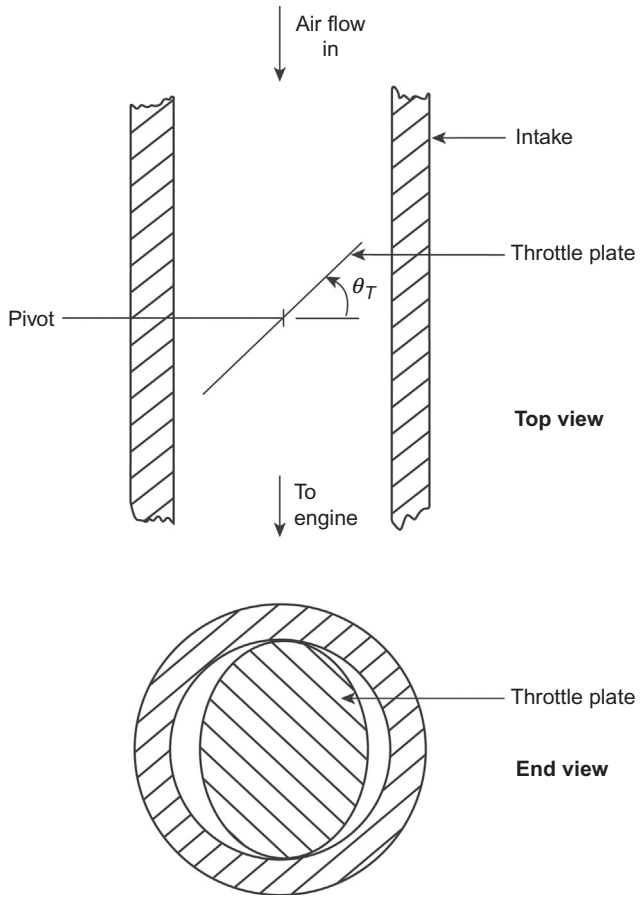


FIG. 4.3 Intake system with throttle plate.

As the driver depresses the accelerator pedal, the throttle angle (θ_T in Fig. 4.3) increases, which increases the cross-sectional area through which the air flows, reducing the resistance to airflow, thereby allowing an increased airflow into the engine. A model for the airflow versus throttle angle and engine RPM is given later in this chapter. The role of fuel control is to regulate the fuel that is mixed with the air so that it increases in proportion to the airflow. As we will see later in this chapter, the performance of the engine is affected strongly by the mixture (i.e., by the ratio of air-to-fuel). However, for any given mixture, the power produced by the engine is directly proportional to the mass flow rate of air into the engine. In the US system of units (in the early days of emission control) as a rough "rule of thumb," an airflow rate of about 6 lb/h produces 1 hp of usable mechanical power at the output of the engine. Metric units have come to be more commonly used, in which engine power is given in kilowatts (kW) and air mass is given in kilograms (kg).

Denoting the power from the engine P_b, the linear model for engine power is given by

$$P_b = K\dot{M}_A$$

where P_b is the power from the engine (hp or kW), $\dot{M}_A$ the mass airflow rate (MAF) (kg/sec) or (slugs/sec), and K the constant relating power to airflow (kW/kg/sec) or (hp/lb/sec). Of course, it is assumed that all parts of the engine, including fuel delivery and ignition timing, are functioning correctly for this relationship to be valid.

We consider next an electronic engine control system that regulates fuel flow to the engine. An electronic engine control system is an assembly of electronic and electromechanical components that continuously varies the fuel and spark settings in order to satisfy government exhaust emission and fuel economy regulations. Fig. 4.4 is a block diagram (at the most abstract level) of a generalized electronic engine control system.

It will be explained later in this chapter that an automotive engine control has both open-loop (OL) and closed-loop (CL) operating modes. As explained in Appendix A, a CL control system requires measurements of certain output variables such that the controller can calculate the state of the system being controlled, whereas an OL system does not. The electronic engine control system receives input electric signals from the various sensors that measure the state of the engine. From these signals, the controller generates output electric signals to the actuators that determine the correct fuel delivery and spark timing.

Models for and performance analysis of automotive engine control system sensors and actuators are discussed in Chapter 5. As mentioned, the configuration and control for an automotive engine control system are determined in part by the set of sensors that are available to measure the variables. In many cases, the sensors available for automotive use involve compromises between performance and cost. In other cases, only indirect measurements of certain variables are feasible. From measurement of these

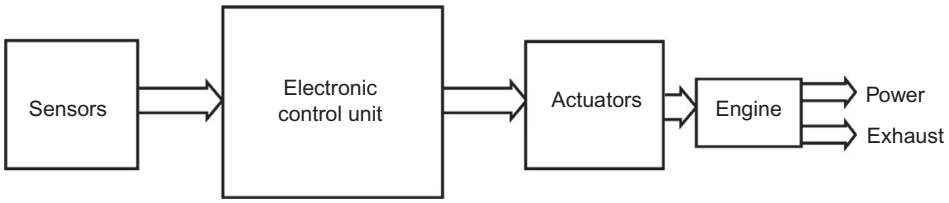


FIG. 4.4 Generic electronic engine control system.

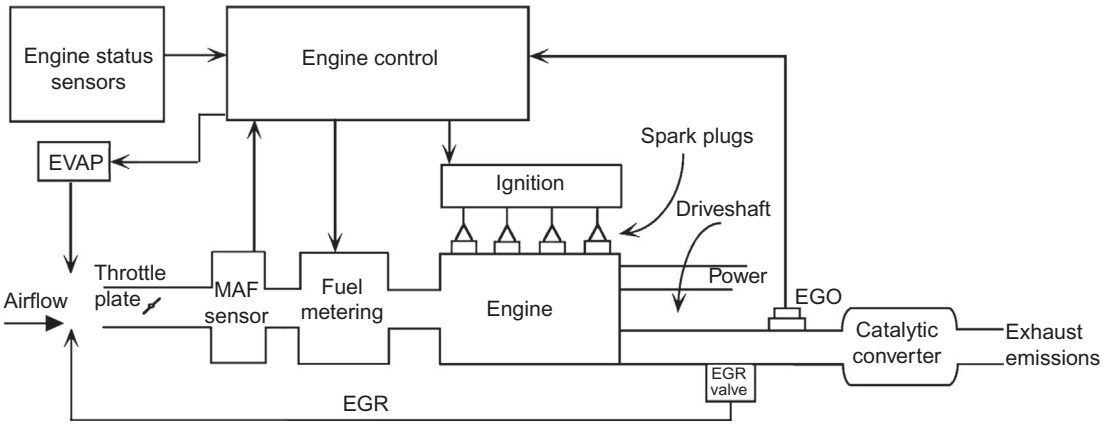


FIG. 4.5 Engine functions and control diagram.

variables, the desired variable is found by computation. Chapter 5 presents explanations of the sensor used for vehicular electronic control (and instrumentation). However, for the present chapter, it is assumed that each depicted sensor performs the necessary measurement.

Fig. 4.5 is a form of overall engine electronic control at a very abstract level.

There is a fuel metering system to set the air-fuel mixture flowing into the engine through the intake manifold. Spark control determines when the air-fuel mixture is ignited after it is compressed in the cylinders of the engine. The power is delivered at the driveshaft and the gases that result from combustion flow out from the exhaust system. In the exhaust system, there is a valve to control the amount of exhaust gas being recirculated back to the input and a catalytic converter to further control emissions. The addition of recirculated exhaust gas to the engine intake and various sensors and actuators depicted in Fig. 4.5 is explained later in this chapter. In addition, there is a subsystem that collects the evaporating fuel vapors in the fuel tank to prevent them from being vented to the atmosphere. These fuel vapors are later sent to the intake system as a small component of fuel being supplied to the engine. This subsystem is denoted EVAP in Fig. 4.5.

At the early stage of development, the electronic engine control consisted of separate subsystems for fuel control, spark control, and exhaust gas recirculation (EGR). The ignition system in Fig. 5.5 is shown as a separate control system, although engine control has evolved toward an integrated digital system (see Chapter 6).

INPUTS TO CONTROLLER

Fig. 4.6 identifies the major physical quantities that are sensed and provided to the traditional electronic controller as inputs. They are as follows:

1. Throttle position sensor (TPS)
2. Mass airflow rate (MAF)
3. Engine temperature (coolant temperature) (CT)

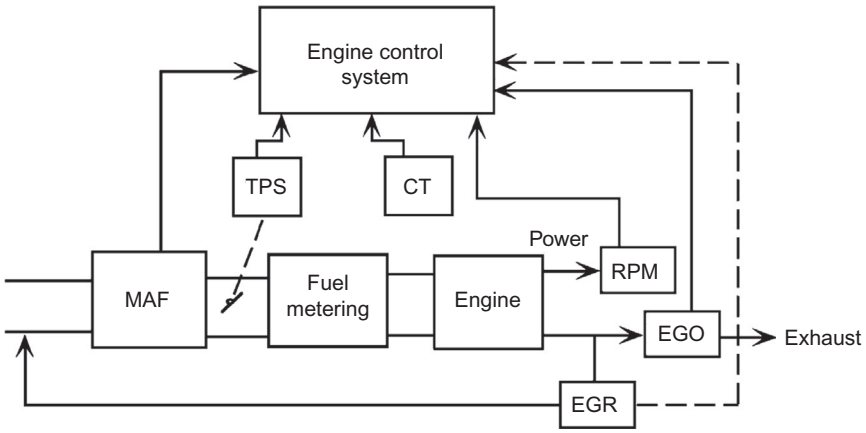


FIG. 4.6 Major controller inputs from engine.

4. Engine speed (RPM) and angular position
5. Exhaust gas recirculation (EGR) valve position
6. Exhaust gas oxygen (EGO) concentration

OUTPUT FROM CONTROLLER

Fig. 4.7 identifies the major physical quantities that are outputs from the controller. These outputs are the following:

1. Fuel metering control
2. Ignition control (dwell and timing)

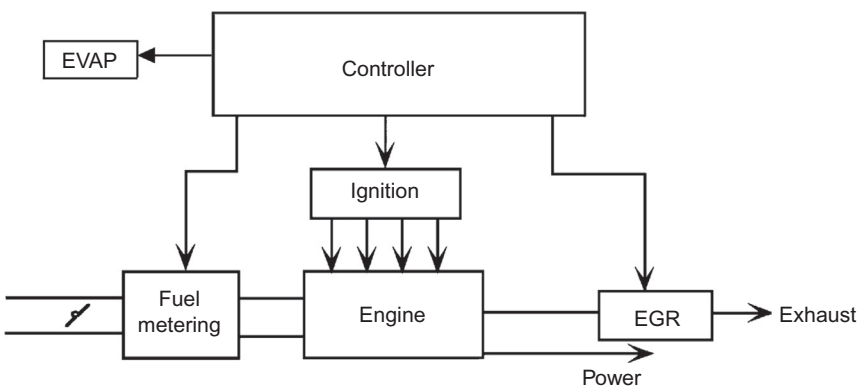


FIG. 4.7 Major controller outputs to engine.

3. Exhaust gas recirculation control
4. Fuel tank evaporative emission control (EVAP)

This chapter discusses the various electronic engine control functions separately and explains how each function is implemented by a separate control system. Chapter 6 shows how these separate control systems are being integrated into one system and are implemented with digital electronics.

BASIC PRINCIPLE OF FOUR-STROKE ENGINE OPERATION

For certain readers of this book, a brief review of engine configuration and operation may be helpful. Although several types of engines have found application as the prime mover in automobiles, the one most commonly used continues to be the multicylinder, four-stroke IC engine as explained earlier in the chapter. The configuration and operation of electric propulsion (e.g., in hybrid vehicle) are discussed in Chapter 5.

The configuration of a single cylinder of an IC engine is depicted in Fig. 4.8A. Mechanical power is produced by the engine in the form of torque acting on the rotating crankshaft. There are four basic engine processes that occur during the two complete revolutions of the crankshaft that occur during any single cycle of operation. This engine configuration includes a component called the piston, which fits within a cylinder and is mechanically linked to the crankshaft by the connecting rod. Airflow into and out of the cylinder is controlled by poppet valves (simply called the valves here). One of these is termed the intake valve and the other the exhaust valve. Additional components of the engine include a so-called intake port system consisting of a system of passageways (e.g., tubes) that direct fuel/air mixture into the engine and a so-called exhaust port system that directs the products of combustion out of the engine. Chapter 6 explains the operation and theory of certain contemporary engines for which fuel is injected directly into the cylinder.

During any single cycle of engine operation (involving two complete rotations of the crankshaft), there are four portions of the crankshaft rotation called strokes. Each stroke corresponds to piston (reciprocating) motion between its highest point (called "top dead center" or TDC) and the lowest point (called "bottom dead center" or BDC). The piston axial displacement between TDC and BDC is $L = 2R$, where R is the radius from the axis of rotation of the crankshaft to the center of the journal. These four strokes of any given engine cycle are termed intake, compression, power, and exhaust strokes. During the intake stroke, the piston moves from TDC to BDC. During most of this stroke, the intake valve is open, and the exhaust valve is closed. During this stroke, air mixed with fuel is pumped into the cylinder by the positive differential pressure between the intake port and the cylinder internal pressure. During the compression stroke, the piston moves from BDC to TDC. Both intake and exhaust valves are closed. For an ideal IC engine, this compression is adiabatic and modeled by the following expression: Eq. (4.1)

$$p_c = V_c^{\gamma} \tag{4.1}$$

where V_c is the cylinder contained volume (between the piston upper surface and the top of the combustion chamber), p_c the combustion chamber pressure, and γ the ratio of specific heat at constant pressure to the specific heat at constant volume. For the intake air/fuel mixture, $\gamma \cong 1.5$ for air/gasoline mixture.

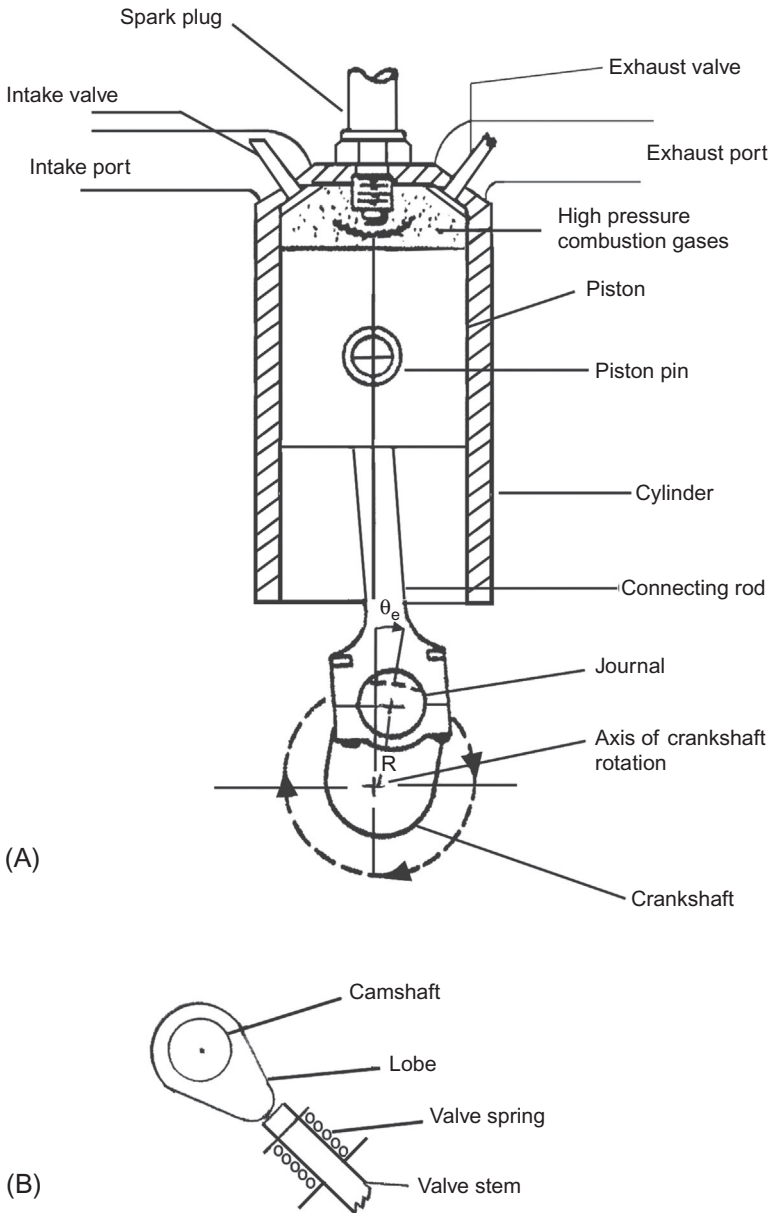


FIG. 4.8 IC engine cylinder and cam actuation mechanism.

The term adiabatic refers to a process with zero heat loss. The compression process for an actual engine is not adiabatic since heat is lost (e.g., through the cylinder sidewalls). The actual function $p_c(V_c)$ for a practical engine is shown graphically later in this chapter.

The difference between combustion chamber maximum volume (piston at BDC) and its minimum volume (at TDC) (which is often called the "clearance volume") is called the cylinder displacement V_D. The ratio of cylinder pressure at TDC to that at BDC is called the compression ratio r. At some point before the piston reaches TDC on the compression stroke, the spark is generated, and combustion of the fuel/air mixture is initiated, and cylinder pressure rises rapidly.

During the next stroke, the power stroke, the cylinder pressure, acting on the piston via the connecting rod, applies a torque to the crankshaft. This expansion ideally would be adiabatic but in fact is not adiabatic due to heat losses (as is demonstrated later from measurements made on an actual engine).

During the final stroke, the exhaust stroke, the piston again moves from BDC to TDC. The exhaust valve is open during most of this stroke, and the products of combustion discussed earlier are pumped out of the cylinder into the exhaust system and released through this system to the atmosphere.

The actual point in the 720 degrees crankshaft rotation angle at which the valves open and close (called valve timing) has traditionally been determined by a mechanism that includes the camshaft and mechanical linkage connecting it to the valves. The camshaft, which is illustrated in Fig. 4.8B, has lobes that force the valves open against the restoring forces of valve springs that otherwise hold the valves closed. The reader should imagine that the valves depicted in Fig. 4.8A extend to the end of the valve stem depicted in Fig. 4.8B. The camshaft is coupled via a gear system to the crankshaft such that it rotates at half the speed of the latter. This mechanism assures that the valves operate synchronously within each engine cycle. During the development period of any new engine design, the optimal valve timing is determined. In Chapter 5, the drive mechanism for the camshaft and the means for rotating it at half the crankshaft angular speed are explained with respect to a system known as variable valve phasing (VVP). For the present, however, the discussion is focused on basic engine processes.

Energy is produced by a four-stroke/four-cycle internal combustion engine only during the power stroke. The energy produced during this stroke must be greater than the energy required for the other strokes and by internal friction losses. Normally, in any well-designed engine, the power stroke energy far exceeds the magnitude of all mechanical losses, thereby yielding net output energy.

A basic method of evaluating the output mechanical energy involves the so-called indicator diagram, which is also a plot of the p_c versus V_c for the entire cycle. Fig. 4.9 represents an indicator diagram for an ideal engine cycle (in the sense of no heat loss to the engine) by the dashed curve and the $p_c(V_c)$ plot for an actual engine by the solid curve.

For the ideal engine cycle, the valves are assumed to open or close at exactly TDC or BDC. Any time delays associated with the gas dynamics of intake and exhaust are taken to be negligible. In Fig. 4.9, point a is at BDC with the cylinder filled with fuel/air mixture. The segment from point a to point b corresponds to the compression stroke. Ignition occurs at point b and the combustion chamber pressure increases instantaneously to point c, which is the beginning of the power stroke. The power stroke is represented by the segment from point c to point d. At point d, the exhaust valve opens and pressure drops to the exhaust system pressure (p_e) at point a. The segment from point a to point e corresponds to the exhaust stroke. At point e, the exhaust valve closes, and the intake valve opens. The intake stroke corresponds to the segment from point e to point a where the cycle began and where

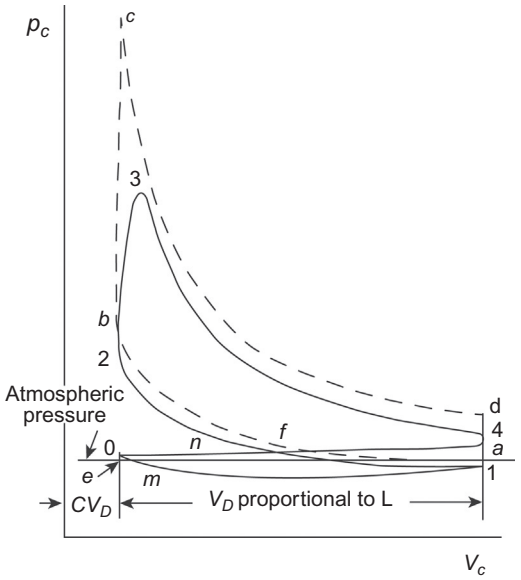


FIG. 4.9 Indicator diagram for a four-stroke engine.

the next engine cycle commences. For this ideal engine cycle, both intake and exhaust gas pressures are taken to be at atmospheric pressure.

The indicator diagram for an actual engine is depicted by the solid curve for which the compression stroke is the segment of the solid curve from point 1 to point 2. Notice that the pressure at point 1 is slightly below atmosphere pressure, which occurs because of pressure losses in the intake system. At point 2, ignition occurs, and the pressure rises to its maximum value at point 3. The expansion from point 3 to point 4 is somewhat different in shape than the ideal adiabatic expansion due to heat losses. Pressure continues to drop after the exhaust valve opens (near point 4) but remains slightly above atmospheric pressure due to "back pressure" in the exhaust system. The exhaust stroke occurs between point 4 and point 0. The intake stroke occurs from point 0 to point 1 at pressure somewhat below atmospheric due to pressure drop across the throttle plate and some pressure losses in the intake system. From point 1, the cycle begins again.

The net energy/cycle (called the indicated energy, W_i) is given by the contour integral around the curve from points 1–6 below:

$$W_i = \oint p_c \, dV_c \tag{4.2}$$

The only positive contribution to this integral comes from the portion from point 3 to point 4 (i.e., the power stroke). The energy/cycle is influenced markedly by the timing of the valve openings and closing as will be explained in Chapter 6. As clearly shown from Fig. 4.9, in any practical engine, the indicator diagram deviates from the ideal as indicated by the continuous curve of Fig. 4.9.

DEFINITION OF ENGINE PERFORMANCE TERMS

Several common terms are used to describe an engine's performance, including the torque and power at various places in the engine and power train, as well as cylinder pressure, crankshaft angular speed, fuel consumption, and various combinations of these as explained below. It is these performance variables that are influenced by the electronic engine control. For an understanding of this controller influence, it is necessary to have the quantitative models for these performance variables as presented below.

TORQUE

Engine *torque* is produced on the crankshaft by the cylinder pressure pushing on the piston during the power stroke. In an IC, engine torque is produced at the crankshaft as explained below. The torque that is applied to the crankshaft is called "indicated torque T_i." The output torque from the engine at the transmission end of the crankshaft differs from T_i due to friction and pumping losses and is called the brake torque (denoted T_b).

For an understanding of the various torques at different points in the power train, it is helpful to refer to Fig. 4.10, which illustrates the geometry of a single cylinder in a four-stroke IC engine.

Fig. 4.10 shows the centerline of the cylinder, which is a line along the cylinder axis through the crankshaft rotational center. The piston is connected via the connecting rod to the crankshaft. The connecting rod is fastened to the piston via the piston pin about which this rod can rotate. The piston pin is offset from the cylinder axis by an amount denoted δ in Fig. 4.10, which improves the torque relative to that which would be produced with $\delta = 0$. During the power stroke, a torque is applied to the crankshaft resulting from the force acting on the piston due to combustion chamber pressure acting through a lever arm, which is proportional to the crank throw R and which varies with crankshaft angular position (θ_e). This torque is known as the indicated torque to distinguish it from other torque acting on the crankshaft and can be computed as explained below.

Fig. 4.10 presents the geometry of the piston, connecting rod, and crankshaft in a way which permits a model for the indicated torque $T_i(\theta_e)$ as a function of crankshaft angle (θ_e) to be developed. In Fig. 4.10, the connecting rod length is denoted L_r, and the radius from the crankshaft axis of rotation to the center of the connecting rod journal is denoted R. It is this radius of the crankshaft rotation that provides the lever arm for the production of indicated torque due to the force on the top of the piston due to combustion chamber pressure ($P_c(\theta_e)$). In many engines, the piston pin is located slightly off the cylinder centerline (C_L) in a plane that is orthogonal to the crankshaft axes of rotation, which benefits torque production. The piston pin offset from the cylinder C_L is denoted δ in Fig. 4.10. The angle between the connecting rod plane of symmetry and the cylinder axis is denoted β. Owing to the piston offset, the indicated torque for a piston on the downstroke is given by

$$T_i(\theta_e) = \frac{p_c(\theta_e)\, AR \sin(\theta_e + \beta)}{\cos\beta} \quad 0 \le \theta_e < \pi \tag{4.3}$$

On the upstroke T_i is given by

$$T_i(\theta_e) = \frac{p_c(\theta_e)\, AR \sin(\theta_e - \beta)}{\cos\beta} \quad \pi \le \theta_e < 2\pi \tag{4.4}$$

where A is the piston cross-sectional area. The factors $R[\sin(\theta_e \pm \beta)]/\cos\beta$ represent the lever arm through which torque is applied to the crankshaft.

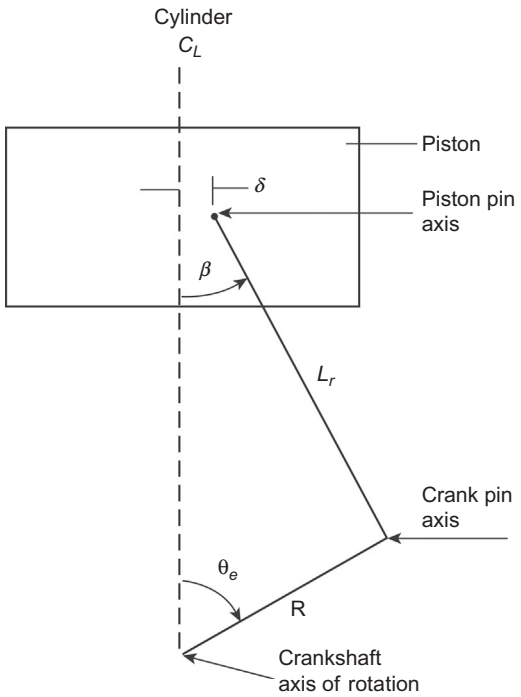


FIG. 4.10 Schematic illustration of cylinder geometry.

The combustion chamber pressure for a representative four-stroke reciprocating IC engine is shown in Fig. 4.11 for a complete engine cycle (720 degrees of crankshaft rotation) beginning at −180 degrees (BDC) for the start of compression and ending at 540 degrees (BDC) at the end of intake stroke. Note that following ignition (point x), the pressure rises abruptly due to combustion reaching a maximum at a point (y) slightly beyond TDC.

The region of positive work for each cycle is indicated in the drawing as the power stroke (i.e., 0–180 degrees). The fluctuations in combustion chamber pressure along with the geometry factor relating p_c to T_i cause T_i to fluctuate with crankshaft angle and of course with time. However, when the engine produces power, the time-average value for T_i (i.e., $\overline{T_i}$) is positive:

$$\overline{T_i} > 0$$

There are other contributors to the total dynamic indicated torque at the crankshaft, including torques due to the reciprocating forces of the piston and connecting rod. The details of the reciprocating torque (T_r) are explained in Chapter 11 and are not relevant to the present discussion, but in general increase quadratically with rotational speed. In addition, there are contributors to the torque at the crankshaft due to internal friction of the rotating and reciprocating components as well as due to pumping of intake and exhaust gases.

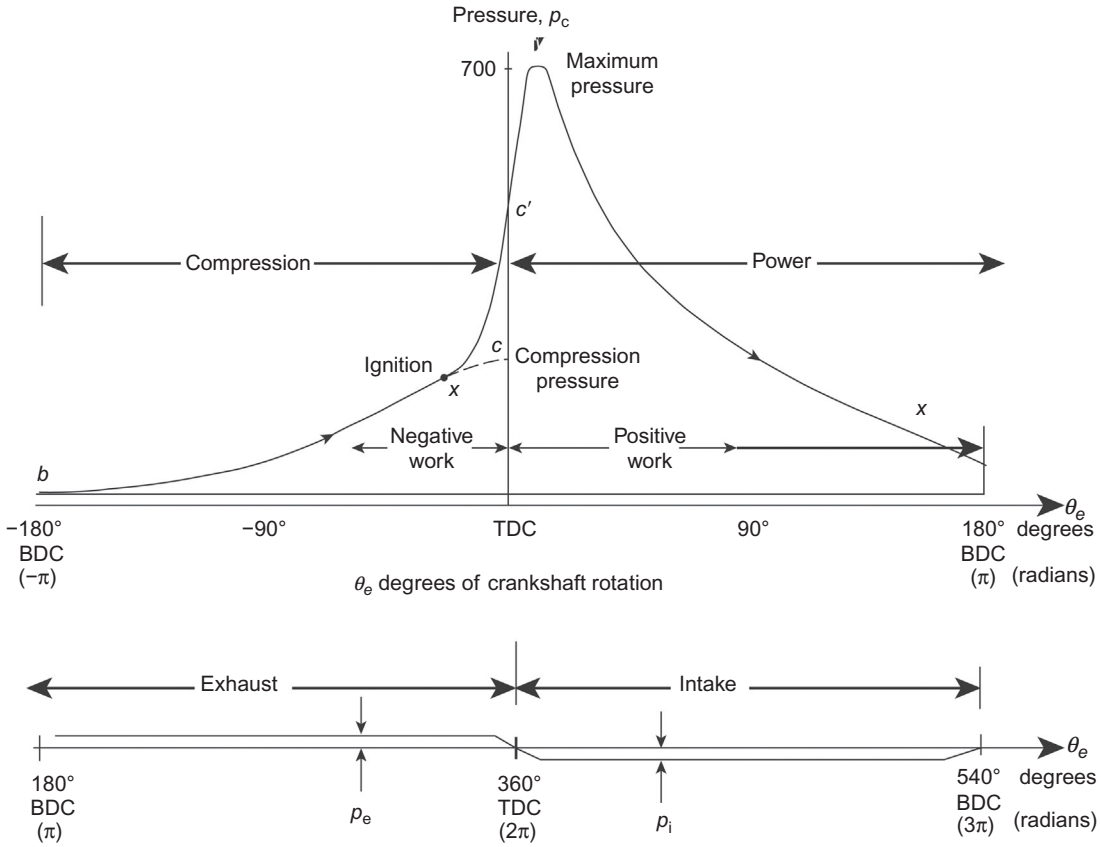


FIG. 4.11 Exemplary plot of $p_c(\theta_e)$.

The indicated torque is the maximum available torque that is applied at each crankshaft segment for the corresponding cylinder. Typically, between each crankshaft "throw" are sleeve bearings that have friction. In addition to friction, there are negative torques applied to the crankshaft owing to the nonzero cylinder pressures—p_e during exhaust and p_i during intake—and a relatively large negative torque associated with compression. The time-average torque averaged over an engine cycle at the crankshaft output end is called the brake torque $\bar{T}_b$ and is given by

$$\bar{T}_b = \bar{T}_c - \bar{T}_{fp} \tag{4.5}$$

where $\bar{T}_{fp}$ is the average torques associated with friction and pumping losses. It is this brake torque acting through the drivetrain that provides the torque to drive the vehicle. The drivetrain includes the transmission and other gear systems (e.g., differential) as explained in Chapter 6.

POWER

One of the most important metrics for engine performance is output power. This power is related to the indicated torque applied to the crankshaft (as explained above). The instantaneous power applied to the crankshaft by the indicated torque is known as the indicated power ($P_i(\theta_e(t))$), given by

$$P_i(t) = T_i(t)\omega_e(t)$$

where

$$\omega_e(t) = \frac{d\theta_e}{dt} \text{ in rad/s} \tag{4.6}$$

The units for $P_i(t)$ are $N \cdot$m/sec (metric) or ft $\cdot$ lb/sec (English units). Normally, it is the average indicated power averaged over N engine cycles $\bar{P}(N)$ that is useful as a metric for engine available power (with θ_e in radians) is given by

$$\bar{P}_i(N) = \frac{1}{4\pi N}\int_0^{4\pi N} P_i(\theta_e)d\theta_e \quad N = \text{integer} \tag{4.7}$$

The appropriate unit for P_i is kW; although in the United States, the popular unit (with the driving public) remains horsepower (hp), where 1 hp = 0.75 kW and 550 ft lb/sec.

The engine output power at the crankshaft is known as the brake power (P_b) since traditionally engine power was measured using a Prony brake. This brake power (in kW or hp) is the difference between indicated power and the power associated with internal power losses due, for example, to friction and pumping of the intake mixture and exhaust gases. Generally, the cycle-averaged friction and pumping power are combined and denoted $\bar{P}_{fp}$. The brake power P_b is given by

$$P_b = \bar{P}_i - \bar{P}_{fp} \tag{4.8}$$

Measurements are readily made of P_{fp} by driving the engine from an external power source such as an electric dynamometer. The latter is an instrumented electric motor/generator having the capacity to absorb all brake power produced by the running engine under test. Normally, instrumentation permits measurements to be made of output torque, angular speed ω_e, and P_b. It is also common practice to evaluate engine performance via the averaged torque at the engine output, which is called "brake torque" and is denoted T_b and which is related to P_b by the expression

$$T_b = P_b/\omega_e \tag{4.9}$$

Another metric of performance for an engine is the so-called mean-effective pressure (*mep*). It is defined as the indicated work done on the piston (W_i) (given in Eq. (4.2)) with units in in·lb divided by displacement volume V_D with units in in^3. As in the case of torques, it is convenient to consider the indicated *mep* (*imep*), which is defined as

$$imep = \frac{W_i}{V_D} \text{ (psi)} \tag{4.10}$$

where $V_D = V_1 - V_2$ is the displacement, V_1 the cylinder maximum volume (at BDC), and V_2 the cylinder minimum volume at TDC; (i.e., clearance volume).

The *imep* (which has the dimensions of pressure) is the value of constant pressure, which, if acting during an engine cycle, would produce the work done on the crankshaft. There is also a friction *mep* (*fmep*):

$$fmep = \frac{W_f}{V_d} \tag{4.11}$$

where W_f is the work done by the friction torque. The most commonly used *mep* is the brake *mep* (*bmep*), which is defined as

$$bmep = imep - fmep$$

It has the units of pressure (e.g., N/m^2 or lb/in^2) and is the value of constant pressure acting over a full engine cycle to produce the output mechanical work/cycle.

FUEL CONSUMPTION

Fuel economy can be measured while the engine delivers power to the dynamometer. The engine is typically operated at a fixed RPM and a fixed brake power (fixed dynamometer load), and the fuel flow rate (in kg/h or lb/h) is measured. The fuel consumption is then given as the ratio of the fuel flow rate $\dot{f}$ to the brake power output (P_b). This fuel consumption is known as the *brake-specific fuel consumption* or BSFC. BSFC is a measurement of the fuel economy of the engine alone and is given by

$$\text{BSFC} = \frac{\dot{f}}{P_b} \tag{4.12}$$

The unit for BSFC is kg/(kW × h) or lb/(hp·h) in British units. By improving the BSFC of the engine, the fuel economy of the vehicle in which it is installed is also improved. It is shown later in this chapter that electronic controls can optimize BSFC.

In gasoline-fueled engines, airflow into the engine at any operating angular speed (RPM) is determined by the throttle angular position. In fact, the throttle is the control by the driver that determines the engine output power.

As explained above, any internal combustion must pump air/fuel into its combustion chamber. If an IC engine were a perfect air pump, then, at wide open throttle, the air volume pumped into the engine for each complete engine cycle (i.e., two complete revolutions) would be its displacement volume V_d:

$$V_d = A_p S_c M \tag{4.13}$$

where A_p is the piston cross-sectional area and S_c the cylinder stroke, which is the distance traveled by the piston from TDC to BDC and M = number of cylinders.

Formally, the volumetric efficiency e_v is defined as the ratio of the mass of fresh mixture (i.e., air and fuel) that is actually pumped into the cylinder during an intake stroke at inlet air density to the mass of this mixture, which would fill the cylinder at the inlet air density. The volumetric efficiency for any given engine is determined empirically and varies with throttle angle, RPM, inlet pressure and temperature, and exhaust pressure (p_e). Assuming initially that all cylinders receive mixture at identical density, the definition of e_v can be expressed by

$$e_v = \frac{2\dot{M}_i}{NV_d\rho_i} \tag{4.14}$$

where $\dot{M}_i$ is the air intake mixture mass flow rate (slugs/sec) (or kg/sec), N the number of revolutions/sec, and ρ_i the inlet mixture density (slugs/ft^3) (or kg/m^3).

The variable ρ_i is the density of the mixture in the intake system downstream from the throttle plate in or near a cylinder inlet port. When inlet air density is defined at this point in the intake system, it provides a measurement of the air pumping efficiency of the cylinder and valves alone. It is this definition that is used for the present discussion. However, it should be noted that volumetric efficiency could be based on the air density at the input to the intake system (i.e., upstream of the throttle plate). With air density taken at this point, the volumetric efficiency is termed the overall volumetric efficiency. Unfortunately, it is not always convenient to measure the density of the inlet mixture that consists of air, fuel, and atmospheric water vapor. On the other hand, since fuel, airflow, and water vapor occupy the same volume and have the same intake volume flow rate $\dot{V}_i$, the following relationship is valid:

$$\begin{aligned} \dot{V}_i &= \frac{\dot{M}_i}{\rho_i} \\ &= \frac{\dot{M}_a}{\rho_a} \end{aligned} \tag{4.15}$$

where $\dot{M}_a$ is the mass flow rate of dry air and ρ_a the inlet density of dry air.

For mixtures of air, water vapor, and gaseous or evaporated fuel, Dalton's law of partial pressures states

$$p_i = p_a + p_f + p_w \tag{4.16}$$

where p_i is the total inlet pressure, p_a the partial pressure of air, p_f the partial pressure of fuel, and p_w the partial pressure of water vapor.

Each constituent (denoted k) of the inlet air mixture behaves as a perfect gas such that

$$\rho_k = \frac{p_k}{RT_i} \tag{4.17}$$

where R is the perfect gas law constant. Also, it can be shown that

$$\frac{p_a}{p_i} = \frac{M_a/m_a}{\left[\dfrac{M_a}{m_a} + \dfrac{M_f}{m_f} + \dfrac{M_w}{m_w}\right]} \tag{4.18}$$

where M_k = mass of constituent k and m_k = molecular mass of constituent k:

$$k = a, f, w$$

The air density in the mixture ρ_a is given by

$$\rho_a = \left(\frac{p_i}{RT_i}\right) \Big/ \left[1 + F_i\left(\frac{m_a}{m_f}\right) + \frac{m_a}{m_w}h\right] \tag{4.19}$$

where $F_i\left(\dfrac{m_f}{m_a}\right)$ is the fuel/air mass ratio, h the ratio of mass water vapor to the mass of air, $m_a = 29$, and $m_w = 18$.

In this form, it is possible to compute inlet air density from measurements of total inlet air pressure (p_i) and inlet absolute air temperature T_i and standard environmental variable measurements (e.g.,

relative humidity). As presently shown, F_i is determined by the fuel control system to achieve certain engine performance requirements.

As explained earlier in this chapter, the engine power is regulated by the driver via an air valve in the form of a movable throttle plate in the intake system. Linkage connects the accelerator pedal to the throttle plate such that it partially restricts airflow into the engine. Typically, the throttle plate is in the form of a circular disk that pivots about a diametric axis in a cylindrical portion of the intake manifold (e.g., see Fig. 4.3). Effectively, the airflow into the engine at any given engine angular speed (RPM) varies in proportion to the opening of this plate as represented by the throttle angle θ_T. This empirically determined volumetric efficiency is a convenient variable that can be used to characterize engine pumping efficiency during the development of a new engine control system and can also be used in fuel control of a production engine as explained later.

ENGINE OVERALL EFFICIENCY

There are numerous ways to characterize the performance of an engine as indicated above. One of the most meaningful of these is the efficiency with which the engine converts the energy available in the fuel (in chemical form) to mechanical work. This efficiency, which we denote η_m, can be evaluated on an engine cycle by engine cycle basis. However, it is more convenient to express η_m as the ratio of the instantaneous mechanical power delivered to the load to the rate of change of available energy in the fuel being delivered:

$$\eta_m = P_m / (Q_f \dot{M}_f)$$

where P_m is the mechanical power delivered to load (kW), $\dot{M}_f = \frac{d}{dt} M_f$ the instantaneous mass flow rate of fuel (kg/sec), and Q_f the energy content of fuel (Joule/kg).

CALIBRATION

The definition of engine *calibration* is the setting of the air/fuel ratio and ignition timing for the engine for any given operating condition. With the new electronic control systems, calibration is determined by the electronic engine control system.

As will be shown later in this chapter, electronic engine control systems are based upon microprocessors or microcontrollers. Under program control, the engine control system determines the correct fuel delivery amount and the ignition timing as a function of driver command (via throttle setting) and other operating variables and parameters. In an exemplary fuel control system, these correct values are found from a table lookup process with interpolation. The calibration tables for any given engine configuration are found empirically as described below. As will also be shown below, an additional component of fuel delivery is determined from a closed loop (CL) portion of the control.

The majority of present-day engines deliver fuel by means of an individual fuel injector (FI) associated with each cylinder. A FI is essentially an electromechanical valve to which fuel, under pressure, is supplied. Chapter 5 explains the configuration and operation of FIs. As will be explained later, each FI delivers fuel in a pulse mode in which fuel quantity is determined primarily by the duration of fuel delivery at a nominally constant delivery rate. Another important engine calibration variable for such systems is the time of fuel delivery relative to cycle for the associated cylinder.

ENGINE MAPPING

The development of any control system comes from knowledge of the plant or system to be controlled. In the case of the automobile engine, this knowledge of the plant (the engine) comes primarily from a process called *engine mapping*.

For engine mapping, the engine is connected to a dynamometer and operated throughout its entire speed and load range. Measurements are made of the important engine variables, while quantities, such as the air/fuel ratio and the spark control, are varied in a known and systematic manner. Such engine mapping is done in engine test cells that have engine dynamometers and complex instrumentation that collects data under computer control. At each operating point, calibration is varied, and performance is measured. An optimum calibration can be found that is a compromise between performance and allowable exhaust gas emission rates under federal regulations.

From the engine mapping, a calibration table can be created of optimum values for later incorporation into the engine control system ROM as explained in Chapter 6. Also, from this mapping, a mathematical model can be developed empirically that explains the influence of every measurable variable and parameter on engine performance. The control system designer can, if desired, select a control configuration, control variables, and control strategy that will satisfy all performance requirements (including stability) as computed from this model and that are within the other design limits such as cost, quality, and reliability. To understand a representative engine control system, it is instructive to consider the influence of control variables on engine performance and exhaust emissions.

EFFECT OF AIR/FUEL RATIO ON PERFORMANCE

Fig. 4.12 illustrates the variation in the performance variables of indicated torque (T_i) BSFC and engine emissions with variations in the air/fuel ratio with fixed spark timing and a constant engine speed.

In this figure, the exhaust gases are represented in brake-specific form. This is a standard way to characterize exhaust gases whose absolute emission levels are proportional to power. The definitions for the brake-specific emissions rates are

$$\text{BSHC} = \text{brake-specific HC concentration} = \frac{r_{HC}}{P_b} \tag{4.20}$$

$$\text{BSCO} = \text{brake-specific CO concentration} = \frac{r_{CO}}{P_b} \tag{4.21}$$

$$\text{BSNO}x = \text{brake-specific NO}x \text{ concentration} = \frac{r_{NOx}}{P_b}$$

where r_{HC} is the HC rate of flow, r_{CO} the CO rate of flow, $r_{\text{NO}x}$ the NOx rate of flow, and P_b the brake power. The indicated torque is denoted T_i in Fig. 4.12.

One specific air/fuel ratio is highly significant in electronic fuel control systems, namely, the *stoichiometric mixture*. The stoichiometric (i.e., chemically correct) mixture corresponds to an air and fuel combination such that if combustion was perfect, all the hydrogen and carbon in the fuel would

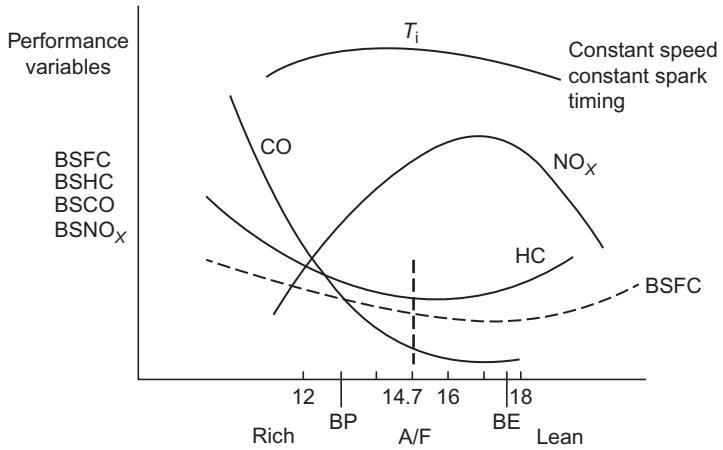


FIG. 4.12 Typical variation of performance with a variation in air/fuel ratio.

be converted by the burning process to H_2O and CO_2. For gasoline, the stoichiometric mixture ratio is 14.7:1.

Stoichiometry is sufficiently important that the fuel and air mixture is often represented by a ratio called the *equivalence ratio*, which is given the specific designation λ. The equivalence ratio is defined as follows:

$$\lambda = \frac{(\text{air/fuel})}{(\text{air/fuel@stoichiometry})}$$

A relatively low air/fuel ratio, below 14.7 (corresponding to $\lambda < 1$), is called a *rich* mixture, and an air/fuel ratio above 14.7 (corresponding to $\lambda > 1$) is called a lean mixture. Emission control is strongly affected by air/fuel ratio or by λ.

Note from Fig. 4.12 that torque (T_i) reaches a maximum in the air/fuel ratio range of 12–14. The exact air/fuel ratio for which torque is maximum depends on the engine configuration, engine speed, and ignition timing. Also note that the CO and unburned hydrocarbons tend to decrease with increasing air/fuel ratios, as one might expect because there is relatively more oxygen available for combustion with lean mixtures than with rich mixtures.

Unfortunately, for the purposes of controlling exhaust emissions, the NO*x* exhaust concentration increases with increasing air/fuel ratios. That is, there is no air/fuel ratio that simultaneously minimizes all regulated exhaust gases.

EFFECT OF SPARK TIMING ON PERFORMANCE

Spark advance is the time before top dead center (TDC) when the spark is initiated. It is usually expressed in number of degrees of crankshaft rotation relative to TDC. Fig. 4.13 reveals the influence of spark timing on brake-specific exhaust emissions with constant speed and constant air/fuel ratio for a representative engine. Note that both NO*x* and HC generally increase with increased advance of spark

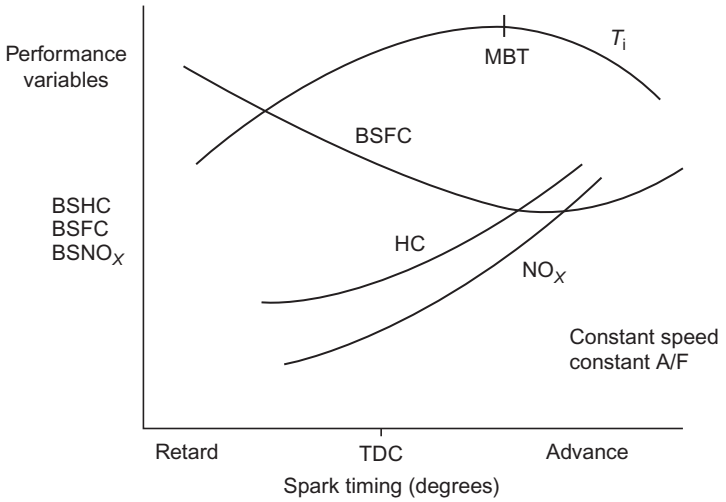


FIG. 4.13 Representative variation of performance with spark timing.

timing. BSFC and torque are also strongly influenced by timing. Fig. 4.13 shows that maximum torque occurs at a particular advanced timing (called advanced for mean best torque) denoted MBT.

Operation at or near MBT is desirable since this spark timing tends to optimize performance. This optimal spark timing varies with RPM. As will be explained, engine control strategy involves regulating fuel delivery at a stoichiometric mixture and varying ignition timing for optimized performance. However, there is yet another variable to be controlled, which assists the engine control system in meeting exhaust gas emission regulations.

EFFECT OF EGR ON PERFORMANCE

Up to this point in the discussion, only the traditional calibration parameters of the engine (air/fuel ratio and spark timing) have been considered. However, by adding another control variable, the undesirable exhaust gas emission of NO*x* can be significantly reduced while maintaining a relatively high level of torque. This new control variable, EGR, consists of recirculating a precisely controlled amount of exhaust gas into the intake. The engine control configuration depicted in Fig. 4.5 shows that EGR is a major subsystem of the overall control system. Its influence on emissions is shown qualitatively in Figs. 4.14 and 4.15 as a function of the percentage of exhaust gas in the intake. Fig. 4.14 shows the dramatic reduction in NO*x* emission when plotted against air/fuel ratio, and Fig. 4.15 shows the effect on performance variables as the percentage of EGR is increased. Note that the emission rate of NO*x* is most strongly influenced by EGR and decreases as the percentage of EGR increases. The HC emission rate increases with increasing EGR; however, for relatively low EGR percentages, the HC rate changes only slightly. Thus, a compromise EGR rate between NO*x* reduction and HC increase is possible in which the benefits of EGR on NO*x* reduction far offset the adverse effect on HC emissions. This compromise amount of EGR varies with engine configuration.

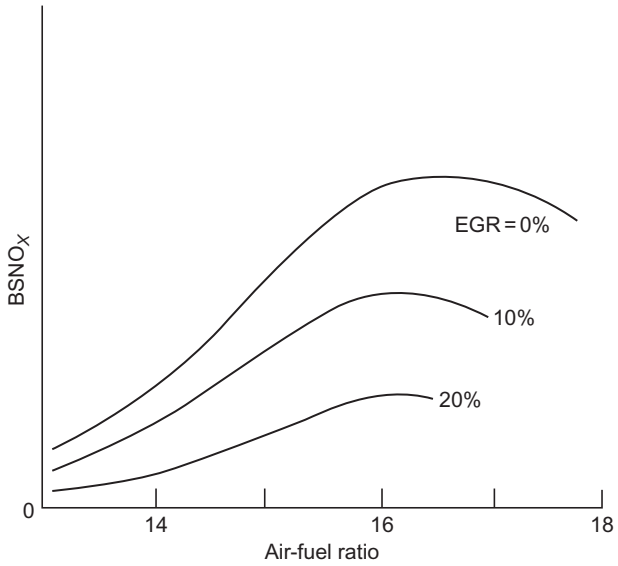


FIG. 4.14 NO_x emission as a function of air/fuel ratios at various EGR%.

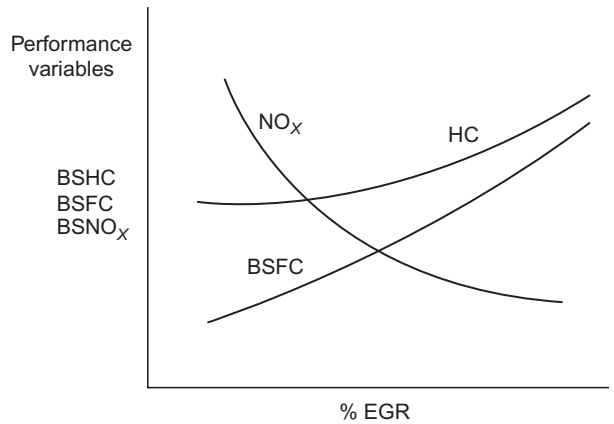


FIG. 4.15 Influence of EGR on brake-specific performance variables.

The mechanism by which EGR affects NO*x* production is related to the peak combustion temperature. Roughly speaking, the NO*x* generation rate increases with increasing peak combustion temperature if all other variables remain fixed. Increasing EGR tends to lower this temperature; therefore, it tends to lower NO*x* generation. It should be noted that EGR, though relatively small, does influence the air partial pressure in the intake mixture. Compensation for this effect is required as explained later.

EXHAUST CATALYTIC CONVERTERS

It is the task of the electronic control system to set the calibration for each engine-operating condition. There are many possible control strategies for setting the variables for any given engine, and each tends to have its own advantages and disadvantages. Moreover, each automobile manufacturer has a specific configuration that differs in certain details from competitive systems. However, this discussion is about a typical electronic control system that is highly representative of the systems for engines used by US manufacturers. This typical system is one that has a catalytic converter in the exhaust system. Exhaust gases passed through this device are chemically altered in a way that reduces tailpipe emissions relative to engine output exhaust. Essentially, the catalytic converter reduces the concentration of undesirable exhaust gases coming out of the tailpipe relative to engine-out gases (the gases coming out of the exhaust manifold).

The EPA regulates only the exhaust gases that leave the tailpipe; therefore, if the catalytic converter reduces exhaust gas emission concentrations, the engine exhaust gas emissions at the exhaust manifold can be higher than the EPA requirements. This has the significant benefit of allowing engine calibration to be set for better performance than would be permitted if exhaust emissions in the engine exhaust manifold had to satisfy EPA regulations. This is the type of system that is chosen for the typical electronic engine control system.

Several types of catalytic converters are available for use on an automobile. The desired functions of a catalytic converter include

1. oxidation of hydrocarbon emissions to carbon dioxide (CO_2) and water (H_2O),
2. oxidation of CO to CO_2,
3. reduction of NO_x to nitrogen (N_2) and oxygen (O_2).

OXIDIZING CATALYTIC CONVERTER

The oxidizing catalytic converter (Fig. 4.16) has been one of the more significant devices for controlling exhaust emissions since the era of emission control began. The purpose of the oxidizing catalyst (OC) is to increase the rate of chemical reaction, which initially takes place in the cylinder as the compressed air-fuel mixture burns, toward an exhaust gas that has complete oxidation of HC and CO to H_2O and CO_2.

The extra oxygen required for this oxidation is often supplied by adding air to the exhaust stream from an engine-driven air pump. This air, called *secondary air*, is normally introduced into the exhaust manifold.

The most significant measure of the performance of the OC is its conversion efficiency:

$$\eta_c = \frac{M_o}{M_{ic}} \tag{4.22}$$

where M_o is the mass flow rate of gas that has been oxidized leaving the converter and M_{ic} is the MAF of gas into the converter.

The conversion efficiency of the OC depends on its temperature. Fig. 4.17 shows the conversion efficiency (expressed as a percent) of a typical OC for both HC and CO as functions of temperature. Above about 300°C, the efficiency approaches 98%–99% for CO and more than 95% for HC.

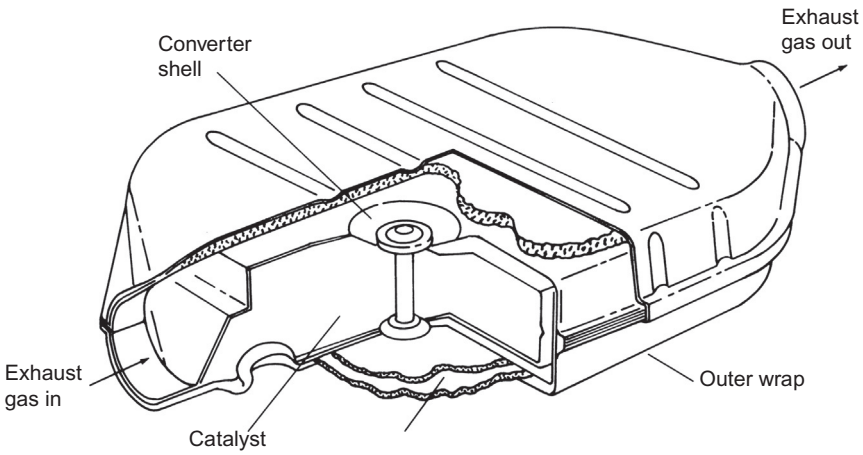


FIG. 4.16 Catalytic converter configuration.

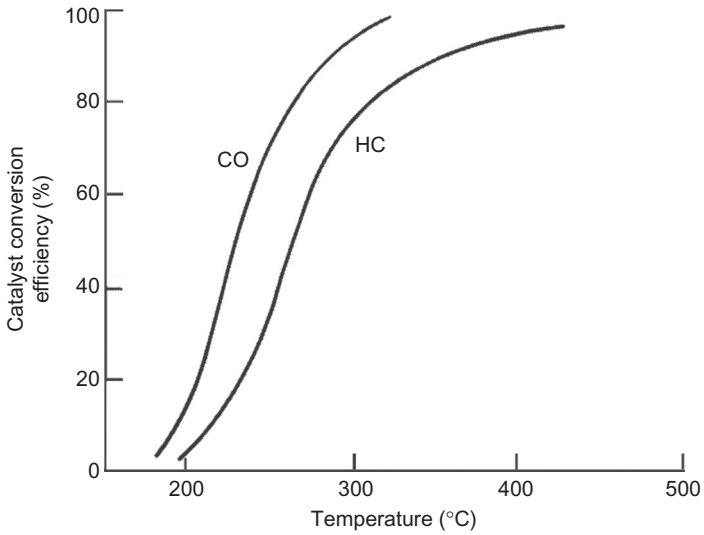


FIG. 4.17 Oxidizing catalyst conversion efficiency versus temperature.

THE THREE-WAY CATALYST

Another catalytic converter configuration that is extremely important for modern emission control systems is called the three-way catalyst (TWC). It uses a specific catalyst formulation containing platinum, palladium, and rhodium to reduce NO*x* and oxidize HC and CO all at the same time. It is called three-way because it simultaneously reduces the concentration of all three major undesirable exhaust

gases. The TWC uses a specific chemical design to reduce all three major emissions (HC, CO, and NO*x*) by ~90%.

The conversion efficiency of the TWC for the three exhaust gases depends mostly on the air/fuel ratio. Unfortunately, the air/fuel ratio, for which NO*x* conversion efficiency is high, corresponds to a very low conversion efficiency for HC and CO and vice versa. However, as shown in Fig. 4.18, there is a very narrow range of air/fuel ratio (called the window and shown as the lined region in Fig. 4.18) in which an acceptable compromise exists between NO*x* and HC/CO conversion efficiencies. The conversion efficiencies within this window are sufficiently high to meet the very stringent EPA requirements established so far.

Note that this window is only about 0.1 air/fuel ratio wide (±0.05 air/fuel ratio) and is centered at stoichiometry. (Recall that stoichiometry is the air/fuel ratio that would result in complete oxidation of all carbon and hydrogen in the fuel if burning in the cylinder were perfect; for gasoline, stoichiometry corresponds to an air/fuel ratio of 14.7.) This ratio and the concept of stoichiometry are extremely important in an electronic fuel controller. In fact, the primary function of most modern electronic fuel control systems is to maintain average air/fuel ratio at stoichiometry. The operation of the three-way catalytic converter is adversely affected by lead. Thus, in automobiles using any catalyst, it is necessary to use lead-free fuel.

Controlling the average air/fuel ratio to the tolerances of the TWC window (for the full-life requirement) requires accurate measurement of MAF and precise fuel delivery and is the primary function of the electronic engine control system. A modern electronic fuel control system can meet these precise fuel requirements. In addition, it can maintain the necessary tolerances for government regulations for over 100,000 mi.

The fundamentals of any electronic engine control system are that it regulates the fuel/air mixture and ignition timing in response to an arbitrary driver input (via the throttle plate). The driver input traditionally includes the accelerator pedal position, which ultimately determines the throttle position. However, the throttle angular position is controlled electronically in contemporary vehicles

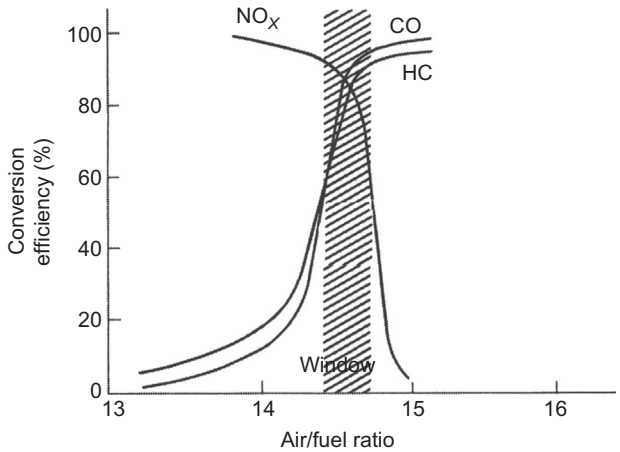


FIG. 4.18 Conversion efficiency of a TWC versus air/fuel.

incorporating an advanced cruise control, as explained in detail in Chapter 7. In addition, electronic throttle control is also implemented in hybrid and autonomous vehicles as explained in Chapters 7 and 12. The electronic engine control system directly determines a corresponding fuel quantity delivered to the engine, which optimizes performance subject to a somewhat complex set of constraints. The constrained optimization involves compromises between the conflicting constraints of emission regulations and required fuel economy. As explained above, there are other control variables (e.g., spark timing EGR) that are part of the constrained optimization process.

ELECTRONIC FUEL CONTROL SYSTEM

For an understanding of the configuration of an electronic fuel control system, refer to the block diagram of Fig. 4.19.

The primary function of this fuel control system is to determine the MAF accurately into the engine. Then, the control system precisely regulates fuel delivery such that the ratio of the mass of air to the mass of fuel in each cylinder is as close as possible to stoichiometry (i.e., 14.7). As will be shown in this chapter, it has two major modes of operation, OL and CL as explained in Appendix A. The components of this block diagram are as follows:

1. Throttle position sensor (TPS)
2. Mass airflow sensor (MAF)
3. Fuel injectors (FI)
4. Ignition systems (IGN)
5. Exhaust gas oxygen sensor (EGO)
6. Engine coolant sensor (ECS)

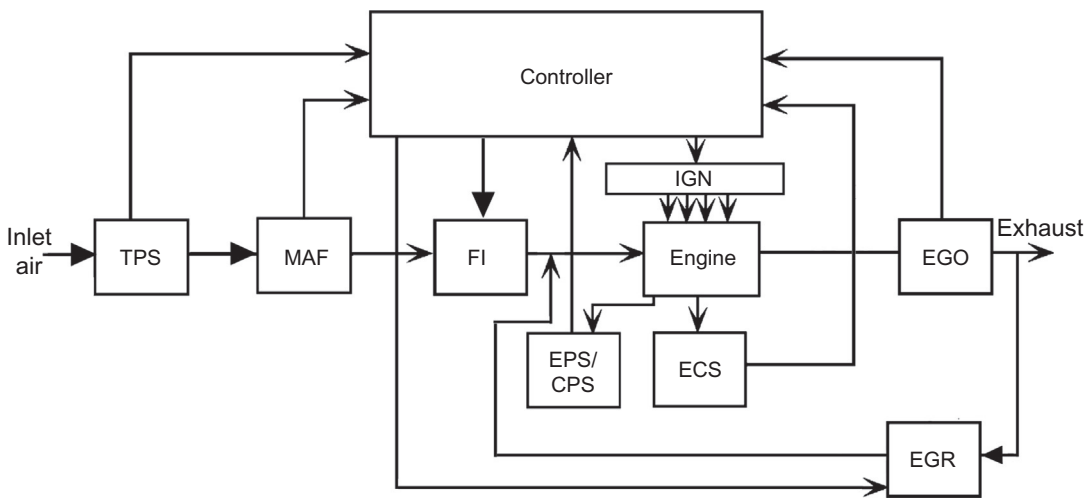


FIG. 4.19 Electronic fuel control configuration.

7. Engine position sensor (EPS)
8. Camshaft position sensor (CPS)
9. Exhaust gas recirculation actuator (EGR)

The EPS has the capability of measuring crankshaft angular speed (RPM) and crankshaft angular position when it is used in conjunction with a stable and precise electronic clock (in the controller) as explained in Chapter 5. The camshaft position sensor typically generates a timing pulse for each camshaft revolution (i.e., one complete engine cycle); the combination of EPS and CPS yields an unambiguous measurement of engine angular position (within each engine cycle) for each cylinder. The CPS sensor is required in a four-stroke/cycle engine since each cycle involves two complete crankshaft revolutions as explained above.

The signals from the various sensors enable the controller to determine the correct fuel flow in relation to the airflow to obtain the stoichiometric mixture. From this calculation, the correct fuel delivery is regulated via FI. In addition, optimum ignition timing is determined and appropriate timing pulses are sent to the ignition module (IGN).

The intake air passes through the individual pipes of the intake manifold to the various cylinders. The set of FIs (one for each cylinder) are each normally located near the intake valve within the corresponding cylinder. As explained in Chapter 5, each FI is an electrically operated valve that is (ideally) either fully open or fully closed. When the valve is closed, there is, of course, no fuel delivery. When the valve is open, fuel is delivered at a fixed rate as set by the FI characteristics and fuel supply pressure. The amount of fuel delivered to each cylinder during engine cycle k ($M_f(k)$) is determined by the length of time τ_k that the FI valve is open. This time is, in turn, computed in the engine controller to achieve the desired air/fuel ratio. Typically, the FI open timing is set to coincide with the time that air is flowing into the cylinder during the intake stroke. However, at relatively low fuel delivery rates (e.g., near closed throttle), the control system must account for the relatively short opening and closing FI dynamic transients response. The control system generates a pulsed electric signal of sufficient amplitude to open the FI valve. The duration of this pulse τ_k regulates the quantity of fuel such that the mass of fuel delivered (M_f) is given by

$$\begin{aligned} M_f(k) &= \int_{t_{k,n}}^{t_{k,n}+\tau_k} \dot{M}_f dt \\ &\cong \dot{M}_o \tau_k \end{aligned} \tag{4.23}$$

where $\dot{M}_f$ is the fuel mass flow rate and $t_{k,n}$ the time of fuel delivery to cylinder n during engine cycle k and $\dot{M}_o$ is the fuel flow rate when the FI is fully open.

It is assumed for this discussion that $M_f(k)$ is the same for all cylinders during any engine cycle.

There is an important property of the catalytic converter that allows for momentary (very short-term) fluctuations of the air/fuel ratio outside the narrow window. As the exhaust gases flow through the catalytic converter, they are actually in it for a short (but nonzero) amount of time, during which the conversions described above take place. Because of this time interval, the conversion efficiency is unaffected by rapid fluctuations above and below stoichiometry (and outside the window) as long as the average air/fuel ratio over time remains within the window centered at stoichiometry provided the fluctuations are rapid enough. A practical fuel control system maintains the average mixture at stoichiometry but has minor (relatively rapid) fluctuations about the average, as explained later in this chapter.

The electronic fuel control system operates in two modes, open loop (OL) and closed loop (CL). The concepts for OL and CL control are explained in Appendix A. In the OL mode (also called feed forward), the MAF $(\dot{M}_a)$ into the engine is measured. Then, the fuel control system determines the quantity of fuel (M_f) to be delivered to meet the required air/fuel ratio.

In the CL control mode (also called feedback), a measurement of the controlled variable is provided to the controller (i.e., it is fed back) such that an error signal between the actual and desired values of the controlled variable is obtained. Then, the controller generates an actuating signal that tends to reduce the error to zero.

In the case of fuel control, the desired variables to be measured are HC, CO, and NOx concentrations. Unfortunately, there is no cost-effective, practical sensor for such measurements that can be built into the car's exhaust system. On the other hand, there is a relatively inexpensive sensor that gives an indirect measurement of HC, CO, and NOx concentrations. This sensor generates an output that depends on the concentration of residual oxygen in the exhaust after combustion. As will be explained in detail in Chapter 5, this sensor is called an *exhaust gas oxygen* (EGO) *sensor*. There, it is shown that the EGO has evolved since its introduction in the earliest electronic control systems for engines with TWC. For the purposes of the present chapter, we consider the simplest model for an EGO sensor. In this simplified model, the EGO sensor output switches abruptly between two voltage levels depending on whether the input air/fuel ratio is richer than or leaner than stoichiometry. Such a sensor is appropriate for use in a limit-cycle type of CL control. Although the EGO sensor is a switching-type sensor, it provides sufficient information to the controller to maintain the average air/fuel ratio over time at stoichiometry, thereby meeting the mixture requirements for optimum performance of the three-way catalytic converter.

In a typical modern electronic fuel control system, the fuel delivery is partly OL and partly CL. The OL portion of the fuel flow is determined by measurement of mass airflow. This portion of the control sets the air/fuel ratio at approximately stoichiometry. A CL portion is added to the fuel delivery to ensure that time-average air/fuel ratio is at stoichiometry (within the tolerances of the window).

There are exceptions to the stoichiometric mixture setting during certain engine-operating conditions, including engine start, heavy acceleration, and deceleration. There are also exceptions due to ambient environmental conditions, particularly engine temperature and ambient air temperature and pressure. These conditions represent a very small fraction of the overall engine-operating times. They are discussed in Chapter 6, which explains the operation of a modern, practical digital electronic engine control system.

ENGINE CONTROL SEQUENCE

The step-by-step process of events in fuel control begins with engine start. During engine cranking, the mixture is set rich by an amount depending on the engine temperature (measured via the engine coolant sensor), as explained in detail in Chapter 6. Generally speaking, the mixture is relatively rich for starting and operating a cold engine as compared with a warm engine. However, the discussion of this requirement is deferred to Chapter 6. Once the engine starts and until a specific set of conditions is satisfied, the engine control operates in the OL mode.

After combustion, the exhaust gases flow past the EGO sensor, through the TWC and out the tailpipe. Once the EGO sensor has reached its operating temperature (typically a few seconds to about 2 min depending upon ambient conditions and the type of sensor used (see Chapter 5)), the EGO sensor signal is input to the controller, and the system begins CL operation.

OL CONTROL

Fuel control for an electronically controlled engine operates OL any time the conditions are not met for CL operation. Among many conditions (which are discussed in detail in Chapter 6) for CL operations, there are some temperature requirements. After operating for a sufficiently long period after starting, a liquid-cooled automotive engine operates at a steady temperature.

However, an engine that is started cold initially operates in OL mode. This operating mode requires, at minimum, measurement of the mass airflow into the engine and a measurement of RPM, as well as measurement of coolant temperature (CT). The MAF measurement in combination with RPM permits computation (by the engine controller) of the mass of air (M_a) drawn into each cylinder during intake for each engine cycle. The correct fuel mass (M_f) that is injected with the intake air is computed by the electronic controller:

$$M_f = r_{fa} M_a \tag{4.24}$$

where r_{fa} is the desired ratio of fuel to air.

For a fully warmed-up engine, this ratio is 1/14.7, which is about 0.068. That is, 1 lb of fuel is injected for each 14.7 lb of air, making the air/fuel ratio 14.7 (i.e., stoichiometry). The desired fuel/air ratio varies with temperature in a known way such that the correct value can be found from the measurement of CT. For a very cold engine, the mixture ratio can go as low as about 2 (i.e., $r_{fa} \cong 0.5$).

Theoretically, if there were no changes to the engine, the sensors, or the FI, an engine control system could operate OL at all times. In practice, owing to errors in the calculation of M_a, variations in manufactured components and to factors such as wear, the OL control would not be able to maintain the mixture at the desired air/fuel ratio if it were used alone. In order to maintain the very precise air/fuel mixture ratio required for emission control over the full-life of the vehicle, the engine controller is operated in CL mode for as much of the time as possible. Compensation for the OL mode variations above is possible via adaptive CL control as explained in Chapter 6.

CL CONTROL

Referring to Fig. 4.20, the control system in CL mode operates as follows. For any given set of operating conditions, the fuel metering actuator provides fuel flow to produce an air/fuel ratio set by the controller output.

This mixture is burned in the cylinder, and the combustion products leave the engine through the exhaust pipe. The EGO sensor generates a feedback signal for the controller input that depends on the EGO concentration. This concentration is a function of the intake air/fuel during the intake portion of the same cycle, which is approximately 1.5 crankshaft revolutions earlier than the time at which the EGO sensor output is measured.

One CL control scheme that has been traditionally used in practice (i.e., limit-cycle control) results in the air/fuel ratio cycling around the desired set point of stoichiometry. The following sections of this chapter present a somewhat simplified version of the traditional CL fuel control in order to develop a model that explains the basic principles involved in such a control system. The practical digital control technology involved in contemporary vehicles is presented in Chapter 6. The important parameters for this type of control include the amplitude and frequency of excursion away from the desired stoichiometric set point. Fortunately, the three-way catalytic converter's characteristics are such that only the short-term time-average air/fuel ratio determines its performance. The variation in air/fuel ratio during

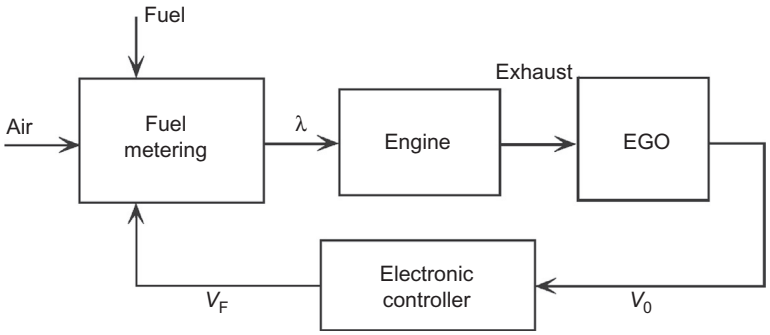


FIG. 4.20 Simplified typical closed-loop fuel control system block diagram.

the limit-cycle operation is so rapid that it has no effect on engine performance or emissions, provided that the average air/fuel ratio remains at stoichiometry.

EGO concentration

The EGO sensor, which provides feedback, will be explained in Chapter 5. In essence, the EGO generates an output signal that depends on the amount of oxygen in the exhaust. This oxygen level, in turn, depends on the air/fuel ratio entering the engine. The amount of oxygen is relatively low for rich mixtures and relatively high for lean mixtures. In terms of equivalence ratio (λ), recall that $\lambda = 1$ corresponds to stoichiometry, $\lambda > 1$ corresponds to a lean mixture with an air/fuel ratio greater than stoichiometry, and $\lambda < 1$ corresponds to a rich mixture with an air/fuel ratio less than stoichiometry. (The EGO sensor is sometimes called a lambda sensor.) Fuel entering each cylinder having a relatively lean mixture (i.e., excess oxygen) results in a relatively high oxygen concentration in the exhaust after combustion. Correspondingly, intake fuel and air having a relatively rich mixture (i.e., low oxygen) result in relatively low oxygen concentration in the exhaust.

For the purposes of the present chapter, a relatively simple continuous time model for an ideal EGO sensor output voltage V_o is given in Eq. (4.25):

$$V_o(t) = V_1 - V_2 \operatorname{sgn}\left[\lambda(t - t_d) - 1\right] \tag{4.25}$$

where t_d is the time delay from input mixture for a given engine cycle to the corresponding exhaust gases reaching the EGO sensor. This time delay that is about three-quarters of the period of an engine cycle (for a given cylinder) varies inversely with engine RPM. The parameters V_1 and V_2 are derived from the pair of actual EGO sensor voltages for $\lambda < 1$ and $\lambda > 1$ (see Chapter 5) and are approximately

$$V_1 \simeq .55, V_2 \simeq .45$$

Although there are many potential control strategies for a switching-type sensor such as the EGO sensor, we will illustrate with a relatively straightforward example. Any control system incorporating a switching sensor (as characterized by the above model) will operate in a form of limit-cycle type of operation. As explained above, the fuel delivered to any given cylinder by its FI during the kth engine cycle ($M_f(k)$) is proportional to the time interval ($\tau_F(k)$) of its binary-valued control electric signal $V_F(k)$ (see Fig. 4.20). In our present example controller, the FI (i.e., actuator) control voltage is given by

$$\begin{aligned} V_F(k) &= V \quad t_{k,n} \le t < t_{k,n} + \tau_F(k) \\ &= 0 \quad t_{k,n} + \tau_F(k) < t < t_{k+1,n} \end{aligned} \tag{4.26}$$

where $t_{k,n}$ is the injection time during kth engine cycle for the nth cylinder.

The FI duration $\tau_F(k)$ consists of an OL component $\tau_o(k)$ plus a CL component τ_{Fc}:

$$\tau_F(k) = \tau_o + \tau_{Fc}(k) \tag{4.27}$$

where $\tau_o(k)$ is calculated based on $\dot{M}_a$ measurement.

CL OPERATION

In the present example, the CL portion of the pulse duration $\tau_F(k)$ for each cycle $\tau_{Fc}(k)$ is a function of the equivalence ratio at the EGO sensor:

$$\tau_{Fc}(k) = \tau_{Fc}(k-1) + \delta\tau \operatorname{sgn}[\lambda(k) - 1] \tag{4.28}$$

Whenever the mixture is lean of stoichiometry (i.e., $\lambda > 1$), the pulse duration increases by $\delta\tau$ from the previous cycle, thereby richening the mixture (and causing λ to decrease toward 1). Correspondingly, whenever $\lambda < 1$ (rich of stoichiometry), the pulse duration is decreased from cycle k to cycle $k+1$. The OL portion of τ_F is a pulse whose duration $\tau_o(k)$ (called base pulse duration) is calculated to yield M_F corresponding to stoichiometric mixture based upon MAF $\dot{M}_a$ measurements. We illustrate the operation of this example fuel control system in Fig. 4.20.

The variable λ is used in the block diagram of Fig. 4.20 to represent the equivalence ratio at the intake manifold. The EGO concentration determines the EGO output voltage (V_o). The EGO output voltage abruptly switches between the lean and the rich levels as the air/fuel ratio crosses stoichiometry. The EGO sensor output voltage V_o is at its higher of two levels for a rich mixture and at its lower level for a lean mixture.

Reduced to its essential features, the engine control system operates as a limit-cycle controller in which the air/fuel ratio cycles up and down about the set point of stoichiometry, as shown by the idealized waveforms in Fig. 4.21.

The air/fuel ratio is either increasing or decreasing; it is never constant. The increase or decrease is determined by the EGO sensor output voltage. Whenever the EGO output voltage level indicates a lean mixture, the controller causes the air/fuel ratio to decrease, that is, to change in the direction of a rich mixture. On the other hand, whenever the EGO sensor output voltage indicates a rich mixture, the controller changes the air/fuel ratio in the direction of a lean mixture.

The electronic fuel controller changes the mixture by changing the duration of the actuating signal to each FI. Increasing this duration causes more fuel to be delivered, thereby causing the mixture to become richer. Correspondingly, decreasing this duration causes the mixture to become leaner. Fig. 4.21B is a plot of the example FI signal duration versus time.

In Fig. 4.21A, the EGO sensor output voltage is at the higher of two levels over several time intervals, including 0–1 and 1.7–2.2. This high voltage indicates that the mixture is rich. The controller causes the pulse duration (Fig. 4.21B) to decrease during this interval. At time of 1 s, the EGO sensor voltage switches low, indicating a lean mixture. At this point, the controller begins increasing the actuating time interval to tend toward a rich mixture. This increasing actuator interval continues until the

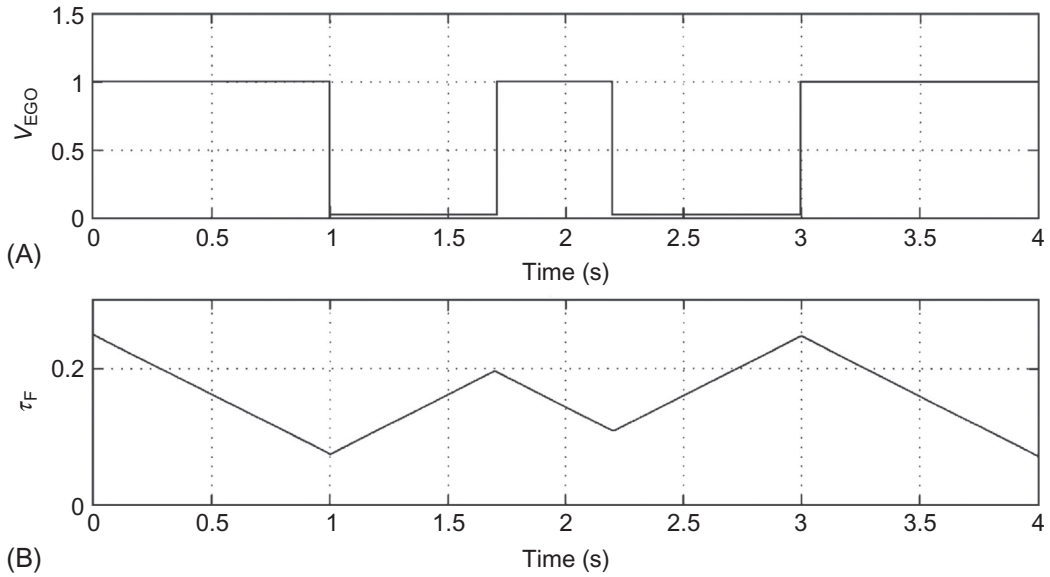


FIG. 4.21 Simplified waveforms in a closed-loop fuel control system. (A) EGO sensor voltage, (B) Fuel injector duration.

EGO sensor switches high, causing the controller to decrease the FI actuating interval. The process continues this way, cycling back and forth between rich and lean around stoichiometry.

The engine controller continuously computes the desired FI actuation duration and maintains the current value in memory. At the appropriate time in the intake cycle, the controller reads the value of the FI duration and generates a pulse of the correct duration to activate the proper FI for the computed time interval τ_F.

One point that needs to be stressed at this juncture is that the air/fuel ratio deviates from stoichiometry. However, the catalytic converter will function as desired as long as the time-average air/fuel ratio is at stoichiometry. The controller continuously computes the average of the EGO sensor voltage. Ideally, the air/fuel ratio should spend as much time rich of stoichiometry as it does lean of stoichiometry. In the simplest case, the average EGO sensor voltage $\bar{V}_{\text{EGO}}$ should be halfway between the rich and the lean values: Eq. (4.29)

$$\bar{V}_{\text{EGO}} = \frac{V_{\text{rich}} + V_{\text{lean}}}{2} \tag{4.29}$$

Whenever this condition is not met, the controller adapts its computation of pulse duration (from EGO sensor voltage) to achieve the desired average stoichiometric mixture. Chapter 6 explains this adaptive control in more detail than is given here.

Frequency and deviation of the fuel controller

It is shown in Appendix A that a limit-cycle controller regulates a system between two limits and that it has an oscillatory behavior; that is, the control variable oscillates about the set point or the desired value for the variable. The simplified fuel controller operates in a limit-cycle mode, and as shown in Fig. 4.21,

the air/fuel ratio oscillates about stoichiometry (i.e., average air/fuel ratio is 14.7). The two end limits are determined by the rich and lean voltage levels of the EGO sensor, by the controller, and by the characteristics of the fuel metering actuator. The time necessary for the EGO sensor to sense a change in fuel metering is known as the transport delay. As engine speed increases, the transport delay decreases.

The frequency of oscillation f_L of this limit-cycle control system is defined as the reciprocal of its period. The period of one complete cycle is denoted T_p, which is proportional to transport delay. Thus, the frequency of oscillation varies inversely with T_p and is given by

$$f_L = \frac{1}{T_p}$$

Furthermore, the transport delay varies inversely with engine speed (RPM). Therefore, the limit-cycle frequency is proportional to engine speed. This is depicted in Fig. 4.22 for a representative typical engine.

Another important aspect of limit-cycle operation is the maximum deviation of air/fuel ratio from stoichiometry. It is important to keep this deviation small because the net TWC conversion efficiency is optimum for stoichiometry. The maximum deviation typically corresponds to an air/fuel ratio deviation of about ± 0.1. Although the air/fuel ratio is constantly deviating up and down, the average value of deviation is held within ± 0.05 of the 14.7:1 ratio. In addition, the limit-cycle frequency and deviation in a practical engine control are influenced by hysteresis in the transfer characteristics $V_F(\lambda)$ for an actual EGO sensor as discussed in Chapter 6.

Generally, the maximum deviation decreases with increasing engine speed because of the corresponding decrease in transport delay. The parameters of the control system are adjusted such that at the worst case the deviation is within the required acceptable limits for the TWC used.

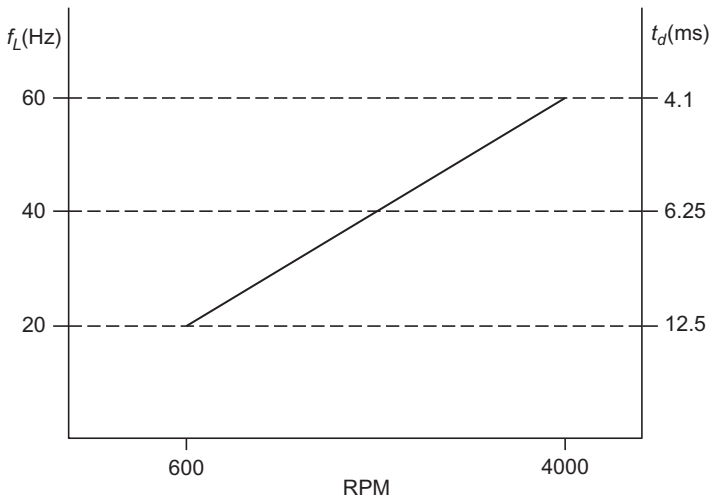


FIG. 4.22 Representative limit-cycle frequency versus RPM.

The preceding discussion applies only to a simplified idealized fuel control system. Chapter 6 explains the operation of practical electronic fuel control systems in which the calculation of FI duration is done numerically in a microprocessor-based engine control system.

ANALYSIS OF INTAKE MANIFOLD PRESSURE

As explained earlier, OL fuel control is based on a measurement of MAF and on regulation of fuel flow to maintain a desired air/fuel ratio. Mass airflow measurement can be accomplished either directly or indirectly via computation based on measurement of other intake variables. For an understanding of this important measurement, it is helpful to consider the characteristics of the intake system and the relationship between the relevant variables.

Fig. 4.23 is a very simplified sketch of an intake manifold. In this simplified sketch, the engine is viewed as an air pump pumping air into the intake manifold.

Whenever the engine is not running, no air is being pumped, and the intake manifold absolute pressure (MAP) is at atmospheric pressure. This is the highest intake MAP for a nonsupercharged engine. (A supercharged engine has an external air pump called a supercharger.) When the engine is running, the airflow is impeded by the partially closed throttle plate. This reduces the pressure in the intake manifold so it is lower than atmospheric pressure; therefore, a partial vacuum exists in the intake.

If the engine were a perfect air pump and if the throttle plate were tightly closed, a perfect vacuum could be created in the intake manifold. A perfect vacuum corresponds to zero absolute pressure. However, the engine is not a perfect pump, and some air always leaks past the throttle plate. (In fact, some air must get past a closed throttle or the engine cannot idle.) Therefore, the intake MAP fluctuates during the stroke of each cylinder and as pumping is switched from one cylinder to the next.

Each cylinder contributes to the pumping action every second crankshaft revolution. For an N-cylinder engine, the frequency f_p, in cycles per second, of the manifold pressure fluctuation for an engine running at a certain RPM is given by

$$f_p = \frac{N \times \text{RPM}}{120}$$

Fig. 4.24 shows manifold pressure fluctuations qualitatively and average MAP.

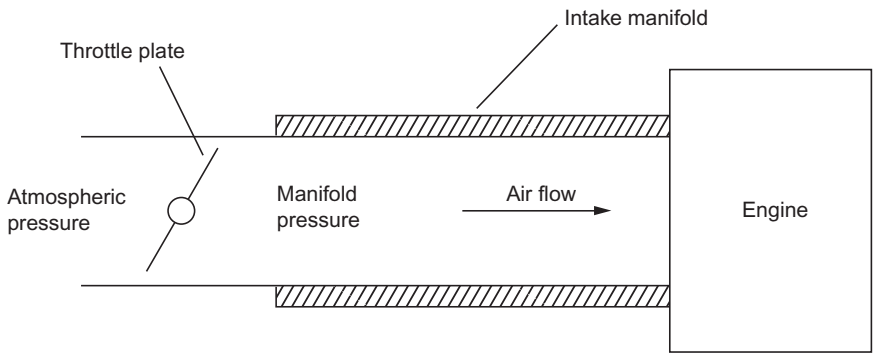


FIG. 4.23 Simplified intake system configuration.

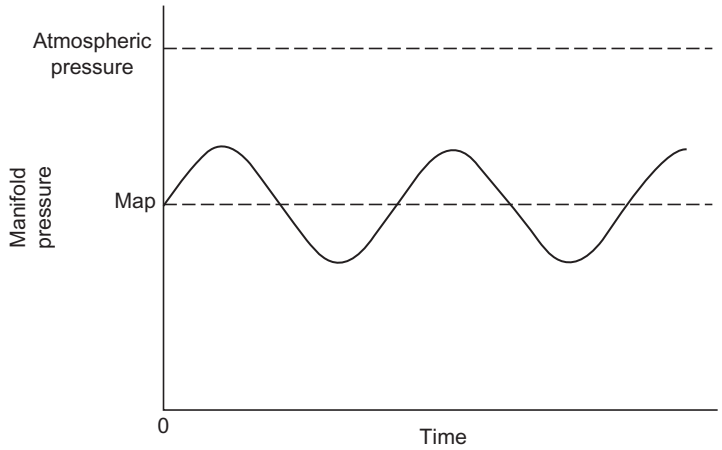


FIG. 4.24 Intake manifold pressure fluctuations.

For a control system application, only average manifold pressure is required. The torque produced by an engine at a constant RPM is approximately proportional to the average value of MAP. The rapid fluctuations in instantaneous MAP are not of interest to the engine controller. Therefore, the manifold pressure measurement method should filter out the pressure fluctuations at frequency f_p and measure only the average pressure. One way to achieve this filtering is to connect the MAP sensor to the intake manifold through a very small diameter tube. The rapid fluctuations in pressure do not pass through this tube, but the average pressure does. The MAP sensor output voltage then corresponds only to the average manifold pressure. Of course, electronic filtering of the MAP sensor voltage is also possible as explained in Appendix A.

MEASURING AIR MASS

A critically important aspect of fuel control is the requirement to measure the mass of air that is drawn into the cylinder (i.e., the air *charge*). The amount of fuel delivered can then be calculated such as to maintain the desired air/fuel ratio. There is no practically feasible way of measuring the mass of air in the cylinder directly. However, the air charge can be determined from the mass flow rate of air into the engine intake since all of this air eventually is distributed to the cylinders (ideally uniform).

There are two methods of determining the mass flow rate of air into the engine. One method uses a single sensor that directly measures MAF. The operation of this sensor is explained in Chapter 5. The other method uses a number of sensors that provide data from which mass flow rate can be computed. This method is known as the *speed-density method*.

Speed-density method

Although the speed-density method of measuring $\dot{M}_a$ has disappeared from contemporary engines, it is, perhaps, worthwhile to review it briefly in part because of its existence in older vehicles and in part to develop an understanding of this engine intake process. The concept for this method is based on the mass density of air as illustrated in Fig. 4.25A.

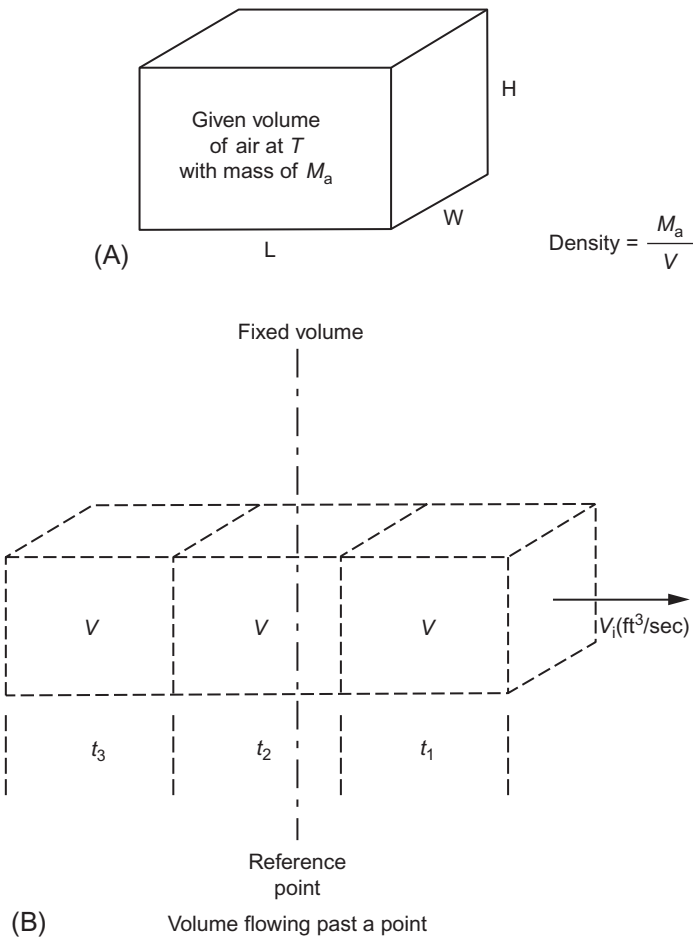


FIG. 4.25 Volume flow rate calculation.

For a given volume of air (V) at a specific pressure (p) and temperature (T) having mass M_a, the density of the air (ρ_a) is given by

$$\rho_a = \frac{M_a}{V} \tag{4.30}$$

This concept can be extended to moving air, as depicted in Fig. 4.25B. Here, air is assumed to be moving through a uniform tube (e.g., the intake pipe for an engine) past a reference point for a specific period of time. This is known as the volume flow rate.

Earlier, it was shown that the MAF $\dot{M}_a$ in the engine is given by

$$\dot{M}_a = \dot{V}_i \rho_a \tag{4.31}$$

where ρ_a is the density of the mixture of fuel air and water vapor: Eq. (4.32)

$$\rho_a = \frac{p_a}{RT_i} \tag{4.32}$$

with p_a being MAP, T_i being inlet air absolute temperature, and R being constant for air and $\dot{V}_i$ is the volume flow rate.

In the above model, we neglect the relatively small contribution to manifold pressure of water vapor and assume that fuel is injected at the intake valve location, which is downstream from the throttle plate.

In Chapter 5, sensors are described for measuring p_a (i.e., MAP sensor) and T_i (i.e., inlet air temperature sensor). Thus, the air density can readily be calculated based on measurements of these sensors. It was also shown earlier in this chapter in the discussion of volumetric efficiency that the air volume flow rate $\dot{V}_i$ into the intake is given by

$$\dot{V}_i = e_v \frac{N}{2} V_D \tag{4.33}$$

Where $N = \dfrac{\text{RPM}}{60}$ V_D is the displacement, and e_v is the volumetric efficiency.

The engine angular speed (N) is readily measured, and the volumetric efficiency is normally measured during the engine mapping process. Tables of e_v versus throttle angle and RPM can be stored in memory for retrieval during a calculation of $\dot{M}_a$ in the engine controller. Thus, via measurement and table lookup, the engine control has sufficient data to calculate (or at least to closely estimate) $\dot{M}_a$ from which fuel delivery quantities are readily computed.

INFLUENCE OF VALVE SYSTEM ON VOLUMETRIC EFFICIENCY

For any given engine configuration, volumetric efficiency is determined by the intake manifold, the valve sizes, and locations, as well as the timing and profile of the cam lobe characteristics. The design of the cam lobe profile determines when the valves open and close and determines the maximum valve-opening (lift). Any given cam profile is optimum only for a relatively narrow range of RPMs and throttle settings. Compromises are made between low-, high-, and midrange RPMs and part throttle versus open or closed throttle.

Ideally, it would be desirable to vary valve timing and lift continuously as the engine operates so as to optimize volumetric efficiency. One technology exists for such variable valve timing (VVT) and is found in certain production vehicles. VVT is also called VVP, which is the preferred terminology in this book.

This technology involves separate camshafts for intake and exhaust valves. These two camshafts are driven via a mechanism that varies the relative timing for intake and exhaust. This mechanism (which includes an electromechanical actuator) and its operation are explained in Chapter 5. There, it is shown that either (or both) intake or exhaust valve timing is varied relative to the engine cycle. The control strategy for regulating VVP is explained in Chapter 6.

In essence, the exhaust valve is open primarily during the exhaust stroke, and the intake valve is open primarily during the intake stroke. Typically in automotive engines, the exhaust valve remains open during the initial portion of the intake valve-opening period. The crankshaft angle over which the two valves are both open (or partially open) is called overlap. Valve overlap permits exhaust action

to assist the intake and improve volumetric efficiency. It also permits some exhaust gas to be mixed with intake gases such the EGR system is at least partially implemented by engine pumping processes. In a variable cam-phasing system, this overlap is minimum at idle and varies with operating conditions to optimize emissions and performance. This topic is discussed in detail in Chapters 5 and 6.

Including EGR

Calculating $\dot{V}_i$ is relatively straightforward in a computer-based control system. Another factor must be taken into account in determining MAF. EGR requires that a certain portion of the charge into the cylinders be exhaust gas. Because of this, a portion of the displacement V_D is exhaust gas. Therefore, the volume flow rate of EGR must be known. A valve-positioning sensor in the EGR valve can be calibrated to provide the flow rate.

From this information for the speed-density method of calculating $\dot{M}_a$, the true volume flow rate of air, $\dot{V}_a$, can be determined by subtracting the volume flow rate of EGR $\dot{V}_{\text{EGR}}$ from $\dot{V}_a$. The total cylinder air charge rate $\dot{V}_a$ is thus given as follows:

$$\dot{V}_a = \dot{V}_i - \dot{V}_{\text{EGR}} \tag{4.34}$$

The volume flow rate of EGR is known from the position of the EGR valve and from engine-operating conditions, as explained in Chapter 6.

Substituting the equation for $\dot{V}_a$, the volume flow rate of air is

$$\dot{V}_a = \frac{NV_D e_v}{2} - \dot{V}_{\text{EGR}} \tag{4.35}$$

Knowing $\dot{V}_a$ and the density ρ_a gives the mass flow rate of air $\dot{M}_a$ as follows:

$$\dot{M}_a = \rho_a \dot{V}_a \tag{4.36}$$

Knowing $\dot{M}_a$, the stoichiometric mass flow rate for the fuel, $\dot{M}_f$, can be calculated as follows:

$$\dot{M}_f = \frac{\dot{M}_a}{14.7} \tag{4.37}$$

Continuing with the discussion of the speed-density method for measuring $\dot{M}_a$, it is the function of the fuel metering actuator to set the fuel mass flow rate at this desired value based on the values of $\dot{V}_a$ and p_a. The control system continuously calculates $\dot{M}_a$ from $\dot{V}_a$ and ρ_a at the temperature and manifold pressure involved and generates an output electric signal to operate the FIs to produce a stoichiometric mass fuel rate. For a practical engine control system, it completes such a measurement, computation, and control signal generation at least once for each cylinder firing.

IDLE SPEED CONTROL

The operation of an automotive engine at idle involves a special consideration. Under idle conditions, there is no input to the throttle from the driver via the accelerator pedal. The engine must produce exactly the torque required to balance all applied load torques from the transmission and any accessories and internal friction and pumping torques in order to run at a steady idle angular speed (RPM). Certain load torques occur as a result of driver action (e.g., change in the transmission selector from park or

neutral to drive or reverse and switching electric loads). However, certain other load torques occur without a direct driver command (e.g., air conditioner clutch actuation).

As in all engine-operating modes, the torque produced by the engine at idle is determined by the mass flow rate of intake air. The electronic fuel control regulates fuel flow to maintain stoichiometry as long as the engine is fully warmed and may briefly regulate fuel to somewhat richer than stoichiometry during cold starts. Normally, at engine idle condition, the electronic engine control is intended to operate the engine at a fixed RPM regardless of load. It does this by regulating mass airflow with the throttle command from the driver at zero. The airflow required to maintain the desired idle RPM must enter the engine via the throttle assembly with the throttle at a small but nonzero angle. Alternatively, some engines are equipped with a special air passage that bypasses the throttle plate. For either method, an actuator is required to enable the electronic engine control system to regulate the idle MAF. Chapter 5 discusses various actuators having application for idle airflow control. For the present discussion, we assume a model for the idle MAF that is representative of the practical actuator configurations discussed in Chapter 5. (Note, in the following analysis, the subscript I is included for all variables and parameters to emphasize that the present system refers to idle speed control.)

Regardless of the idle air bypass configuration, the mass airflow at idle condition (which we denote $\dot{M}_{aI}$) is proportional to the displacement of a movable element that regulates the size of the aperture through which the idle air flows (e.g., the throttle angle θ_T or its equivalent x_T in an idle bypass structure). For the purposes of the present discussion, we assume that the engine indicated torque at idle T_{iI} is given by

$$T_{iI} = K_I \dot{M}_{aI} \tag{4.38}$$

where K_I is the constant for the idle air system; we further assume that $\dot{M}_{aI}$ varies linearly with the position of the idle bypass variable x_I:

$$\dot{M}_{aI} = K_m x_I \tag{4.39}$$

where x_I is the opening in the idle bypass passage way and K_m the constant for this structure.

Typically, the movable element in the idle air bypass structure incorporates a spring that acts to hold $x_I = 0$ in the absence of any actuation. The actuation force (or torque) acts on the force (torque) of this spring and the internal force (torque) in accelerating the mass m_I (or moment of inertia for rotating air bypass configuration) of the movable elements and the friction force (torque). We assume, for the present, a linear model for the actuator motion:

$$m_I \ddot{x}_I + d_I \dot{x}_I + k_I x_I = K_a u \tag{4.40}$$

where d_I is the viscous friction constant, k_I the spring rate of restoring spring, u the actuator input signal, and K_a the actuator constant.

It is also necessary for this discussion of idle speed control to have a model for the relationship between indicated torque and engine angular speed at idle. To avoid potential confusion with other frequency variables, we adapt the notation Ω_I for the crankshaft angular speed of idle (rad/sec). This variable is given by Eq. (4.41)

$$\Omega_I = \pi \frac{\text{RPM}_I}{30} \tag{4.41}$$

Where $\text{RPM}_I = \text{RPM}$ at idle

In general for relatively small changes in Ω_I, the load torques (including friction and pumping torques) can be represented by the following linear model:

$$T_L(\Omega_I) = R_e \Omega_I$$

where R_e is essentially constant for a given engine/load configuration at a particular operating temperature. The indicated torque at idle T_{iI} has the following approximate linear model:

$$T_i \cong J_e \dot{\Omega}_I + T_L(\Omega) \tag{4.42}$$

where J_e is the moment of inertia of engine and load rotating components.

Using the Laplace transform methods of Appendix A, it is possible to obtain the engine transfer function at idle $H_{eI}(s)$:

$$H_{eI}(s) = \frac{\Omega_I(s)}{T_i(s)} \tag{4.43}$$

$$= \frac{1}{J_e s + R_e} \tag{4.44}$$

Similarly, the transfer function for the idle speed actuator dynamics $H_{aI}(s)$ is given by

$$\begin{aligned} H_{aI}(s) &= \frac{x_I(s)}{u(s)} \\ &= \frac{K_a}{m_I(s^2 + 2\zeta_I \omega_I s + \omega_I^2)} \end{aligned} \tag{4.45}$$

Where $\omega_I = \sqrt{k_I / m_I}$

$$\zeta_I = \frac{d_I}{2 m_I \omega_I}$$

These transfer functions can be combined to yield the transfer function (in standard form) of the idle speed control "plant" $H_{pI}(s)$:

$$H_{pI}(s) = \frac{\Omega_I(s)}{u(s)} \tag{4.46}$$

$$= \frac{K_a K_m K_I}{J_e m_I \left[(s^2 + 2\zeta \omega_I s + \omega_I^2) \left(s + \frac{R_e}{J_e} \right) \right]} \tag{4.47}$$

where u is the control variable that is sent to the actuator.

Open loop control of idle speed is not practical owing to the large variations in load and parameter changes due to variations in operating environmental conditions. On the other hand, CL control is well suited to regulating idle speed to a desired value. Fig. 4.26 is a block diagram of such an idle speed control system.

Using the analysis procedures of Appendix A and denoting the idle speed set point Ω_s, it can be shown that the idle speed control CL transfer function H_{CLI} is given by

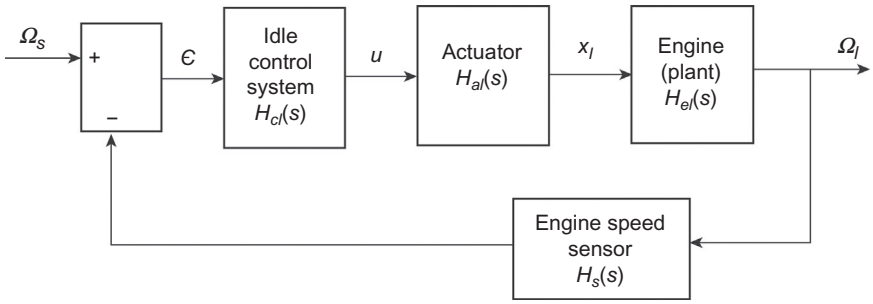


FIG. 4.26 Idle speed control system block diagram.

$$H_{CLI}(s) = \frac{\Omega_I(s)}{\Omega_S(s)} = \frac{H_{CI}(s)H_{pI}}{I + H_s(s)H_{CI}(s)H_{pI}(s)} \tag{4.48}$$

where H_{cI} is the transfer function for the idle speed controller and $H_s(s)$ the transfer function for the crankshaft speed sensor.

In Appendix A, there were three control strategies introduced, P, PI, and PID. Of these, the proportional only (P) is undesirable since it has a nonzero steady-state error between Ω_I and its desired value (Ω_s). It was also shown in Appendix A that a proportional-integral (PI) control had zero steady-state error but could potentially yield an unstable CL system. However, depending upon the system parameters, there are ranges of values for both the proportional gain (K_p) and integral gain (K_I) for which stable operation is possible and for which the idle speed control system has acceptable performance. The controller transfer function for PI control is given by

$$H_{cI}(s) = K_p + \frac{K_I}{s} = K_p\left(\frac{s + s_0}{s}\right) \tag{4.49}$$

For the purpose of illustrating exemplary idle speed control performance, we assume the following set of parameters:

$$\begin{aligned} \zeta_I &= 0.5 \\ \omega_I &= 25\,\text{rad/s} \\ {}_{\omega e} &= R_e/J_e = 10\,\text{rad/s} \\ K_{\text{num}} &= K_a K_m K_I = 250 \\ K_{\text{den}} &= J_e m_I = 0.05 \\ s_0 &= K_I/K_p = 10 \end{aligned}$$

The forward transfer function $H_F(s)$ is defined by the following expression:

$$H_F(s) = H_{cI}(s)H_{pI}(s) = \frac{K_{\text{num}}(s+s_0)}{K_{\text{den}}[(s^3+2\zeta\omega_I s^2+\omega_I^2 s)(s+\omega_e)]} \tag{4.50}$$

The present analysis is simplified by assuming a perfect angular speed sensor such that $H_s(s)=1$. In this case, the CL idle speed control transfer function ($H_{CLI}(s)$) is given by Eq. (4.51)

$$H_{CLI}(s) = \frac{K_p H_F(s)}{1+K_p H_F(s)} \tag{4.51}$$

The influence of proportional gain on stability of this CL idle speed control can be evaluated via root locus techniques as explained in Appendix A. Fig. 4.27 is a plot of the root locus for this idle speed control with the assumed parameters.

It can be seen from this figure that the CL poles all begin in the left half complex plane and are all stable. However, as K_p increases, a pair of poles cross over into the right half complex plane and are unstable. Using the MATLAB "data cursor" function under the tools bar on the root locus plot, it can be seen that for $K_p=1.2$, the poles that migrate to the right-hand side of the complex plane are stable and have a damping ratio of about 25%.

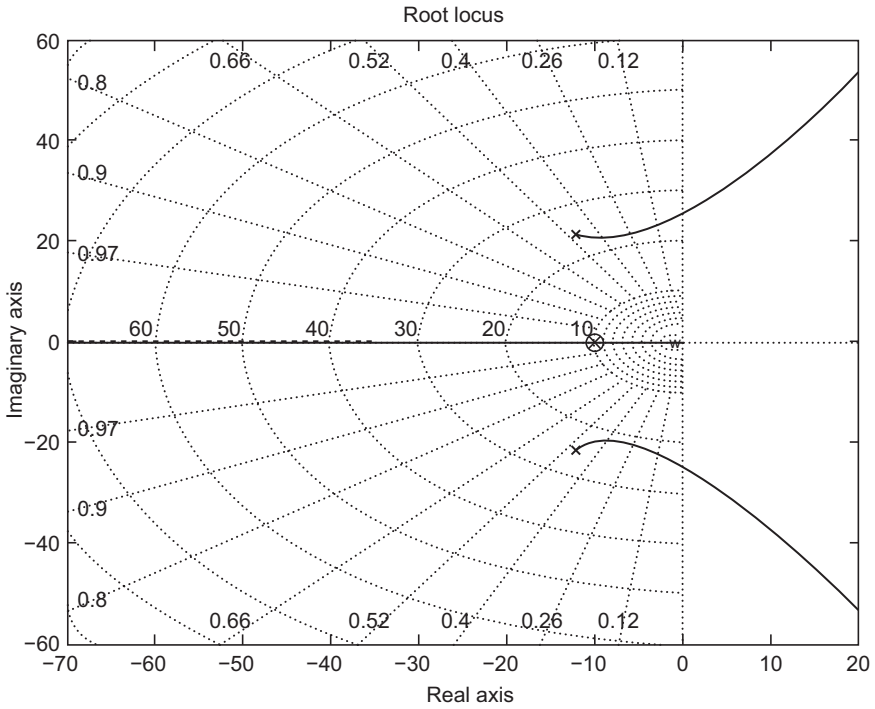


FIG. 4.27 Root locus for idle speed control.

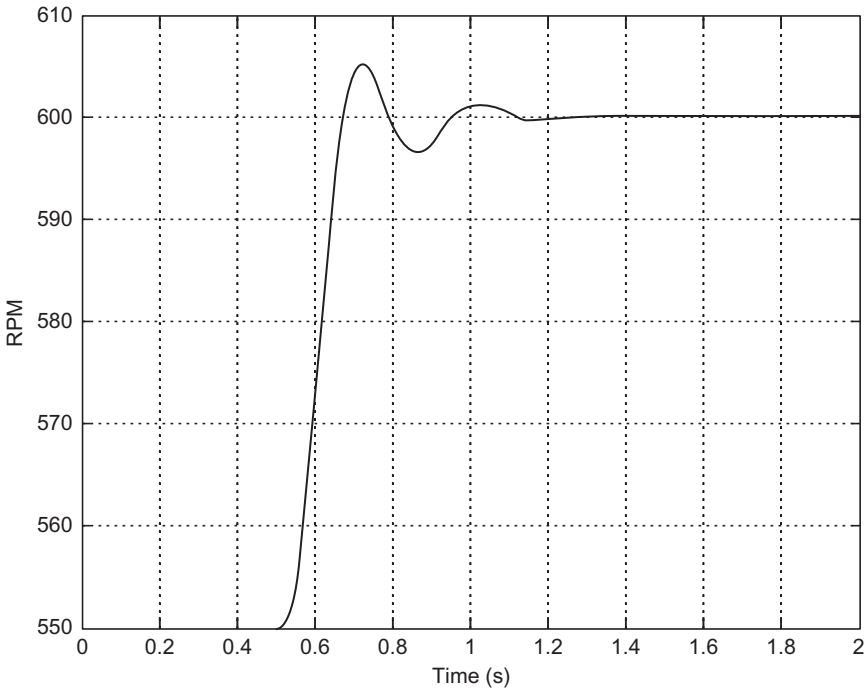


FIG. 4.28 Step response of idle speed control.

Using this value for K_p (i.e., $K_p = 1.2$), the CL dynamic response for the system was examined by commanding a step change in RPM from an initial 550–600 RPM at $t = 0.5$ s. Fig. 4.28 is a plot of the dynamic response of engine idle speed (in RPM) to this command input.

It can be seen that the idle speed reaches the command RPM after a brief transient response with zero steady-state error.

The parameters used in this idle speed control simulation are not necessarily representative of any particular engine. Rather, they have been chosen to illustrate characteristics of this important engine control function. In Chapter 6 where digital engine (power train) control is discussed, a discrete-time control is modeled.

ELECTRONIC IGNITION

The engine ignition system exists solely to provide an electric spark to ignite the mixture in the cylinder. As explained earlier in this chapter, the engine performance is strongly influenced by the spark timing relative to the engine position during the compression stroke (see also Appendix A). The spark advance (relative to TDC) is determined in the electronic engine control based on a number of

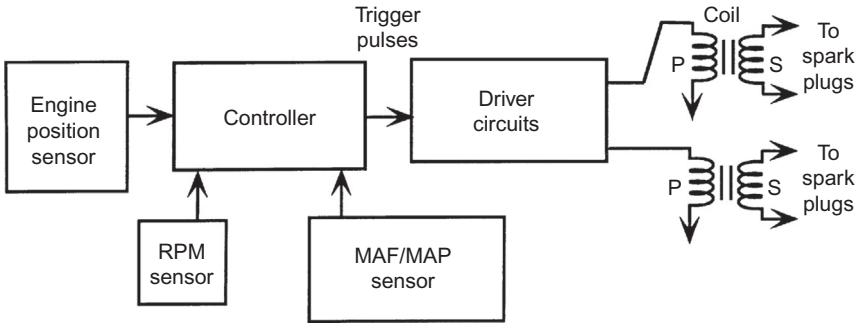


FIG. 4.29 Electronic ignition system configuration.

measurements made by sensors. As will be explained in Chapter 6, the optimum spark advance varies with the intake manifold pressure, RPM, and temperature.

However, in order to generate a spark at the correct spark advance, the electronic engine control must have a measurement of the crankshaft angular position within an engine cycle. The engine position measurement is determined by a sensor coupled to the camshaft and another coupled to the crankshaft as explained by Chapter 5.

Electronic ignition can be implemented as part of an integrated system or as a stand-alone ignition system. A block diagram for the latter system is shown in Fig. 4.29.

Based on measurements from the sensors for engine position, mass airflow or manifold pressure, and RPM, the electronic controller computes the correct spark advance for each cylinder. At the appropriate time, the controller sends a trigger signal to the driver circuits, thereby initiating spark. Before the spark occurs, the driver circuit sends a relatively large current through the primary (P) of the coil. When the spark is to occur, a trigger pulse is sent to the driver circuit to interrupt the current in the primary. A very high voltage is induced at this time in the secondary (S) of the coil. The physical mechanism by which this interrupted primary current causes the high voltage in the coil secondary windings is explained in Chapter 5. This high voltage is applied to the spark plugs, causing them to fire. In those cases for which a coil is associated with two cylinders, one of the two cylinders will be in this compression stroke. Combustion will occur in this cylinder, resulting in power delivery during its power stroke. The other cylinder will be in its exhaust stroke, and the spark will have no effect. Most engines have an even number of cylinders, and there can be a separate driver circuit and coil for each pair of cylinders.

Before proceeding with a discussion of contemporary discrete-time digital control of the complete power train, however, it is necessary to explain and develop models for the critically important components of a control system, sensors and actuators. Chapter 5 is devoted to these important components.

CHAPTER

SENSORS AND ACTUATORS

5

CHAPTER OUTLINE

Automotive Control System Applications of Sensors and Actuators 184
- Variables to be Measured 185
- Airflow Rate Sensor 186
- Pressure Measurements 191
- Engine Crankshaft Angular Position Sensor 194
- Magnetic Reluctance Position Sensor 195
- Hall-Effect Position Sensor 205
- Optical Crankshaft Position Sensor 208

Throttle Angle Sensor 211
Temperature Sensors 213
Typical Coolant Sensor 214
Sensors for Feedback Control 215
- Exhaust Gas Oxygen Sensor 215
- Oxygen Sensor Improvements 220

Knock Sensors 221
Angular Rate Sensor 223
LIDAR 227
Digital Video Camera 229
Flex-Fuel Sensor 235
- Oscillator Methods of Measuring Capacitance 239
- Acceleration Sensor 244

Automotive Engine Control Actuators 247
- Fuel Injection 251
- Exhaust Gas Recirculation Actuator 253

Variable Valve Timing 254
- VVP Mechanism Model 257

Electric Motor Actuators 258
- Two-Phase Induction Motor 263
- Brushless DC Motors 266

Stepper Motors 268
Ignition System 268
- Ignition Coil Operations 269

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-12-810434-7.00005-3

Copyright © 2017 Elsevier Inc. All rights reserved.

The previous chapter introduced two critically important components found in any electronic control system: sensors and actuators. This chapter explains the operation of the sensors and actuators used throughout a modern car. Special emphasis is placed on sensors and actuators used for power train (i.e., engine and transmission) applications since these systems often employ the largest number of such devices. However, this chapter will also discuss sensors found in other subsystems on modern cars.

In any control system, sensors provide measurements of important plant variables in a format suitable for the digital control system (often called an ECU). Similarly, actuators are electrically operated devices that regulate inputs to the plant that directly controls its output. For example, as we shall see, fuel injectors are electrically driven actuators that regulate the flow of fuel into an engine for engine control applications.

In Appendix A, it is explained that fundamentally an electronic control system uses measurements of the plant variable being regulated in the closed-loop mode of operation. The measured variable is compared with a desired value (set point) for the variable to produce an error signal. In the closed-loop mode, the electronic controller generates output electrical signals that regulate inputs to the plant in such a way as to reduce the error to zero. In the open-loop mode, it uses measurements of the key input variable to calculate the desired control variable. Automotive instrumentation (as described in Appendix A) also requires measurement of some variable. For either control or instrumentation applications, such measurements are made using one or more sensors. However, since control applications of sensors demand more accurate sensor performance models, the following discussion of sensors will focus on control applications. The reader should be aware, however, that many of the sensors discussed below also can be used in instrumentation systems.

As will be shown throughout the remainder of this book, automotive electronics has many examples of electronic control in virtually every subsystem. Modern automotive electronic control systems use microcontrollers based on microprocessors (as explained in Chapter 4) to implement almost all control functions. Each of these subsystems requires one or more sensors and actuators in order to operate.

AUTOMOTIVE CONTROL SYSTEM APPLICATIONS OF SENSORS AND ACTUATORS

In any control system application, sensors and actuators are, in many cases, the critical components for determining system performance. This is especially true for vehicular control system applications. The availability of appropriate sensors and actuators dictates the design of the control system and the type of function it can perform.

The sensors and actuators that are available to a control system designer are not always what the designer wants, because the ideal device may not be commercially available at acceptable costs. For this reason, special signal processors or interface circuits often are designed to adapt an available sensor or actuator, or the control system is designed in a specific way to fit available sensors or actuators. However, because of the large potential production run for automotive control systems, it is often worthwhile to develop a sensor for a particular application, even though it may take a long and expensive research project to do so.

Although there are many subsystems on automobiles that operate with sensors and actuators, we begin our discussion with a survey of the devices for power train control. To motivate the discussion of engine control sensors and actuators, it is helpful to review the variables measured (sensors) and the

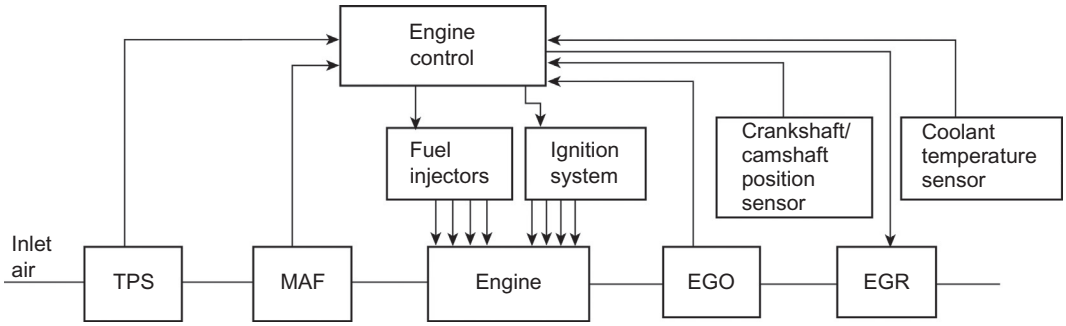


FIG. 5.1 Representative electronic engine control system.

controlled variables (actuators). Fig. 5.1 is a simplified block diagram of a representative electronic engine control system illustrating most of the relevant sensors used for engine control.

As explained in Chapter 6, the position of the throttle plate, sensed by the throttle position sensor (TPS), directly regulates the airflow into the engine, thereby controlling output power. A set of fuel injectors (one for each cylinder) delivers the correct amount of fuel to a corresponding cylinder during the intake stroke under the control of the electronic engine controller to maintain the fuel/air mixture at stoichiometry within a narrow tolerance band. A fuel injector is, as will presently be shown, one of the important actuators used in automotive electronic application. The ignition control system fires each spark plug at the appropriate time under control of the electronic engine controller. The exhaust gas recirculation (EGR) is controlled by yet another output from the engine controller. All critical engine control functions are based on measurements made by various sensors connected to the engine in an appropriate way. Computations made within the engine controller based on these inputs yield output signals to the actuators. We consider inputs (sensors) to the control system first, and then, we will discuss the outputs (actuators).

VARIABLES TO BE MEASURED

The set of variables sensed for any given power train is specific to the associated engine control configuration. Space limitations for this book preclude a complete survey of all power train control systems and relevant sensor and actuator selections for all car models. Nevertheless, it is possible to review a set of possible sensors, which is done in this chapter, and to present representative examples of practical digital control configurations, which is done in the next chapter.

The set of variables sensed in engine control includes the following:

1. Mass airflow (MAF) rate
2. Exhaust gas oxygen concentration
3. Throttle plate angular position
4. Crankshaft angular position/RPM
5. Camshaft angular position
6. Coolant temperature

7. Intake air temperature
8. Ambient air pressure
9. Ambient air temperature
10. Manifold absolute pressure (MAP)
11. Differential exhaust gas pressure (relative to ambient)
12. Vehicle speed
13. Transmission gear selector position
14. Actual transmission gear in use
15. Various pressures

In addition to measurements of the above variables, engine control is also based on the status of the vehicle as monitored by a set of switches. These switches include the following:

1. Air conditioner clutch engaged
2. Brake on/off
3. Wide open throttle
4. Closed throttle
5. Transmission gear selection

AIRFLOW RATE SENSOR

In Chapter 4, we showed that the correct operation of an electronically controlled engine operating with government-regulated exhaust emissions requires a measurement of the mass flow rate of air $(\dot{M}_a)$ into the engine. Throughout this book, the over dot in this notation implies time rate of change. The majority of cars produced since the early 1990s use a relatively simple and inexpensive mass airflow rate (MAF) sensor. This is normally mounted as part of the intake air assembly, where it measures airflow into the intake manifold. It is a ruggedly packaged, single-unit sensor that includes solid-state electronic signal processing. In operation, the MAF sensor generates a continuous signal that varies as a function of true mass airflow $\dot{M}_a$.

Before explaining the operation of the MAF, it is, perhaps, helpful to review the characteristics of the inlet airflow into an engine. It has been shown in Chapter 4 that a four-stroke reciprocating engine functions as an air pump with air pumped sequentially into each cylinder every two crankshaft revolutions. The dynamics of this pumping process are such that the airflow consists of a fluctuating component (at half the crankshaft rotation frequency times the number of cylinders) superposed on a quasisteady component. This latter component is a constant only for constant engine operation (i.e., steady power at constant RPM such as might be achieved at a constant vehicle speed on a level road). However, automotive engines rarely operate at absolutely constant power and RPM. The quasisteady component of airflow changes with load and speed. It is this quasisteady component of $\dot{M}_a(t)$ that is measured by the MAF for engine control purposes. One way of characterizing this quasisteady-state component is as a short-term time average over a time interval τ (which we denote $\dot{M}_{a\tau}(t)$) where

$$\dot{M}_{a\tau}(t) = \frac{1}{\tau} \int_{t-\tau}^{t} \dot{M}_a(t')dt' \tag{5.1}$$

The integration interval (τ) must be long enough to suppress the time-varying component at the lowest cylinder pumping frequency (e.g., idle RPM) yet short enough to preserve the transient characteristics of airflow associated with relatively rapid throttle position changes.

Alternatively, the quasisteady component of mass airflow can be represented by a low-pass-filtered version of the instantaneous flow rate. Appendix A explains that a low-pass filter (LPF) can be characterized (in continuous time) by an operational transfer function ($H_{\mathbf{LPF}}(s)$) of the form

$$H_{\mathrm{LPF}}(s) = \frac{b_o + b_1 s + \cdots b_m s^m}{a_o + a_1 s + \cdots a_n s^n} \tag{5.2}$$

where the coefficients determine the response characteristics of the filter. The filter bandwidth effectively selects the equivalent time interval over which mass airflow measurements are averaged. Of course, in practice with a digital power train control system, mass airflow measurements are sampled at discrete times, and the filtering is implemented as a discrete time transformation of the sampled data (see Appendix B).

A typical MAF sensor is a variation of a classic airflow sensor that was known as a hot-wire anemometer and was used, for example, to measure wind velocity for weather forecasting and for various scientific studies. In the typical MAF, the sensing element is a conductor or semiconductor thin-film structure mounted on a substrate. On the air inlet side, a honeycomb flow straightener is mounted that "smoothes" the airflow (causing nominally laminar airflow over the film element).

The concept of such an airflow sensor is based upon the variation in resistance of the two-terminal sensing element with temperature. A current is passed through the sensing element supplying power to it, thereby raising its temperature and changing its resistance. When this heated sensing element is placed in a moving airstream (or other flowing gas), heat is removed from the sensing element as a function of the mass flow rate of the air passing the element and the temperature difference between the moving air and the sensing element. For a constant supply current (i.e., heating rate), the temperature at the element changes in proportion to the heat removed by the moving airstream, thereby producing a change in its resistance. A convenient model for the sensing element resistance (R_{SE}) at temperature (T) is given by Eq. (5.3)

$$R_{\mathrm{SE}}(T) = R_o + k_T(T - T_{\mathrm{ref}}) \tag{5.3}$$

where R_o is the resistance at some reference temperature T_{ref} (e.g., 0°C) and $k_{\boldsymbol{T}}$ is the resistance/temperature coefficient. The current i_2 flowing through R_{SE} causes its temperature to rise above ambient temperature such that $T = T_a + \Delta T$. The relationship between ΔT and $\dot{M}_a$ is explained later in this section of the chapter for a conducting sensing element, $k_{\boldsymbol{T}} > 0$, and for a semiconducting sensing element, $k_{\boldsymbol{T}} < 0$.

The mass flow rate of the moving airstream is measured via a measurement of the change in resistance. There are many potential methods for measuring mass airflow via the influence of mass airflow on the sensing element resistance. One such scheme involves connecting the element into a so-called bridge circuit as depicted in Fig. 5.2.

In the bridge circuit, three resistors (R_1, R_2, and R_3) are connected as depicted in Fig. 5.2 along with a resistive sensing element denoted $R_{\mathrm{SE}}(T)$. This sensing element consists of a thin film of conducting (e.g., Ni) or semiconducting material that is deposited on an insulating substrate. The voltages V_1 and V_2 (depicted in Fig. 5.2) are connected to the inputs of a relatively high-gain differential amplifier.

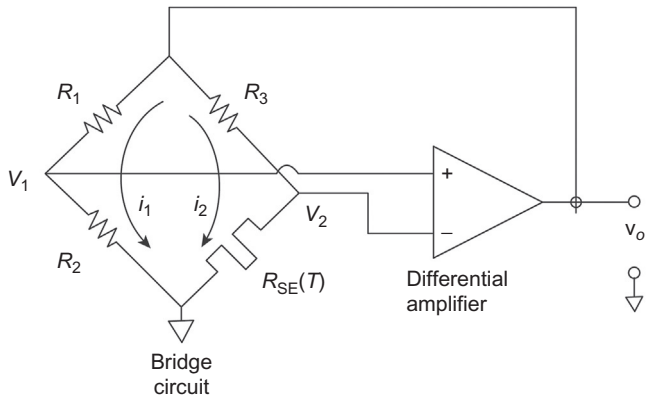


FIG. 5.2 Mass airflow sensor.

The output voltage of this amplifier v_o is connected to the bridge (as shown in Fig. 5.2) and provides the electrical excitation for the bridge. This voltage is given by

$$v_o = G(V_1 - V_2) \tag{5.4}$$

where G is the amplifier voltage gain.

In this bridge circuit, only that sensing element is placed in the moving airstream whose mass flow rate is to be measured. The other three resistances are mounted such that they are at the same ambient temperature (T_a) as the moving air.

The combination of bridge circuit and differential amplifier forms a closed loop in which the temperature difference ΔT between the sensing element and the ambient air temperature remains fixed independent of T_a (which for an automobile can vary by more than 100°F). We discuss the circuit operation first and then explain the compensation for variation in T_a.

For the purposes of this explanation of the MAF operation, it is assumed that the input impedance at both differential amplifier inputs is sufficiently large that no current flows into either the + or − input. With this assumption, the differential input voltage ΔV is given by

$$\Delta V = V_1 - V_2 \tag{5.5}$$

$$= v_o \left[\frac{R_2}{R_1 + R_2} - \frac{R_{SE}}{R_{SE} + R_3} \right] \tag{5.6}$$

However, since it has been shown that $v_o = G\Delta V$, the following equation can be shown to be valid:

$$\frac{1}{G} = \left[\frac{R_2}{R_1 + R_2} - \frac{R_{SE}}{R_{SE} + R_3} \right] \tag{5.7}$$

In the present MAF sensor configuration, it is assumed (as is often found in practice) that $G \gg 1$. For sufficiently large G, from Eq. (5.7), we can see that R_{SE} is given approximately by

$$R_{SE}(T) = \frac{R_2 R_3}{R_1} \tag{5.8}$$

In this case, it can be shown using Eq. (5.3) that the temperature difference between the sensing element and the ambient air is given approximately by

$$K_T \Delta T = \frac{R_2 R_3}{R_1} - [R_0 + K_T(T_a - T_{\text{ref}})] \tag{5.9}$$

where T_{ref} is an arbitrary reference temperature.

This temperature difference can be made independent of ambient temperature T_a by the proper choice of R_3, which is called the temperature-compensating resistance. In one such method, R_3 is made with the same material but possibly with a different structure as the sensing element such that its resistance is given by

$$R_3(T_a) = R_{3o} + K_{T3}(T_a - T_{\text{ref}}) \tag{5.10}$$

where R_{3o} is the resistance of R_3 at $T_a = T_{\text{ref}}$ and K_{T3} is the temperature coefficient of R_3.

The sensing element temperature difference ΔT is given by

$$K_T \Delta T = \left(\frac{R_2 R_{3o}}{R_1} - R_0\right) + \left(\frac{R_2}{R_1} K_{T3} - K_T\right)(T_a - T_{\text{ref}}) \tag{5.11}$$

If the sensor is designed such that

$$\frac{R_2 K_{T3}}{R_1} = K_T$$

then ΔT is independent of T_a and is given by Eq. (5.12)

$$\Delta T = \frac{1}{K_T}\left[\frac{R_2 R_{3o}}{R_1} - R_o\right] \tag{5.12}$$

This temperature difference is determined by the choice of circuit parameters and is independent of amplifier gain for sufficiently large gain (G).

The preceding analysis has assumed a steady mass airflow (i.e., $\dot{M}_a = \text{constant}$). The mass airflow into an automotive engine is only approximate constant for certain driving conditions, so it is useful to consider the MAF sensor dynamic response to time-varying $\dot{M}_a$. The combination of bridge circuit and differential amplifier has essentially instantaneous dynamic response to changes in $\dot{M}_a$. The dynamic response of the MAF of Fig. 5.2 is determined by the dynamic temperature variations of the sensing element. Whenever the mass airflow rate changes, the temperature of the sensing element changes. The voltage v_o changes, thereby changing the power P_{SE} dissipated in the sensing element in such a way as to restore ΔT to its equilibrium value. An approximate model for the dynamic response of ΔT to changes in $\dot{M}_a$ is given by

$$\Delta \dot{T} + \frac{\Delta T}{\tau_{\text{SE}}} = \alpha_1 P_{\text{SE}} - \alpha_2 \dot{M}_a \tag{5.13}$$

where $P_{\text{SE}} = i_2^2 R_{\text{SE}}$:

$$P_{\text{SE}} = \left(\frac{v_o}{R_{\text{SE}} + R_3}\right)^2 R_{\text{SE}}$$

In Eq. (5.13), $i_2 =$ current shown in Fig. 5.2

τ_{SE} = sensing element time constant and where, α_1 and α_2 are constants for the sensing element configuration.

The Laplace methods of analysis in Appendix A are not applicable for solving this nonlinear differential equation for the exact time variation of T_{SE}. However, a well-designed sensing element has a sufficiently short time constant τ_{SE} such that the variation in ΔT is negligible (i.e., $\Delta T \backsimeq$ constant). In this case, the change in power dissipation from the zero airflow condition is given by

$$\alpha_1 \left[P_{SE}\left(\dot{M}_a\right) - P_{SE}(0)\right] = \alpha_2 \dot{M}_a \tag{5.14}$$

It can be shown from Eq. (5.14) that MAF sensor output voltage varies as given below:

$$\mathrm{v}_o\left(\dot{M}_a\right) = \left[\mathrm{v}_o^2(0) + K_{MAF}\dot{M}_a\right]^{1/2} \tag{5.15}$$

where K_{MAF} is the constant for the MAF configuration.

As an example of this variation, Fig. 5.3 is a plot of the sensor voltage versus airflow for a production MAF sensor. This example sensor uses a Ni film for the sensing element.

The conversion of MAF to voltage is nonlinear, as indicated by the calibration curve depicted in Fig. 5.3, for the example MAF sensor. Fortunately, a modern digital engine controller can convert the analog bridge output voltage directly to mass airflow by simple computation. As will be shown in Chapter 6, in which digital engine control is discussed, it is necessary to convert analog sensor voltage from the MAF to a digital format. The analog output of the differential amplifier can be sampled and converted to digital format using an A/D converter (see Chapter 3). The engine control system can calculate $\dot{M}_a$ from v_o using the known functional relationship $\mathrm{v}_o\left(\dot{M}_a\right)$.

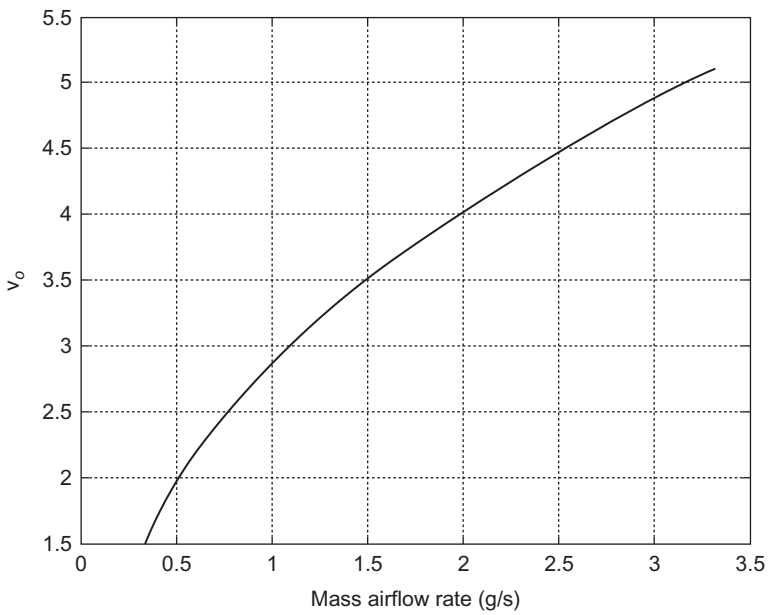


FIG. 5.3 Output voltage for example MAF versus mass flow rate g/s.

PRESSURE MEASUREMENTS

There are numerous potential applications for the measurement of pressure (both pneumatic and hydraulic) at various points in the modern automobile, including ambient air pressure, intake manifold absolute pressure, tire pressure, oil pressure, coolant system pressure, transmission actuation pressure, and several others. Essentially, in all such measurements, the basis for the measurement is the change in an electrical parameter or variable (e.g., resistance and voltage) in a structure that is exposed to the pressure. Space limitations prevent us from explaining all of the many pressure sensors used in a vehicle. Rather, we illustrate pressure-type measurements with the specific example of intake manifold absolute pressure (MAP). Although it is obsolete in contemporary vehicles, the speed-density method (discussed in Chapter 4) of calculating mass airflow in early emission regulation vehicles used such a MAP sensor. The following (MAP) sensor discussion is representative of other pressure sensors in that it illustrates the change in circuit parameters with pressure.

Strain gauge MAP sensor

One relatively inexpensive MAP sensor configuration is the silicon-diaphragm diffused strain-gauge sensor shown in Fig. 5.4. This sensor uses a silicon chip that is $\sim$3 mm^2. Along the outer edges, the chip is $\sim$250 μm (1 μm $= 10^{-6}$ m) thick, but the center area is only 25 μm thick and forms a diaphragm. The edge of the chip is sealed to a Pyrex plate under vacuum, thereby forming a vacuum chamber between the plate and the center area of the silicon chip.

A set of sensing resistors is formed around the edge of this chamber, as indicated in Fig. 5.4. The resistors are formed by diffusing a doping impurity into the silicon. External connections to these resistors are made through wires connected to the metal bonding pads. This entire assembly is placed in a sealed housing that is connected to the intake manifold by a small-diameter tube. Manifold pressure applied to the diaphragm causes it to deflect.

Diaphragm deflection in response to an applied pressure results in a small elongation of the diaphragm along its surface. The elongation of any linear isotropic material of length L corresponds to the length becoming $L+\delta L$ in response to applied pressure. For linear deformation, $\delta L \ll L$. The elongation is quantitatively represented by its strain $\in$, which is given by

$$\in = \frac{\delta L}{L} \tag{5.16}$$

In any diaphragm made from a linear material, the strain is proportional to the applied pressure (p):

$$\in = K_D p \tag{5.17}$$

where K_D is a constant that is determined by the diaphragm configuration (e.g., its shape and area exposed to p and its thickness).

The resistance of the sensing resistors changes in proportion to the applied manifold pressure by a phenomenon that is known as *piezoresistivity*. Piezoresistivity occurs in certain semiconductors so that the actual resistivity ρ (the reciprocal of conductivity as explained in Chapter 2) changes in proportion to the strain. The strain induced in each resistor is proportional to the diaphragm deflection, which, in turn, is proportional to the pressure on the outside surface of the diaphragm. For a MAP sensor, this pressure is the manifold absolute pressure.

An electrical signal that is proportional to the manifold pressure is obtained by connecting the resistors in a circuit called a Wheatstone bridge (similar to that for the MAF sensor), as shown in

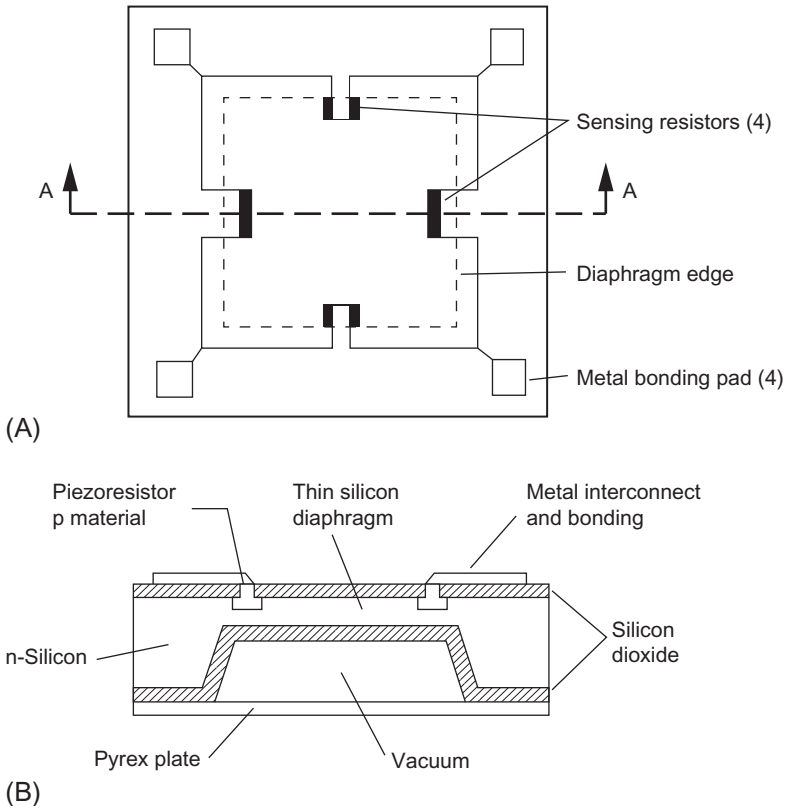


FIG. 5.4 Exemplary manifold pressure sensor configuration. (A) Top view and (B) section A-A.

the schematic of Fig. 5.5A. The voltage regulator holds a constant DC voltage (V_s) across the bridge. The resistors diffused into the diaphragm are denoted R_1, R_2, R_3, and R_4 in Fig. 5.5A. When there is no strain on the diaphragm, all four resistances are equal, and the bridge is balanced, which means that the voltage between points A and B is zero. When manifold pressure changes, it causes these resistances to change in such a way that R_1 and R_3 increase by an amount that is proportional to pressure; at the same time, R_2 and R_4 decrease by an identical amount. This unbalances the bridge, and a net difference voltage is present between points A and B. The differential amplifier generates an output voltage proportional to the difference between the two input voltages (which is, in turn, proportional to the pressure), as shown in Fig. 5.5B.

We illustrate the operation of this sensor with the following model. The voltage at point A is denoted V_A and at point B as V_B. The resistances R_1 and R_3 are given by

$$R_n(\epsilon) = R_o + R_\epsilon \epsilon \quad n = 1,3 \tag{5.18}$$

where

$$R_\epsilon = \left.\frac{dR}{d\epsilon}\right|_{\epsilon=o} > 0 \tag{5.19}$$

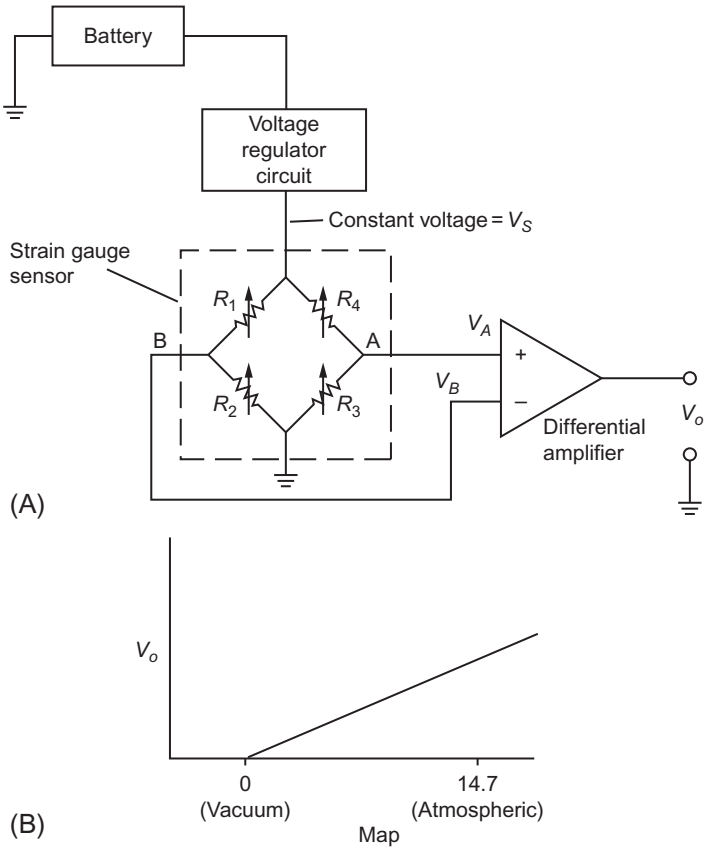


FIG. 5.5 Example MAP sensor circuit. (A) Circuit and (B) V_o versus MAP.

For resistances R_2 and R_4, the model for resistance is given by

$$R_m(\in) = R_o - R_\in \in \quad m = 2,4 \tag{5.20}$$

The voltages V_A and V_B are given, respectively, by

$$V_A = V_S\left(\frac{R_3}{R_3 + R_4}\right) = V_S\frac{(R_o + R_\in \in)}{2R_o} \tag{5.21}$$

$$V_B = V_S\left(\frac{R_2}{R_2 + R_4}\right) = V_S\frac{(R_o - R_\in \in)}{2R_o} \tag{5.22}$$

The voltage difference $V_A - V_B$ is given by

$$V_A - V_B = V_S\frac{R_\in \in}{R_o} \tag{5.23}$$

The differential amplifier output voltage (V_o) is given by

$$V_o = G_A(V_A - V_B) \tag{5.24}$$

$$= G_A \frac{V_S R_\in \in}{R_o} \tag{5.25}$$

where G_A is the amplifier voltage gain. Since the sensor strain is proportional to pressure, the output voltage is also proportional to the applied pressure:

$$V_o = G_A \frac{V_S R_\in}{R_o} K_D p$$

This pressure signal can be input to the digital control system via sampling and an analog-to-digital converter (see Appendix B).

ENGINE CRANKSHAFT ANGULAR POSITION SENSOR

Another important measurement for electronic engine control is the angular position of the crankshaft relative to a reference position. The crankshaft angular position is often termed the "engine angular position" or simply "engine position." It will be shown that the sensor for measuring crankshaft angular position can also be used to calculate its instantaneous angular speed. It is highly desirable that this measurement be made without any mechanical contact with the rotating crankshaft. Such noncontacting measurements of any rotating shafts (i.e., in engine or drivetrain) can be made in a variety of ways, but the most common of these in automotive electronics use magnetic or optical phenomena as the physical basis. Magnetic means of such measurements are generally preferred in engine applications since they are unaffected by oil, dirt, or other contaminants. There are other applications in vehicular systems of a sensor capable of measuring angular position/velocity.

The principles involved in measuring rotating shafts are illustrated by this example that is one of the most significant applications for engine control (the measurement of crankshaft angular position or angular velocity (i.e., RPM)). Imagine the engine as viewed from the rear, as shown in Fig. 5.6.

On the rear of the crankshaft is a large, circular steel disk called the *flywheel* that is connected to and rotates with the crankshaft A point on the flywheel is denoted the flywheel mark, as shown in Fig. 5.6. A reference line is taken to be a line through the crankshaft axis of rotation and a point (b) on the engine block. For the present discussion, the reference line is taken to be a horizontal line. The crankshaft angular position is the angle between the reference line and the line through the axis and the flywheel mark.

Imagine that the flywheel is rotated so that the mark is directly on the reference line. This is an angular position of 0 degree. For our purposes, assume that this angular position corresponds to the no. 1 cylinder at top dead center (TDC) on either intake or power strokes. As the crankshaft rotates, this angle increases from 0 to 360 degrees in one revolution. However, one full engine cycle from intake through exhaust requires two complete revolutions of the crankshaft; that is, one complete engine cycle corresponds to the crankshaft angular position going from 0 to 720 degrees. During each cycle, it is important to measure the crankshaft position relative to the reference for each cycle in each cylinder. This information is used by the electronic engine controller to set ignition timing and to set the fuel injector pulse timing.

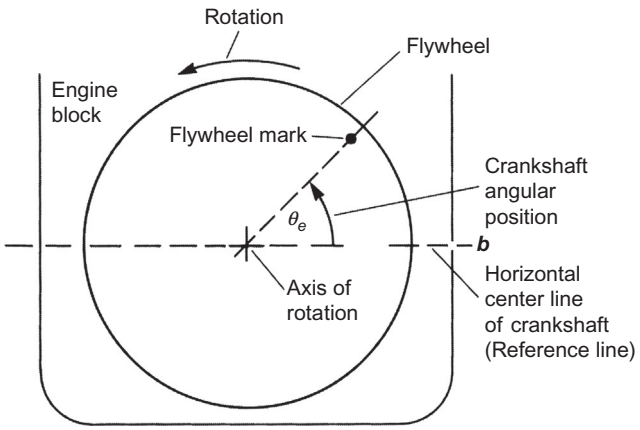


FIG. 5.6 Illustration of crankshaft angular position representation.

In automobiles with electronic engine control systems, angular position θ_e can be sensed on the crankshaft directly or on the camshaft. Recall that the piston drives the crankshaft directly, while the valves are driven from the camshaft. The camshaft is driven from the crankshaft through a 1:2 reduction drivetrain, which can be gears, belt, or chain. Therefore, the camshaft rotational speed is one-half that of the crankshaft, so the camshaft angular position goes from 0 to 360 degrees for one complete engine cycle. Either or both of these sensing locations can be used in electronic control systems. Although the crankshaft location is potentially superior for accuracy because of torsional and gear backlash errors in the camshaft drivetrain, many production systems locate this sensor such that it measures camshaft position. For the measurement of engine position via a crankshaft sensor, an unambiguous measurement of the crankshaft angular position relative to a unique point in the cycle for each cylinder requires some measurement of camshaft position and crankshaft position. Typically, it is sufficient to sense camshaft position at one point in a complete revolution. Traditionally, engine position involved measuring crankshaft position directly in combination with measuring camshaft position. In principle, it is sufficient for engine control purposes to measure crankshaft/camshaft position at a small number of fixed points. The number of such measurements (or samples), for example, could be determined by the number of cylinders.

MAGNETIC RELUCTANCE POSITION SENSOR

One noncontacting engine sensor configuration that measures crankshaft position directly (using magnetic phenomena) is illustrated in Fig. 5.7. This sensor consists of a permanent magnet with a coil of wire wound around it. A steel disk that is mounted on the crankshaft (usually in front of the engine) has tabs that pass between the pole pieces of this magnet. In Fig. 5.7 for illustrative purposes, the steel disk has four protruding tabs, which is the minimum number of tabs for an eight-cylinder engine. In general, there are N tabs where N is determined during the design of the engine control system. The passage of each tab could correspond, for example, to the TDC position of a cylinder, although other reference positions are also possible. The crankshaft position θ at all other times in the engine cycle are given by

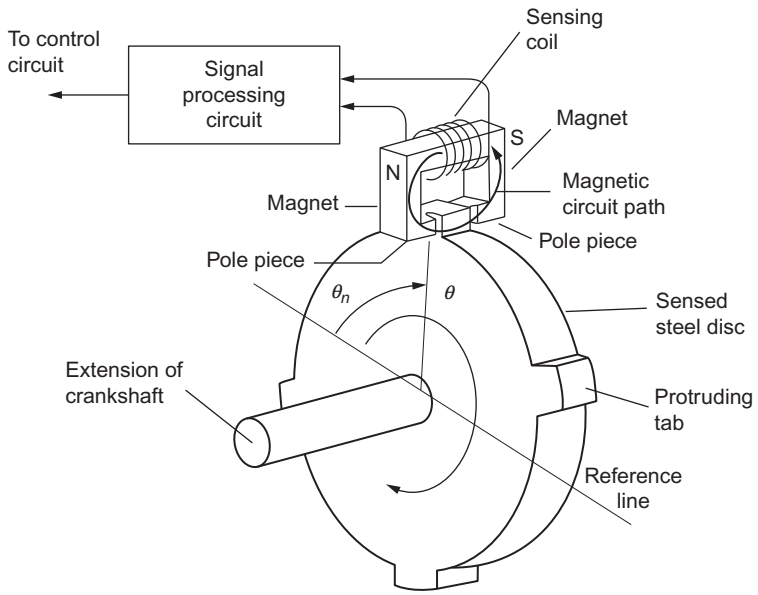


FIG. 5.7 Magnetic crankshaft angular position sensor configuration.

$$\theta - \theta_n = \int_{t_n}^{t} \omega(t)dt \quad t_n < t < t_{n+1} \tag{5.26}$$

where θ_n is the angular position of the nth tab relative to a reference line, t_n is the time of passage of the nth tab associated with the reference point for the corresponding cylinder during the nth engine cycle, and ω is the instantaneous crankshaft angular speed. Of course, the times t_n are determined in association with camshaft reference positions. The camshaft sensor provides a reference point in the engine cycle that determines the index n above. The precision in determining engine position within each cycle for each cylinder is improved by increasing the number of tabs on the disk.

The sensor in Fig. 5.7 (and any magnetic sensor) incorporates one or more components of its structure that are of a ferromagnetic material such as iron, cobalt, or nickel or any of the class of manufactured magnetic materials (e.g., ferrites). Performance analysis and/or modeling of automotive sensors based upon magnetic phenomena, strictly speaking, requires the determination of the magnetic fields associated with the configuration. The full, precise, and accurate determination of the magnetic field distributions for any sensor configuration is beyond the scope of this book. However, approximate analysis of such magnetic fields for structures having relatively simple geometries is possible with the introduction of the following simplified theory for the associated magnetic field distributions.

The magnetic field in a material is described by a pair of field quantities that can be compared with the voltage and current of an ordinary electric circuit. One of these quantities is called the *magnetic field intensity vector* $\bar{H}$. It exerts a force analogous to voltage. The response of the magnetic circuit

to the magnetic field intensity is described by the second vector, which is called *magnetic flux density vector* $\bar{B}$, which is analogous to current. In these two quantities, the overbar indicates that each is a vector quantity.

The structure of any practical magnetic sensor (which provides noncontact measurement capability) will have a configuration that consists, at least, in part of ferromagnetic material. Ferromagnetism is a property of the transition metals (iron, cobalt, and nickel) and certain alloys and compounds made from them. Magnetic fields in these materials are associated with electron spin for each atom. Physically, such materials are characterized by small regions called domains, each having a magnetic field associated with it due to the parallel alignment of the electron spins (i.e., each domain is effectively a tiny permanent magnet). If no external magnetic field is applied to the material, the magnetic field directions of the domains are randomly oriented, and the material creates no permanent external magnetic field. Whenever an external magnetic field is applied to a ferromagnetic material, the domains tend to be reoriented such that their magnetic fields tend to align with the external field, thereby increasing the external magnetic flux density in the direction of the applied magnetic field intensity.

Fig. 5.8 illustrates the functional relationship of the scalar magnitudes $B(H)$ for a typical ferromagnetic material having a configuration such as is depicted in Fig. 5.7.

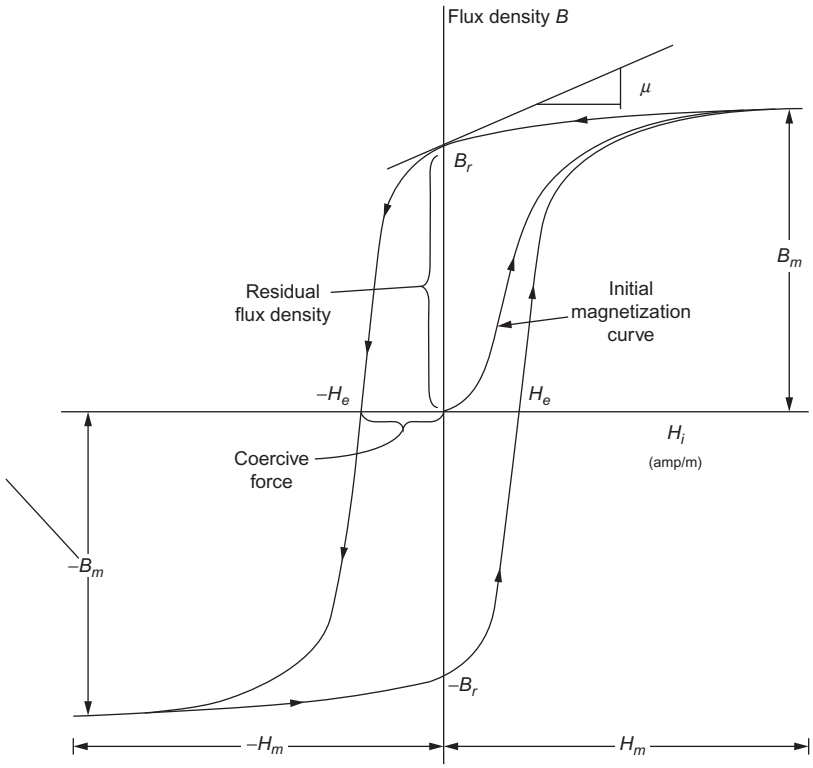


FIG. 5.8 Magnetization curve for exemplary ferromagnetic material.

The externally applied magnetic field intensity H_i is created by passing a current through the coil of N turns. If the material is initially unmagnetized and the current is increased from zero, the $B(H_i)$ follows the portion labeled "initial magnetization curve." The arrows on the curves of Fig. 5.8 indicate the direction of the change in H_i. The contribution of the ferromagnetic material to the flux density is called magnetization M and is given by

$$M = \frac{B}{\mu_o} - H_i \tag{5.27}$$

where μ_o is the magnetic permeability of free space.

For a sufficiently large applied H_i (e.g., $H_i > H_m$), all of the domains are aligned with the direction of H and B saturates such that $(B - B_m) = \mu_o(H_i - H_m)$, where H_m and B_m are depicted in Fig. 5.8. If the applied field is reduced from saturation to zero, the ferromagnetic material has a nonzero flux density denoted B_r in Fig. 5.8, and the corresponding magnetization M_r (called remanent magnetization) causes the material to become a permanent magnet. Essentially, all ferromagnetic materials exhibit hysteresis in the $B(H)$ relationship as depicted in Fig. 5.8. Certain ferromagnetic materials have such a large remanent magnetization that they are useful in providing a source of magnetic field for some automotive sensors. The structure depicted in Fig. 5.7 is such a sensor.

Normally, in automotive sensors, the signals involved correspond to relatively small incremental changes in B and H about a steady value. For example, the sensor of Fig. 5.7 operates with small B and H incremental changes about the remanent magnetization such that B is given approximately by

$$B = B_r + \mu H_i \tag{5.28}$$

where

$$\mu = \left.\frac{dB}{dH_i}\right|_{H_i=0} \tag{5.29}$$

is the incremental permeability of the ferromagnetic materials.

The straight line of Fig. 5.8 passing through $B = B_r$ and $H_i = 0$ has slope μ as defined above.

Ferromagnetic materials have very high incremental permeability relative to nonmagnetic materials. For sensor regions that can be described by the scalar model (i.e., $B = \mu H$), the incremental permeability is given by

$$\mu = \mu_r \mu_o$$

where μ_o is the permeability of free space and μ_r is the relative permeability of the material. For any ferromagnetic material $\mu_r \gg 1$.

From electromagnetic theory, there is an important fundamental equation, which is useful in the present analysis of any magnetic automotive sensor. That equation relates the contour integral of $\bar{H}$ along a closed contour C and is given by

$$\oint_C \bar{H} \cdot \overline{d\ell} = I_T \tag{5.30}$$

where I_T is the total current passing normal to and through the surface enclosed by C and where $\overline{d\ell}$ is a differential length along contour C. This integral equation will be shown to be useful for analyzing magnetic automotive sensors of the type depicted in Fig. 5.7.

Another relationship that is useful for developing the model for a magnetic sensor is continuity of the normal component of $\bar{B}$ at the interface of any two materials. This continuity is expressed by the relationship

$$\bar{B}_1 \cdot \hat{n} = \bar{B}_2 \cdot \hat{n} \tag{5.31}$$

where $\bar{B}_1$ and $\bar{B}_2$ are the magnetic flux densities in two materials at their interface and $\hat{n}$ is the unit vector normal to the surface at the interface. These two important fundamental equations are used in the modeling of the sensor of Fig. 5.7 and other similar magnetic sensors.

The path for the magnetic flux of the sensor of Fig. 5.7 is illustrated in Fig. 5.9.

In Fig. 5.9, g_c is the width of the gap in the pole piece, and t_T is the thickness of the steel disk. For a configuration such as is shown in Fig. 5.9, the lines of constant magnetic flux follow paths as indicated in the figure. The following notation is used:

$\bar{B}_m$ is the flux density within the ferromagnetic material.
$\bar{H}_m$ is the magnetic field intensity within the ferromagnetic material.
$\bar{B}_g$ is the flux density within air gaps.
$\bar{H}_g$ is the magnetic field intensity within air gaps.

From Eq. (5.30) above, the following equation can be written for the contour shown in Fig. 5.9:

$$\int_C \bar{H} \cdot \overline{d\ell} \cong H_g g_a + H_m L_m \tag{5.32}$$

where g_a is the total air gap length along contour C, L_m is the total length along contour C within the material, and C is the closed path along line of constant B.

We consider first the open-circuit case in which $I_T = 0$. In this case, the air gap magnetic field intensity H_g is given by

$$H_g \cong -H_m L_m / g_a \tag{5.33}$$

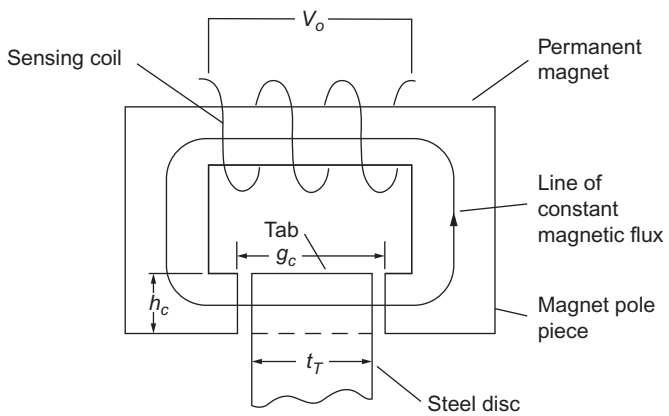


FIG. 5.9 Magnetic circuit of the sensor of Fig. 5.7.

From Eq. (5.31), the following equation can be written for the interface between the ferromagnetic material and the air gap:

$$\bar{B}_g \cdot \hat{n}_g = \bar{B}_m \cdot \hat{n}_m$$

However, since the lines of magnetic flux are normal to this interface,

$$\bar{B}_g \cdot \hat{n}_g = B_g$$

and

$$\bar{B}_m \cdot \hat{n}_m = B_m$$

or

$$B_g = B_m$$

That is, the magnetic flux density for the configuration of Fig. 5.9 is constant along the path denoted therein. Within the material, the following relationship is valid:

$$\begin{aligned} B_m &= \mu_o(H_m + M_r) \\ &= B_g \\ &= \mu_o H_g \end{aligned} \tag{5.34}$$

where M_r is the remanent magnetization of the pole piece. Thus, we can write

$$H_m = H_g - M_r \tag{5.35}$$

For a magnetized ferromagnetic material $M_r \gg H_g$ such that

$$H_m \cong -M_r \tag{5.36}$$

Combining Eqs. (5.29), (5.30), the flux density is given by

$$\begin{aligned} B_g &= \mu_o H_g \\ &= \mu_o \frac{M_r L_m}{g_a} \end{aligned} \tag{5.37}$$

Eq. (5.37) shows that the magnitude of B around the contour C varies inversely with the size of the air gap along that path. Note that when one of the tabs of the steel disk is located between the pole pieces of the magnet, a large part of the gap between the pole pieces is filled by the steel. The total air gap g_a in this case is given by $g_a = g_c - t_T$. On the other hand, when a tab is not positioned between the magnet pole pieces, the total air gap is g_c. Since B varies inversely with the size of the air gap for the configuration of Fig. 5.8, it is much larger whenever any of the tabs is present than when none are present. Thus, the magnitude of the magnetic flux that "flows" through the magnetic circuit depends on the position of the tab, which, in turn, depends on the crankshaft angular position.

The magnetic flux is least when none of the tabs is near the magnet pole pieces. As a tab begins to pass through the gap, the magnetic flux increases. It reaches a maximum when the tab is located symmetrically between the pole pieces and then decreases as the tab passes out of the pole piece region. In any control system employing a sensor such as that of Fig. 5.7, the position of maximum magnetic flux has a fixed relationship to TDC for one of the cylinders.

An approximate model for the sensor configuration of Fig. 5.7 is developed as follows using the model developed above for $B(g_a)$. The terminal voltage V_o (according to Faraday's law) is given by the time rate of change of the magnetic flux linking the N turns of the coil:

$$V_o = N\frac{d\Phi}{dt}$$

where

$$\Phi = \int_{A_c} B ds$$

$$= \frac{\mu_o M_r L_m A_c}{g_a}$$

where

$$A_c = h_c \mathrm{w}_c$$

where w_c is the width of the magnet normal to the page.

The integral is taken over the cross-sectional area of the coil A_c (i.e., orthogonal to the contour of constant flux density). However, since the flux density is essentially constant around this contour C, the integral can be taken in the gap.

When the tabs are far away from the magnetic piece, the flux density magnitude is approximately given by

$$B = \frac{\mu_o M_r L_m}{g_c}$$

and g_c is the pole piece gap.

In this case, the magnetic flux Φ is given to close approximation by

$$\Phi \approx \frac{\mu_o M_r L_m h_c \mathrm{w}_c}{g_c} \tag{5.38}$$

When the tab moves between the pole pieces, the flux increases roughly in proportion to the projected overlap of the tab and gap cross-sectional areas, reaching a maximum when the tab is symmetrically located between the pole faces. The value for Φ when the tab is located symmetrically is given approximately by

$$\Phi = \frac{\mu_o M_r L_m h_c \mathrm{w}_c}{(g_c - t_T)} \tag{5.39}$$

The sensor terminal voltage, which is proportional to the time derivative of this flux, reaches a maximum and then crosses zero at the point when the tab is centered between the pole pieces. It then decreases and is antisymmetric about the center point as depicted in Fig. 5.10. The zero crossing of this voltage pulse is a convenient point for crankshaft and camshaft position measurements.

In the theory of electromagnetism, the ratio Φ/M for a structure such as is depicted in Fig. 5.8 is known as "permeance" $\wp$ with its reciprocal known as "reluctance" and is denoted $\mathfrak{R}$, which is given by

$$\wp = \mathfrak{R}^{-1} = \frac{\mu_o L_m h_c \mathrm{w}_c}{g_a}$$

Since the air gap g_a varies with the position of the steel disk in the sensor depicted in Fig. 5.7, this sensor is often termed a "variable reluctance sensor." It is, in fact, an inductive variable reluctance sensor since its output voltage is generated only when the magnetic flux changes with time.

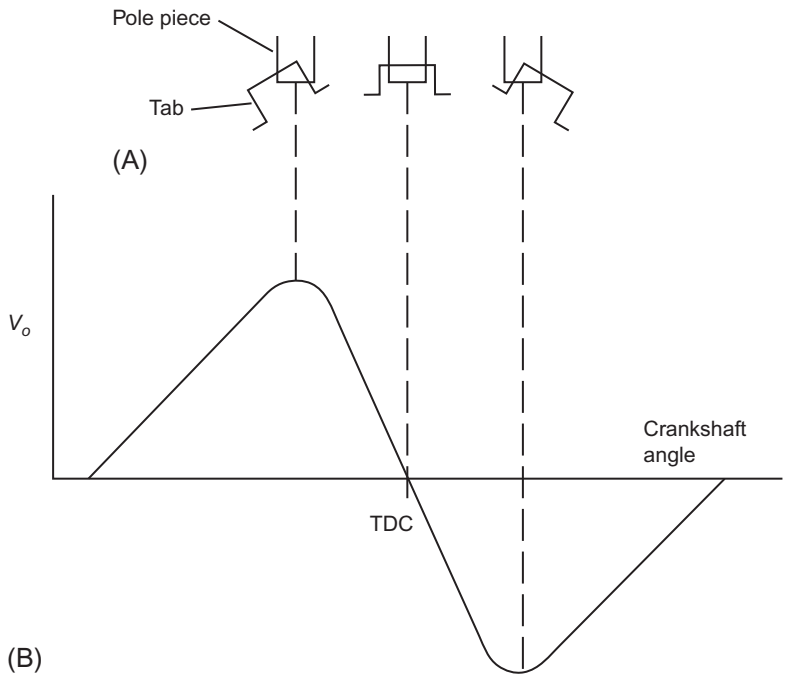


FIG. 5.10 Variable reluctance sensor voltage. (A) Position of tab and pole piece and (B) output voltage.

One of the disadvantages of the inductive type of variable reluctance sensor as depicted in Fig. 5.7 is that it only produces a nonzero voltage when the shaft is moving. Static engine timing such as was used in preemission-regulated vehicles is impossible with this type of variable-reluctance-type sensors. However, it will be shown later in this chapter that there are noncontacting magnetic position sensor configurations that are capable of static timing.

Another disadvantage of the inductive variable reluctance angular position sensor is the variation in the zero-crossing point with angular speed due to the impedance characteristics of the sensor. The precise timing requirements of modern digital engine control require that some compensation be made for the slight variation in timing reference of this sensor due to its source impedance. Fig. 5.11 gives an equivalent circuit for this sensor in which the open-circuit voltage source is represented by the voltage waveform of Fig. 5.10B. In this figure, L_s represents the inductance of the coil, which varies somewhat with steel disk angular position. The source resistance (R_s) is primarily the physical resistance of the coil wire but includes a component due to energy losses in the magnetic material.

Typically, these parameters are determined empirically for any given sensor configuration. The load impedance (resistance) of the signal processing circuitry is denoted R_ℓ. When the sensor of Fig. 5.7 is connected to signal processing circuitry, the exact zero-crossing point of its terminal voltage can potentially vary as a function of RPM. The variation in zero-crossing point is associated with the phase shift of the circuit of Fig. 5.11. At any sinusoidal frequency ω, the approximate phase shift $\varphi(\omega)$ between v_o and v_ℓ is given by

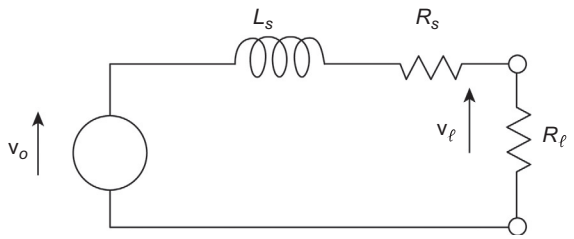


FIG. 5.11 Equivalent circuit for variable reluctance sensor.

$$\phi = -\tan^{-1}\left(\frac{\omega L_s}{R_s + R_\ell}\right)$$

where L_s is the inductance when the tab lies within the pole piece. The exact variation with RPM can be determined empirically such that compensation for this error can be done in the electronic engine control system. Compensation for such variations in the zero-crossing point is important for precise fuel delivery and ignition timing as explained in Chapter 6.

Fig. 5.7 illustrates a sensor having a ferromagnetic disk with four protruding tabs, which is a useful configuration for an eight-cylinder engine. However, engine position can readily be measured with the number of tabs being more than one-half the number of cylinders. For crankshaft position measurement, it is only necessary for the angular position of the tabs relative to crankshaft reference line position to be known. In fact, the precision and accuracy of crankshaft position can theoretically be improved with an increase in the number of tabs.

On the other hand, an increase in the number of tabs for a practical sensor increases the sensor excitation frequency (ω_s) for a given crankshaft angular speed. This increased excitation frequency increases the phase shift $\varphi(\omega_s)$ of the signal applied to a load resistance (R_ℓ) by an amount given by

$$\phi(\omega_s) = -\tan^{-1}\left(\frac{n\omega_s L_s}{R_s + R_\ell}\right) \quad n = 1,2\ldots$$

for each harmonic (n) component of the sensor output voltage. Typically, the crankshaft angular position is sensed at the zero crossings of the sensor output voltage as explained above. The phase shift associated with the sensor inductance introduces errors in this zero-crossing point relative to the actual tab center. However, this phase error is reduced by increasing load resistance (R_ℓ). Any compensation for this error via calculation in the digital engine control system is a unique process for any specific sensor/signal processing configuration.

Engine angular speed sensor

An engine angular speed sensor is needed to provide an input for the electronic controller for several functions. The crankshaft angular position sensor discussed previously can be used to measure engine speed. The reluctance sensor is used in this case as an example; however, any of the other position sensor techniques could be used as well. Refer to Fig. 5.7 and notice that the four tabs will pass through the sensing coil once for each crankshaft revolution.

For each crankshaft revolution, there are four voltage pulses of a waveform depicted qualitatively in Fig. 5.10B. For a running engine, the sensor output consists of a continuous stream of such voltage pulses. We denote the time of the nth zero crossing of voltage V_o (corresponding to TDC for a cylinder) as t_n. With this notation, the sensor output voltage is characterized by the following relationships:

$$\begin{aligned} V_o(t_n) &= 0 \\ \left.\frac{dV_o}{dt}\right|_{t=t_n} &< 0 \end{aligned} \tag{5.40}$$

The crankshaft angular speed ($\omega_e(t)$ in rad/sec) is given by

$$\omega_e(t) = \frac{2\pi}{M(t_{n+1} - t_n)} \tag{5.41}$$

where M = number of tabs (four in the example illustrated in Fig. 5.6). Thus, a measurement of the time between any pair of successive zero crossings of V_o can be used by a digital controller to calculate crankshaft angular speed.

One convenient way to measure this time interval is via the use of a binary counter and a high-frequency oscillator (clock). A high-frequency clock is a required component for the operation of a microprocessor/microcontroller as described in Chapter 3. A digital subsystem is readily configured to start counting the clock at time t_n and stop counting at t_{n+1}. The contents of the binary counter will contain the binary equivalent of B_c where

$$B_c = f_c(t_{n+1} - t_n) \tag{5.42}$$

Then, in one scheme, the time from t_{n+1} to t_{n+2} can be used for the digital control to access B_c for later computation of ω_e.

Control of this counting process can be implemented with a circuit known as a zero-crossing detector (ZCD). This circuit responds to the zero-crossing event at each t_n by producing an output pulse V_{ZCD} of the form

$$\begin{aligned} V_{\text{ZCD}} &= V_1 \quad t_n \le t < t_n + \tau_{\text{ZCD}} \\ &= V_2 \quad t_n + \tau_{\text{ZCD}} < t < t_{n+1} \end{aligned} \tag{5.43}$$

where the time interval $\tau_{\text{ZCD}} \ll (t_{n+1} - t_n)$ at all engine speeds and V_1 is a voltage that corresponds to binary 1 in a digital system and V_2 to binary 0.

The ZCD pulse can be used to control an electronic switch (gate) to alternately supply oscillator pulses to the binary counter or stop the counting. The ZCD, gate, and counter can be implemented by ad hoc dedicated circuitry (see Chapter 2) or within the controller/microprocessor.

Timing sensor for ignition and fuel delivery

As explained above, the combination of crankshaft and camshaft angular position measurements is sufficient to unambiguously determine the instantaneous position in the cycle for each cylinder. The measurement of engine position via crankshaft and camshaft position sensors (and its use in timing fuel delivery and ignition) is described in Chapter 6. Normally, it is sufficient to measure camshaft position at a single fixed point in each camshaft revolution. Such a measurement of camshaft position is readily achieved by a magnetic sensor similar to that described above for the crankshaft position measurement.

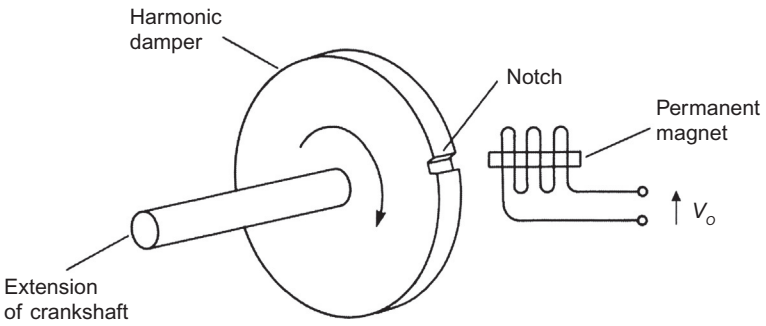


FIG. 5.12 Exemplary camshaft angular sensor configuration.

This sensor detects a reference point on the angular position of the camshaft that defines the beginning of a complete engine cycle. Once this reference point has been detected, crankshaft position measurements (as described above) provide sufficient information for timing fuel injection pulses and ignition.

In one scheme, a variable reluctance sensor is located near a ferromagnetic disk on the camshaft. This disk has a notch cut as shown in Fig. 5.12 (or it can have a protruding tab). The disk provides a low-reluctance path (yielding high magnetic flux) except when the notch aligns with the sensor axis. Whenever the notch aligns with the sensor axis, the reluctance of this magnetic path is increased because the permeability of air in the notch is very much lower than the permeability of the disk. This relatively high reluctance through the notch causes the magnetic flux to decrease and produces a change in sensor output voltage.

As the camshaft rotates, the notch passes under the sensor once for every two crankshaft revolutions. The magnetic flux abruptly decreases, then increases as the notch passes the sensor. This generates a pulse in the sensor output voltage V_o that can be used in electronic control systems for timing purposes. For the configuration depicted in Fig. 5.12, the sensor output voltage resembles that of Fig. 5.9 with a polarity reversal; that is, the output voltage satisfies the conditions:

$$V_o < 0 \quad t < T_{\text{notch}}$$

$$V_o > 0 \quad t > T_{\text{notch}}$$

where T_{notch} is the time at which the notch is symmetrically located along the magnet axis. The precise camshaft angular location is determined by the zero crossing of the sensor output voltage.

HALL-EFFECT POSITION SENSOR

As mentioned previously, one of the main disadvantages of the magnetic reluctance sensor is its lack of output when the engine is not running. A crankshaft position sensor that avoids this problem is the Hall-effect position sensor. This sensor can be used to measure either camshaft position or crankshaft position.

A Hall-effect position sensor is shown in Fig. 5.13. This sensor is similar to the reluctance sensor in that it employs a steel disk having protruding tabs and a magnet for coupling the disk to the sensing

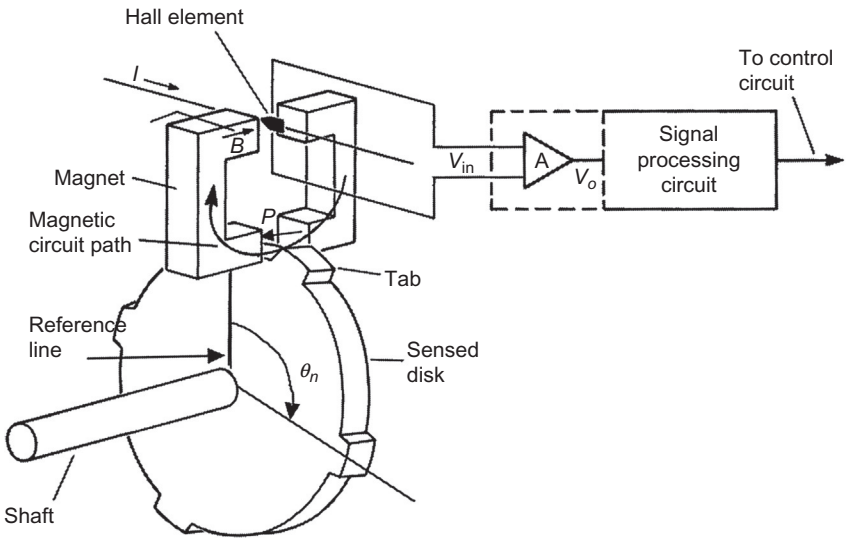


FIG. 5.13 Representative Hall-effect sensor configuration.

element. Another similarity is that the steel disk varies the reluctance of the magnetic path as the tabs pass between the magnet pole pieces. This sensor is useful for measuring the angular position θ of any shaft (e.g., crankshaft) relative to a reference line. Its operation depends upon a phenomenon known as the Hall effect. For convenience, this reference line is the intersection of the vertical plane of symmetry of the magnet with the flat surface of the disk. In Fig. 5.13, θ_n is the angle between the reference line and the center of the nth tab as shown.

The Hall-effect

The Hall element is a thin, flat slab of semiconductor through which a current I caused by an applied external potential V_s is flowing. Fig. 5.14 depicts a Hall element in the form of a semiconductor slab of length L_x, width L_y, and depth d that has an applied voltage V_s with current I.

In this configuration, there is a uniform magnetic field in the z direction (i.e., normal to the page). Although the electric field intensity $\bar{E}_s$ due to the applied voltage V_s is a function of position in the material, for a relatively long, thin slab of semiconductor (i.e., $L_x \gg L_y \gg d$), it is nearly uniform over much of the sample and given approximately by

$$\bar{E}_s \cong \frac{V_s}{L_x}\hat{x} = E_x\hat{x} \tag{5.44}$$

where $\hat{x}$ is a unit vector in the x-direction. The concentrations of electrons and holes in this material are denoted n and p, respectively. In the absence of the magnetic field, the current that would flow is given by

$$I = \int_o^{L_y}\int_o^d J_x dy dz \tag{5.45}$$

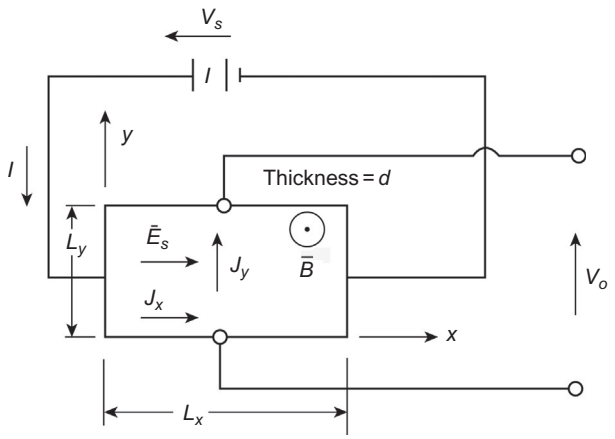


FIG. 5.14 Schematic illustration of Hall-effect sensor.

where J_x is the current density (i.e., the current/unit area across any y- and z-plane):

$$J_x = q[n v_{ex} + p v_{hx}] \tag{5.46}$$

where $v_{\text{ex}} = \mu_e E_x =$ electron drift velocity; $v_{\text{ex}} = \mu_h E_x =$ hole drift velocity, and where n and p are the electron and hole concentrations; μ_e and μ_n are the electron and hole mobilities, respectively.

However, when the magnetic flux density (B) is nonzero, there is a force acting on the electrons and holes known as the Lorentz force $\bar{F}_{Le}$ (electrons) and $\bar{F}_{Lh}$ (holes), which are proportional to the vector product of $\bar{B}$ and velocities $\bar{v}_e$ and $\bar{v}_h$:

$$\begin{aligned} \bar{F}_{Le} &= q\bar{v}_e \times \bar{B} \\ \bar{F}_{Lh} &= q\bar{v}_h \times \bar{B} \end{aligned} \tag{5.47}$$

where $\bar{B} = B_z \hat{z}$ and $\hat{z}$ is the unit vector in the z direction.

This Lorentz force acts on the electrons and holes causing them to drift in the y direction with velocity components v_{ey} (electrons) and v_{hy} (holes) creating a current flow in this direction represented by current density J_y:

$$J_y = q\left[p v_{hy} + n v_{ey}\right]$$

If (as is the usual case) the input impedance of the differential amplifier A in Fig. 5.13 is extremely large, $J_y \simeq 0$, which means that $p v_{hy} = -n v_{ey}$. The charge carriers will drift orthogonal to J_x and B_z creating an electric field E_y whose strength cancels the Lorentz force.

The strength of this y-directed electric field is given by

$$E_y = R_H J_x B_z$$

where R_H is the Hall-effect coefficient.

The terminal voltage of the sensor V_o is given by

$$\begin{aligned} V_o &= \int_0^{L_y} E_y dx \\ &\cong E_y L_y = R_H J_x B_z L_y \end{aligned}$$

Thus, the Hall-effect sensor generates an open-circuit voltage that is proportional to the x-directed current density J_x and to the magnetic flux density B_z.

The operation of the angular position sensor configuration depicted in Fig. 5.13 is based upon the variation of magnetic flux density normal to the Hall element and its relationship to the terminal voltage V_o derived above. Recall that the magnetic flux density is essentially constant along a closed path through the magnetic pole pieces and across the two gaps.

This flux density has a relatively low magnitude for all shaft positions for which the protruding tabs are away from the lower gap shown in Fig. 5.13. As a tab approaches this gap, it begins to fill the gap with a ferromagnetic material having a much higher magnetic permeability than air. The magnitude of the flux density increases in proportion to the projected overlap area of the tab on the magnet pole face (i.e., the face orthogonal to the magnetic path). This magnetic flux density reaches a maximum when any of the tabs is symmetrically located within the magnet's lower gap. If the angular position of the nth tab is denoted θ_n (as shown in Fig. 5.13), then the terminal voltage V_o of the sensor has a waveform as depicted in Fig. 5.15; that is, the terminal voltage reaches a maximum whenever $\theta_n = 0$ $(n = 1,2,\ldots,N)$ where $N =$ number of tabs. Thus, this sensor produces a voltage pulse of the general waveform of Fig. 5.15 each time a tab passes through the gap. As in the case of the active variable reluctance sensor discussed above, if this sensor is used for crankshaft position measurement, it must be combined with a camshaft angular position sensor (possibly also a Hall-effect sensor) for unambiguous timing within each engine cycle.

Shielded-field sensor

Fig. 5.16A shows another concept that uses the Hall-effect element in a way different from that just discussed. In this method, the Hall element is normally exposed to a magnetic field and produces an output voltage. When one of the tabs passes between the magnet and the sensor element, the low-reluctance values of the tab and disk provide a path for the magnetic flux that bypasses the Hall-effect sensor element, and the sensor output drops to near zero. Note in Fig. 5.16B that the waveform is just the opposite of the one in Fig. 5.15 in the sense of high versus low voltages.

OPTICAL CRANKSHAFT POSITION SENSOR

In a sufficiently clean environment, a shaft position can also be sensed using optical techniques. Fig. 5.17 illustrates such a system. Again, as with the magnetic system, a disk is directly coupled to the crankshaft. This time, the disk has holes in it that correspond to the number of tabs on the disks

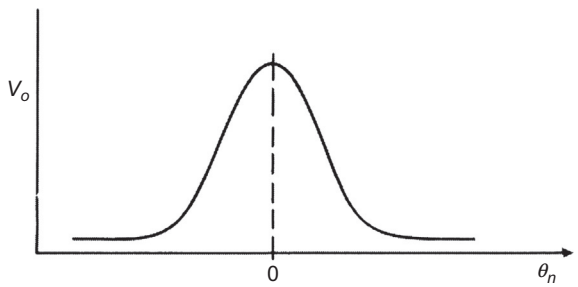


FIG. 5.15 Hall sensor output voltage waveform.

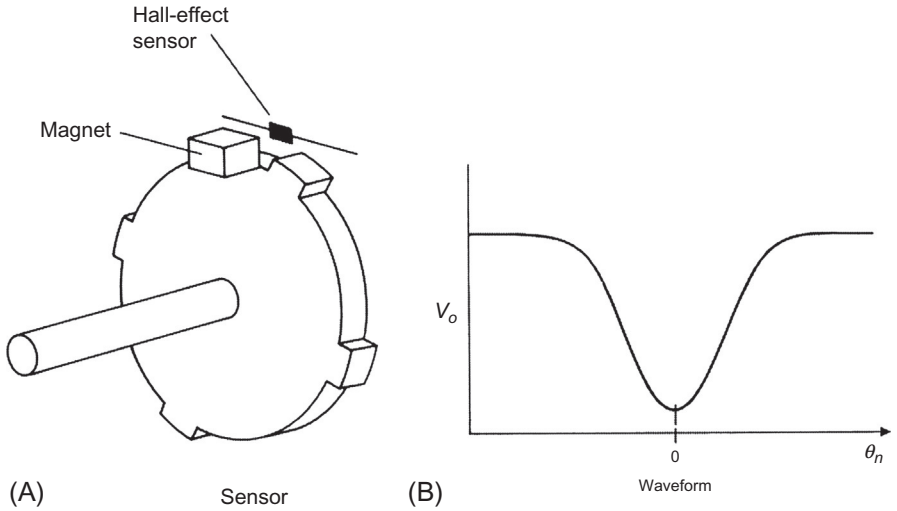


FIG. 5.16 Shielded-field Hall-effect sensor. (A) Sensor configuration; (B) Sensor output voltage.

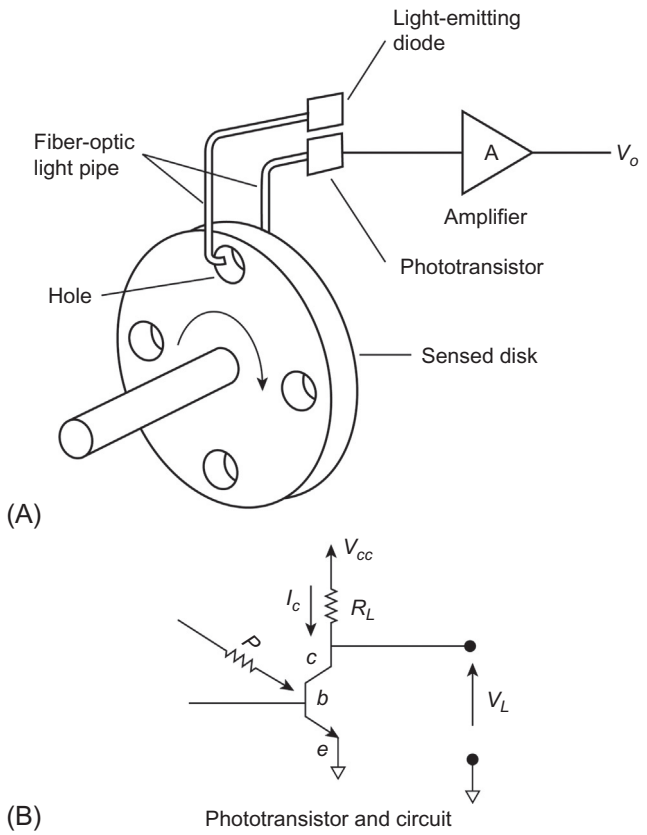


FIG. 5.17 Optical angular position sensor. (A) Sensor configuration; (B) Sensor circuit.

of the magnetic systems. Mounted on each side of the disk are fiber-optic light pipes. The hole in the disk allows transmission of light through the light pipes from the light-emitting diode (LED) source to the phototransistor used as a light sensor. Light would not be transmitted from source to sensor when there is no hole because the solid disk blocks the light. On the other hand, whenever a disk hole is aligned with one of the fiber-optic light pipes, light from the LED passes through the disk to the phototransistor.

The light-emitting diode used as a light source for this sensor has an increasing number of other applications in automotive systems including lighting (e.g., brake lights, turn signals, and instrumentation displays). The theory of operation of the LED is explained in Chapter 9. LEDs are made from a variety of semiconductor materials and are available in wavelength regions from infrared through ultraviolet depending upon the material, fabrication, and excitation voltage. There is even now a white-light LED.

The other important component of the optical position sensor of Fig. 5.17A is the phototransistor. A bipolar phototransistor has essentially the configuration of a conventional transistor having collector, base, and emitter regions. However, instead of injecting minority carriers into the base region via an electrical source (i.e., via base current i_b), the received light performs this function. The phototransistor is constructed such that light from a source is focused onto the junction region. The energy bandgap of the base region ΔE_g (i.e., the gap in allowable electron energy from the top of the valence band to the bottom of the conduction band; see Chapter 2) determines the wavelength of light to which the phototransistor responds.

Fig. 5.17B depicts an NPN phototransistor and its grounded emitter circuit configuration. The collector-base junction is reverse biased. Incoming light of illumination level P is focused by a lens arrangement onto the base (b) region of the phototransistor. When photons of the incoming light are absorbed in the base region, they create charge carriers that are equivalent to the base current of a conventional bipolar transistor. As explained in Chapter 2, increases in base region carriers cause the collector-emitter current to increase. Consequently, the collector current I_c varies linearly with P and is given by

$$I_c = I_o + \beta\gamma P \tag{5.48}$$

where β = grounded emitter current gain and γ = conversion constant from light intensity to equivalent base current.

The load voltage V_L is given by

$$\begin{aligned} V_L &= V_{cc} - I_c R_L \\ &= V_{cc} - R_L(I_o + \beta\gamma P) \end{aligned} \tag{5.49}$$

Each time a hole in the disk passes the fiber-optic light path depicted in Fig. 5.17A, the load voltage will be a high-to-low voltage pulse. The amplifier can be configured with a negative voltage gain such that its output will be a positive voltage pulse at the time any hole passes the optical path. These voltage pulses can be used to obtain the angular position of a rotating shaft (e.g., crankshaft) in a way similar to the magnetic position sensors explained above.

One of the problems with optical sensors is that they must be protected from dirt and oil; otherwise, the optical path has unacceptable transmissivity. On the other hand, they have the advantages that they can sense position without the engine running and that the pulse amplitude is essentially constant with variation in speed.

THROTTLE ANGLE SENSOR

Still another variable that must be measured for electronic engine control is the throttle plate angular position. In most automobiles, the throttle plate is linked mechanically to the accelerator pedal and moves with it. When the driver depresses the accelerator pedal, this linkage causes the throttle plate angle to increase, allowing more air to enter the engine and thereby increasing engine power.

Measurement of the instantaneous throttle angle is important for control purposes, as will be explained in Chapter 6. Most throttle angle sensors are essentially potentiometers. A *potentiometer* consists of a resistor with a movable contact, as illustrated in Fig. 5.18.

The basis for the throttle angle position sensor is the influence of geometric size and shape on the resistance of a conductive material. To illustrate this relationship, consider the resistance of a long section of a conductor of length L with a uniform cross-sectional area A with a voltage V_S applied at the ends along the long axis. As long as the lateral dimensions are small compared with length (i.e., $\sqrt{A} \ll L$), the current density is essentially uniform across the cross-sectional area. The current density magnitude of a current flowing through this area J is related to the electric field intensity magnitude E along the conductor long axis by Eq. (5.50)

$$J = \sigma E \tag{5.50}$$

where σ is the conductivity of the material. The total current through the conductor I for uniform J is given by

$$\begin{aligned} I &= \int_A J ds \\ &\cong JA \end{aligned} \tag{5.51}$$

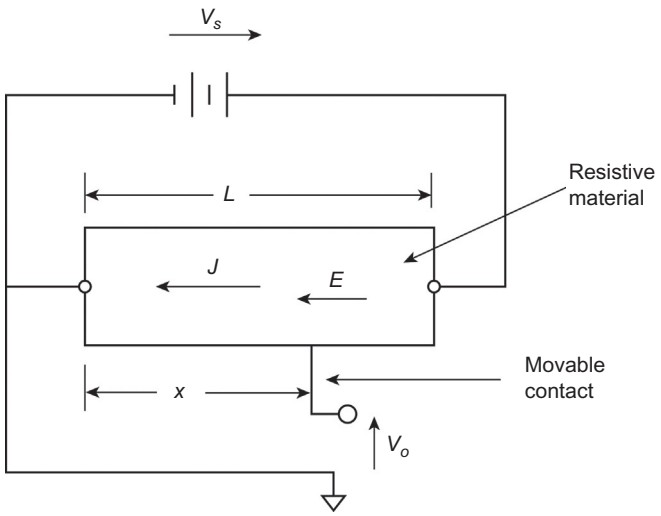


FIG. 5.18 Potentiometer schematic circuit.

where the integral is taken over the cross-sectional area of the conductive material. Furthermore, the terminal voltage at the conductor ends is given by

$$\begin{aligned} V_s &= \int_o^L E dx \\ &\cong EL \end{aligned} \tag{5.52}$$

where the x-coordinate is along the long axis. The voltage relative to ground at contact point x, which is denoted V_o in Fig. 5.18, varies linearly with position x:

$$\begin{aligned} V_o(x) &= \int_o^x E dx \qquad 0 \le x \le L \\ &= \frac{V_S x}{L} \end{aligned} \tag{5.53}$$

The resistance R of this conductor is defined as

$$\begin{aligned} R &= \frac{V}{I} \\ &= \frac{EL}{\sigma EA} \end{aligned} \tag{5.54}$$

$$R = \frac{L\rho}{A} \tag{5.55}$$

where $\rho = 1/\sigma =$ material resistivity (ohm m).

Consider now a resistive material formed in a segment of a circle of radius r as depicted in Fig. 5.19. Let the radial dimension and the thickness of the material be uniform and small compared with the circumferential distance along the arc ($r\alpha$). A movable metallic contact that pivots about the center of the circular arc makes contact with the resistive material at an angle α (measured from a line through the center and the grounded end of the resistive material). The opposite end of the material (at an angle $\alpha_{\max}$) is connected to a constant voltage V_s. A structure such as that depicted in Fig. 5.19 is known as a rotary potentiometer (or just as a potentiometer). Let the total resistance from the end of the material that is connected to V_s be denoted R_p and the resistance from the movable contact to ground at any angle α be denoted $R(\alpha)$. With the assumptions of uniform geometry given above, this resistance varies linearly with arc length $r\alpha$. Thus, the resistance $R(\alpha)$ can be shown to be given by

$$R(\alpha) = \frac{R_p \alpha}{\alpha_{\max}} \tag{5.56}$$

The current I flowing into this potentiometer is given by

$$I = \frac{V_s}{R_p} \tag{5.57}$$

The open-circuit voltage at the movable contact $V(\alpha)$ is given by

$$\begin{aligned} V(\alpha) &= IR(\alpha) \\ &= \frac{V_s}{R_p} R(\alpha) \\ &= V_s \frac{\alpha}{\alpha_{\max}} \end{aligned} \tag{5.58}$$

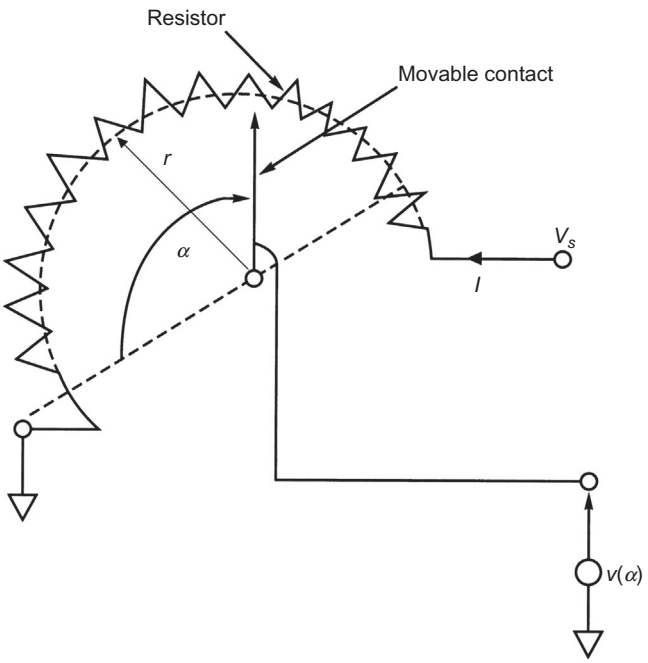


FIG. 5.19 Throttle angle sensor, a potentiometer.

A potentiometer is made by connecting the movable contact to a shaft at the pivot point whose axis is orthogonal to the plane of the conductor. If this shaft is mechanically connected to another rotary shaft (e.g., the throttle plate pivot shaft), the configuration of Fig. 5.19 is a sensor for measuring the angular position (α) of that other shaft. In the case of the throttle plate shaft, this potentiometer constitutes a throttle angle sensor in which the voltage $V(\alpha)$ provides a measurement of the throttle angle and thereby yields a measurement of the driver command for engine power. For digital engine control, the voltage $V(\alpha)$ must be converted to digital format using an analog-to-digital converter. As explained elsewhere in this book, there are other vehicular applications in which angular position (to a maximum that is $<2\pi$) is measured via a potentiometer.

TEMPERATURE SENSORS

Temperature (T) is an important parameter throughout the automotive system. In the operation of an electronic fuel control system, it is vital to know the temperature of the coolant, the temperature of the inlet air, and the temperature of the exhaust gas oxygen sensor (a sensor to be discussed in the next section). Several sensor configurations are available for measuring these temperatures, but we can illustrate the basic operation of most of the temperature sensors by explaining the operation of a typical coolant sensor. The temperature sensor for any given application is designed to meet the expected temperature range. For example, a coolant, temperature sensor experiences far lower temperatures than a sensor exposed to exhaust gases.

TYPICAL COOLANT SENSOR

A typical coolant sensor, shown in Fig. 5.20, consists of a thermistor mounted in a housing that is designed to be inserted in the coolant stream. This housing is typically threaded such that it seals the assembly against coolant leakage.

A thermistor is a two-terminal semiconductor whose resistance varies inversely with its temperature. The theory of operation is based upon the influence of temperature on the charge carrier concentrations that, in turn, depend upon the difference in energy between the valence and conduction band and are an exponential function of temperature. The resistance of a thermistor is a nonlinear function of temperature that can be modeled over a given temperature range by a polynomial function of T.

However, a relatively commonly used model that is valid over the range of coolant temperatures represents the thermistor resistance R_T as a logarithmic function of T is given by

$$\ln(R_T) = \frac{A}{T} - B \tag{5.59}$$

where, for an exemplary sensor, the coefficients are approximately $A \cong 5000$ and $B \cong 3.96$, and T is absolute temperature (K).

The sensor is typically connected in an electrical circuit like that shown in Fig. 5.21, in which the coolant temperature sensor resistance is denoted R_T. This resistance is connected to a reference voltage through a fixed resistance R. The sensor output voltage, V_T, is given by the following equation:

$$V_T = V\frac{R_T}{R + R_T} \tag{5.60}$$

Combining Eqs. (5.59), (5.60) yields the following equation for temperature T:

$$T = A/\{B + \ln[V_T R/(V - V_T)]\}$$

The terminal voltage V_T is input to the digital engine control system (e.g., via an A/D converter) where R_T is computed from V_T. Then, temperature is obtained using the model for $R_T(T)$ given above or another model (e.g., polynomial).

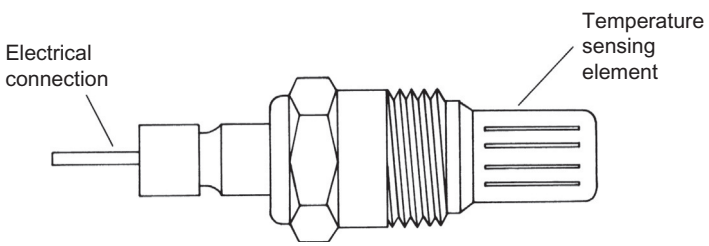


FIG. 5.20 Coolant temperature sensor.

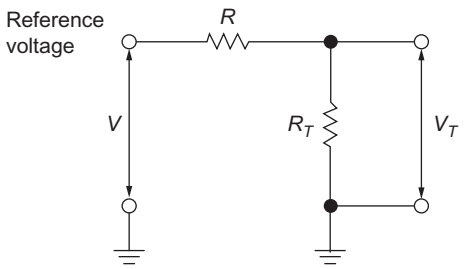


FIG. 5.21 Temperature sensor circuit.

SENSORS FOR FEEDBACK CONTROL

The sensors that we have discussed until now have been part of the open-loop (i.e., feed forward) control. Next, we consider sensors that are appropriate for feedback engine control. Recall from Chapter 4 that feedback control for fuel delivery is based on maintaining the air/fuel ratio at stoichiometry (i.e., 14.7:1 for gasoline). The primary sensor for fuel control is the exhaust gas oxygen sensor.

EXHAUST GAS OXYGEN SENSOR

Recollect from Chapter 4 that the amount of oxygen in the exhaust gas is used as an indirect measurement of the intake air/fuel ratio. As a result, one of the most significant automotive engine sensors in use today is the exhaust gas oxygen (EGO) sensor. This sensor is often called a *lambda sensor* from the Greek letter lambda (λ), which is commonly used to denote the equivalence ratio (as defined in Chapter 4):

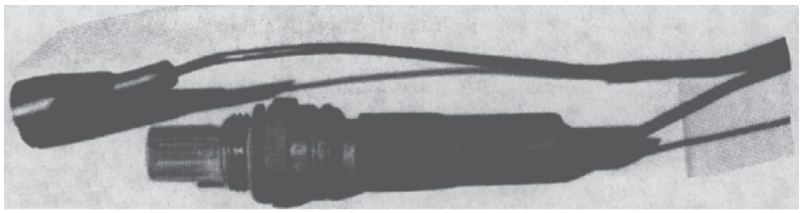
$$\lambda = \frac{(\text{air/fuel})}{(\text{air/fuel @ stoichiometry})} \tag{5.61}$$

Whenever the air/fuel ratio is at stoichiometry, the value for λ is 1. When the air-fuel mixture is lean, the condition is represented by $\lambda > 1$. Conversely, when the air-fuel mixture is rich, the condition is represented by ($\lambda < 1$).

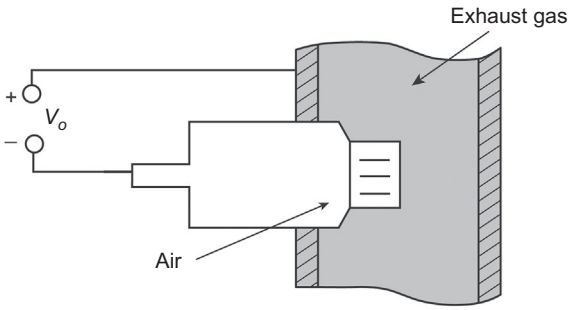
The two types of EGO sensors that have been used are based on the use of active oxides of two types of materials. One uses zirconium dioxide (ZrO_2), and the other uses titanium dioxide (TiO_2). The former is traditionally the most commonly used type. Fig. 5.22A is a photograph of a traditional ZrO_2 EGO sensor. Fig. 5.22B schematically depicts the mounting of the sensor on the exhaust system. Fig. 5.22C schematically shows the structure of the individual components and the way in which the exhaust gas acts on the EGO sensor.

In essence, the EGO sensor consists of a thimble-shaped section of ZrO_2 with thin platinum electrodes on the inside and outside of the ZrO_2. The inside electrode is exposed to air, and the outside electrode is exposed to exhaust gas through a porous protective overcoat.

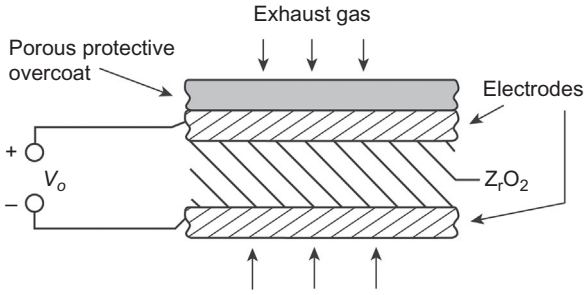
A simplified explanation of EGO sensor operation is based on the distribution of oxygen ions. Oxygen ions have two excess electrons such that the ions are negatively charged. The ZrO_2 has a tendency to attract the oxygen ions, which accumulate on the ZrO_2 surface just inside the platinum electrodes.



(A)



(B) Sensor mounted in exhaust manifold



(C) Inside the sensor tip

FIG. 5.22 Illustration of EGO sensor. (A) Picture of exemplary EGO sensor; (B) Illustrative installation; and (C) Exposure to exhaust gas.

The platinum plate on the air reference side of the ZrO_2 is exposed to a much higher concentration of oxygen ions than the exhaust gas side. The air reference side becomes electrically more negative than the exhaust gas side; therefore, an electric field exists across the ZrO_2 material and a voltage, V_o, results. The polarity of this voltage is positive on the exhaust gas side and negative on the air reference side of the ZrO_2. The magnitude of this voltage depends on the concentration of oxygen in the exhaust gas and on the sensor temperature.

The quantity of oxygen in the exhaust gas is represented by the oxygen partial pressure. Basically, this partial pressure is that proportion of the total exhaust gas pressure slightly above (but nearly at atmospheric pressure) that is due to the concentration of oxygen in the composite exhaust gas. The

exhaust gas oxygen partial pressure for a rich mixture varies over the range of 10^{-16}–10^{-32} of atmospheric pressure. The oxygen partial pressure for a lean mixture is roughly 10^{-2} atm. Consequently, for a rich mixture, there is a relatively low oxygen concentration in the exhaust and a higher EGO sensor output voltage. For a fully warmed EGO sensor, the output voltage is about 1 V for a rich mixture and about 0.1 V for a lean mixture.

Desirable EGO characteristics

The EGO sensor characteristics that are desirable for the type of limit-cycle fuel control system that was discussed in Chapter 4 are as follows:

1. Abrupt change in voltage at stoichiometry
2. Rapid switching of output voltage in response to exhaust gas oxygen changes
3. Large difference in sensor output voltage between rich and lean mixture conditions
4. Stable voltages with respect to exhaust temperature

Switching characteristics

The switching time for the EGO sensor also must be considered in control applications. An ideal characteristic for a limit-cycle controller is shown in Fig. 5.23. The arrow pointing down indicates the change in V_o as the air/fuel ratio was varied from rich to lean. The up arrow indicates the change in V_o as the air/fuel ratio was varied from lean to rich. Note that this EGO sensor has switching characteristics with hysteresis. A model for the ideal EGO sensor was used in Chapter 4 for explaining closed-loop fuel control in which the hysteresis was taken to be negligible.

Fig. 5.24 depicts the actual sensor voltage/equivalence ratio characteristics for a common commercially available (fully warmed) EGO sensor. Comparing this sensor's characteristics to that of the ideal sensor characteristics shows that the voltage drop from a rich mixture to lean has a finite slope and occurs on the lean side of stoichiometry. Furthermore, the EGO sensor terminal voltage is a continuous function of λ. This voltage is also a continuous function of λ for a lean to rich transfer but has a very steep slope at $\lambda = 1$.

Temperature affects switching times and output voltage. Switching times at two temperatures are shown in Fig. 5.25. Note that the time per division is twice as much for the display at 350°C as at 800°C.

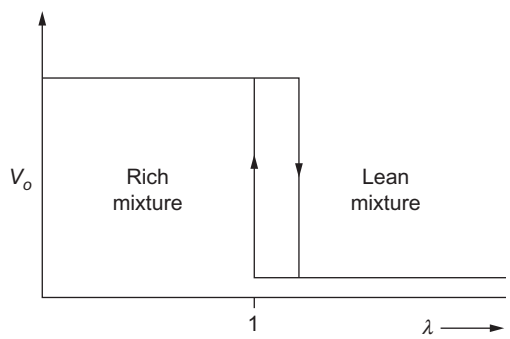


FIG. 5.23 Switching characteristics of ideal EGO sensor.

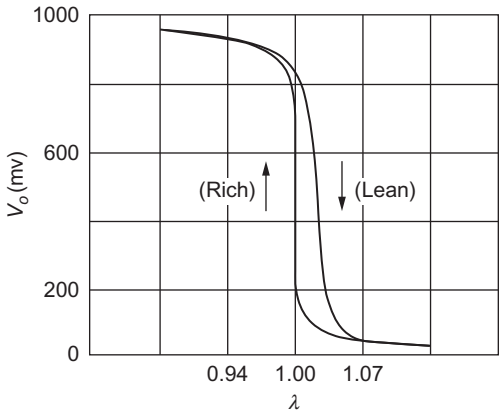


FIG. 5.24 Commercial EGO sensor voltage versus λ.

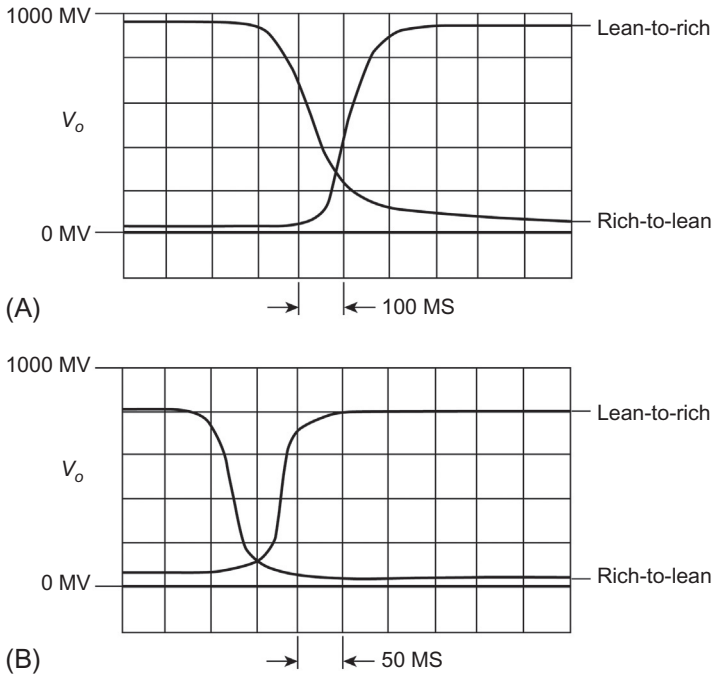


FIG. 5.25 EGO sensor switching transients. (A) At 350°C and (B) at 800°C.

This means that the switching times are roughly 0.1 s at 350°C, whereas at 800°C, they are about 0.05 s. This is a 2:1 change in switching times due to changing temperature.

The temperature dependence of the EGO sensor output voltage is very important. The graph in Fig. 5.26 shows the temperature dependence of an EGO sensor output voltage for lean and rich mixtures and for two different load resistances 5 and 0.83 MΩ. The EGO sensor output voltage for a rich mixture is in the range of about 0.80–1.0 V for an exhaust temperature range of 350–800°C. For a lean mixture, this voltage is roughly in the range of 0.05–0.07 V for the same temperature range.

Under certain conditions, the fuel control using an EGO sensor will be operated in open-loop mode, and for other conditions, it will be operated in closed-loop mode (as will be explained in Chapter 6). The EGO sensor should not be used for control at temperatures below about 300°C because the difference between rich and lean voltages decreases rapidly with temperature in this region.

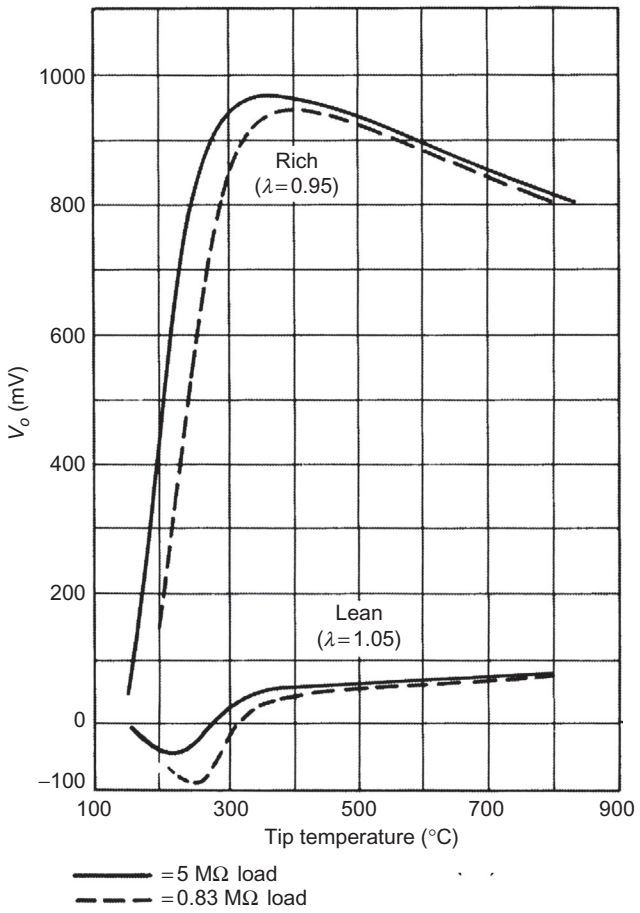


FIG. 5.26 EGO sensor temperature characteristics.

This important property of the sensor is partly responsible for the requirement to operate the fuel control system in the open-loop mode at low exhaust temperatures. Closed-loop operation with the EGO output voltage used as the error input cannot begin until the EGO sensor temperature exceeds about 300°C. Open-loop mode operation is undesirable since exhaust emission regulation is not as reliable as closed-loop operation particularly as a vehicle ages and engine parameters can change. Although it is important to hasten the change from open- to closed-loop operation (particularly during a cold engine start), the EGO sensor voltage must be sufficient for closed-loop operation.

OXYGEN SENSOR IMPROVEMENTS

Improvements have also been made in the exhaust gas oxygen sensor, which remains today the primary sensor for closed-loop operation in cars equipped with the three-way catalyst. As we have seen, the signal from the oxygen sensor is not useful for closed-loop control until the sensor has reached a temperature of about 300°C. Typically, the temperature of the sensor is too low during the starting and engine warm-up phase, and it can also be too low during relatively long periods of deceleration. It is desirable to return to closed-loop operation in as short a time as possible. Thus, the oxygen sensor must reach its minimum operating temperature in the shortest possible time.

An improved exhaust gas oxygen sensor has been developed that incorporates an electric heating element inside the sensor, as shown in Fig. 5.27. This EGO sensor is known as the heated exhaust gas oxygen, or HEGO, sensor. The heat current is automatically switched on and off depending on the engine operating condition. When available in a vehicle configuration, an exhaust gas temperature sensor can closely estimate the HEGO temperature. Heating can then be applied as necessary to reach closed-loop operation as soon as possible. The heating element is made from resistive material and derives heat from the power dissipated in the associated resistance. The HEGO sensor is packaged in such a way that this heat is largely maintained within the sensor housing, thereby leading to a relatively rapid temperature rise.

Normally, the heating element needs only be turned on for cold-start operations. Shortly after engine start, the exhaust gas has sufficient heat to maintain the EGO sensor at a suitable temperature.

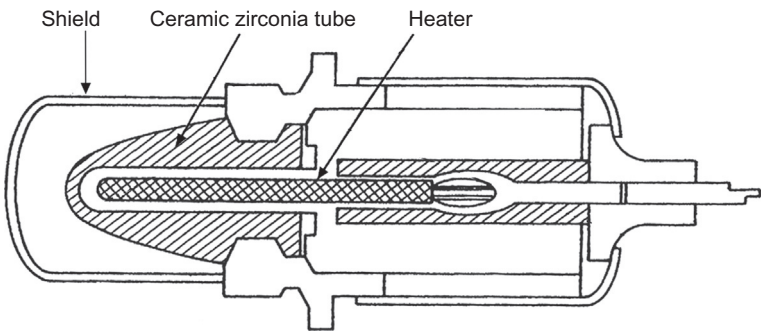


FIG. 5.27 Heated EGO sensor configuration.

KNOCK SENSORS

Another sensor having applications in closed-loop engine control is the so-called knock sensor. As explained in Chapter 6, this sensor is employed in closed-loop ignition timing to prevent undesirable knock. Although a more detailed explanation of knock is given in Chapter 6, for the purposes of this chapter, it can be described generally as a rapid rise in cylinder pressure during combustion. It does not occur normally, but only under special conditions. It occurs most commonly with high manifold pressure and excessive spark advance and at relatively high combustion temperatures. It is important to detect knock and avoid excessive knock; otherwise, there may be damage to the engine.

As will be explained in Chapter 6, one way of controlling knocking is to sense when knocking begins and then retard the ignition until the knocking stops. A key to the control loop for this method is a knock sensor.

Knock sensors fundamentally detect impulsive acoustic signals associated with the rapid pressure rise of cylinder pressure. The phenomenon is called knock because the acoustic signal associated with it is in the audio range and sounds like a "knock." It is characterized by a short, relatively intense, pulse followed by rapidly decaying oscillations in the few kilohertz range depending on engine configuration. The associated cylinder pressure waveform is depicted in Chapter 6 in Fig. 6.20.

The configuration of a representative knock sensor using magnetostrictive techniques is shown in Fig. 5.28A. *Magnetostriction* is a phenomenon whereby the magnetic properties of a material depend on stress (due to an applied force). When sensing knock, the magnetostrictive rods, which are in a magnetic field, change the flux field in the coil due to knock-induced forces. In Fig. 5.28A, the forces associated with knock cylinder pressure are transmitted through the mounting frame to the magnetostrictive rods. Magnetostriction is a property of ferromagnetic materials, which were introduced in the discussion of the sensor of Fig. 5.7. Recall that a ferromagnetic material is physically made up of individual domains in which the magnetic fields associated with the electron spins within a domain are all aligned in a given direction.

Wherever an external magnetic field is applied, the domains are reoriented such that their axes tend to be parallel with the applied field. The reorientation of the magnetic domains induces a strain within the material, which slightly changes its size and shape.

Conversely, these same materials when magnetized and when subject to stress/strain due to an applied external force change magnetic permeability μ. It is this latter effect (known as reverse magnetostriction) that is of interest in a knock sensor.

Although magnetostriction is strictly speaking an anisotropic phenomenon, for the purposes of the present discussion, a typical magnetostrictive material in a knock sensor is fabricated with relatively long thin rods. In this case, only the permeability change along the axis is of importance and can be treated as an isotropic scalar permeability μ_R (rather than tensor) model as given below.

In Fig. 5.28A, the small magnet creates a magnetic field having a magnetic flux density $\bar{B}$ in the form of closed loops passing through the magnet, the magnetostrictive rods, and the coil of N turns and return through the magnetically "soft" (i.e., relatively high magnetic permeability) magnetic shell (i.e., see Fig. 5.28B). These loops are basically lines of constant flux density magnitude. The strength of this flux density is determined by the magnet and the permeability of the magnetostrictive rods μ_R. The magnetic permeability of the shell μ_s is assumed to satisfy the inequality: $\mu_s \gg \mu_R$.

A simplified model for the amplitude of the flux density is given by

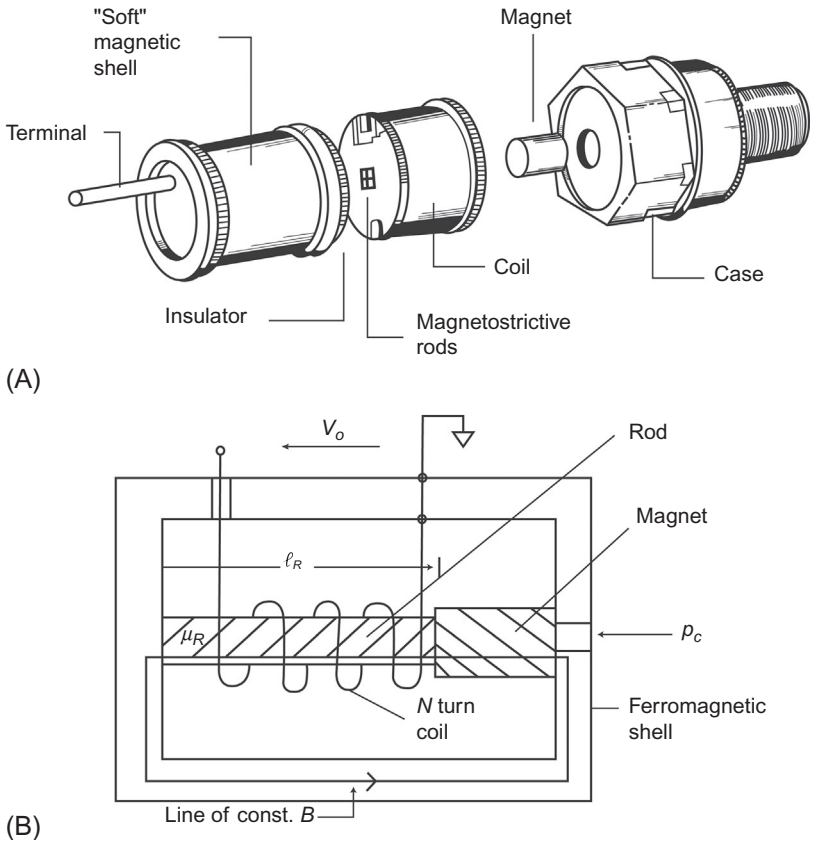


FIG. 5.28 Knock sensor configuration. (A) Knock sensor configuration; (B) Magnetic field illustration.

$$B \cong \frac{M\mu_R}{\ell_R} \tag{5.62}$$

where ℓ_R is the length of the magnetostrictive rods and M is the remanent magnetization that is a constant for the magnet. The total magnetic flux Φ through the rods is approximately given by

$$\begin{aligned} \Phi &= \int_{A_R} B ds \\ &\cong \mathrm{BA_R} \end{aligned} \tag{5.63}$$

where ds is a differential area in a plane orthogonal to the rod long axis and A_R is its entire cross-sectional area. In a typical sensor, B is nearly uniform over the rod area A_R. In this case, the total magnetic flux is given approximately by

$$\Phi = \frac{MA_R\mu_R}{\ell_R} \tag{5.64}$$

The sensor terminal voltage V_o is given by Eq. (5.65)

$$\begin{aligned} V_o &= N\frac{d\Phi}{dt} \\ &= \frac{NMA_R}{\ell_R}\frac{d\mu_R}{dt} \end{aligned} \tag{5.65}$$

where N = number of turns of the coil.

The time derivative of μ_R is due to magnetostriction in the rods. An approximate model for μ_R is given by

$$\mu_R = \mu_1 + \mu_2\sigma_R \tag{5.66}$$

where σ_R is the stress induced in the rod by knock forces, which are transmitted to the rods by the frame, and where μ_1 and μ_2 are constants for the magnetostrictive rod material.

During normal combustion, $d\sigma_R/dt$ is relatively small. However, during knock, this time derivative is relatively large and is proportional to the knock cylinder pressure fluctuations. Thus, the sensor terminal voltage contains a term that is proportional to knock intensity. This voltage is used to sense excessive knock (see Chapter 6). Other sensors use piezoelectric crystals or the piezoresistance of a doped silicon semiconductor. Whichever type of sensor is used, it forms a closed-loop system that retards the ignition to reduce the knock detected at the cylinders. Systems using knock sensors are explained in Chapter 6.

The problem of detecting knock is complicated by the presence of other vibrations and noises in the engine. Normally, signal processing in the form of filters "tuned" to the knock frequency of the specific engine configuration enhances the detection of knock (see Chapter 6).

ANGULAR RATE SENSOR

An important sensor having multiple applications in vehicle motion control, as explained in later chapters (e.g., Chapters 7 and 10), is a sensor that is capable of measuring the angular rate of the vehicle body relative to an inertial reference frame. Measurement of the angular roll rate and angular yaw rate are related to control of vehicular roll and yaw. For example, both have applications in safety-related control systems. In addition, the measurement of angular pitch rate has potential for vehicle suspension control.

The aerospace industry has had the need to measure angular rate of an aircraft (or satellite) for many decades. The aerospace angular rate sensors are implemented with instrumented gyroscopes that yield highly accurate measurements. However, owing to the relatively high cost for gyroscope-based angular rate sensors, they are not economically viable for automotive applications.

On the other hand, a relatively low-cost angular rate sensor that is based on the dynamic motion of a tuning-fork-type structure has found much application in vehicle electronic control system. The physical configuration of a representative vehicular angular rate sensor is depicted in Fig. 5.29. The structure depicted in Fig. 5.29 is fabricated with a thin layer of quartz (SiO_2). It consists of a pair of tuning-fork-shaped structures that are effectively mounted on a support. However, the entire structure shown in Fig. 5.29 is fabricated as a single piece. The slender z-directed pieces are called tines, since they actually physically resemble and oscillate (when driven electrically) the same way as the tines of a

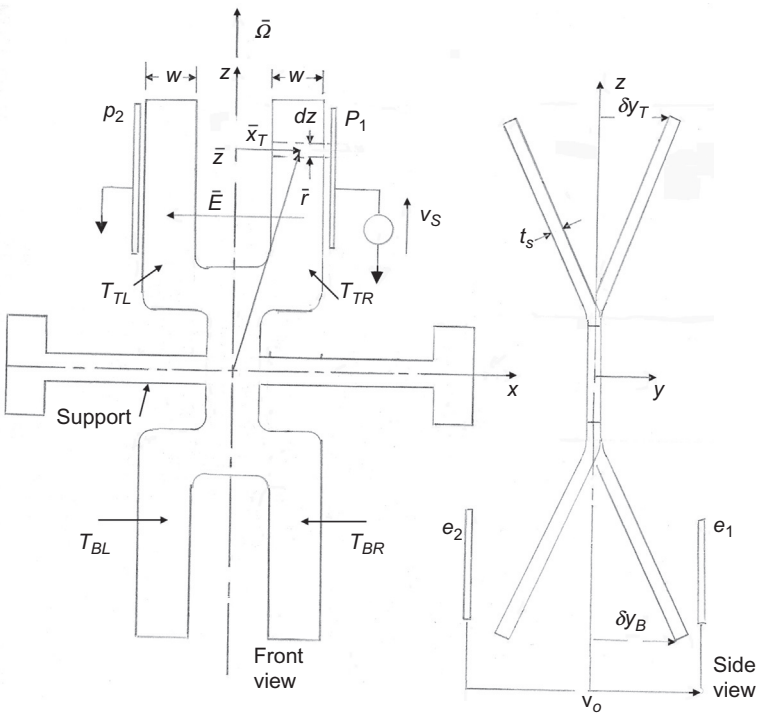


FIG. 5.29 Angular rate sensor configuration.

traditional musical tuning fork. The top tines are denoted T_{TL} and T_{TR} for the left and right tines, respectively. Similarly, the bottom tines are T_{BL} and T_{BR}, respectively.

The top two tines are electromechanically excited into oscillation by a sinusoidally varying electric field denoted $\bar{E}$, which is a vector parallel to the x-axis. In the example configuration, this electric field is produced by a voltage source ($v_S(t)$) that is applied to a pair of planar electrodes p_1 and p_2.

The motion of the tines in response to an electric field results from the piezoelectric property of the quartz material from which this structure is fabricated. An important property of piezoelectric materials is that the internal molecular structure is that the centers of positive and negative charges of a crystal cell are displaced a small amount resulting in a small electric dipole (i.e., + and − charges separated by a small distance d). The application of an electric field lengthens or shortens d (depending on polarity) in proportion to the strength of $\bar{E}$. This dimensional change results in internal stresses that cause the tines to oscillate in the x- and z-plane toward and away from each other. These tines (T_{TL} and T_{TR}) are called the drive tines.

The voltage $v_S(t)$ in the example sensor is given by

$$v_S(t) = V_S \sin \omega t$$

The driven tines that experience the piezoelectric tone oscillate at the same frequency. The dynamic analysis of the motion of the tines is presented in Appendix C. There, the dynamic motion of the driven

tines is derived for a differential length dz of either T_{TL} or T_{TR} at a distance $\bar{z}$ from the origin of the coordinate system at the center of the support derived from basic mechanics. The center of this differential length is at a distance x_T from the z-axis. The vector position of this point from the origin of the coordinate system is denoted $\bar{r}$ and is given by

$$\bar{r} = \bar{z} + x_T \hat{x} \tag{5.67}$$

where $\hat{x}$ = unit vector in the x-direction. The angular motion to be sensed by this angular rate sensor is rotation about the z-axis at a vector angular velocity $\bar{\Omega}$. In Appendix C, it is shown that the relative acceleration of this point $\left(\ddot{\bar{r}}\right)$ is given by

$$\ddot{\bar{r}} = \ddot{\bar{x}}_T + \bar{\Omega} \times (\bar{\Omega} \times \bar{x}_T) + \dot{\bar{\Omega}} \times \bar{x}_T + 2\bar{\Omega} \times \dot{\bar{x}}_T \tag{5.68}$$

The first three terms are the absolute acceleration of the point, and the fourth term is the Coriolis acceleration. The differential force $d\bar{F}$ acting on the differential segment of the tine is given by

$$d\bar{F} = dm\ddot{\bar{r}} \tag{5.69}$$

where

$$dm = \rho_Q w t_S dz$$

$$= \text{mass of the differential segment}$$

$$\rho_Q = \text{quartz mass density}$$

and where w and t_S are depicted in Fig. 5.29. This last force is in a direction orthogonal to the plane of the sensor element and is in opposite sense for T_{TL} and T_{TR} and is responsible for the out-of-plane deflection of the tines as depicted in the side view of Fig. 5.29.

The total force on each tine due to the Coriolis acceleration can be found by integrating the differential force over the length of each tine and results in a couple on the center segment connecting the upper and lower tines $\bar{M}$, which is proportional to the vector product $\bar{\Omega} \times \dot{\bar{x}}_{Tt}$ and is given by

$$\bar{M} = K\bar{\Omega} \times \dot{\bar{x}}_{Tt} \tag{5.70}$$

where $\dot{\bar{x}}_{Tt}$ = tip velocity of the tine and K = constant for the structure of the tines.

The structure of the sensor configuration is such that this moment is transmitted to the lower tines that cause them to oscillate with an amplitude proportional to the couple $\bar{M}$. The displacement of the lower tines is depicted in the side view of Fig. 5.29. The amplitude of the tine tips δy_T is proportional to the product of the amplitude Ω and $\dot{x}_T$ when

$$\Omega = \|\bar{\Omega}\|$$

$$\dot{x}_T = \|\dot{\bar{x}}_T\|$$

that is

$$\delta y_T = \gamma_T \Omega \dot{x}_T$$

where γ_T = constant. The tine x-velocity component $\dot{x}_T$ is proportional to the excitation voltage $v_s(t)$. It should be noted that while the δy_T in Fig. 5.29 is greatly exaggerated, it is only for illustrative purposes.

The motion of the lower tines (called the sensing tines) depicted in this figure can be measured via a pair capacitive electrodes one associated with each sensing tine. The internal dipolar electric fields induce charges of opposite polarity on the two electrodes (denoted e_1 and e_2 in Fig. 5.29), which results

in a differential voltage $v_o(t)$ that is proportional to the instantaneous lower tine deflection δy_B that, in turn, is proportional to δy_T. As a result, the output voltage is given by

$$v_o = \gamma_B \Omega v_S(t) \tag{5.71}$$

where γ_B = constant for the structure of the angular rate sensor.

The sensor output voltage is, in effect, an amplitude-modulated version of the excitation voltage $v_s(t)$ with the amplitude that is proportional to the angular velocity amplitude $\Omega(t)$. A measurement of $\Omega(t)$ can be achieved using the double tuning fork configuration of Fig. 5.29 by demodulating $v_o(t)$. A circuit for accomplishing this demodulation is depicted in Fig. 5.30.

In this figure, the excitation voltage and sensed voltage are sent to an electronic multiplier denoted EM in Fig. 5.30, which generates an output voltage $v_p(t)$ that is proportional to the instantaneous product of v_s and v_o:

$$\begin{aligned} v_p(t) &= K_p v_S v_o \\ &= K_p v_S \gamma_B [(\sin \omega t)(\Omega \sin \omega t)] \\ &= K_p v_S \gamma_B \Omega \sin^2 \omega t \\ &= K_p \frac{V_S}{2} \gamma_B \Omega (1 - \sin(2\omega t)) \end{aligned} \tag{5.72}$$

where K_p is a constant for the multiplier circuit.

The low-pass filter (LPF) of Fig. 5.30 has a sinusoidal frequency response $H_{\text{LPF}}(j\omega)$ (as explained in Appendix A) given by

$$H_{\text{LPF}}(j\omega) = \frac{H_o}{1 + j\dfrac{\omega}{\omega_{\text{LPF}}}}$$

where, H_o = passband amplitude, ω_{LPF} = filter bandwidth (3 at B), $\omega_{\text{LPV}} \ll \omega$.

The filter bandwidth is such that the LPF suppresses the 2ω frequency component of v_p and passes the low-frequency component. The LPF output voltage v_Ω is given by

$$v_\Omega = \frac{H_o K_p V_S \gamma_B}{2} \Omega \tag{5.73}$$

Thus, the double-tine tuning fork structure in combination with the circuit of Fig. 5.30 is a sensor for measuring the angular velocity Ω of rotational motion about the z-axis of the structure as depicted in Fig. 5.29. Contemporary vehicles have the potential need for multiple vehicle angular rate sensors.

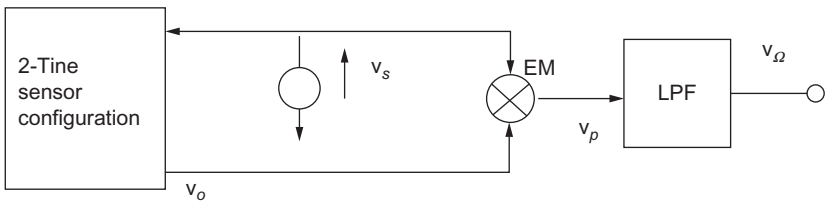


FIG. 5.30 Angular rate sensor demodulator.

LIDAR

One of the increasingly important sensor systems for use in vehicle applications is called LIDAR, which is an acronym for "light detection and ranging." It is an optical carrier frequency equivalent of RADAR and performs the same functions. It can be used to detect the presence of objects in the region surrounding a vehicle at distances that are within the range of a potential collision. Many of the applications are involved in vehicle safety systems as discussed in Chapter 10 and for autonomous vehicles as discussed in Chapter 12 and for maintaining safe intervehicle distances in advanced cruise control as discussed in Chapter 7. A properly designed LIDAR sensor system can supplement the vehicle environmental information obtained visually by the driver and can be the only source of such information for fully autonomous vehicles.

In contemporary vehicles, there are numerous configurations of LIDAR systems. The LIDAR system incorporates one or more (depending on its configuration) optical sources that scan the regions surrounding the vehicle. Also, depending on the LIDAR configuration, an optical source can be (and often is) a laser diode whose theory of operation is explained in Chapter 2. In addition to the optical source, a LIDAR sensor system requires optical detectors that have very fast response times for detecting short-duration optical pulses. In addition, to be able to gather information about the vehicle environment, there must be a means of scanning the light source over the region being covered. The actual beam width of light from a laser diode is small, which means that the LIDAR system can have good angle resolution both in azimuth and elevation angles when scanned over the region of its coverage.

The range resolution for a pulsed LIDAR is determined by the resolution in time of the received pulses relative to their time of transmission. The LIDAR operation of detecting an object depends on its reflectance of the incident light pulses. Let the time of transmission of the kth optical pulse be denoted $t_T(k)$, and the time that the reflected pulse is received by the optical detector be denoted $t_R(k)$. The time difference of δt_k is proportional to the range $r(k)$ to the object surface from the light source, which reflects the light that is given by

$$\begin{aligned}\delta t(k) &= t_T(k) - t_R(k) \quad k = 0,1,2,\ldots \\ &= \frac{2r(k)}{c}\end{aligned} \tag{5.74}$$

where

$$\begin{aligned}c &= \text{speed of light} \\ &= 3 \times 10^8 \text{m/s in air}\end{aligned}$$

For an ideal pulsed LIDAR, the transmitted optical intensity $I_T(k)$ is given by

$$\begin{aligned}I_T(k) &= I_o \quad t_T \le t \le t_T(k) + \tau \\ &= 0 \quad t_T(k) + \tau \le t \le t_T(k+1)\end{aligned} \tag{5.75}$$

where, I_o = amplitude of the transmitted intensity; τ = pulse duration.

The received optical intensity $I_R(k)$ is given by

$$\begin{aligned}I_R(k) &= \rho I_o e^{-2\alpha r(k)} + I_n \quad T_T(k) + \delta t(k) \le t < T_T(k) + \delta t(k) + \tau \\ &= I_n \quad T_T(k) + \delta t(k) + \tau \le t < T_T(k+1) + \delta t(k)\end{aligned} \tag{5.76}$$

where I_n is the intensity of any other optical source that happens to be propagating along the instantaneous LIDAR path toward the optical detector during the time of the kth pulse. In the above equation, the parameter ρ is the object reflectivity at laser frequency, and α is the attenuation constant of the optical beam along the path to and from the object and the laser. This parameter is strongly influenced by atmospheric conditions (e.g., clear air versus rain, sleet, snow, and dust). Due to the exponential character of the attenuation, the received intensity amplitude varies over an extremely large dynamic range over the intended domain of $r(k)$ for any given LIDAR system.

For vehicular applications, the maximum intended range for LIDAR under maximal attenuation and minimal reflectivity calls for a very sensitive optical detector in the receiver portion of the system. There are certain photodiodes that have high sensitivity including so-called avalanche photodiodes. In such diodes, the photoionization releases charge carriers that, via collisions with neutral atoms, create increasing charge carriers. For such photodiodes, the current that flows due to $I_R(k)$ is effectively amplified by the avalanche process, thereby increasing the receiver sensitivity. The current pulse from each received LIDAR pulse passes through a high-resistance component creating a proportional voltage that is amplified by a high-gain amplifier (see Chapter 2). It is important to note that the information about the object range is contained in the leading edge of the received pulse and is independent of the amplitude provided that the pulse amplitude is sufficiently large to exceed the amplitude of interference (I_n) or noise.

One of the most important requirements for LIDAR when applied to a vehicle safety system (e.g., automatic braking) or to upper-level autonomous vehicles is the surveillance of the vehicle environment with sufficient resolution to take required action. This resolution of LIDAR is due, in part, to scanning of the environment with the laser source or sources. There are numerous techniques for achieving the required scanning. One such technique incorporates a rotating mirror. For a full 360 degrees view, the scanning mirror and laser assembly need to be mounted on the highest point on the vehicle (e.g., the roof of the vehicle). For scanning in both azimuth and elevation, one method involves a mirror that oscillates about an axis that is on a lateral vehicle axis. The maximum deviation of the mirror as it oscillates corresponds to the deviation in elevation for the LIDAR sensor system. The light that is reflected by the oscillating mirror is directed to a mirror that rotates about a vertical axis. The rotating mirror is inclined at an angle such that the light that it reflects covers the elevation region required for the LIDAR sensor.

To be useful as a sensor for surveillance of the vehicular environment, the LIDAR system must incorporate an optical receiver that detects the reflected pulses. A block diagram of an illustrative LIDAR sensor system is depicted in Fig. 5.31.

In Fig. 5.31, the oscillating mirror is denoted OM, and the rotating mirror is denoted RM. Also depicted is a fixed, partially reflecting mirror denoted PR and a photodetector that is denoted PD. The PR mirror (e.g., 50% reflecting and 50% transmitting) passes light from the source I_S to the OM and RM where, in this example, $I_T = \frac{1}{2} I_S$. The received light is reflected by RM, OM, and PR to the PD receiver.

The PD generates a pulsed signal (denoted V_R in Fig. 5.31) that is sent to a digital signal processor (DSP). Also sent to the DSP are the measurements of the azimuth angle of RM about the z-axis θ and the elevation angle ϕ from a pair of sensors. The signal processing unit generates the three-dimensional position at times t_k of the reflecting surface ($r_k\, \theta_k\, \phi_k$) using the range algorithm described above and the measured angles of RM and OM. Such a system is capable of producing a "point cloud" of data representing the LIDAR system surveillance of the vehicle environment. These data are then sent to the

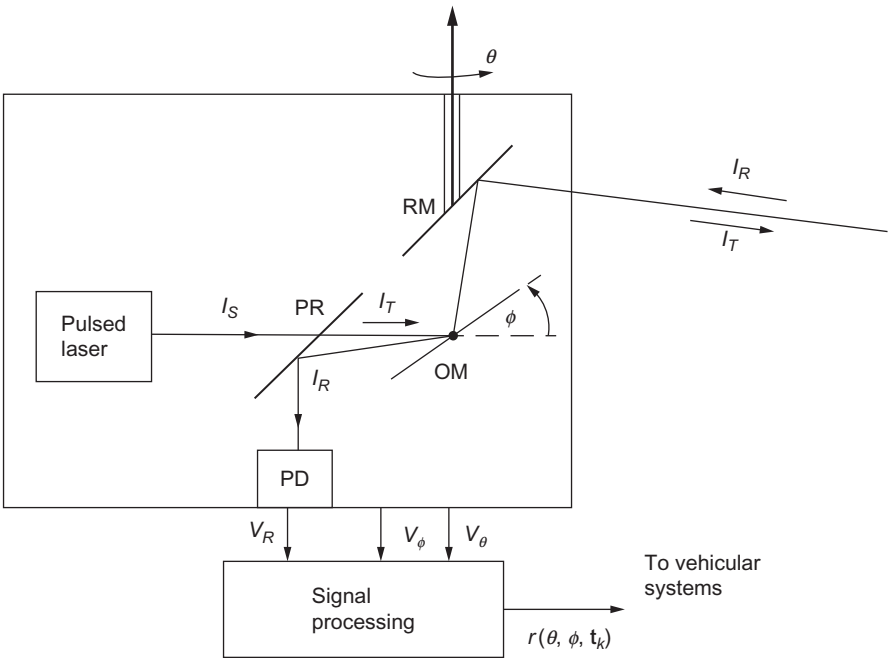


FIG. 5.31 Illustrative LIDAR block diagram.

vehicle control systems that initiate action (e.g., braking, steering, and deceleration) as called for by the control algorithms. The details of such actions are vehicle- and control-system-specific, but illustrative examples are given in Chapters 10 and 12.

There are other technical means for achieving the scanning that is necessary for the LIDAR sensor. One such mean involves multiple lasers, each with a specific angle toward which the light propagates. Scanning can be achieved by sequentially switching the lasers a single one at a time. By scanning in this way, the same result as achieved by the mirror system can be accomplished by the switched laser scanning method.

DIGITAL VIDEO CAMERA

In addition to LIDAR as a sensor system for surveillance of the region surrounding a vehicle, there are digital video cameras that can provide the necessary data for various control applications. There are many applications of these cameras that are discussed later in this book including blind-spot detection, lane following or changing with automatic steering systems, and obstacle detection for collision avoidance systems. In this section, we present the basic theory of operation of digital video cameras.

The basic configuration of such a camera is depicted in Fig. 5.32.

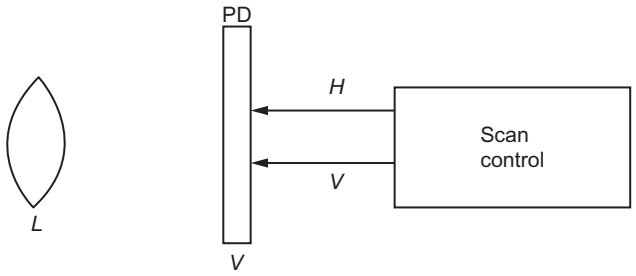


FIG. 5.32 Illustration of digital camera configuration.

In this figure, the block labeled PD consists of a planar array of photodetecting semiconductor elements that interact with the light projected onto the surface by a lens of lens system denoted L in Fig. 5.32. The lens projects a two-dimensional real image of objects within the field of view (FOV) of the lens onto the surface of the PD that is positioned at the image plane of L.

The two-dimensional image on the PD is converted to a single-dimensional signal v(t) called the video, which is a function of time by a process that is called scanning. At each of the tiny picture elements or pixels in the PD array, there are separate semiconductor elements for the three primary colors (red R, green G, and blue B). The scanning mechanism must scan all RGB pixel semiconductor elements. One means of separating the three primary colors in a camera is to cover the array of photodetecting elements with a matching cover array of filters passing R, G, or B. The photodetecting element beneath each of these colors generates a response to the intensity in each of the three colors. The scanning process involves sequentially sampling each consecutive pixel in a row/column at a fixed clock rate. At the end of scanning each row/column, the next adjacent row/column in the PD is scanned. The process continues until all rows and columns have been scanned in a single cycle. The camera can be used for obtaining video of images with motion by repeating the scanning cycle continuously at a rate compatible with vision perception (e.g., 30 Hz or more).

One technology for implementing the PD structure that has scanning capability is a so-called charge-coupled device (CCD). In a CCD-type digital camera, each pixel is in effect a photosensitive capacitor with associated charge transfer gates (electrodes). A simplified configuration of a photosensitive capacitor is depicted in Fig. 5.33.

The PD itself is made of a slab of p-type Si covered with an insulating thin layer of SiO_2. An optically transparent gate G is deposited on the SiO_2 layer. The incident light of intensity I enters the depletion region. Each photon creates a hole-electron pair. A voltage V_G is applied to the gate relative to the ground electrode creating an electric field vector $\bar{E}$ that attracts the electrons to the region immediately near G. The total number of electrons determined by I and by the time V_G is nonzero. Let $V_G(t)$ be

$$\begin{aligned} V_G(t) &= V_O \quad T_k < t < t_k + T \\ &= 0 \quad t_k + T < t < t_{k+1} \end{aligned} \tag{5.77}$$

where

$$t_k = kT_F$$

T_F = picture frame time interval, T = photon collecting interval, and $T_F > T$.

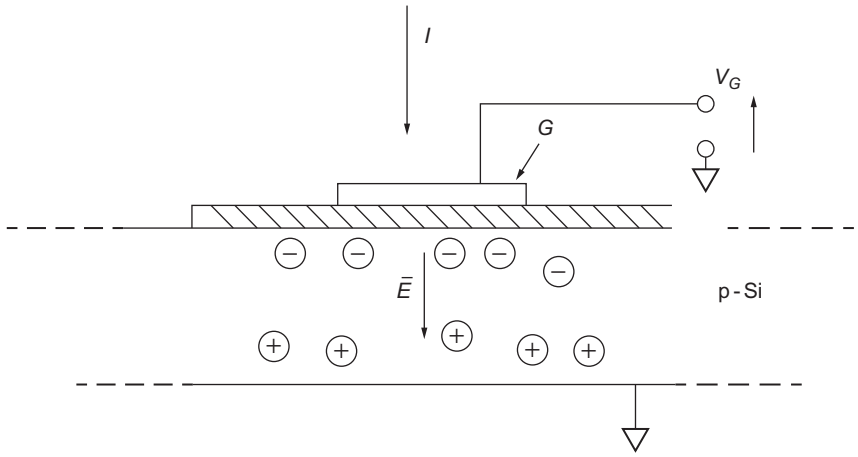


FIG. 5.33 Illustrative photosensitive capacitor configuration.

The total charge, which is proportional to the number of electrons near the gate (N_p), collected during a picture frame within the region surrounding the gate G is given by

$$N_p(k) = \gamma \int_{t_k}^{t_k+T} I(t)dt \tag{5.78}$$

where

$$\gamma = \text{constant for the structure}$$

When $V_G = 0$, the charge stops accumulating. In essence, the gate voltage serves a "shuttering" action.

The scanning for a column of photon-generated charge for each pixel can be implemented by one of many methods, each of which depends on the PD configuration. We illustrate scanning with a configuration that is called "interline." The following description of a CCD scanning method is illustrative and not exactly based on any existing camera configuration. However, it explains some of the basic operations involved in scanning the array of charge distributions obtained during a shuttering step. For this illustrative configuration, each column of charge accumulation pixels is adjacent to a column of CCD pixels that have an opaque cover and do not generate charge from incident photons. That is, the charge accumulation and CCD columns are interchanged along the face of the PD.

For this illustrative interline configuration, the CCD columns have three gates for each pixel that is adjacent to the corresponding photosensitive pixel. The combination of a photosensitive column of pixels and its adjacent CCD column constitutes a so-called channel. Each channel configuration is separated from the adjacent channels by an insulating layer of oxides that are created during the PD fabrication process. These regions, which are called channel stops, prevent charge from crossing channels and cause the charges to be moved along a channel during the scanning process.

The CCD scanning portion of each channel for the scanning example considered here consists of three gates for each pixel, as depicted for a portion of one channel in Fig. 5.34.

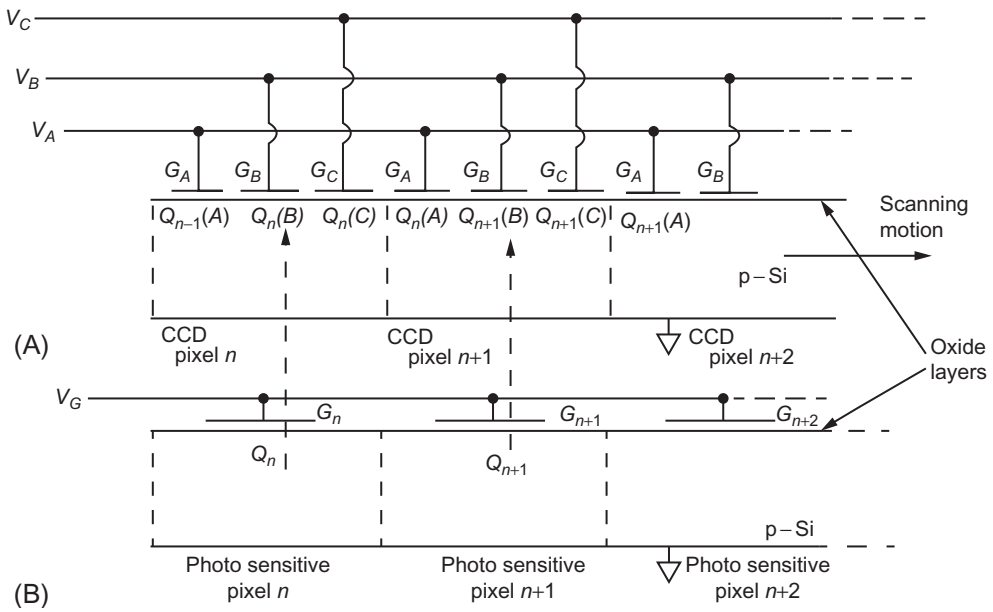


FIG. 5.34 Illustration of scanning CCDs (A) and adjacent photosensitive pixels (B).

In this figure, the two columns in (a) and (b) are actually physically side by side and constitute a single channel.

Each of the three gates on any CCD scanning column is activated by voltages such that G_A is connected to V_A, G_B to V_B, and G_C to V_C. The three voltages have pulse waveforms and are at the same frequency but have different phases as depicted in Fig. 5.35.

The time variable δt is the differential time following the shuttering interval of the frame k:

$$\delta t = t - (t_k + T) \qquad t_k + T < t \leq t_{k+1} + T \tag{5.79}$$

The scanning interval τ_1, which is the time required to move Q_n from CCD pixel n to CCD pixel $n+1$, is divided into three intervals of equal duration such that

$$\begin{aligned} T_B &= \tau_1/3 \\ T_C &= 2\tau_1/3 \end{aligned} \tag{5.80}$$

At any given time within each scanning cycle when one of the voltages is high, there is charge in any pixel only under the gate. For example, in pixel n of Fig. 5.34, the charge denoted $Q_n(B)$ is the position of the total charge within that pixel during the time interval for which V_B is high (i.e., $0 \leq \delta t \leq T_B$). During that time interval, there is (theoretically) no charge under G_A or G_C in any pixel. During the next subinterval of the scanning cycle, V_C is high, and the charge moves from under G_B to under G_C and is denoted $Q_n(C)$. During the final interval of a scan cycle, voltage V_A is high, and the charge in each pixel moves under G_A. At this point, the charge created in photosensitive pixel n has moved into the first region of pixel $n+1$ as illustrated in Fig. 5.34 by $Q_n(A)$.

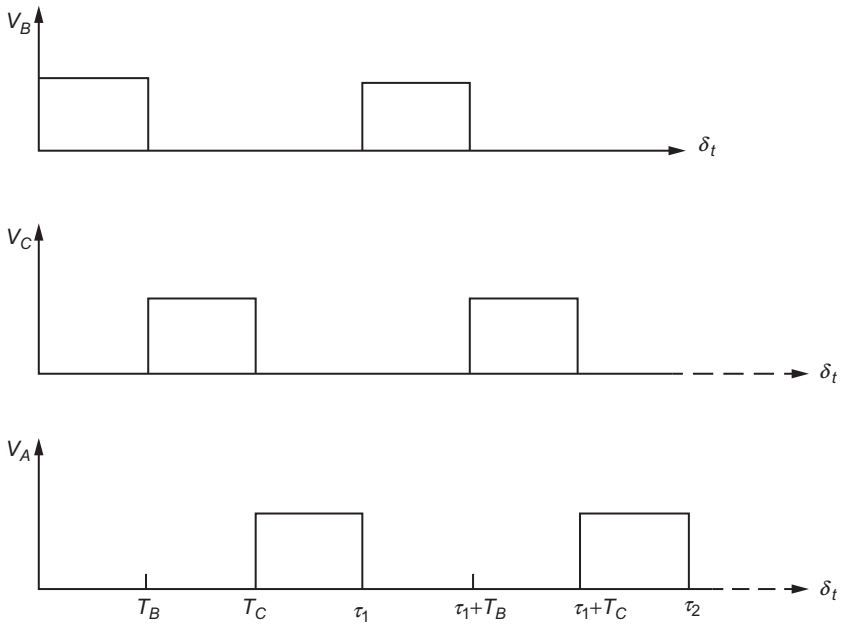


FIG. 5.35 Timing diagram for scanning voltages.

All columns are shifted synchronously such that all rows of charge move toward the end of the columns (to the right in Fig. 5.35) simultaneously.

At the end of all of the columns is another CCD scanning structure similar to the column but oriented orthogonal to the column CCD part of each channel. This scanning CCD receives photogenerated charge from each column, thereby receiving in each of its pixel elements the charge from the channel to which it is connected and is termed a row scanning CCD. Once a given row has transferred to this CCD, the charge is shifted toward an end where it constitutes row scanning and yields a charge sequence that is converted to the video signal as explained later.

Assuming there are M rows of pixels in the PD, the time to scan all columns to the row scanning CCD corresponds to the video frame time interval T_F. The time interval to shift all charge in columns from row n to row $n+1$ in the column CCD is τ_1. Thus, for M rows, the complete scan time T_M is given by

$$T_M = (M+1)\tau_1$$

During the first one-third of the initial scanning interval $0 \leq \delta t < \tau_1$, the voltage V_B is high, and the photogenerated in pixel n is transferred to the region under gate G_B for that pixel. During the next subinterval $T_B \leq \delta t < T_C$, only voltage V_C is high, and V_A and V_B are low. This high V_C voltage shifts the charge to the region under gate G_C in pixel n. Then, during the final interval $T_C \leq \delta t < \tau_1$, only voltage V_A is high. This transfers the charge to the region under gate G_A in pixel $n+1$. During the next scan cycle in the period $\tau_1 \leq \delta t < \tau_2$, the same set of moves occurs except that the first step is for the charge to

move from under gate G_A to under G_B instead of from the photocapacitor. For all of the next charge coupling cycles from $m\tau_1 \leq \delta t < (m+1)\tau_1$ with $m = 1,2,\ldots M$, the charges move in sequence to the right as depicted in Fig. 5.34.

In the example configuration, the G_C gate of pixel M for each column is adjacent to a G_B gate of the row scanning CCD. During the final step in each column scan, this charge is transferred from the end of each column into a horizontally oriented CCD.

During row scanning in the horizontal CCD, a similar configuration for each pixel having three gates/pixel is activated by a set of three voltages having the same sequence as depicted in Fig. 5.35. However, the row scanning must be completed before the succeeding column scan step. The frequency of the column scanning F_C (i.e., the frequency of V_A V_B V_C) is given by

$$F_C = \frac{1}{\tau_1} \tag{5.81}$$

The row scanning frequency F_R must be at least satisfy

$$F_R = NF_C \tag{5.82}$$

where N = number of columns.

For this set of scanning frequencies, all N-charge packets that are transferred to the row scanning CCD are transferred out before the next row of charge is supplied to that CCD, since all of the charge being transferred into it simultaneously from each column. Furthermore, with the column scanning frequency F_C given above, all charges that were put into each photonsensitive pixel in each channel will have been scanned out.

To be useful, for creating a video signal that corresponds to the image on PD for each shuttering interval, the charges leaving the row scanning CCD must be converted to a voltage that is proportional to the charge. A circuit that accomplishes this process is called a charge amplifier. An example circuit diagram for such a charge-to-voltage converter that is implemented with an FET and an op-amp is depicted in Fig. 5.36.

One of the major issues in obtaining a voltage that is proportional to the charge being transferred to the output node is the relatively small magnitude of the charge. A typical PD will supply charge with a

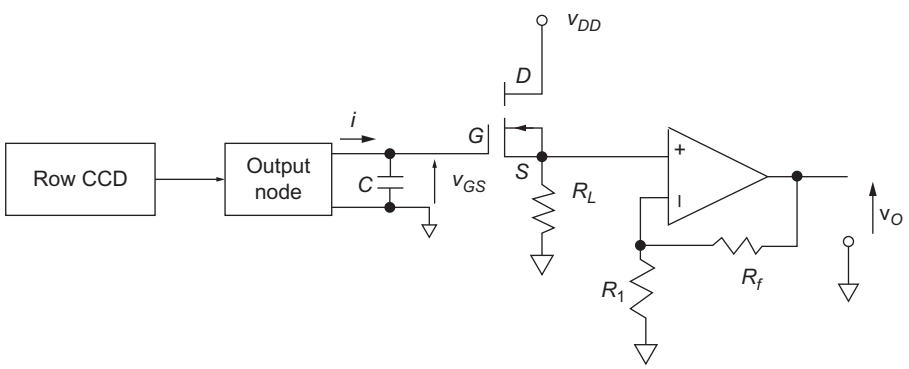


FIG. 5.36 Example CCD output amplifier.

maximum value of only 100,000–200,000 electrons. The output node supplies a voltage to the FET source-follower amplifier v_{GS}, which is given by

$$v_{GS} = \frac{Q_{n,m}}{C} \tag{5.83}$$

where $Q_{n,m}$ = photon-generated charge from row n, column m, and C = net capacitance from G to ground, which contains charge $Q_{n,m}$.

The voltage that is developed across the FET source load resistance R_L is amplified with a noninverting op-amp circuit. The output voltage $v_o(t)$ is an analog version of the scanned image. For most applications, the analog video is converted to a digital format via A/D conversion.

For color video, the PD must have pixel elements that respond to the RGB. One method of achieving the color separation is to incorporate R, G, and B filters in a cover for the PD and to have photosensitive elements for each color in each pixel. The photogeneration and scanning are accomplished as explained above. There are numerous vehicular applications as mentioned in the introduction to this section. The discussions of each application in the various chapters of this book make reference to this section.

FLEX-FUEL SENSOR

Another sensor, which has evolved in recent years, has the capability to measure the composition of a mixture of gasoline and ethanol that is called flex fuel. As explained in Chapter 6 and based on the theory presented in Chapter 4 concerning the control of fuel delivered to an engine, for a vehicle equipped with an engine that can run on flex fuel, the composition of the fuel must be measured for the engine control system to determine the quantity of fuel delivered to the engine. In this section, we explain the theory of operation of a sensor for measuring this composition, that is, the fraction of ethanol in gasoline (η_{eF}). A flex-fuel sensor (FFS) is essentially a capacitor having capacitance that is a unique function of fuel composition. Circuitry for measuring the capacitance $C(\eta_{eF})$ yields the data necessary for the engine control to determine fuel delivery.

In order to explain the theory of an FFS, it is helpful to briefly review some of the basic elements of electrostatic field theory. Electrostatic field theory is analogous in many respects to magnetic field theory in that it is characterized by a pair of field vectors that have distributions in space determined by charge distributions and conducting boundaries.

The relationship between these two field vectors is determined by the surrounding medium. The most basic configuration for explaining these two vectors is a point charge of magnitude q (units = coulombs) located at the origin of a spherical coordinate system. One of the vectors that is called the electric flux density vector and at a point of vector position $\bar{r}$ is denoted $\bar{D}(\bar{r})$ and is given by

$$\bar{D}(\bar{r}) = \frac{q}{4\pi r^2}\hat{r} \tag{5.84}$$

where $\hat{r}$ = unit vector directed radially away from the origin and
$r = \|\bar{r}\|$ = distance to the given point from the origin.

An electric flux density vector exists in association with any distribution of electric charges (e.g., on a conductor). The theory for calculating the distribution in space of $\bar{D}$ for any arbitrary charge distribution is well covered in books that are devoted to electromagnetic field theory, but it is not required for the simplified FFS analysis presented below.

One of the important field properties of an electric flux density vector that is very useful in computing the capacitance of an FFS is the integral of the normal component of $\bar{D}$ over a closed surface that is equal to the total charge enclosed by the surface. This integral can be expressed as follows:

$$Q_T = \int_S \bar{D} \cdot \hat{n} ds \tag{5.85}$$

where S is the closed surface,
$\hat{n}$ = outer normal unit vector to the surface at differential surface element ds, and
Q_T = total charge enclosed by the surface.

The other important vector in electrostatic field theory is the electric field intensity vector, which is denoted $\bar{E}$. This vector is related to $\bar{D}$ by the properties of the material in which the change distribution responsible for generating $\bar{D}$ is located. For an isotropic homogeneous material (IHM) such as is found in an FFS, these vectors are related as given below:

$$\bar{E} = \frac{\bar{D}}{\epsilon} \tag{5.86}$$

where ϵ = dielectric constant for the material (also called permittivity). For a vacuum, ϵ is denoted ϵ_o. For an IHM, this parameter is given by

$$\epsilon = \epsilon_o \epsilon_r \tag{5.87}$$

where ϵ_r = relative dielectric constant (dimensionless) for the material.

For the two components of flex fuel, ϵ_r is given by

$$\begin{aligned} \epsilon_g &= \epsilon_r \text{ for gasoline} \\ &\cong 2 \end{aligned} \qquad \begin{aligned} \epsilon_e &= \epsilon_r \text{ for ethanol} \\ &\cong 25 \end{aligned} \tag{5.88}$$

The electric field intensity vector provides the fundamental basis for calculating the difference in voltage between a pair of oppositely charged conductors that are separated in space by an insulating material having dielectric constant ϵ as depicted in Fig. 5.37.

The voltage difference between these two conductors δV is given by

$$\delta V = -\int_{C_S} \bar{E} \cdot \overline{d\ell} \tag{5.89}$$

where C_S = contour in space from the $-Q$ conductor to the $+Q$ charged conductor, $\overline{d\ell}$ vector differential length element along C_S (i.e., tangent to C).

This equation is also important in analyzing the capacitance C of a flex-fuel sensor. In any such sensor, there is a pair of circuit connections (i.e., wires) such as are denoted ω_1 and ω_2.

In the configuration depicted in Fig. 5.38, the pair of coaxial cylindrical conductors C_r and C_2, with leads w_1 and w_2 form a capacitor. The capacitance of any such configuration is denoted C and is given by

$$C = \frac{Q}{\delta V}$$

The two integral equations given above can be used to model the capacitance of an FFS in a somewhat simplified version of a common FFS configuration depicted in Fig. 5.38.

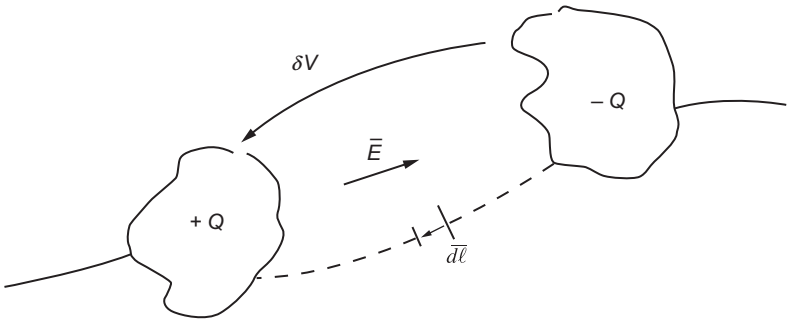


FIG. 5.37 Illustration of a pair of oppositely charged conductor.

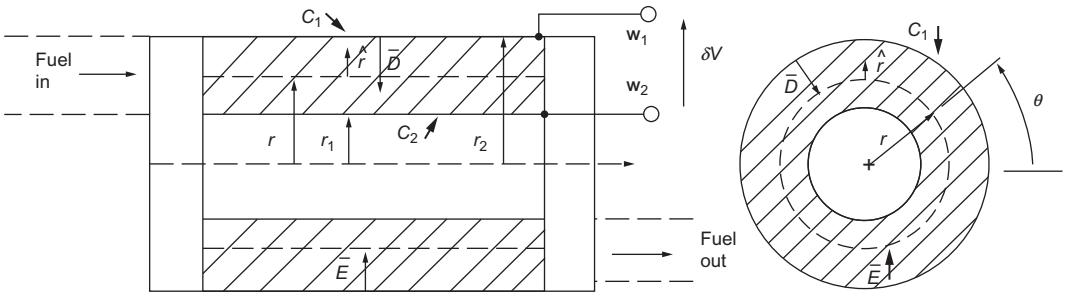


FIG. 5.38 Simplified flex-fuel sensor configuration.

The FFS configuration depicted in this figure consists of a pair of coaxial cylindrical conductors. The flex fuel, the composition of which is to be measured, is contained in the space between the two conductors. The fluid connectors that allow the fuel to pass into the region between the conducting cylinders are not specifically depicted since it is not essential for deriving the sensor model and because it is manufacturer-specific. The inner cylinder radius is denoted r_1 and the outer radius r_2. The endcaps that form a portion of this representative structure are made from insulating materials. A pair of leads (w_1, w_2) are connected respectively to the outer and inner cylindrical conductors. These leads provide the connection to the circuitry what is involved in the measurement of capacitance which ultimately yields the fuel composition.

The flex fuel contained between the two cylindrical conductors is an isotropic homogeneous material with a relative dielectric constant which we denote $\in_{FF}$. Any electrical structure in which a pair of conductors is separated by an insulating material essentially forms a capacitor.

The capacitance of this FFS structure is denoted C_{FF} and is given by

$$C_{FF} = \frac{Q}{\delta V} \tag{5.90}$$

where Q is the magnitude of the charge on the two cylinders with $Q > 0$ assumed on the outer cylinder and $-Q$ on the inner cylinder. For simplicity of analysis, the geometry is assumed to be such that the

sensor length $\ell \gg r_2$. In this case, the field vectors are essentially radially directed, except for a relatively small region near the endcaps.

The analysis begins with the surface integral of the flux density vector. The surface S for this integral is a coaxial cylinder of radius r ($r_1 \leq r \leq r_2$). The integral of $\bar{D}$ over this surface yields the charge $-Q$ on the inner conductor is given by

$$-Q = \int_S \bar{D} \cdot \hat{n} ds \tag{5.91}$$

where

$$\bar{D} = -D_r \hat{r}$$
$$\hat{n} = \hat{r}$$
$$ds = r d\theta dz$$

With the assumption of radially directed $\bar{D}$, the flux density on surface S is constant for which the equation for Q becomes

$$\begin{aligned} -Q &= -D_r \int_o^{2\pi} \int_o^{\ell} r d\theta dz \\ &= -2\pi D_r \ell r \end{aligned} \tag{5.92}$$

Solving for D_r yields,

$$D_r = \frac{Q}{2\pi \ell r}$$

The electric field intensity vector is computed from D_r as follows:

$$\begin{aligned} \bar{E} &= \frac{\bar{D}}{\in_{FF} \in_o} \\ &= -\frac{D_r}{\in_{FF} \in_o} \hat{r} \\ &= -\frac{Q\,\hat{r}}{2\pi \in_{FF} \in_o r \ell} \end{aligned} \tag{5.93}$$

The voltage difference between the conductor δV can be computed from the contour integral of $\bar{E}$ over a contour C that is a straight line radially directed such that the vector differential length $\overline{d\ell}$ is given by

$$\overline{d\ell} = dr\hat{r}$$

The voltage difference δV is given by

$$\begin{aligned} \delta V &= -\int_{r_1}^{r_2} \bar{E} \cdot \hat{r}\, dr \\ &= \frac{Q}{2\pi \in_o \in_{FF} \ell} \int_{r_1}^{r_2} \frac{dr}{r} \\ &= \frac{Q}{2\pi \in_{FF} \in_o \ell} \ell n \left(\frac{r_2}{r_1} \right) \end{aligned} \tag{5.94}$$

The FFS sensor capacitance C_{FF} is given by

$$C_{FF} = \frac{Q}{\delta V} = \frac{2\pi \in_{FF} \in_o \ell}{\ell n\left(\frac{r_2}{r_1}\right)} \tag{5.95}$$

The factor C_{FF} varies by a large amount depending on the ethanol fraction of the flex fuel that means that $C_{FF}(\eta_{eF})$ varies significantly with η_{eF}. A measurement of C_{FF}, which is straight forward, yields the value for η_{eF} as explained in the section of Chapter 6 that is devoted to flex-fuel vehicles.

A measurement of C_{FF} yields a measurement of $\in_{FF}$ since all parameters of C_{FF} are known. In Chapter 6, it is shown that the value of $\in_{FF}$ contains the information required to determine flex-fuel composition (i.e., η_{eF}). Thus, the FFS depicted in Fig. 5.38 provides the necessary fuel composition for engine control from a measurement of C_{FF}.

OSCILLATOR METHODS OF MEASURING CAPACITANCE

There are numerous methods of measuring capacitance that are readily adaptable to vehicular environments. One such method that is routinely used involves connecting the capacitor to an oscillator circuit for which the frequency of oscillation is uniquely determined by its capacitance. A representative example circuit is a so-called astable multivibrator that is one of three operating modes of a 555 timer IC. The schematic of an astable multivibrator that is connected to C_{FF} is depicted in Fig. 5.39.

The 555 timer circuit has a limited range of capacitance values that can be measured with the circuit of Fig. 5.39. It may not be practical to use this circuit for measuring C_{FF} depending on the FFS

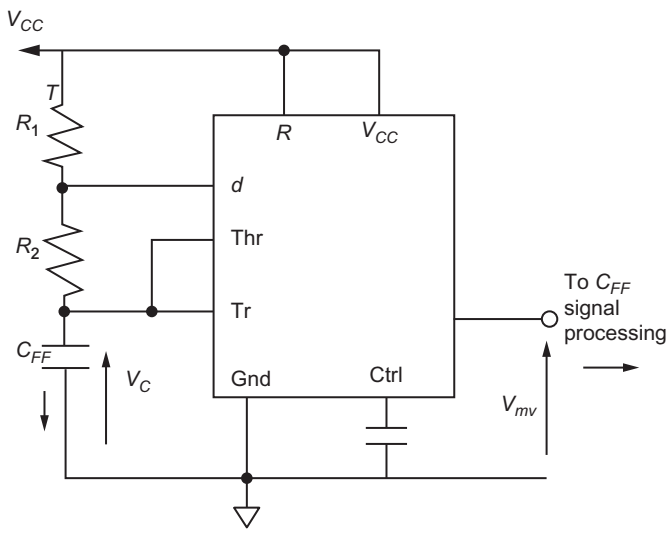


FIG. 5.39 Circuit for measuring C_{FF}.

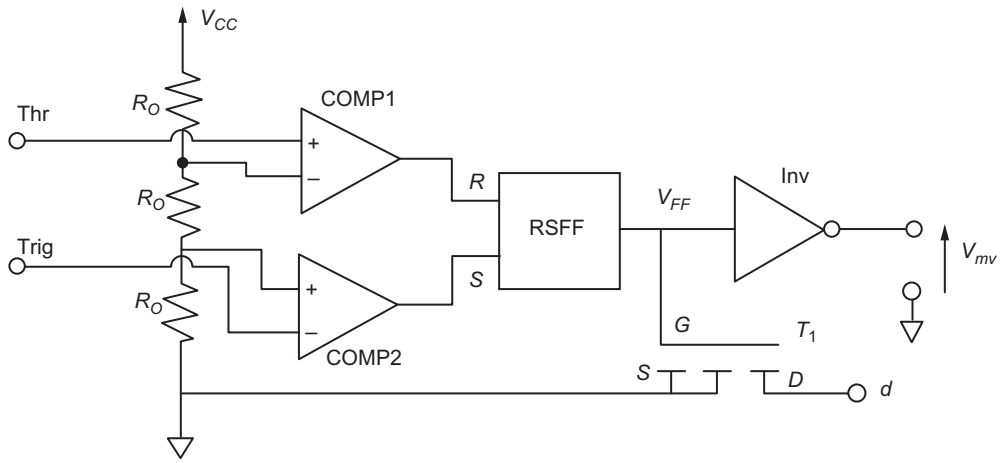


FIG. 5.40 Block diagram of representation 555 timer circuit.

configuration. However, this circuit illustrates an oscillator method of measuring capacitance. A block diagram of the internal circuitry of a typical 555 timer IC is depicted in Fig. 5.40. A similar block diagram is presented in Chapter 2 but is repeated here for convenience.

The circuit pin labels are V_{CC} = supply voltage, V_C = voltage across C_{FF}, Thr = threshold, Tr = trigger input, d = discharge input, OUT = circuit output, and V_{FF} = R-S FF output voltage. The block diagram components are the following:

COMP1 and COMP2 = comparators
R-S FF = set-reset flip-flop
T_1 = n-channel enhancement FET
Inv = logical inverter

The operation of this multivibrator is controlled by the state of the R-S FF that causes the d input to switch between essentially open circuit and short circuit. The circuit oscillates such that V_C charges $V_{CC}/3$ to $\frac{2}{3}V_{CC}$ as set by the three identical resistors, each of resistance R_o that forms a two-level voltage divider circuit. Whenever V_{FF} is in its low state (ideally 0 V), transistor T_1 is effectively an open circuit to the d input. During this period, the capacitor C_{FF} is charging through resistors R_1 and R_2. When $V_C = \frac{2}{3}V_{CC}$, COMP 2 switches and the R-S FF is switched to the high state. This causes the d input to be nearly zero, and C_{FF} discharges through R_2 until $V_C = V_{CC}/3$. At this point, COMP 1 switches, and the R-S FF switches such that V_{FF} is low and the d input is nearly an open circuit. At this point, C_{FF} begins charging again until it reaches $\frac{2}{3}V_{CC}$ and the cycle is complete. The output voltage switches high when V_{FF} is low and vice versa. This output waveform is rectangular and is given by

$$\begin{aligned} V_{mv}(t) &= V_H \quad t_k \le t < t_k + T_H \\ &= V_L \quad t_k + T_H \le t < t_{k+1} \end{aligned} \tag{5.96}$$

where

V_{H} = high state voltage
V_{L} = low state voltage
T_{H} = high level period
T_L = low level period
t_k = time of kth cycle of oscillation.

During the capacitor charging interval $(t_k \le t < t_k + T_H)$, the capacitor satisfies the following differential equation:

$$(R_1 + R_2)i_C + V_C = V_{CC} \tag{5.97}$$

where

$$i_c = C_{FF}\frac{dV_C}{dt} \quad t_k \le t < t_k + T_H \quad k = 0,1,2,\ldots \tag{5.98}$$

The solution to this differential equation during the charging period is given by

$$V_C(t) = V_{CC}\left(1 - e^{-t/\tau_c}\right) \quad t_k \le t < t_k + T_H$$

where

$$\tau_c = C_{FF}(R_1 + R_2) \tag{5.99}$$

Similarly, during the discharge period, the equation becomes

$$R_2 i_C + V_C = 0$$

The solution for this interval is left as an exercise for the reader.

The periods can be determined from the voltage V_C. The voltage $V_C(t_k + T_H) = 2\dfrac{V_{CC}}{3}$ at which point the d input causes V_{mv} to switch to V_L and the capacitor begins discharging.

The high-level time interval T_H can be computed from the ratio:

$$V_C(t_k + T_H)/V_C(t_k)$$

$$V_C(t_k) = \frac{V_{CC}}{3} = V_{CC}\left(1 - e^{-t_k/\tau_c}\right)$$

$$V_C(t_k + T_H) = \frac{2V_{CC}}{3} = V_{CC}\left(1 - e^{-(t_k + T_H)/\tau_c}\right) \tag{5.100}$$

The ratio $V_C(t_k + T_H)/V_C(t_k)$ is given by

$$\frac{1 - e^{-\frac{(t_k + T_H)}{\tau_c}}}{1 - e^{-t_k/\tau_c}} = 2 \tag{5.101}$$

Noting that $1 - e^{-t_k/\tau_C} = \dfrac{1}{3}$ and $1 - e^{-\left(\frac{t_k + T_H}{\tau_c}\right)} = 2/3$, the value T_H can be found by substitution of the above and simplifying algebraically from the result:

$$e^{-T_H/\tau_C} = 1/2$$

$$\begin{aligned} T_H &= \ell n(2)\tau_c \\ &= \ell n(2)(R_1 + R_2)C_{FF} \end{aligned} \tag{5.102}$$

Similarly, it can be shown that the low interval T_L is given by

$$T_L = \ell n(2)R_2C_{FF} \tag{5.103}$$

The frequency of oscillation f is given by the reciprocal of the total period $T_H + T_L$:

$$\begin{aligned} f &= \frac{1}{T_L + T_H} \\ &= [\ell n(2)(R_1 + 2R_2)C_{FF}] \end{aligned} \tag{5.104}$$

Thus, C_{FF} can be measured by measuring T_H, T_L or f yielding a measurement of fuel composition η_{eF} as explained in Chapter 6 where the measurement of fuel composition is necessary for engine control in flex-fuel vehicles. This measurement can be accomplished by the main engine control or by a dedicated microprocessor as a part of the sensor that can compute C_{FF} via algorithms, as exemplified above, and compute η_{eF} from C_{FF} and output this fuel composition to the main engine control. As explained in Chapter 6 in detail, a fuel temperature measurement is also required, which is typically implemented with a thermistor that is located as part of the FFS structure.

An example of capacitance measurement when accomplished in the FFS sensor system that is equipped with a microprocessor and that generates a digital output to the main FFS control computer involves measuring the duration of the high state of the multivibrator output signal. For illustrative purposes, it is assumed that V_{mv} high and low voltages correspond to logic high and logic low, respectively, of the C_{FF} measurement circuit.

A representative measurement circuit is depicted in Fig. 5.41.

In this block diagram, the sensor is controlled by a microprocessor-based control system with internal stored programs. This example system operates by first measuring the time interval T_H and then computing C_{FF} from this measurement. The FFS control outputs the fuel composition η_{eF} to the main FI control system that controls fuel delivery to the engine as explained in Chapter 6.

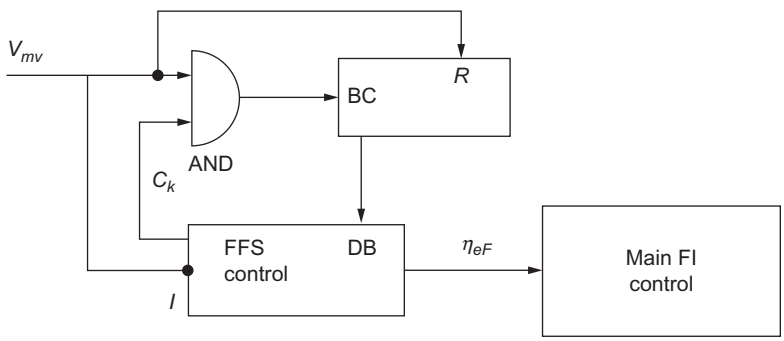


FIG. 5.41 Example C_{FF} measurement block diagram.

The measurement of T_H is accomplished by counting clock (C_k) pulses at frequency f_c for this duration in a binary counter. At the transition of V_{mV} from low to high, the binary counter is reset such that counting c_k pulses occur during the time interval:

$$t_k \leq t < t_k + T_H(k)$$

The number of pulses counted during oscillator cycle k is given by

$$N_k = \{\lfloor f_c\, T_H(k) \rfloor\} \tag{5.105}$$

In this example measurement system, it is assumed that the C_k signal is internally generated in the FFS control. This control receives an interrupt input (I) from the multivibrator output that indicates to the FFS control that the high interval has ended. At this time, the FFS control reads the value of N_k.The FFS control is programmed to compute C_{FF} from this measurement of T_H.:

$$C_{FF} = \frac{N_k}{f_c \ell n(2)(R_1 + R_2)} \tag{5.106}$$

The value of η_{eF} is found via table lookup of $\eta_{eF}(C_{FF})$, and this value is outputted to the main FF computer.

To this point in the discussion of FFS, it has been implicitly assumed that the sensor is a purely capacitive circuit element. However, as is the case of any circuit component, there are electrical properties that require a more complex equivalent circuit to analytically model the device. In the case of an FFS, the equivalent circuit is depicted in Fig. 5.42.

In this equivalent circuit, the capacitance C_{FF} is shunted by a resistance R_C. The physical origin of this resistance is the bulk electrical conductivity of flex fuel. Although both constituent components of this fuel are essential electrical insulators, the mixture can have a nonzero conductivity partly due to fuel additives yielding an equivalent shunt resistance R_C. Although the flex-fuel conductivity is not zero, it is relatively small corresponding to a relatively large value for resistance R_C.

The influence on measurement of C_{FF} (and thus $\eta_{eF)}$ can be made negligible by proper design of the measuring circuit. For example, the resistances R_o in the multivibrator of Fig. 5.46 can be chosen such that $R_o \ll R_C$. In this case, the analysis of the oscillator circuit for T_H, T_C, and f is given below based on flex-fuel conductivity.

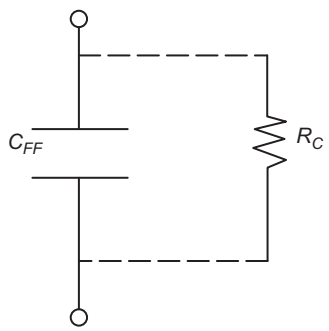


FIG. 5.42 FFS equivalent circuit.

The conductivity of the flex fuel is expressed by the conductivity K_{ff} of the flex-fuel mixture. With respect to Fig. 5.42, the resistance of the sensor R_C is given by

$$R_C = \frac{V}{I_C} \tag{5.107}$$

where I_C is the current flowing through the fuel. This current is uniformly distributed circumferentially around the concentric cylinders and can be computed from the vector current density $\bar{J}_C$ as given by

$$I_C = \oint_S \bar{J}_C \cdot \hat{n}\, ds \tag{5.108}$$

where

$$\begin{aligned} \bar{J}_C &= K_{ff}\bar{E} \\ &= J_r \hat{r} \end{aligned}$$

That is, the current flows radially between the cylinders such that I_C is given by

$$\begin{aligned} I_C &= \frac{2\pi r \ell K_{ff} D_r}{\epsilon_{FF}} \\ &= \frac{K_{ff} Q}{\epsilon_{FF}} \epsilon_{FF} \end{aligned} \tag{5.109}$$

Combining the equation for I_C with the capacitance equation yields

$$R_C = \frac{\epsilon_{FF}}{K_{ff} C_{FF}} \tag{5.110}$$

It can be shown that the analysis of the oscillator circuit in which there is a shunt resistance R_C is similar to the analysis of the circuit of Fig. 5.42. However, the T_H, T_L, and frequency are modified by R_C such that

$$\begin{aligned} T_H &\simeq \frac{(R_1 + R_2) C_{FF}\, \ell n(2)}{1 + (R_1 + R_2)/R_C} \\ T_L &\cong \frac{\ell n(2)(R_1 + 2R_2) C_{FF}}{1 + R_2/R_C} \\ f &= [T_H + T_L]^{-1} \end{aligned} \tag{5.111}$$

The two equations for T_H and T_L can be solved for both C_{FF} and R_C in the FFS control of Fig. 5.41. The capacitance C_{FF} is measured via a measurement of T_H using the exemplary system of Fig. 5.41 and then finding $\eta_{eF}(C_{FF})$ (e.g., via table lookup).

ACCELERATION SENSOR

There are several applications of sensors for measuring vehicle acceleration along body axes directions. These applications are discussed in Chapter 7 on vehicle motion control and in Chapter 10 on vehicle safety-related systems. There are multiple technologies for measuring acceleration, virtually all of which are based on Newton's law of force F and acceleration a for a given mass m:

$$F = ma$$

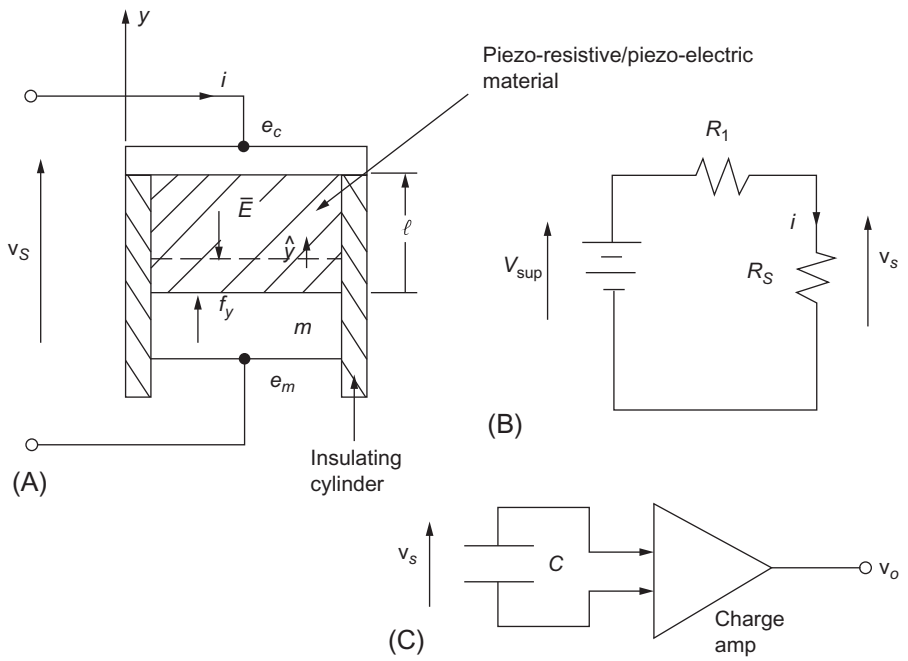


FIG. 5.43 Example single-axis acceleration sensor. (A) Illustrative configuration; (B) Piezoresistive Sensor circuit; (C) Piezoelectric sensor circuit.

Among the many vehicular applications of an acceleration sensor, there is a requirement to measure a steady state (i.e., constant magnitude acceleration). For example, Chapter 10 explains a directional stability system in which measurement of acceleration along the vehicle lateral axis (i.e., a_y) is required for a maneuver at a constant speed along a constant radius curve in a road. This acceleration is constant along that path at a constant speed.

A sensor configuration that can generate a steady voltage that is proportional to this acceleration is depicted in Fig. 5.43A. This sensor measures acceleration along a single (y) axis.

This sensor configuration consists of a cylindrical mass m within a cylinder fabricated from an electrically insulating material. A metallic endcap is fixed to the insulating cylinder. Between the mass m and the endcap is a piezoresistive material that is attached, as shown, to the endcap and the cylindrical mass. The entire structure is attached rigidly to the vehicle whose acceleration along the y-axis (a_y) is to be measured.

The mass m applies a force f_y to the piezoresistive material where

$$f_y = ma_y \tag{5.112}$$

As explained earlier in this chapter, a force applied to a piezoresistive material changes the resistivity ρ of the material in proportion to the strain induced by the applied force.

For convenience, it is assumed that the mass m is made from a conducting material such that it and the endcap form a pair of electrodes ($e_m\, e_c$), to which wires are attached. The sensor is wired in a circuit (which is depicted in Fig. 5.43B) with a supply voltage V_{sup} applied to the series connection of a load resistance R_1 and the resistance R_S of the piezoresistive material.

The piezoresistive material resistivity ρ can be modeled as follows:

$$\rho = \rho_o + K_S f_y \tag{5.113}$$

where ρ_o = resistivity with zero force applied and K_S = constant for the material.

The applied voltage v_S across the sensor configuration due to V_{sup} creates an electric field vector $\bar{E}$ as depicted in Fig. 5.43. It is assumed for simplifying the theory of the sensor that $\bar{E}$ is given by

$$\bar{E} = -\frac{v_s}{\ell}\hat{y} \tag{5.114}$$

and is uniform over the cross section of the piezoresistive material. The current density $\bar{J}$ is given by

$$\bar{J} = \frac{\bar{E}}{\rho} \tag{5.115}$$

The current flowing through the piezoresistive structure i is given by the integral of the current density over a surface parallel to the x and z axes within the piezoresistive material:

$$\begin{aligned} i &= -\int_{S_p} \bar{J}\cdot\hat{y}\,ds \\ &= \frac{v_s S_p}{\ell\rho} \end{aligned} \tag{5.116}$$

where S_p = cross-sectional area of the piezoresistive material. The resistance R_S of the sensor is given by

$$\begin{aligned} R_S &= \frac{v_s}{i} \\ &= \frac{\ell\rho}{S_p} \\ &= \frac{\ell}{S_p}\left(\rho_o + K_S f_y\right) \end{aligned} \tag{5.117}$$

This resistance can be written in a simple model:

$$R_S = R_o + K_a a_y \tag{5.118}$$

where $R_o = \dfrac{\ell\rho_o}{S_p}$

$$K_a = \frac{\ell K_s m}{S_p}$$

The sensor resistance has a component that is proportional to acceleration that is present even for steady (DC) components of the acceleration. The voltage across the sensor in the circuit of Fig. 5.43 is given by

$$v_S = \frac{V_{sup}\left(R_o + K_a a_y\right)}{R_1 + R_o + K_a a_y} \tag{5.119}$$

In the exemplary circuit, the resistance R_1 is sufficiently large that v_s is given approximately by

$$v_s \simeq V_{sup}\left(\frac{R_o + K_a a_y}{R_1}\right) \tag{5.120}$$

For a vehicle system requiring a measurement of a_y, the voltage v_s can be sampled and converted to digital format via a standard A/D converter.

For vehicular electronic systems that do not require a measurement of a steady acceleration component (e.g., crash sensors as described in Chapter 10), an alternate configuration can be fabricated in which the piezoresistive material is replaced by a piezoelectric material. In such a material, the atomic structure is such that a dipolar electric field is created that is proportional to any strain/stress applied to the material. Quartz is an example of a material in which an internal electric field $\bar{E}$ is produced that is proportional to the internal strain.

We consider an exemplary configuration, such as that depicted in Fig. 5.43A, having a piezoelectric material (e.g., quartz) between the mass and endcap instead of a piezoresistive material. Quartz is an electrical insulator material having a dielectric constant $\in_Q$. The structure of Fig. 5.43 with quartz or some other piezoelectric material actually forms a capacitor with the mass and endcaps forming the electrodes. The force due to acceleration for the configuration of Fig. 5.43 produces an electric field that (for the purposes of simplifying the explanation of the theory of this acceleration sensor) is assumed to be uniform over the piezoelectric material and given by

$$\begin{aligned} \bar{E} &= -K_p f_y \hat{y} \\ &= -K_p m\, a_y \hat{y} \end{aligned} \tag{5.121}$$

The terminal voltage v_s for this sensor is given by

$$\begin{aligned} v_S &= -\int_o^{\ell} \bar{E} \cdot \hat{y}\, dy \\ &= K_p m \ell\, a_y \end{aligned} \tag{5.122}$$

If this sensor is connected to a very high input impedance amplifier, the voltage produced at its output is linearly proportional to acceleration of the sensor along its y-axis. For a piezoelectric capacitive-type sensor, the amplifier connected to the terminals is often a charge amplifier of the type described earlier in this chapter with respect to the readout circuit of a digital camera. In theory, the piezoelectric acceleration sensor could measure a steady acceleration, but in practice, any nonzero conductivity in the material could discharge the capacitance to some extent.

AUTOMOTIVE ENGINE CONTROL ACTUATORS

In addition to the set of sensors, electronic engine control is critically dependent on a set of actuators to control air/fuel ratio, ignition, and EGR. Each of these devices will be discussed separately.

In general, an actuator is a device that receives an electrical input (e.g., from the engine controller) and produces an output of a different physical form (e.g., mechanical or thermal or other). Examples of actuators include various types of electric motors, solenoids, and piezoelectric force generators. In automotive electronic systems, the solenoid is a very commonly used device because it is relatively simple and inexpensive.

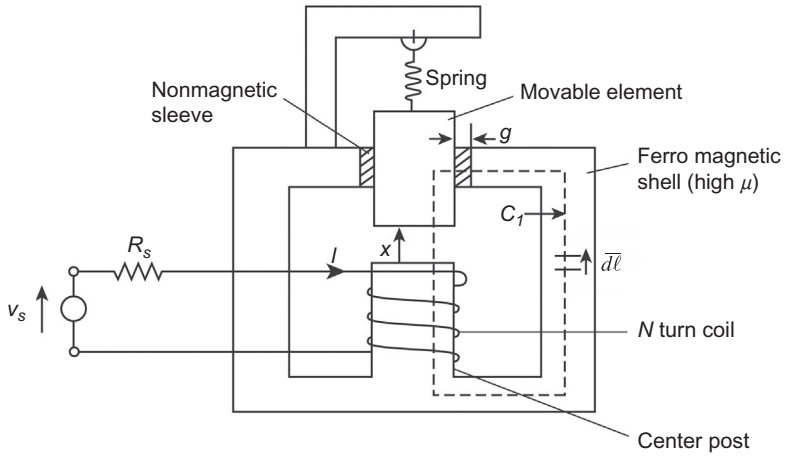


FIG. 5.44 Solenoid configuration.

The solenoid is used in applications ranging from precise fuel control to mundane applications such as electric door locks. A solenoid is, in essence, a powerful electromagnet having a configuration generally similar to that illustrated schematically in Fig. 5.44. The solenoid consists of a fixed cylindrical steel (i.e., ferromagnetic) frame with a movable steel element. A coil having N turns is wound around the steel frame, forming a powerful electromagnet.

Unlike the magnetic sensors explained above in which the source of magnetic field is a permanent magnet, the source of the magnetic field in the solenoid is the current I that flows through the coil. The lines of constant magnetic flux density B form closed contours such as denoted C_1 in Fig. 5.44. These contours include a segment through the center post and movable plunger, a segment directed radially in the upper and lower portions and then through the outer shell. The contour C_1 is in a plane that passes through the axis of symmetry of the cylindrical solenoid structure. Note that any contour such as C_1 passes through the ferromagnetic material and the nonmagnetic sleeve of thickness g and nonmagnetic air gap of length x.

Fig. 5.44 also shows a movable element that is in the form of a cylinder of high-permeability (μ) ferromagnetic material. This movable element is held away from the center post by a spring. The other end of this spring is attached to a structure that is fixed rigidly to the ferromagnetic shell. This shell is normally cylindrical in shape and coaxial with the center post.

The value of the magnetic field intensity $\bar{H}$ can be related to the total current I that passes through the surface enclosed by C_1 using one of the fundamental equations (Eq. 5.30) given above:

$$I_T = \oint_{C_1} \bar{H} \cdot \overline{d\ell} \tag{5.123}$$

where $\overline{d\ell}$ is a differential vector along the contour C_1. The magnetic flux density magnitude B is constant along any contour C_1. The magnitude of $\bar{H}$ along any contour C_1 is given by

$$H = \frac{B}{\mu}$$

where

$$\mu = \mu_o \text{ in the sleeve and gap}$$
$$= \mu_r \mu_o \text{ in the ferromagnetic material.}$$

The relative permeability (μ_r) in the ferromagnetic material is so large that in the ferromagnetic material H is negligibly small. The contour integral above reduces approximately to

$$I_T \simeq H_g(x+g) \tag{5.124}$$

where H_g is the magnitude of H in the air and sleeve material and is given by

$$H_g = \frac{B}{\mu_o}$$

To a close approximation, H and B are essentially constant over the center post cross-sectional area. The total current I_T is given by

$$I_T = NI$$
$$= \frac{B}{\mu_o}(x+g)$$

The total flux linking the N turn coil λ_T is given by

$$\lambda_T = N \iint_{A_c} B ds$$

where ds is the differential area in the cross section of the post and where the integral is taken over the cross-sectional area of the post A_C. With the assumption of essentially uniform B over the post, the total flux linking the coil becomes

$$\lambda \cong NBA_C$$
$$= \mu_o \frac{N^2 I A_c}{x+g}$$

The important circuit parameter that characterizes the electrical model for the coil, which is called its inductance L, is defined as

$$L = \frac{\lambda}{I}$$
$$= \frac{\mu_o N^2 A_C}{(x+g)}$$

The terminal voltage v_o of a two-terminal device having inductance L is given by

$$v_o = \frac{d(LI)}{dt} \tag{5.125}$$

For a fixed inductor, this model becomes familiar:

$$v_o = L\frac{dI}{dt} \quad \text{(for constant L)} \tag{5.126}$$

However, for a magnetic actuator such as is used in automotive electrical systems (e.g., the example solenoid), this inductance varies with the position of any movable element.

For the purposes of modeling an actuator, the primary focus is on determining the dynamic response (i.e., movement of the plunger) to an applied electrical signal. At any instant, the total energy is the sum of the magnetic and mechanical energy. As a simplification (without loss of generality), it is convenient to assume a lossless electromechanical system. In this case, the electric energy put into the system is stored in the magnetic field. If electrical power is supplied to the system at constant x, the instantaneous stored magnetic energy (W_m) is given by

$$W_m = \int_0^I \lambda_T(i,x)di \tag{5.127}$$

$$= \frac{1}{2}L(x)I^2 \tag{5.128}$$

If the total energy stored in the magnetic field is denoted W_m, conservation of energy requires that

$$\frac{dW_m}{dt} = IL\frac{dI}{dt} - f_e\frac{dx}{dt} \tag{5.129}$$

where the first term is the instantaneous electrical power P_e into the system and the second term represents time rate of change of mechanical energy due to the mechanical force of electrical origin f_e applied to the movable element. The negative sign on the second term indicates that the mechanical energy is taken from the stored magnetic energy and is applied to the movable element. In this model, both λ_T and x are independent variables. For our assumed conservative system, the force of electrical origin (f_e) for the solenoid of Fig. 5.43 is given by

$$f_e = \frac{\partial W_m}{\partial x} \tag{5.130}$$

$$= -\frac{\mu_o N^2 A_C I^2}{2(x+g)^2} \tag{5.131}$$

Note that the minus sign indicates a force that is reducing x and that it varies inversely with x.

The solenoid of Fig. 5.44 is mechanically unstable in the sense that a current of sufficient strength causes f_e to increase as an inverse quadratic function of x, whereas the spring force countering f_e varies linearly with x. In any solenoid configured as in Fig. 5.44, the movable element will accelerate toward the fixed post, stopping abruptly only when $x=0$. In any practical solenoid, the plunger actually bounces away from the post and oscillates briefly with decaying amplitude (typically at a very high frequency). Normally, the nonmagnetic sleeve provides sufficient mechanical damping to rapidly damp out any "bounce." It should be noted that the introduction of the model for the solenoid using stored energy is useful for explaining other types of actuators (e.g., motors to be discussed later in automotive electronic systems).

It is, perhaps, worthwhile to extend the static model developed above for the solenoid to develop the dynamic equations. First, however, we simplify the notation for the flux linkage to the following:

$$\lambda(I,x) = \frac{L_o I}{(1+x/g)} \tag{5.132}$$

where $L_o = \dfrac{\mu_o A_C N^2}{g}$

and where L_o is the inductance of the solenoid at $x=0$. Summing the voltages around the loop formed by the source v_s, R_s and the solenoid terminals yields

$$\begin{aligned} v_s &= IR_s + \frac{d\lambda}{dt} \\ &= IR_s + \frac{L_o}{(1+x/g)}\frac{dI}{dt} - \frac{L_o I}{g(1+x/g)^2}\frac{dx}{dt} \end{aligned} \tag{5.133}$$

The first term on the right-hand side of Eq. (5.133) is the voltage drop across the source resistance R_s. The second term is the familiar voltage due to the instantaneous inductance $L(x)$, and the final term is a voltage that is induced by the moving plunger.

Next, we write the mechanical equation of motion of the plunger:

$$f_e = M\frac{d^2x}{dt^2} + D\frac{dx}{dt} + K_s(x-\ell)$$

where M is the plunger mass, D is the mechanical damping force due to the plunger motion (that is here taken to linear), K_s is the spring rate of the spring that holds the movable element in its extended position, and ℓ is the spring length in the absence of the mechanical force of electrical origin f_e. This force has been derived above, and using the new notation yields the mechanical equation of motion for the plunger:

$$-\frac{1}{2}\frac{L_o I^2}{g(1+x/g)^2} = M\frac{d^2x}{dt^2} + D\frac{dx}{dt} + K_s(x-\ell) \tag{5.134}$$

Since these equations are nonlinear in I and x, they cannot be solved by the Laplace operator method of Appendix A. However, modern computer simulation (e.g., MATLAB/SIMULINK) provides a means of calculating $I(t)$ and $x(t)$ once the numerical parameters for the structure are known. However, one aspect of this model has not been considered. That aspect is the bounce of the plunger at the point where $x=0$ during the initial motion of the plunger. The model for bounce involves the elasticity of the mechanical stop and the damping of the nonmagnetic sleeve. It will be shown in the next chapter that the details of this bounce are not normally relevant to the operation of an automotive solenoid type actuator and will not be further explored here.

This abrupt motion of the movable element is essentially in the form of a mechanical switching action such that the solenoid tends to be either in its rest position as held by the spring (i.e., $x=\ell$) or against the center post (i.e., at $x=0$). The movable element is typically connected to a mechanism that is correspondingly moved by the snap action of this element. Applications of solenoids in automotive electronics include fuel injectors and EGR valves.

FUEL INJECTION

A fuel injector is (in essence) a solenoid-operated valve. The valve opens or closes to permit or block fuel flow to the engine. The valve is attached to the movable element of the solenoid and is switched by the solenoid activation.

Fuel injector signal

Consider an idealized fuel injector as shown in Fig. 5.45, in which the injector is open when the applied voltage is on and is closed when the applied voltage is off.

In this configuration, a solenoid has a movable element in the form of a pintle with a conical tip that fits into a conical section forming a nozzle. A spring holds the pintle such that the nozzle is closed.

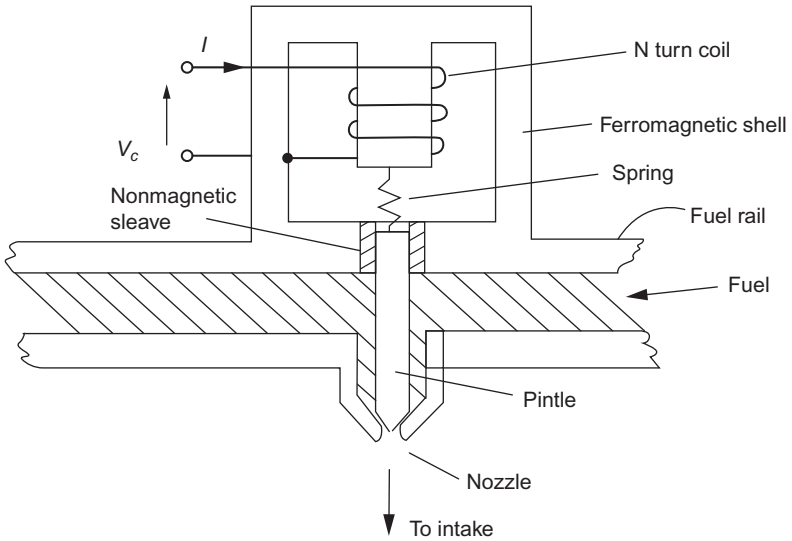


FIG. 5.45 Simplified fuel injector configuration.

Behind this nozzle is a small fuel-filled chamber holding fuel under pressure that is supplied by a tube known as the fuel rail. With no control voltage applied (i.e., $V_c = 0$) and no current I flowing, the spring holds the pintle in a closed position such that no fuel flows through the nozzle. With voltage of sufficient amplitude applied, the solenoid pulls the pintle out of its seat, and fuel flows through the nozzle into the intake system.

Once the pintle is pulled fully toward the solenoid center post, the fuel flow rate through the nozzle is constant for a given regulated fuel pressure and nozzle geometry. Therefore, except for brief transient periods, the quantity of fuel injected into the airstream is proportional to the time the valve is open. The control current that operates the fuel injector is pulsed on and off to deliver precise quantities of fuel.

For most contemporary vehicles, fuel injection takes place in the intake port for each cylinder such that the fuel spray is directed along with intake air flowing past the intake valve during the intake stroke. The control voltage V depicted in Fig. 5.46 is the terminal voltage applied to the fuel injector by the electronic engine control system. Fig. 5.46A and B depicts idealized binary-valued voltage levels that are "on" or "off."

However, it has been shown above that the terminal voltage of a solenoid is characterized by a nonlinear model and the plunger/pintle is similarly characterized by a nonlinear dynamic model. On the other hand, the actual opening and closing pintle/plunger transient response normally represents a relatively short period compared with the "on" time t even under idle conditions (i.e., low duty cycle). In the idealized situation depicted in Fig. 5.46, the fuel is assumed to flow at an essentially constant rate (i.e., $\dot{M}_f$ = constant) for a constant fuel rail pressure. In this situation, the mass of fuel delivered to a cylinder during any given engine cycle $M_f(k)$ is given by

$$M_f(k) = \dot{M}_f t_k$$

where t_k = "on" time for the kth engine cycle. For a pulse train fuel injector control voltage signal, the ratio of "on" time t to the period of the pulse T ("on" time plus "off" time) is called the *duty cycle* δ_{FI}. This is

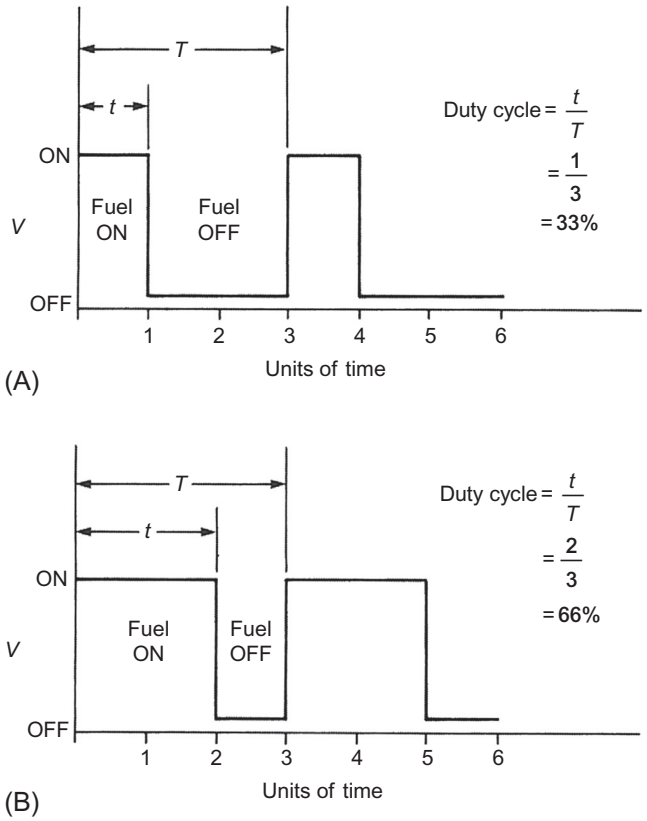


FIG. 5.46 Fuel injector terminal voltage. (A) Duty cycle for relatively high A/F and (B) duty cycle for relatively low A/F.

shown in Fig. 5.46. The fuel injector is energized for time t_k to allow fuel to spray from the nozzle into the airstream going to the intake manifold. The injector is de-energized for the remainder of the period. For a constant fuel rail/pressure, the quantity of fuel supplied during the kth engine cycle $[M_f(k)]$ is proportional to δ_{FI}. Therefore, a low duty cycle, as seen in Fig. 5.46A, is used for a relatively high air/fuel ratio (lean mixture), and a high duty cycle (Fig. 5.46B) is used for a relatively low air/fuel ratio (rich mixture).

EXHAUST GAS RECIRCULATION ACTUATOR

In Chapter 4, it was explained that exhaust gas recirculation (EGR) is used to reduce NOx emissions. The amount of EGR is regulated by the engine controller, as explained in Chapter 6. When the correct amount of EGR has been determined by the controller based on measurements from the various engine control sensors, the controller sends an electrical signal to the EGR actuator. Typically, this actuator is a variable-position valve that regulates the EGR as a function of intake manifold pressure and exhaust gas pressure.

Although there are many EGR configurations, only one representative example will be discussed to explain the basic operation of this type of actuator. The example EGR actuator is shown schematically in Fig. 5.47.

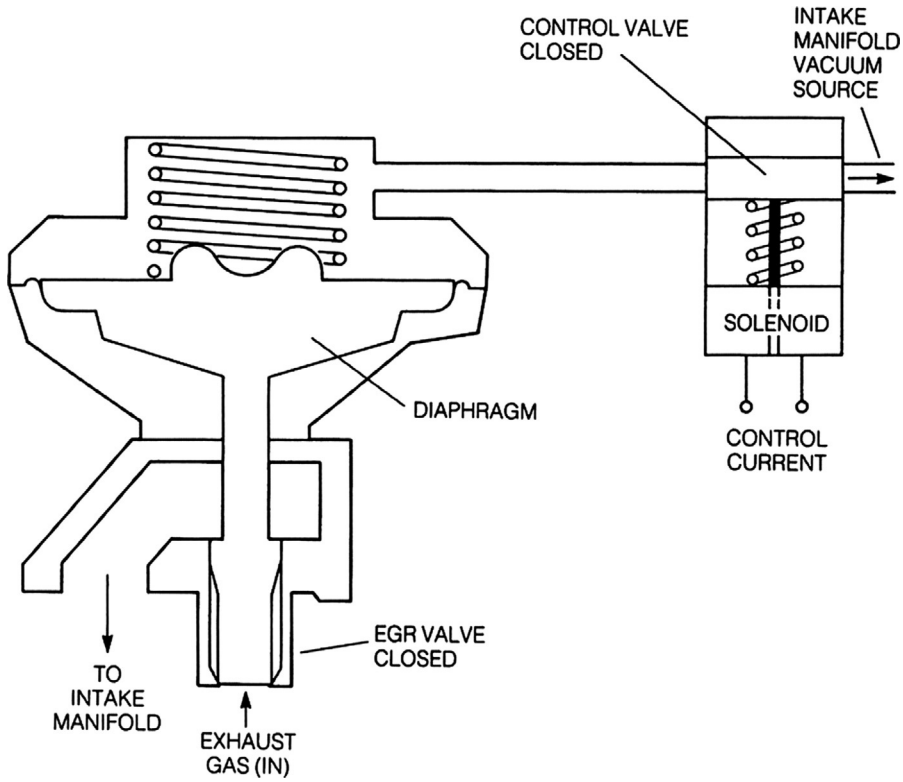


FIG. 5.47 EGR actuator.

This actuator is a vacuum-operated diaphragm valve with a spring that holds the valve closed if no vacuum is applied. The vacuum that operates the diaphragm is supplied by the intake manifold and is controlled by a solenoid-operated valve. This solenoid valve is controlled by the output of the control system.

This solenoid operates essentially in the same manner as that explained in the discussion on fuel injectors. Whenever the solenoid is energized (i.e., by current supplied by the control system flowing through the coil), the EGR valve is opened by the applied vacuum.

The amount of valve opening is determined by the average pressure on the vacuum side of the diaphragm. This pressure is regulated by pulsing the solenoid with a variable-duty-cycle electrical control current. The duty cycle (see discussion on fuel injectors) of this pulsing current controls the average pressure in the chamber that affects the diaphragm deflection, thereby regulating the amount of EGR.

VARIABLE VALVE TIMING

In the discussion of the four-stroke IC engine, it was explained that the intake and exhaust valves were opened by a mechanism that is driven from the camshaft. It was explained that the intake valve is opened during the intake stroke and closed otherwise. Similarly, the exhaust valve is opened during

the exhaust stroke. The exact time during the engine cycle at which these valves open and close is determined by the profile of the camshaft lobes.

The engine performance (including power output and exhaust emissions) is determined partly by the timing of these openings and closings relative to top dead center (TDC) and bottom dead center (BDC) and by the amount of opening (valve lift). It has long been known that optimal cam timing and lift vary with engine operating conditions (i.e., load and RPM). The design of a cam profile has been a compromise that yields acceptable performance over the entire engine operating envelope.

A long-sought goal for the four-stroke IC engine has been the ability to continuously vary valve timing and lift to achieve optimum performance at all operating conditions. In this chapter, variation in the opening and closing of valves relative to a fixed point in the engine cycle (e.g., TDC) is termed variable valve phasing (VVP). It is appropriate to use such a term since the relative time of occurrence of multiple events in any cyclic process is often called phase.

After a considerable development period, various mechanisms have come into production automotive engines for varying valve phasing under electronic control. Significant improvements in volumetric efficiency are possible with VVP. For example, if the exhaust valve closing is delayed relative to BDC and relative to intake valve opening, there is a portion of the cycle in which both valves are open simultaneously (known as valve overlap). The gas dynamics of the exhaust gas leaving the cylinder and intake air entering the cylinder are such that volumetric efficiency is improved by this valve overlap. The optimum overlap varies with operating conditions, and electronic control (with a suitable actuator) is required to achieve this optimum. VVP can be achieved by regulating the timing of either or both the exhaust or intake valves.

A representative example of the mechanism for valve phasing is depicted in Fig. 5.48. Fig. 5.48A is a front view of the engine. Both camshafts are driven via sprocket gears that are, in turn, driven by a sprocket gear mounted at the end of the crankshaft. These sprocket gears can be coupled via a chain (or a timing belt or possibly by a gear system) to a gear on the crankshaft.

In this hypothetical example, each cam sprocket includes a housing within that is a helical spline gear that engages an inner gear connected rigidly to this sprocket gear. Fig. 5.48B shows an exploded view of the helical gear and the camshaft sprocket gear. The camshaft is connected to the helical spline and rotates with it relative to the sprocket as the helical gear moves axially. This conversion of axial displacement to relative rotary motion is responsible for advancing and retarding the exhaust camshaft relative to the exhaust camshaft sprocket.

The helical gear is moved axially by engine lubricating oil acting on a pair of pistons within cylinders located at either end of the helical gear. Oil under pressure is supplied to a pair of sealed chambers, the ends of which are the helical gear (acting as a piston). The axial displacement of the helical gear is regulated by a variable-duty-cycle solenoid-activated control valve that is itself regulated by the engine electronic control system. By regulating the axial displacement of the helical gear, the engine control system controls the relative phasing of the exhaust and intake camshafts.

An alternate cam phasing mechanism is depicted in Fig. 5.48C. This mechanism incorporates extended vanes (V) on the camshaft (C). The vanes are located within recesses in the camshaft gear G. Gear teeth T are circumferentially located around the gear. Although only three gear teeth are shown in the figure, they extend around the periphery of the gear and engage the camshaft drive chain/belt/gears. The vanes, which have the same thickness as the gear, fit tightly into the recesses. Rotation of the camshaft/vanes within the recesses provides the rotational movement that is responsible for variable valve phasing.

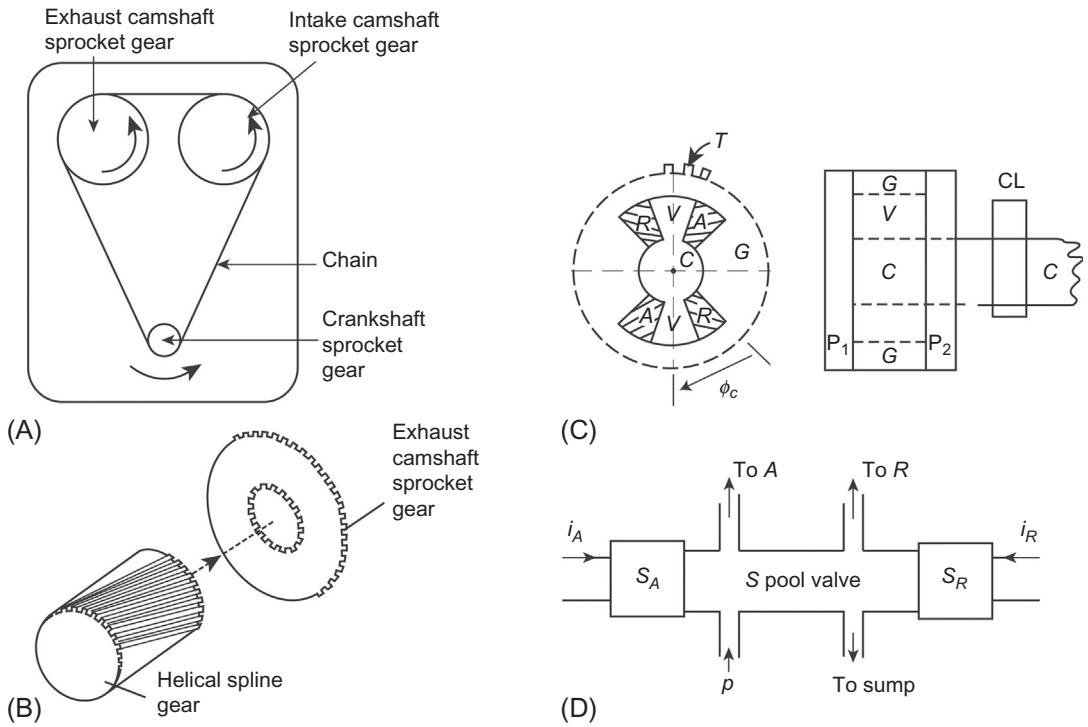


FIG. 5.48 Representative WP mechanisms. (A) Camshaft/crackshaft coupling; (B) Spline VVP adjustment; (C) Differential pressure VVP configuration; (D) VVP control valve.

The recesses are larger circumferentially than the vanes such that chambers *A* and *R* are formed between the gear and the vanes. Oil, under pressure, is supplied to these sets of chambers differentially filling them completely such that the volume of one chamber increases as the other decreases. The pressure in the chambers is maintained by a pair of covering plates P_1 and P_2, which provide sealing of the chambers. A single cam lobe (CL) is depicted on the camshaft extension from the variable valve phasing mechanisms. Such a cam lobe is present for each cylinder operated by the camshaft. Although they are not shown in Fig. 5.48C, there are passageways that supply the oil under pressure.

Fig. 5.48D depicts a spool valve assembly that permits the pressurized oil to be sent to either the *A* or *R* chambers and allows the displaced oil on the opposite side (i.e., the nonpressurized side) to return to the engine oil sump. When pressurized oil is supplied to the *A* chambers and released from the *R* chambers, the camshaft rotates clockwise as shown in Fig. 6.33C, thereby advancing camshaft phase. The reverse is true when the pressurized oil is supplied to the *R* chambers and the oil displaced from the *A* chambers returns to the engine oil sump. Once the desired cam phasing has been achieved, this spool valve is centered, and the oil is blocked from further movement. The camshaft phase is rigidly maintained under so-called "hydraulic lock" conditions. This hydraulic locking is important to maintain the desired phasing because the camshaft itself is subjected to the reaction torque from the valve actuation. The forces acting on each cam lobe include the compression of the valve springs (i.e., the springs that

hold the valves closed) and inertial forces due to acceleration of the valves and any mechanism required to operate them. These latter forces predominate for RPM above a certain level depending on spring rate and the mass of the valve actuation mechanism.

The spool valve actuation is implemented via one or more electromechanical actuators (depending upon system configuration), which are typically solenoids. Fig. 5.48D depicts a pair of solenoids SA and SR. The VVP control comes from the electronic engine control unit (ECU) in the form of currents i_A and i_R. In one implementation, the current(s) that regulate spool valve position are variable-duty-cycle electrical pulse signals, as described earlier in this chapter.

VVP MECHANISM MODEL

Next, an approximate model is developed for the VVP mechanism that has been described qualitatively above. This model is used in Chapter 6 to explain the operation of the VVP under power train control. In the implementation shown in Fig. 5.48C and D, the spool valve is centered via a spring when $i_A = 0$ and $i_R = 0$. In this condition, the mechanism is in hydraulic lock, and the cam phase is constant. Whenever the pulsed current i_A is supplied to S_A, the pressurized oil p is supplied to chamber A, and the displaced oil from chamber R is sent to the oil sump causing the camshaft phase to advance. The displacement of the spool valve within its housing (x_A) is proportional to the duty cycle δ_A of the pulsed current i_A.

The pressure in chamber A denoted p_A (for a given supply pressure from the main oil galley) is proportional to spool valve displacement, which is proportional to δ_A. The model for chamber A pressure is given by

$$p_A = k_{pA}\delta_A$$

The torque acting on the camshaft to advance the camshaft phasing T_c is proportional to the pressure differential between the A and R chambers. Since the oil in chamber R (for nonzero current i_A) is returned to the oil sump, the pressure in chamber R (i.e., p_R) is slightly above oil sump pressure (i.e., atmospheric pressure). The torque acting on the camshaft to advance the cam phase T_c is given by

$$T_c = k_c\delta_A \quad \text{for } i_A \text{ nonzero} \tag{5.135}$$

where k_c is the constant for the geometry and for constant oil supply pressure. Similarly, whenever pulsed current i_R (having duty cycle δ_R) is sent to solenoid SR, the spool valve position is negative, and pressurized oil is sent to chamber R (causing the camshaft phase to retard). The corresponding torque acting on the camshaft is negative and given by

$$T_c = -k_c\delta_R \quad \text{for } i_R \text{ nonzero} \tag{5.136}$$

With proper design, such a system can be modeled as a linear actuator in the following form:

$$T_c = k_c u \tag{5.137}$$

where u is the control signal from electronic engine control system:

$$\begin{aligned} u &= \delta_A \quad i_A \text{ nonzero} \\ &= -\delta_R \quad i_R \text{ nonzero} \end{aligned} \tag{5.138}$$

The torque applied to the camshaft results in dynamic angular movement of the camshaft $\phi_c(t)$ measured relative to the camshaft drive gear (G; see Fig. 5.34C). It is convenient to represent this dynamic motion with the following approximate linear model:

$$J_c\ddot{\phi}_c + B_c\dot{\phi}_c = T_c \tag{5.139}$$

$$= k_c u \tag{5.140}$$

where J_c = moment of inertia of the components that rotate relative to the gear and B_c = viscous damping coefficient for VVP mechanism.

The VVP actuator mechanism can now be modeled as a transfer function $H_p(s)$ (see Appendix A), which is given by

$$H_p = \frac{\phi_c(s)}{u(s)} = \frac{k_a}{s(s+s_o)} \tag{5.141}$$

where

$$k_a = \frac{k_c}{J_c}$$

$$s_o = \frac{B_c}{J_c}$$

Parameters for a hypothetical VVP mechanism are as follows:

$$k_a = 2600$$
$$s_o = 17$$

The transfer function $H_p(s)$ is the plant for a VVP control system that is incorporated as a function in the engine/power train control system for an engine with VVP. The electronic control for this VVP is described and analyzed in Chapter 6.

ELECTRIC MOTOR ACTUATORS

Perhaps the most important electromechanical actuator in automobiles is an electric motor. Electric motors have long been used on automobiles beginning with the starter motor, which uses electric power supplied by a storage battery to rotate the engine at sufficient RPM that the engine can be made to start running. Motors have also been employed to raise or lower windows and position seats and for actuators on airflow control at idle (see Chapter 6). In recent times, electric motors have been used to provide the vehicle primary motive power in hybrid or electric vehicles.

There are a great number of electric motor types that are classified by the type of excitation (i.e., DC or AC), the physical structure (e.g., smooth air gap or salient pole), and the type of magnet structure for the rotating element (rotor), which can be either a permanent magnet or an electromagnet. However, there are certain fundamental similarities between all electric motors, which are discussed below. Still another distinction between types of electric motors is based upon whether the rotor receives electrical excitation from sliding mechanical switch (i.e., commutator and brush) or by induction. Regardless of motor configuration, each is capable of producing mechanical power due to the torque applied to the rotor by the interaction of the magnetic fields between the rotor and the stationary structure (stator) that supports the rotor along its axis of rotation.

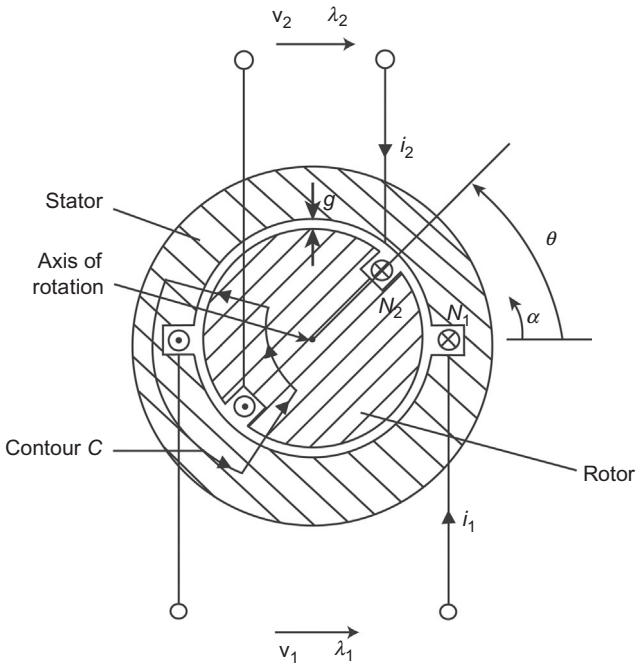


FIG. 5.49 Schematic representation of electric motor.

It is beyond the scope of this book to consider a detailed theory of all motor types. Rather, we introduce basic physical structure and develop analytic models that can be applied to all rotating electromechanical machines. Furthermore, we limit our discussion to linear, time-invariant models, which are sufficient to permit performance analysis appropriate for most automotive applications.

We introduce the structures of various electric motors with Fig. 5.49, which is a highly simplified sketch depicting only the most basic features of the motor.

This motor has coils wound around both the stator (having N_1 turns) and the rotor (having N_2 turns), which are placed in slots around the periphery in an otherwise uniform gap machine. In this simplified drawing, only two coils are depicted. In practice, there are more than two with an equal number in both the stator and rotor. Each winding in either stator or rotor is termed a "pole" of the motor. Both stator and rotor are made from ferromagnetic material having a very high permeability (see discussion above on ferromagnetism). It is worthwhile to develop a model for this simplified idealized motor to provide the basis for an understanding of the relatively complex structure of a practical motor. In Fig. 5.49, the stator is a cylinder of length ℓ, and the rotor is a smaller cylinder supported coaxially with the stator such that it can rotate about the common axis. The angle between the planes of the two coils is denoted θ, and the angular variable about the axis measured from the plane of the stator coil is denoted α. The radial air gap between rotor and stator is denoted g. It is important in the design of any rotating electric machine (including motors) to maintain this air gap as small as is practically feasible since the strength of the associated magnetic fields varies inversely with g. The terminal voltages

of these two coils are denoted v_1 and v_2. The currents are denoted i_1 and i_2, and the magnetic flux linkage for each is denoted λ_1 and λ_2, respectively. Assuming for simplification purposes that the slots carrying the coils are negligibly small, the magnetic field intensity H is directed radially and is positive when directed outward and negative when directed inward.

The terminal excitation voltages are given by

$$v_1 = \dot{\lambda}_1$$
$$v_2 = \dot{\lambda}_2$$

The magnetic flux density in the air gap B_r is also radially directed and is given by

$$B_r = \mu_0 H_r \tag{5.142}$$

where μ_o is the permeability of air.

This magnetic flux density is continuous through the ferromagnetic structure, but because the permeability of the stator and rotor (μ) is very large compared with that of air, the magnetic field intensity inside both the rotor and stator is negligibly small:

$H \simeq 0$ inside ferromagnetic material.

The contour integral along any path (e.g., contour C of Fig. 5.49) that encloses the two coils is given by

$$I_T = \oint_C \bar{H}.\overline{d\ell} = 2gH_r(\alpha) \tag{5.143}$$

The magnetic flux density $B_r(\alpha)$ is also directed radially and is given by

$$B_r(\alpha) = \mu_o H_r(\alpha)$$

This magnetic field intensity is a piecewise continuous function of α as given below:

$$\begin{aligned} 2gH_r(\alpha) &= N_1 i_1 - N_2 i_2 \quad 0 \le \alpha < \theta \\ &= N_1 i_1 + N_2 i_2 \quad \theta < \alpha < \pi \\ &= -N_1 i_1 + N_2 i_2 \quad \pi < \alpha < \pi + \theta \\ &= -N_1 i_1 - N_2 i_2 \quad \pi + \theta < \alpha < 2\pi \end{aligned}$$

The magnetic flux linkage for the two coils λ_1 and λ_2 are given by

$$\begin{aligned} \lambda_1 &= N_1 \int_o^{\pi} B_r(\alpha) \ell R_r d\alpha \\ \lambda_2 &= N_2 \int_{\theta}^{\pi+\theta} B_r(\alpha) \ell R_r d\alpha \end{aligned} \tag{5.144}$$

where R_r is the rotor radius.

It is assumed in the integrals for λ_1 and λ_2 that the so-called fringing magnetic flux outside of the axial length ℓ of the rotor/stator is negligible. Using the concept of inductance for each coil as introduced in the discussion about solenoids, this flux linkage can be written as a linear combination of the contributions from i_1 and i_2:

$$\lambda_1 = L_1 i_1 + L_m i_2 \tag{5.145}$$

$$\lambda_2 = L_m i_1 + L_2 i_2 \tag{5.146}$$

where

$$L_1 = N_1^2 L_o = \text{self inductance of coil 1} \tag{5.147}$$

$$L_2 = N_2^2 L_o = \text{self inductance of coil 2} \tag{5.148}$$

$$L_o = \frac{\mu_o \ell R_r \pi}{2g} \tag{5.149}$$

The parameter L_m is the mutual inductance for the two coils that is defined as the flux linkage induced in each coil due to the current in the other divided by that current and is given by

$$\begin{aligned} L_m &= L_o N_1 N_2 \left(1 - \frac{2\theta}{\pi}\right) \quad 0 < \theta < \pi \\ &= L_o N_1 N_2 \left(1 + \frac{2\theta}{\pi}\right) \quad -\pi < \theta < 0 \end{aligned}$$

The above formulas for these inductances provide a sufficient model to derive the terminal voltage/current relationships and the electromechanical models for motor performance calculations. The self-inductances for each coil are independent of θ, but the mutual inductance varies with θ such that $L_m(\theta)$ is a symmetrical function of θ. It can be formally expanded in a Fourier series in θ having only cosine terms in odd harmonics as given below:

$$L_m(\theta) = M_1 \cos(\theta) + M_3 \cos(3\theta) + M_5 \cos(5\theta) + \cdots \tag{5.150}$$

In any practical motor, there will be a distribution of windings such that the fundamental component M_1 predominates; that is, the mutual inductance is given approximately by

$$L_m \simeq M \cos(\theta) \tag{5.151}$$

For notational convenience, the subscript 1 on M_1 is dropped. Any motor made up of multiple matching pairs of coils in the stator and rotor will have a set of terminal relations in the flux linkages for the stator and rotor λ_s and λ_r, respectively, given by

$$\begin{aligned} \lambda_s &= L_s i_s + M i_r \cos\theta \\ \lambda_r &= L_r i_r + M i_s \cos\theta \end{aligned}$$

The torque of electrical origin acting on the rotor T_e is given by

$$T_e = -\frac{\partial W_{mM}}{\partial \theta}$$

where, for a linear lossless system, the mutual coupling energy W_{mM} is

$$W_{mM} = -i_s i_r L_m(\theta)$$

The torque T_e is given by

$$T_e = -i_s i_r M \sin\theta$$

The mechanical dynamics for the motor are given by

$$T_e = J_r \frac{d^2\theta}{dt^2} + B_r \frac{d\theta}{dt} + C_c \operatorname{sign}\left(\frac{d\theta}{dt}\right)$$

where J_r is the rotor moment of inertia about its axis, B_v is the rotational damping coefficient due to rotational viscous friction, and C_c is the coulomb friction coefficient.

It is of interest to evaluate the motor performance by calculating the motor mechanical power P_m for a given excitation. Let the excitation of the stator and rotor be from ideal current sources such that

$$\begin{aligned} i_s &= I_s \sin(\omega_s t) \\ i_r &= I_r \sin(\omega_r t) \\ \theta(t) &= \omega_m t + \gamma \end{aligned} \tag{5.152}$$

where ω_m is the rotor rotational frequency (rad/s) and γ expresses an arbitrary time-phase parameter. The motor power is given by

$$P_m = T_e \omega_m \tag{5.153}$$

$$= -\omega_m I_s I_r M \sin(\omega_s t) \sin(\omega_r t) \sin(\omega_m t + \gamma) \tag{5.154}$$

This equation can be rewritten using well-known trigonometric identities in the form

$$\begin{aligned} P_m = -\frac{\omega_m I_s I_r M}{4} \{ &\sin[(\omega_m + \omega_s - \omega_r)t + \gamma] + \sin[(\omega_m - \omega_s + \omega_r)t + \gamma] \\ &- \sin[(\omega_m + \omega_s + \omega_r)t + \gamma] - \sin[(\omega_m - \omega_s - \omega_r)t + \gamma] \} \end{aligned} \tag{5.155}$$

The time average value of any sinusoidal function of time is zero. The only conditions under which the motor can produce a nonzero average power are given by the frequency relationships below:

$$\omega_m = \pm\omega_S \pm \omega_r \tag{5.156}$$

For example, whenever $\omega_m = \omega_s + \omega_r$, the motor time average power Pm_{av} is given by

$$P_{m_{av}} = \frac{\omega_m I_s I_r M}{4} \sin\gamma \tag{5.157}$$

In such a motor, an equilibrium operation will be achieved when $P_{m_{av}} = P_L$ where $P_L =$ load power. Thus, the phase between rotor and stator fields is given by

$$\sin\gamma = \frac{4P_L}{\omega_m I_s I_r M} \tag{5.158}$$

provided

$$P_L \leq \frac{\omega_m I_s I_r M}{4} \tag{5.159}$$

The above frequency conditions (Eq. 5.156) are fundamental to all rotating machines and are required to be satisfied for any nonzero average mechanical output power. Each different type of motor has a unique way of satisfying the frequency conditions. We illustrate with a specific example, which has been employed in certain hybrid vehicles. This example is the induction motor. However, before proceeding with this example, it is important to consider an issue in motor performance. Normally, electric motors that are intended to produce substantial amounts of power (e.g., for hybrid vehicle application) are polyphase machines; that is, in addition to the windings associated with stator excitation, a polyphase machine will have one or more additional sets of windings that are excited by the same frequency but at different phases. Although three-phase motors are in common use, the analysis of a two-phase

induction motor illustrates the basic principles of polyphase motors with a relatively simplified model and is assumed in the following discussion.

TWO-PHASE INDUCTION MOTOR

A two-phase motor has two sets of windings displaced at 90 degrees in the θ direction and excited by currents with a 90 degrees phase for both stator and rotor. A so-called balanced two-phase motor will have its coil excited by currents i_{as} and i_{bs} for phases a and b, respectively, where

$$\begin{aligned} i_{as} &= I_s \cos(\omega_s t) \\ i_{bs} &= I_s \sin(\omega_s t) \end{aligned} \tag{5.160}$$

The rotor is also constructed with two sets of windings displaced physically by 90 degrees and excited with currents i_{ar} and i_{br} having 90 degrees phase shift:

$$\begin{aligned} i_{ar} &= I_r \cos(\omega_r t) \\ i_{br} &= I_r \sin(\omega_r t) \end{aligned} \tag{5.161}$$

A two-phase induction motor is one in which the stator windings are excited by currents given above (i.e., i_{as} and i_{bs}). The rotor circuits are short-circuited such that $v_{ar} = v_{br} = 0$, where v_{ar} is the terminal voltage for windings of phase a and v_{br} is the terminal voltage for the b phase. The currents in the rotor are obtained by induction from the stator fields. By extension of the analysis of the single-phase excitation, the terminal flux linkages are given by

$$\begin{aligned} \lambda_{as} &= L_s i_{as} + M i_{ar} \cos\theta - M i_{br} \sin\theta \\ \lambda_{bs} &= L_s i_{bs} + M i_{ar} \sin\theta + M i_{br} \cos\theta \\ \lambda_{ar} &= L_r i_{ar} + M i_{as} \cos\theta + M i_{bs} \sin\theta \\ \lambda_{br} &= L_r i_{br} - M i_{as} \sin\theta + M i_{bs} \cos\theta \end{aligned} \tag{5.162}$$

The torque T_e and instantaneous power P_m for the two-phase induction motor are given by

$$\begin{aligned} T_e &= M[(i_{ar} i_{bs} - i_{br} i_{as}) \cos\theta - (i_{ar} i_{as} + i_{br} i_{bs}) \sin\theta] \\ P_m &= \omega_m M I_s I_r \sin[(\omega_m - \omega_s + \omega_r)t + \gamma] \end{aligned} \tag{5.163}$$

The average power P_{av} is nonzero when $\omega_m = \omega_s - \omega_r$ and is given by

$$P_a = \omega_m M I_s I_r \sin\gamma$$

Since the rotor terminals are short-circuited, we have

$$\frac{d\lambda_{ar}}{dt} = \frac{d\lambda_{br}}{dt} = 0 \tag{5.164}$$

The two rotor currents, thus, satisfy the following equations:

$$\begin{aligned} 0 = R_r i_{ar} + L_r \frac{di_{ar}}{dt} + M I_s \frac{d}{dt}[&\cos(\omega_s t)\cos(\omega_m t + \gamma) \\ &+ \sin(\omega_s t)\sin(\omega_m t + \gamma)] \end{aligned} \tag{5.165}$$

$$\begin{aligned} 0 = R_r i_{br} + L_r \frac{di_{br}}{dt} + M I_s \frac{d}{dt}[&-\cos(\omega_s t)\sin(\omega_m t + \gamma) \\ &+ \sin(\omega_s t)\cos(\omega_m t + \gamma)] \end{aligned} \tag{5.166}$$

where R_r and L_r are the resistance and self-inductance of the two sets of (presumed) identical structure. These equations can be rewritten as

$$L_r\frac{di_{ar}}{dt}+R_r i_{ar}=MI_s(\omega_s-\omega_m)\sin[(\omega_s-\omega_m)t-\gamma] \tag{5.167}$$

$$L_r\frac{di_{br}}{dt}+R_r i_{br}=-MI_s(\omega_s-\omega_m)\cos[(\omega_s-\omega_m)t-\gamma] \tag{5.168}$$

The current i_{br} is identical to i_{ar} except for a 90 degrees phase shift as can be seen from Eqs. (5.167), (5.168). Note that the current for both phases are at frequency ω_r where

$$\omega_r=(\omega_s-\omega_m)$$

Thus, the induction motor satisfies the frequency condition by having currents at the difference between excitations and rotor rotational frequency. The current i_{ar} is given by

$$i_{ar}=\frac{(\omega_s-\omega_m)MI_s}{\sqrt{R_r^2+(\omega_s-\omega_m)^2L_r^2}}\cos[(\omega_s-\omega_m)t-\alpha] \tag{5.169}$$

where

$$\alpha=-\left(\frac{\pi}{2}+\gamma+\beta\right)\text{ and }\beta=\tan^{-1}\left[\frac{(\omega_s-\omega_m)}{R_r}L_r\right] \tag{5.170}$$

The current in phase b is identical except for a 90 degrees phase shift. Substituting the currents for rotor and stator into the equation for torque T_e yields the remarkable result that the this torque is independent of θ and is given by

$$T_e=\frac{(\omega_s-\omega_m)M^2R_rI_s^2}{R_r^2+(\omega_s-\omega_m)^2L_r^2} \tag{5.171}$$

The mechanical output power P_m is given by

$$\begin{aligned}P_m&=\omega_m T_e\\&=\left[\frac{\omega_s^2M^2I_s^2}{(R_r/s)^2+\omega_s^2L_r^2}\right]\left(\frac{1-s}{s}\right)R_r\end{aligned}$$

where s is called slip and is given by

$$s=\frac{\omega_s-\omega_m}{\omega_s} \tag{5.172}$$

The induction machine has three modes of operation as characterized by values of s. For $0<s<1$, it acts as a motor and produces mechanical power. For $-1<s<0$, it acts like a generator, and mechanical input power to the rotor is converted to output electrical power. For $s>1$, the induction machine acts like a brake with both electrical input and mechanical input power dissipated in rotor ${i_r}^2R_r$ losses. Because of its versatility, the induction motor has great potential in hybrid/electric vehicle propulsion applications. However, it does require that the control system incorporates solid-state power switching electronics to be able to handle the necessary currents. Moreover, it requires precise control of the excitation current.

The application of an induction motor to provide the necessary torque to move a hybrid or electric vehicle is influenced by the variation in torque with rotor speed. Examination of Eq. (5.114) reveals that

the motor produces zero torque at synchronous speed (i.e., $\omega_m - \omega_s$). The torque of an induction motor initially increases from its value at $\omega_m = 0$ reaches a maximum torque ($T_{\max}$) at a speed $\omega_m = \omega_m^*$ when

$$0 \leq \omega_m^* \leq \omega_s$$

The torque has a negative slope given by

$$\frac{dT_e}{d\omega_m} < 0 \quad \omega_m > \omega_m^*$$

Normally, an induction motor is operated in the negative slope region of $T_m(\omega_m)$ (i.e., $\omega_m^* > \omega_m < \omega_s$) for stable operation. Equilibrium is reached at a motor rotational speed ω_m at which the motor torque T_e and load torque T_L are equal, that is, $T_e(\omega m) = T_L(\omega m)$.

This point is illustrated for a hypothetical load torque that is a linear function of motor speed such that the load torque is given by

$$T_L = K_L \omega_m \tag{5.173}$$

Fig. 5.50 illustrates the motor and load torques for a load that varies linearly with ω_m.

For convenience of presentation, Fig. 5.50 presents normalized motor torque and load torque normalized to the maximum torque $T_{\max}$ where

$$T_{\max} = \max_{\omega_m} (T_e(\omega_m)) \tag{5.174}$$

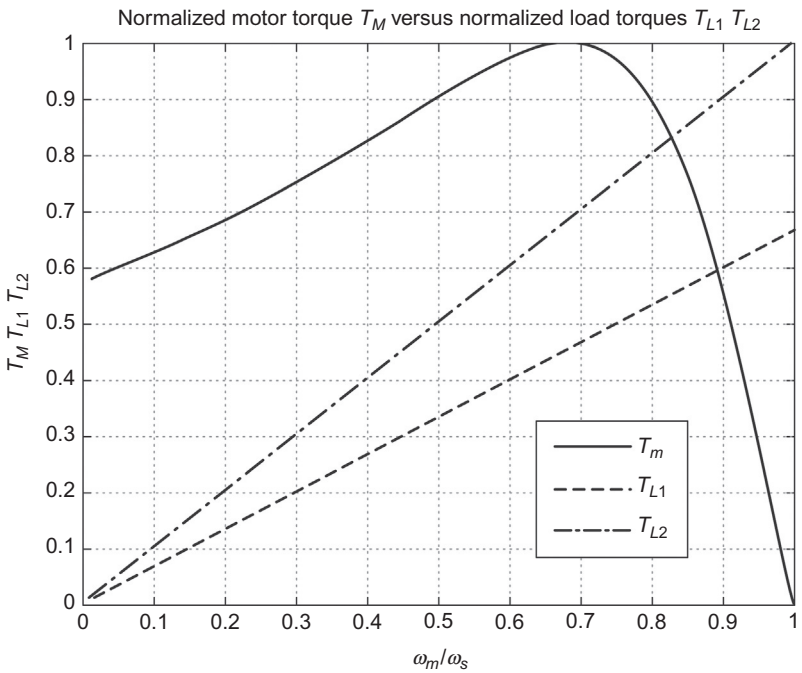


FIG. 5.50 Normalized torque T_m versus normalized load torques T_{L1} T_{L2}.

This maximum occurs at $\omega_m = \omega_m^*$, which, for the present hypothetical normalized example, is given by

$$\frac{\omega_m^*}{\omega_s} \cong 0.68$$

Fig. 5.50 also presents two load torques normalized to $T_{\max}$:

$$T_{L1} = K_{L1}\omega_m / T_{\max}$$
$$T_{L2} = K_{L2}\omega_m / T_{\max}$$

where $K_{L2} > K_{L1}$.

The normalized motor torque T_m is defined by the following:

$$T_m = T_e / T_{max}$$

The operating motor speed for these two load torques are the two intersection points ω_{01} and ω_{02} where

$$T_m(\omega_{01}) = T_{L1}(\omega_{01})$$
$$T_m(\omega_{02}) = T_{L2}(\omega_{02})$$

These two intersection points are the steady-state operating conditions for the two load torques. The higher of the two loads has a steady-state operating point lower than the first (i.e., $\omega_{02} < \omega_{01}$).

Chapter 6 discusses the control of an induction motor that is used in a hybrid electric vehicle. There, the model for load torque versus vehicle operating conditions is developed.

BRUSHLESS DC MOTORS

Next, we consider a relatively new type of electric motor known as a brushless DC motor. A brushless DC motor is not a DC motor at all in that the excitation for the stator is AC. However, it derives its name from physical and performance similarity to a shunt-connected DC motor with a constant field current. An example of this type of motor incorporates a permanent magnet in the rotor and electromagnet poles in the stator as depicted in Fig. 5.51. Traditionally, permanent magnet rotor motors were generally only useful in relatively low-power applications. Recent development of some relatively powerful rare-earth magnets and the development of high-power switching solid-state devices have substantially raised the power capability of such machines.

The stator poles are excited such that they have magnetic N and S poles with polarity as shown in Fig. 5.51 by currents I_a and I_b. These currents are alternately switched on and off from a DC source at a frequency that matches the speed of rotation. The switching is done electronically with a system that includes an angular position sensor (S) attached to the rotor. This switching is done so that the magnetic field produced by the stator electromagnets always applies a torque on the rotor in the direction of its rotation.

The torque $\bar{T}_m$ applied to the rotor by the magnetic field intensity vector $\bar{H}$ created by the stator windings is given by the following vector product:

$$\bar{T}_m = \gamma(\bar{M} \times \bar{H}) \tag{5.175}$$

where $\bar{M}$ is the magnetization vector for the permanent magnet and γ is the constant for the configuration.

The direction of this torque is such as to cause the permanent magnet to rotate toward parallel alignment with the driving field $\bar{H}$ (which is proportional to the excitation current). The magnitude of the torque T_m is given by

$$T_m = \gamma MH \sin(\theta)$$

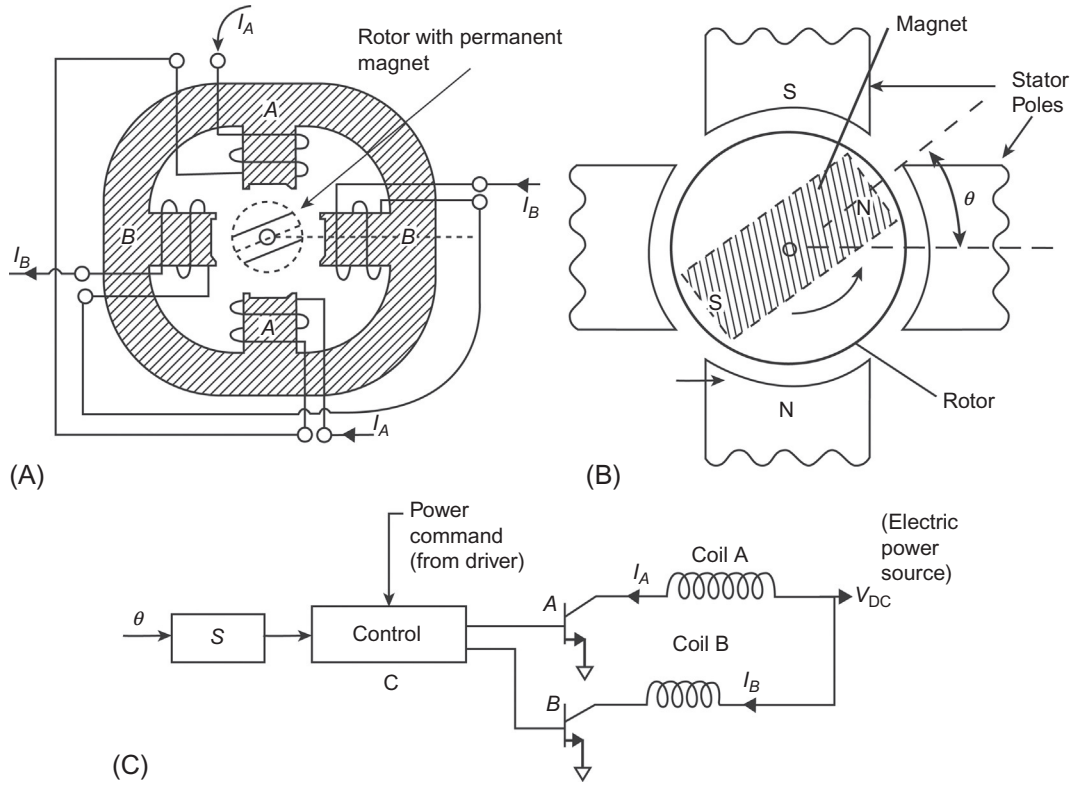


FIG. 5.51 Brushless DC motor. (A) Two-pole motor configuration; (B) Magnetic poles; (C) Motor control block diagram.

where M = magnitude of $\bar{M}$, H = magnitude of $\bar{H}$, and θ = angle between $\bar{M}$ and $\bar{H}$.

If the permanent magnet rotor were allowed to rotate in a static magnetic field, it would only turn until $\theta = 0$ (i.e., alignment).

In a brushless DC motor, however, the excitation fields are alternately switched electronically such that a torque is continuously applied to the rotor magnet. In order for this motor to continue to have a nonzero torque applied, the stator windings must be continuously switched synchronous with rotor rotation. Although only two sets of stator windings are shown in Fig. 5.51 (i.e., two-pole machine), normally there would be multiple sets of windings, each driven separately and synchronously with rotor rotation. In effect, the sequential application of stator currents creates a rotating magnetic field that rotates at rotor frequency (ω_r).

A simplified block diagram of the two-pole motor control system for the motor of Fig. 5.51A and B is shown in Fig. 5.51C. A sensor S measures the angular position θ of the rotor relative to the axes of the magnetic poles of the stator. A controller determines the time for switching currents I_a and I_b on and the duration. The switching times are determined such that a torque is applied to the rotor in the direction of rotation.

At the appropriate time, transistor A is switched on, and electric power from the onboard DC source (e.g., battery pack) is supplied to the poles A of the motor. The duration of this current is regulated by

controller C to produce the desired power (as commanded by the driver). After rotating approximately 90 degrees, current I_b is switched on by activating transistor B via a signal sent by controller C.

The rotor permanent magnet is equivalent to an electromagnet with DC excitation (i.e., $\omega_r = 0$). The frequency at which the currents to the stator coils are switched is always synchronous with the speed of rotation. Thus, the frequency condition for the motor is satisfied since $\omega_s = \omega_m$. This speed is determined by the mechanical load on the motor and the power commanded by the controller. As the power command is increased, the controller responds by increasing the duration of the current pulse supplied to each stator coil. The power delivered by the motor is proportional to the fraction of each cycle that the current is on (i.e., the so-called duty cycle).

STEPPER MOTORS

The configuration of Fig. 5.51 is similar in form to another important motor having automotive applications, which is called a stepper motor. Normally, a stepper motor has application where torque loads are relatively low. Chapter 6 discusses the application of a stepper motor in an engine idle speed control system. In most cases, the stepper motor output employs a reduction gear system in which the gear output shaft rotates at only a fraction of the stepper motor output shaft.

A stepper motor of the configuration depicted in Fig. 5.51 has excitation currents i_A and i_B that are sequences of nonoverlapping pulses. The relative phasing of the pulses determines the direction of motor rotation. The motor rotates a fixed angular increment for each pair of pulses i_A and i_B. Very precise angular position control is obtained for a stepper motor by the number and relative phasing of pairs of such pulses. A control system can advance the load placed on the stepper motor-gear system by a specified amount via the number of output pulses sent to the motor. Feedback via a position sensor of the load movement can be used in conjunction with the output pulses to assure the desired displacement of the load object on the motor/gear system.

The speed of motion of the output shaft is proportional to the pulse frequency of the sequences of pulses on i_A and i_B. However, any such stepper motor has an upper bound on this speed such that the driving pulses are nonoverlapping in time.

IGNITION SYSTEM

The equivalent of an actuator for the ignition system on an engine is the combination of the spark plug, the ignition coil, and driver electronic circuits. This is the subsystem that receives the electrical signal from the engine controller and delivers as its output the spark that ignites the mixture near the end of the compression stroke.

Fig. 5.52 is a block diagram schematic drawing illustrating this subsystem. The primary circuit of the coil (depicted as the left portion P of the coil in Fig. 5.52) is connected to the battery and through a power transistor to ground. For convenience, the collector, emitter, and base are denoted C, E, and B, respectively (see Chapter 2). The coil secondary S is connected to one or more spark plugs, as explained in Chapter 6. A model for the operation of the ignition system is developed next.

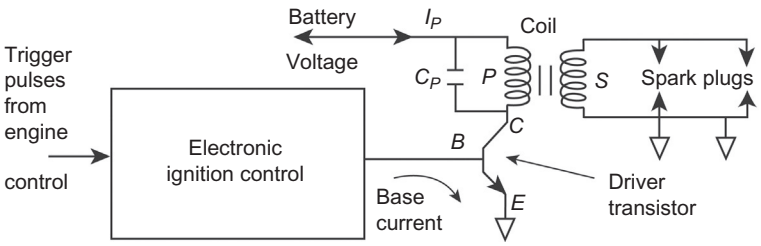


FIG. 5.52 Electronic ignition block diagram.

IGNITION COIL OPERATIONS

The ignition coil is a structure in which a pair of windings (primary P and secondary S) is wound around a ferromagnetic core. This core forms a closed magnetic path linking (ideally) all P and S turns. In contemporary automotive electronic systems, there is often a single coil for each spark plug or for each pair of spark plugs.

Fig. 5.53 depicts a functional model for the ignition system in which the ignition coil is represented by a structure having the topology of a transformer. Denoting the number of turns in the secondary N_s and in the primary N_p for any practically useful ignition coil $N_s \gg N_p$. Although this figure depicts a single spark plug and coil, there must be, of course, a separate circuit for each cylinder (or pair of cylinders).

The engine control unit (ECU) controls the operation of the ignition system via a control signal (e_b) that is applied to the base of transistor Q. Whenever base current $i_b = 0$, the transistor is in a cutoff condition and its collector current $i_c \cong 0$. At the appropriate time (as determined from angular position measurements of the crankshaft and camshaft), the ECU outputs a signal that causes the transistor to switch from cutoff to saturation (see Chapter 2). In saturation, the transistor emitter/collector resistance $R_{ec} = R_{on}$ where R_{on} is a small but nonzero resistance. The collector current I_p that flows under saturation conditions can be shown to satisfy the following differential equation:

$$V_b = R_{on} I_p + L_p \frac{dI_P}{dt} \tag{5.176}$$

where V_b is the vehicle power bus voltage and L_p is the primary coil inductance. Using the Laplace transform methods of Appendix A, the current I_p through the coil (equal to the collector current) primary can be shown to be

$$I_p(t) = \frac{V_b}{R_{on}}\left(1 - e^{-t/\tau_c}\right)$$

where

$$\tau_c = L_p / R_{on}$$

Normally, the ECU will generate a control signal e_b of sufficient duration that I_p has essentially reached the steady-state value of V_b/R_{on}. The duration of this control signal is known as "dwell." The end of this dwell period corresponds to the time that spark is to occur for the given cylinder. At this time, the ECU switches off the control signal, and the coil primary current drops rapidly to 0 (as influenced by the capacitor C_p in Fig. 5.52).

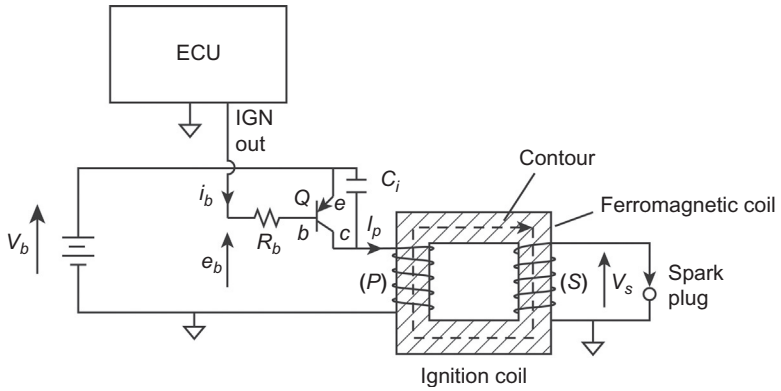


FIG. 5.53 Simplified electronic ignition circuit.

The physical process of creating the spark is the very large coil secondary voltage V_s that is given by

$$V_s = N_s \frac{d\Phi}{dt}$$

where Φ is the total magnetic flux through the core, magnetically linking the N_s turns of the coil secondary.

From the discussions of magnetic field theory for various magnetic sensors and actuators earlier in this chapter, it was shown that the lines of constant magnetic flux density within the ferromagnetic coil core are parallel to the contour C depicted in Fig. 5.53. It can be shown that the total magnetic flux is proportional to the coil primary current I_p. At the end of the dwell period, this flux will have reached a saturation value Φ_s given approximately by

$$\Phi_s \cong \frac{\mu_c A_c \mathrm{V}_b}{\ell_c R_{on}} N_p$$

where μ_c is the core permeability, ℓ_c is the distance around contour C, N_p is the number of turns of the primary coil, and A_c is the core cross-sectional area (normal to C).

The secondary voltage ($V_s(t)$) is a short-duration, very large peak amplitude voltage pulse. The large amplitude of this pulse results from the relatively large value for N_s and the very large time rate of change of Φ when the transistor is "switched" off. The capacitor C_i, which is shown in Fig. 5.53, is partly responsible for the very large rate of change of I_p at the end of dwell.

The generation of the high voltage necessary for ignition based upon magnetic induction using a coil such as is described above has been used to create the ignition spark in various circuits since the earliest days of the four-stroke spark-ignited IC engine. This method is likely to continue well into the future for this engine type.

With the background in sensors and actuators from this chapter, it is now possible to discuss the various automotive control systems. Separate chapters are devoted to each of the vehicular technology areas.

CHAPTER

DIGITAL POWERTRAIN CONTROL SYSTEMS

6

CHAPTER OUTLINE

Introduction 272
Digital Engine Control 272
Digital Engine Control Features 274
Control Modes for Fuel Control 277
- Engine Start 278
- Open-Loop Mode 278
- Acceleration/Deceleration 278
- Idle Mode 279
- Engine Control Configuration 279
- Engine Crank 281
- Engine Warm-Up 281
- Open-Loop Control 283
- Closed-Loop Control 284
- Acceleration Enrichment 288
- Deceleration Leaning 289
- Idle Speed Control 289

Discrete Time Idle Speed Control 291
EGR Control 294
Variable Valve Timing Control 296
Turbocharging 302
Direct Fuel Injection 306
Flex Fuel 308
Electronic Ignition Control 309
- Closed-Loop Ignition Timing 312
- Spark Advance Correction Scheme 317

Integrated Engine Control System 318
- Secondary Air Management 319
- Evaporative Emissions Canister Purge 319
- Automatic System Adjustment 319
- System Diagnosis 320

Summary of Control Modes 321
- Engine Crank (Start) 321

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-12-810434-7.00006-5

Copyright © 2017 Elsevier Inc. All rights reserved.

Engine Warm-Up 321
Open-Loop Control 322
Closed-Loop Control 322
Hard Acceleration 322
Deceleration and Idle 323
Automatic Transmission Control 323
Torque Converter Lock-Up Control 329
Differential and Traction Control 329
Hybrid Electric Vehicle Powertrain Control 331

INTRODUCTION

Traditionally, the term *powertrain* has been used to include the engine, transmission, differential, and drive axle/wheel assemblies. With the advent of electronic controls, the powertrain also includes the electronic control system (in whatever configuration it has). In addition to engine control functions for emission regulation, fuel economy, and performance, electronic controls are also used in the automatic transmission to select shifting as a function of operating conditions. Moreover, certain vehicles employ electronically controlled clutches in the differential (transaxle) for traction control. Electronic controls for these major powertrain components can be either separate (i.e., one for each component) or integrated system regulating the powertrain as a unit.

This latter integrated control system has the benefit of obtaining optimal vehicle performance within the constraints of exhaust emission and fuel economy regulations. Each of the control systems is discussed separately beginning with electronic engine control. Then, a brief discussion of integrated powertrain follows. Several new powertrain technologies that were not presented in the seventh edition of this book have been added to this chapter. This chapter concludes with a discussion of hybrid electric vehicle (HEV) control systems in which propulsive power comes from an IC engine or an electric motor or a combination of both. The proper balance of power between these two sources is a complex function of operating conditions and governmental regulations.

DIGITAL ENGINE CONTROL

Chapter 4 discussed some of the fundamental issues involved in electronic engine control. This chapter explores some practical digital control systems. There is, of course, considerable variation in the configuration and control concept from one manufacturer to another. However, this chapter describes representative control systems that are not necessarily based on the system of any given manufacturer, thereby giving the reader an understanding of the configuration and operating principles of a generic representative system. As such, the systems in this discussion are a compilation of the features used by several manufacturers.

In Chapter 4, engine control was discussed with respect to continuous-time representation. In fact, modern engine control systems, such as the ones discussed in this chapter, are digital. A typical engine control system incorporates a microprocessor and is essentially a special-purpose computer (or microcontroller).

Electronic engine control has evolved from a relatively rudimentary fuel control system employing discrete analog components to the highly precise fuel and ignition control achieved through microprocessor-based integrated digital electronic powertrain control. The motivation for development of the more sophisticated digital control systems has been the increasingly stringent exhaust emission and fuel economy regulations that have evolved recently. It has proved to be cost effective to implement the powertrain controller as a multimode computer-based system to satisfy these requirements.

A multimode controller operates in one of many possible modes and, among other tasks, changes the various calibration parameters as operating conditions change in order to optimize performance. To implement multimode control in analog electronics, it would be necessary to change hardware parameters (e.g., via switching systems) to accommodate various operating conditions. In a computer-based controller, however, the control law and system parameters are changed via program (i.e., software) control. The hardware remains fixed, but the software is reconfigured in accordance with operating conditions as determined by sensor measurements and switch inputs to the controller.

This chapter will explain how the microcontroller under program control is responsible for generating the electrical signals that operate the fuel injectors and trigger the ignition pulses. This chapter also discusses secondary functions (including management of secondary air that must be provided to the catalytic converter, EGR regulation, and evaporative emission control) that have not been discussed in detail before.

All digital systems are inherently discrete-time model based. That is, rather than modeling systems or subsystems on a continuous-time basis, all processes are characterized at discrete times t_k where

$$\begin{aligned} t_k &= kT_s \quad k = 1,2,3 \\ T_s &= \text{sample time} \end{aligned} \tag{6.1}$$

The time interval between successive sample times is the period during which the control system performs the necessary computations to perform its function. The theoretical basis for discrete-time system modeling and analysis has been explained in Appendix B. However, as explained in Chapter 4, the majority of automotive control or instrumentation systems employ some analog sensors and actuators (or in the instrumentation case, displays).

In Chapter 5, it was shown that a number of sensors and actuators are analog devices that are modeled as functions of continuous-time t. As described in Appendix B, measurements made by continuous-time sensors are sampled at times t_k to obtain the necessary discrete-time system input. When representing the sampled data from a continuous-time sensor having an output terminal voltage $V_0(t)$, the notation used here to represent the kth sample of V_0 is V_k where

$$V_k = V_0(t_k) \tag{6.2}$$

It is, perhaps, worthwhile at this point to illustrate the operation of a digital control system with a simple example. Although certain automotive control requires measurements from multiple sensors (i.e., with multiple inputs) to perform a specific task, our illustration considers the example of a single-input, single-output (SISO) linear system. Let the input to the controller at time t_k be x_k and the output corresponding to this and other previous inputs be denoted y_k. It should be noted that y_k is output from the digital system at a time delayed from the x_k owing to the nonzero computation time. As explained in

Appendix B, one form for the relationship between the input and output of a linear SISO digital system is given by the recursive algorithm below:

$$y_k = \sum_{m=0}^{M} a_m x_{k-m} - \sum_{n=1}^{N} b_n y_{k-n} \tag{6.3}$$

The coefficients a_m and b_n are chosen by the designer to perform a specific task. It should be noted that for a purely linear system with continuous-time sensor and actuator, it is possible to develop the control function relating input and output using continuous-time techniques. Then, the discrete-time coefficients can be obtained from this continuous-time function by a discretization process as described in Appendix B.

The trend in contemporary automotive electronic systems is to perform multiple control operations using an integrated digital system based upon a microprocessor/microcontroller. Furthermore, it is an aspect of a digital system that nonlinear transformations and/or calculations are handled as well as linear ones. In addition, various components and/or subsystems are interconnected via a digital communication network, which is termed an in-vehicle network (IVN). Chapter 9 presents detailed discussions of various IVNs in use in contemporary vehicles. This type of interconnection architecture as employed in powertrain control is deferred to that chapter.

DIGITAL ENGINE CONTROL FEATURES

Recall from Chapter 4 that one primary purpose of the electronic engine control system is to regulate the mixture (i.e., air-fuel), the ignition timing, and the EGR. Virtually, all major manufacturers of cars sold in the United States (both foreign and domestic) use the three-way catalytic converter for meeting exhaust emission constraints. For such cars with gasoline only fuel, the air/fuel ratio is held as closely as possible to the stoichiometric value of about 14.7 for as much of the time as possible. Ignition timing and EGR are controlled separately to optimize performance and fuel economy.

Fig. 6.1 illustrates the primary components of an electronic engine control system. In this figure, the engine control system is a microcontroller, typically implemented with a specially designed microprocessor or microcontroller and operating under program control. Chapter 3 presents a discussion of the contemporary programming environment for vehicular electronic systems (AUTOSAR). In this chapter, the algorithms for powertrain control are presented. These algorithms are representative of those that are incorporated as program modules. Spark plugs for this four-cylinder example are denoted S.P.

Often, the controller incorporates hardware for the multiply/divide operation and ROM (see Chapter 3). The hardware multiply greatly speeds up the multiplication routines, which are generally cumbersome and slow when implemented by a subroutine in the software. The associated ROM contains the program for each mode and calibration parameters and lookup tables. The microcontroller under program control generates output electrical signals to operate the fuel injectors so as to maintain the desired mixture and ignition to optimize performance. For a given engine output power (as commanded by the driver via the accelerator pedal), the correct mixture is obtained by regulating the quantity of fuel delivered into each cylinder during the intake stroke in accordance with the corresponding intake air mass, as explained in Chapter 4.

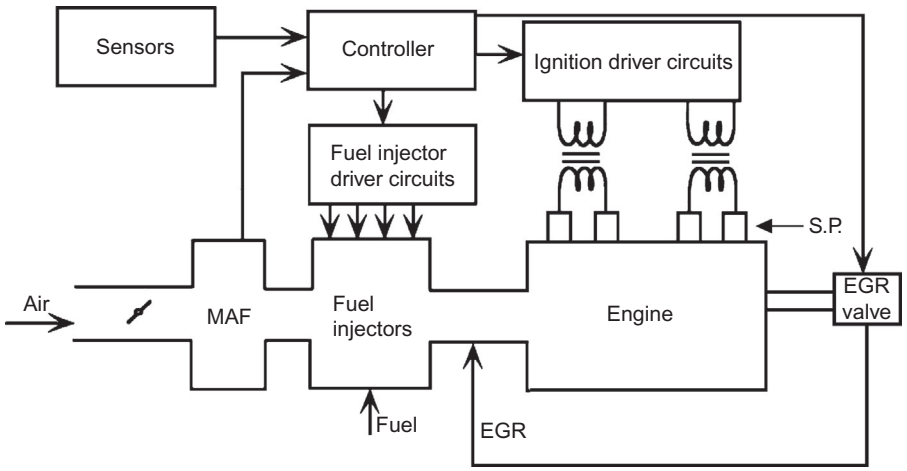


FIG. 6.1 Components of an electronically controlled engine.

With respect to the fuel control function, the digital engine control system obtains a measurement of mass airflow typically using a mass airflow (MAF) sensor. As shown in Chapter 5, the MAF sensor generates an output terminal voltage v_o given by

$$v_o(t) = f_m(\dot{M}_a) \tag{6.4}$$

where $\dot{M}_a$ is the instantaneous mass airflow rate into the engine intake system (kg/s).

As explained in Chapter 5, the function f_m for a representative production MAF sensor is given by

$$v_o(\dot{M}_a) = \sqrt{v_o^2(0) + K_{\text{MAF}}\dot{M}_a}$$

However, a digital fuel control system can invert a nonlinear function to obtain the value $\dot{M}_a$ of mass airflow:

$$\dot{M}_a = f_m^{-1}(v_o) \tag{6.5}$$

As explained in Chapter 4, the intake to the engine includes EGR and air. As will be shown below, the digital engine control system is able to determine the EGR mass flow rate $\dot{M}_{\text{EGR}}$ since it controls the flow of EGR. In certain cases, the EGR rate is determined from a differential pressure sensor (DPS). Thus, the correction for $\dot{M}_{\text{EGR}}$ in the MAF sensor output is a straightforward computation.

An ideal engine control would determine the mass of air drawn into the mth cylinder during the nth engine cycle $M_a(n, m)$. This ideal controller would instantaneously inject fuel with a uniform distribution at the end of the intake process for this cylinder to achieve a uniform stoichiometric mixture throughout the cylinder in preparation for compression ignition and power generation. This ideal fuel injection is being achieved in some contemporary engines by a direct injection, as explained later in this chapter.

A suboptimal fuel injection that is very close to ideal is achieved by well-designed multiport fuel injection in which fuel is injected during the intake stroke with an injector that sprays fuel into the

intake port close to the intake valve. As will be shown later in this chapter, closed-loop fuel control provides sufficient regulation of mixture to meet the strictest emission regulations. It will also be shown later in this chapter that fuel control operates in several possible modes. However, before proceeding to this discussion, it is helpful to explain some of the basic issues in the development of the final system configuration and fuel control algorithms.

In practice, an MAF sensor is placed somewhere in the upstream end of the engine intake system of tubes that direct airflow to the individual cylinders. Typically, this intake system (called "the intake manifold") is designed to achieve as uniform as possible a distribution among all cylinders over the broadest possible operating range. For the present discussion, it is helpful to assume that a uniform distribution of air is achieved for each engine cycle.

At any instant t, the total mass of air pumped into the engine during the previous engine cycle of duration T_e (corresponding to crankshaft rotation through 4π radians) is given by

$$M_{aT}(t) = \int_{\theta_e(t)-4\pi}^{\theta_e(t)} \dot{M}_a(\theta_e)d\theta_e \tag{6.6}$$

where $\theta_e(t)$ is the crankshaft instantaneous angular position at time t, and T_e is the period of an engine cycle at the instantaneous RPM

$$T_e = \frac{120}{\text{RPM}}$$

For simplification and without serious loss of generality, it is convenient to assume that the engine is operating at a steady load and RPM. According to our assumptions, the amount of air drawn into any given cylinder (m) during the nth engine cycle $M_a(n, m)$ is given by

$$M_a(n,m) = \frac{M_{aT}}{M_c} \quad m = 1,2,\ldots,M_c \tag{6.7}$$

where M_c is the number of cylinders.

Note that if the RPM and load are changing but at a slow enough rate, then for at least the period of one cycle, the above model is sufficiently accurate to compute the desired fuel delivery for a stoichiometric mixture.

The fuel mass to be supplied to cylinder m during the nth engine cycle $F(n, m)$ is given by

$$F(n,m) = \frac{M_a(n,m)}{R_{a/f}} \tag{6.8}$$

where $R_{a/f}$ is the desired ratio of mass of air to mass of fuel. As explained below, the correct $R_{a/f}$ depends upon the control operating mode. It is desirable that $R_{a/f}$ for gasoline fuel be at stoichiometry (i.e., $R_{a/f}=14.7$) for as much of the engine-operating period as possible for optimum exhaust emission regulation.

As explained in Chapter 5, fuel delivery in contemporary engines is provided by fuel injectors. It should be recalled that a fuel injector is a solenoid-operated valve that is opened by an electrical control signal at the proper time in the engine cycle for a period of time $\tau_f(n, m)$ (for cylinder m during cycle n) that is computed in the digital engine control system. It was also explained in Chapter 5 that fuel under a regulated pressure is available on the upstream side of the fuel injector valve via the fuel rail.

The fuel flow rate $\dot{M}_f$ is a function of the fuel rail pressure and the open area of the valve and the displacement of the pintle by the solenoid. These latter two parameters are fixed by the structure of the fuel injector. The quantity of fuel delivered by the fuel injector $F(n, m)$ for the mth cylinder during the nth engine cycle is given by

$$F(n,m) = \int_{t_{n,m}}^{t_{n,m}+\tau_F(n,m)} \dot{M}_f dt \tag{6.9}$$

where $t_{n,m}$ is the beginning time of fuel delivery control binary signal, $t_{n,m}+\tau_F(n, m)$ is the end of fuel injection period, and $\dot{M}_f$ is the fuel flow rate for fuel injector.

It is common practice in contemporary engine design to place the fuel injector near to the intake valve such that the fuel spray during the fuel injector open period is directed into the cylinder through the intake valve opening. The binary fuel injection control voltage is timed such that fuel is delivered during an optimal portion of the intake stroke.

The fuel injector opening and closing dynamics are sufficiently short except for very small $F(n, m)$ that the fuel delivery is given approximately by

$$F(n,m) \cong \dot{M}_f \tau_F(n,m) \tag{6.10}$$

Although Eq. (6.9) gives the correct calculation of the fuel delivery, for the purpose of simplifying explanations of fuel control, the model given in Eq. (6.10) is sufficiently accurate for discussion of fuel control operation.

It should be noted that for steady load and RPM, typically τ_F should be constant; however, for varying load and accelerating/decelerating engine τ_F may vary with both n and m. Consequently, the notation τ_F retains both indices.

CONTROL MODES FOR FUEL CONTROL

The engine control system is responsible for controlling fuel and ignition for all possible engine-operating conditions. However, there are a number of distinct categories of engine operation, each of which corresponds to a separate and distinct operating mode for the engine control system. The differences between these operating modes are sufficiently great that a different software routine may be used for each. The control system must determine the operating mode from the existing sensor data and call the particular corresponding software routine. We begin with a qualitative survey of system operation in the various control modes and later present formal models.

For a typical engine, there are at least seven different engine-operating modes that affect fuel control, engine crank, engine warm-up, open-loop control, closed-loop control, hard acceleration, deceleration, and idle. The program for mode control logic determines the engine-operating mode from sensor data and timers.

In the earliest versions of electronic fuel control systems, the fuel metering actuator typically consisted of one or two fuel injectors mounted near the throttle plate so as to deliver fuel into the throttle body. These throttle body fuel injectors (TBFIs) were in effect an electromechanical replacement for the carburetor. Requirements for the TBFI were such that they only had to deliver fuel at the correct average flow rate for any given mass airflow rate. Mixing of the fuel and air and distribution to the individual cylinders took place in the intake manifold system.

The more stringent exhaust emission regulations of recent years have demanded more precise fuel delivery than can normally be achieved by TBFI. These regulations and the need for improved performance have led to timed sequential port fuel injection (TSPFI). In such a system, there is a fuel injector for each cylinder that is mounted so as to spray fuel directly into the intake of the associated cylinder (except for direct cylinder injection engines).

For the purposes of the present discussion, fuel delivery is assumed to be TSPFI (i.e., via individual fuel injectors located so as to spray fuel directly into the intake port and timed to coincide with the intake stroke). Airflow measurement is via a MAF sensor. Some engine control systems involve vehicle speed sensors and various switches to identify brake on/off and the transmission gear, depending on the particular control strategy employed. The discussion of engine control begins with a qualitative summary of control modes and then continues with a detailed quantitative explanation of the operation of each.

ENGINE START

When the ignition key is switched on initially or when an electronic engine control calls for start, the mode control logic automatically selects an engine start control scheme that provides the correct temperature-dependent air/fuel ratio required for starting the engine. Once the engine RPM rises above the cranking value, the controller identifies the "engine started" mode and passes control to the program for the engine warm-up mode. This operating mode typically keeps the air/fuel ratio relatively low to prevent engine stall during cool or cold weather until the engine coolant temperature rises above some minimum value. The instantaneous desired air/fuel is a function of coolant temperature and ambient conditions. The particular value for the minimum coolant temperature is specific to any given engine type and, in particular, to the fuel metering system. (Alternatively, in earlier engine control systems, the low air/fuel ratio was maintained for a fixed time interval following start, depending on start-up engine temperature.)

OPEN-LOOP MODE

When the coolant temperature rises sufficiently, the mode control logic directs the system to operate in the open-loop control mode until the EGO sensor warms up enough to provide accurate readings. The condition for transition from open-loop mode to closed loop is detected by monitoring the EGO sensor's output for voltage readings above a certain minimum rich air/fuel mixture voltage set point (see Chapter 5 for EGO sensor voltage characteristics). When the sensor has indicated rich at least once and after the engine has been in open loop for a specific time, the control mode selection logic selects the closed-loop mode for the system. (Note, other criteria may also be used.) The engine remains in the closed-loop mode unless the EGO sensor fails, and the transition in V_{EGO} does not occur for a certain length of time or a hard acceleration or deceleration occurs. If the sensor fails, it is readily detectable, and the control mode logic selects the open-loop mode again. The closed-loop mode is discussed in detail in a later section of this chapter.

ACCELERATION/DECELERATION

During hard acceleration or heavy engine load, the control mode selection logic chooses a scheme that provides a rich air/fuel mixture for the duration of the acceleration or heavy load. This scheme has the capability to provide maximum torque, but depending on driver demand, suboptimal emissions control

and relatively poor fuel economy regulation as compared with a stoichiometric air/fuel ratio may occur. After the need for enrichment has passed, control is returned to either open-loop or closed-loop mode, depending on the control mode logic conditions that exist at that time. During periods of deceleration, the air/fuel ratio might be increased to reduce emissions of HC and CO due to unburned excess fuel. However, enleanment is limited to an air/fuel that avoids excess NO*x* production. During extreme deceleration with a closed throttle, the fuel delivery ceases, and no fuel is supplied to the engine of contemporary vehicles. This process of zero fuel delivery for closed throttle deceleration is discussed in Chapter 7 in the section on vehicle motion control.

IDLE MODE

When idle conditions are present, control mode logic passes system control to the idle speed control mode. In this mode, the engine speed is controlled to reduce engine roughness and stalling that might occur because the idle load has changed due to air-conditioner compressor operation, alternator operation, or gearshift positioning from park/neutral to drive, although stoichiometric mixture is used if the engine is warm. A detailed model and performance analysis of idle speed control is presented later in this chapter. As explained in the introduction, the complete powertrain control has a number of subsystems each of which is presented separately in this chapter. Depending upon the vehicle model, the individual subsystems can have a separate individual digital control, or the control for each subsystem can be implemented in a combined single unit control. For convenience in this chapter, the control units are treated as individual subsystems. We begin with electronic engine control.

ENGINE CONTROL CONFIGURATION

As explained above, in modern engine control systems, the controller is a special-purpose digital computer built around a microprocessor or microcontroller. An exemplary configuration of a typical modern digital engine control system is depicted in Fig. 6.2.

The controller also includes ROM containing the main program (of several thousand lines of code). This ROM is accessed by the engine control system via address bus A and receives data via data bus D (see Chapter 3). There is also a section of ROM containing parameter values for specific control modes and tables of data for various control functions as explained later in this chapter. The sensor signals are connected to the controller via an input/output (I/O) subsystem. Similarly, the I/O subsystem provides the output signals to drive the fuel injectors (shown as the fuel metering block of Fig. 6.2) and to trigger pulses to the ignition system (described later in this chapter). In addition, this microprocessor-based control system includes hardware for sampling and analog-to-digital conversion such that all sensor measurements are in a format suitable for reading by the microprocessor. (*Note*: See Chapter 3 for a detailed discussion of these components.)

With reference to Fig. 6.2, the sensors that measure various engine variables for the most basic control that are depicted in the figure are the following:

Mass airflow sensor (MAF)
Engine temperature as represented by coolant temperature (CT)
One or two heated exhaust gas oxygen sensor(s) (HEGO)
Crankshaft angular position and RPM sensor (CPS)

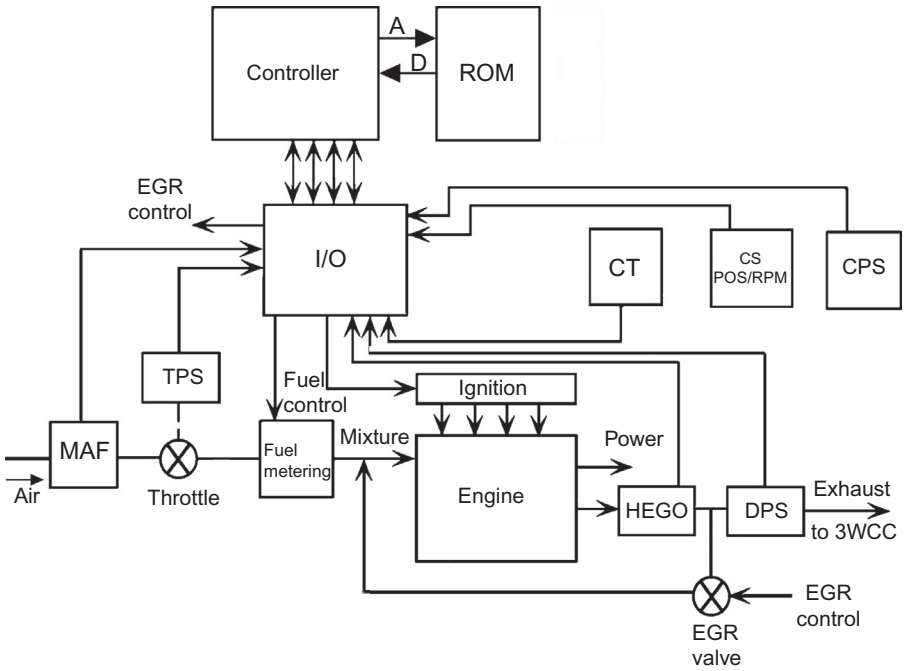


FIG. 6.2 Digital engine control system diagram.

Camshaft position sensor for determining start of each engine cycle (CS POS/RPM)
Throttle position sensor (TPS)
Differential pressure sensor (exhaust to intake) for EGR control (DPS)

Depending upon the individual engine configuration, there are other sensors that are associated with a specific technology that are explained later in this chapter when that technology is discussed (e.g., fuel composition for flex-fuel vehicles).

Other sensors that were used on older model cars and might be used in certain contemporary vehicles that are not given in Fig. 6.2 include the following:

Manifold pressure sensor (MAP)
Inlet air temperature (IAT)
Ambient air pressure (AAP)
Ambient air temperature (AAT)

The control system selects an operating mode based on the instantaneous operating condition as determined from the sensor measurements. Within any given operating mode, the desired air/fuel ratio $(A/F)_d$ is selected. The controller then determines the quantity of fuel to be injected into each cylinder during each engine cycle. This quantity of fuel depends on the particular engine-operating condition and the controller mode of operation, as will presently be explained.

ENGINE CRANK

While the engine is being cranked, the fuel control system must provide an intake air/fuel ratio of anywhere from 2:1 to 12:1, depending on engine temperature. The lowest value for $[A/f]_d$ would be applied for very cold temperatures. The correct air/fuel ratio (i.e., $[A/F]_d$) is selected from an ROM lookup table with interpolation (as explained later in this chapter) as a function of coolant temperature. Low temperatures affect the ability of the fuel metering system to atomize or mix the incoming air and fuel properly to achieve combustion. At low temperatures, the fuel tends to form into large droplets in the air, which do not burn as efficiently as tiny droplets. The larger fuel droplets tend to increase the apparent air/fuel ratio, because the amount of usable fuel (on the surface of the droplets) in the air is reduced; therefore, the fuel metering system must provide a decreased air/fuel ratio to provide the engine with a more combustible air/fuel mixture. During engine crank, the primary issue is to achieve engine start as rapidly as possible. Once the engine is started, the controller switches to an engine warm-up mode.

ENGINE WARM-UP

While the engine is warming up and before the EGO or HEGO sensor has reached the temperature for closed-loop operation, an enriched air/fuel ratio relative to stoichiometry is still needed to keep it running smoothly, but the required air/fuel ratio changes as the temperature increases. Until closed-loop operation is possible, the fuel control system stays in the open-loop mode, but the air/fuel ratio commands continue to be altered due to the temperature changes. The emphasis in this control mode is on rapid and smooth engine warm-up. Fuel economy and emission control may be still a secondary concern. Although the period of open-loop operation is relatively short in contemporary vehicles, a brief discussion of desired A/F based on temperature is included.

A diagram illustrating the lookup table selection of desired air/fuel ratios is shown in Fig. 6.3. Essentially, the measured coolant temperature (T_c) is converted to an address for the lookup table with interpolation as described below. This address is supplied to the ROM table via the system address bus

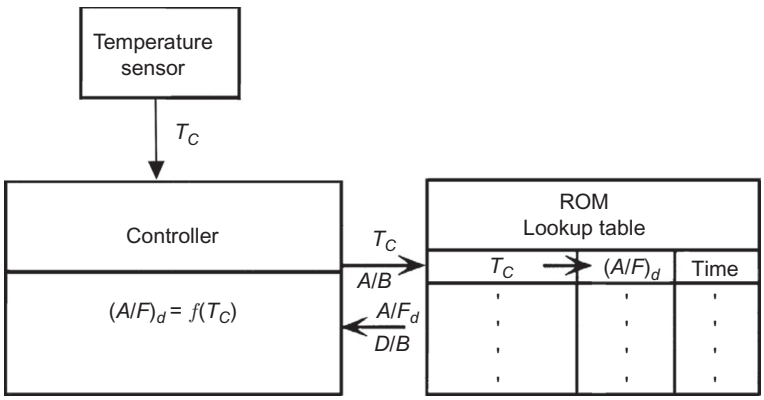


FIG. 6.3 Illustration of table lookup.

(A/B). The data stored at this address in the ROM are desired air/fuel ratio $(A/F)_d$ for the temperature. These data are sent to the controller via the system data bus (D/B).

The term lookup table refers to obtaining an output variable y that is a function of one or more inputs. It provides an alternative to calculation based upon a model (e.g., a polynomial model). It is often applied to empirically obtained data (e.g., from engine mapping) in which the optimum value of a variable (e.g., air/fuel) has been determined from measurements for various values of the independent variables. It is inherently limited to a finite number of discrete points in the relevant range for the independent variables.

On the other hand, during actual engine operation, these same independent variables are continuous and rarely coincide perfectly with the stored values. In this case, the output variable corresponding to these independent variables is obtained by a process called interpolation. This process involves fitting the region between two successive data points with a function (normally linear). We illustrate linear interpolation with a two-dimensional data set. Let y_n be the value of a dependent variable (e.g., air/fuel) at independent data sensor output point x_n where $n = 1,2,\ldots,N$. Let x be a measurement of independent variable (e.g., coolant temperature) for which the corresponding dependent variable y (e.g., desired air/fuel) is sought. Also, let x_m and x_{m+1} be the nearest tabulated data points in the table in which $x_m < x < x_{m+1}$. The corresponding tabulated values for the dependent variable are y_m and y_{m+1}. For linear interpolation, it is assumed that y varies linearly with x over the domain $x_m \leq x \leq x_{m+1}$. The slope S over this domain is given by

$$\begin{aligned} S &= \frac{dy}{dx} \\ &= \frac{y_{m+1} - y_m}{x_{m+1} - x_m} \end{aligned} \tag{6.11}$$

The linearly interpolated value for y is given by

$$\begin{aligned} y &= y_m + S(x - x_m) \\ &= y_m + \left(\frac{y_{m+1} - y_m}{x_{m+1} - x_m}\right)(x - x_m) \end{aligned} \tag{6.12}$$

Alternatively, it is possible to obtain a polynomial model that gives the best fit to measured data in a least-squared error sense. Let an empirical data set be given by $\{x_n \text{ and } y_n, n = 1,2,\ldots,N\}$. The polynomial that best represents this data is of the form

$$y = a_0 + a_1 x + a_2 x^2 + \cdots a_M x^M \quad M < N \tag{6.13}$$

The mean squared error between this polynomial and the data is given by

$$\text{MSE} = \sum_{n=1}^{N} [y(x_n) - y_n]^2 / N \tag{6.14}$$

There are many computer programs for finding the coefficient set $\{a_m, m = 0,1,\ldots,M\}$ such that MSE is minimized. For example, the MATLAB function polyfit (x_n, y_n, M) returns the coefficient set a_m (of order M), which yields the least MSE for the given data set. In this case, the digital engine control can calculate the desired dependent variable for any given measurement of the independent variable x. The choice between table lookup with interpolation and polynomial calculation can be assessed by

the quality of fit of the polynomial to the data given by the MSE for the best polynomial fit and by the relative complexity of the two methods. The set of coefficients for any given data are normally determined during the development of an engine control system. These coefficients are stored in ROM such that the determination of y for any measurement x (during normal engine operation) is readily implemented in the control system using Eq. (6.13) and the stored values for $\{a_m\}$ for the polynomial model method and by Eq. (6.12) for the lookup table and interpolation method.

Returning to the discussion of coolant temperature for setting $(A/F)_d$, there is always the possibility of a coolant temperature failure. Such a failure could result in excessively rich or lean mixtures, which can seriously degrade the performance of both the engine and the three-way catalytic converter (3 WCC). One scheme that can circumvent a temperature sensor failure involves having a time function to limit the duration of the engine warm-up mode. The nominal time to warm the engine from cold soak at various temperatures is known. The controller is configured to switch from engine warm-up mode to an open-loop (warmed-up engine) mode after a sufficient time by means of an internal timer. The on-board diagnosis method for detecting a failed temperature sensor is explained in Chapter 11.

OPEN-LOOP CONTROL

For a warmed-up engine, the controller will operate in an open loop if the closed-loop mode is not available for any reason. For example, the engine may be warmed sufficiently, but the EGO sensor may not provide a usable signal. In any event, as soon as possible, it is important to have a stoichiometric mixture to minimize exhaust emissions.

It was shown above that the quantity of fuel to be delivered to cylinder m during the nth engine cycle can be computed from MAF sensor measurements and can be regulated by means of a fuel injector pulse duration $\tau_F(n, m)$. For the present, it is helpful to assume that intake air is uniformly distributed to all M cylinders. In this case, the fuel injector open duration is given by

$$\tau_F(n, m) = \tau_F(n) \quad \forall m \tag{6.15}$$

This quantity of fuel is actually delivered to each cylinder during the open-loop mode and is often termed the "base pulse duration." Until conditions permit closed-loop mode of fuel control, the fuel quantity is determined from MAF measurements. As a means of denoting open-loop operation, the notation for base pulse duration is $\tau_b(n)$:

$$\tau_F(n)\big|_{(\text{open loop})} = \tau_b(n) \tag{6.16}$$

Corrections of the base pulse width occur whenever any conditions affect the accuracy of the fuel delivery. For example, low-battery voltage might affect the pressure in the fuel rail that delivers fuel to the fuel injectors and the pintle lift of the fuel injectors, thereby reducing $\dot{M}_f$. Corrections to the base pulse width are then made using the actual battery voltage (V_{bat}) in the form of a multiplicative factor $C(V_{\text{bat}})$. In this case, the injection duration $\tau_F(n)$ is given by

$$\tau_F(n) = C(V_{\text{bat}})\tau_b(n)$$

The value of the correction factor, which can readily be determined empirically during engine control development and stored as tabulated data in a lookup table with V_{bat} as an input to the tabulated data, can be determined from the stored values and with interpolation as explained above.

CLOSED-LOOP CONTROL

Perhaps the most important adjustment to the fuel injector pulse duration comes when the control is in the closed-loop mode. In the open-loop mode, the accuracy of the fuel delivery is dependent upon the accuracy of the measurements of the important variables (e.g., MAF). However, any component of a given physical system is susceptible to changes with operating conditions (e.g., temperature) or with time (aging or wear of components). Such failures or degradation of sensor/actuator calibration can adversely affect exhaust emissions in the open-loop mode.

To avoid degraded emission control, it is important for the control system to switch to the closed-loop mode as soon as possible and to remain in this mode for as much of the engine operation as possible. The closed-loop mode can only be activated when the EGO (or HEGO) sensor is sufficiently warmed. Recall from Chapter 5 that for a fully warmed EGO sensor, the output voltage of the sensor is high ($\sim$1 V) when the exhaust oxygen concentration is low (i.e., for a rich mixture relative to stoichiometry). The EGO sensor voltage is low ($\sim$0.1 V) whenever the exhaust oxygen concentration is high (i.e., for a mixture that is lean relative to stoichiometry).

As an illustration of the transition from open-loop mode to closed-loop mode, it is assumed that fuel delivery during the engine warming period is leaner than stoichiometry such that the equivalence ratio $\lambda > 1$. In addition, for simplicity of explanation, it is assumed that engine RPM remains constant. As explained in Chapter 5 for $\lambda > 1$, the EGO sensor voltage is at its low level (e.g., $v_{\text{EGO}} \backsimeq 0.1$). As the engine warms, the air density decreases, and in the illustrative example, fuel delivery remains fixed such that the mixture becomes richer and λ decreases. When the mixture transitions across stoichiometry (i.e., $\lambda = 1$), the EGO sensor output voltage increases to its higher value (e.g., $v_{\text{EGO}} \backsimeq 1$). The controller senses that this transition has occurred and switches to the closed-loop mode. However, the open-loop mode period before a transition to closed loop is relatively short for contemporary vehicles compared with earlier vehicles with the incorporation of heated EGO sensors HEGO (see Chapter 5).

Appendix A presented a discussion of the theory of the closed-loop control of a dynamic system in which a measurement of the dynamic system output variable that is being regulated/controlled is compared with the desired value. The controller produces an input to the plant that changes the output variable in such a way as to minimize the error between actual and desired output. Ideally, control of exhaust emissions would require a sensor for measuring the concentration of each regulated gas component in the engine exhaust as explained in Chapter 4. A large body of theory (both linear and nonlinear) exists that is applicable in the design of a control system provided a sensor exists that can yield an accurate measurement with sufficient bandwidth of the variable being regulated.

However, as explained in Chapters 4 and 5, no cost-effective sensor for measuring these regulated exhaust gases is available for production vehicles. On the other hand, as explained in Chapter 4, the use of a three-way catalytic converter enables tailpipe emissions to be controlled within regulatory limits provided the intake mixture remains sufficiently close to stoichiometry. Furthermore, it was explained that the exhaust gas oxygen concentration changes abruptly as the mixture transitions from rich to lean or from lean to rich at stoichiometry. As explained in Chapter 5, the EGO sensor generates an output voltage that follows exhaust gas concentration. A model for the EGO sensor voltage as a function of exhaust equivalence ratio (λ) was given in Chapter 5.

Unfortunately, a measurement of a switching output variable is compatible only with a limit-cycle controller. None of the linear control theory of Appendix A including design, performance analysis, and stability is applicable to a limit-cycle control system. Although such theory exists for a limit-cycle controller, this theory is beyond the scope of this book. However, as will be shown below, it is possible to

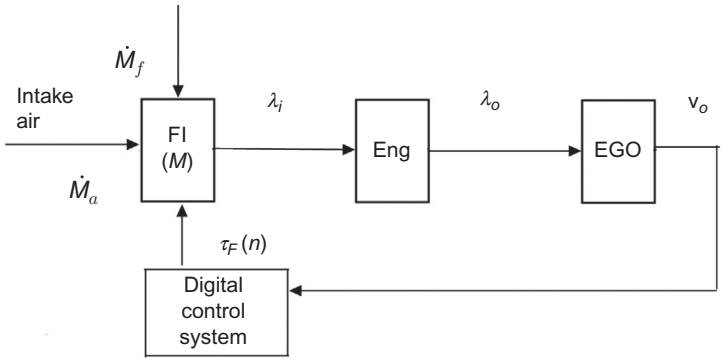


FIG. 6.4 Closed-loop control block diagram.

develop a dynamic simulation model for a limit-cycle fuel control system. Using this simulation, it is possible to investigate the influence of various physical and design parameters on the system performance.

A block diagram depicting the configuration for the closed-loop fuel control system is depicted in Fig. 6.4. In this figure, the engine (Eng) receives fuel and air mixture in the intake system via the *M* fuel injectors (denoted FI—one for each cylinder).

The mixture flowing into the engine is represented by the intake equivalence ratio (λ_i). This mixture is determined by the intake mass airflow rate $\left(\dot{M}_a\right)$ and the fuel injector pulse duration $\tau_F(n)$ for the nth engine cycle as explained above. The exhaust equivalence ratio λ_o can be modeled as a time-delayed version of λ_i where the time delay is modeled below. The exhaust gas oxygen concentration is a function of λ_o such that the output voltage v_o of the EGO sensor can be represented in the ideal case by a binary model as given below. Closed-loop fuel control consists of determining $\tau_F(n)$ as a function of the EGO sensor output voltage. The closed-loop fuel control is illustrated with an exemplary model for computing the pulse duration as a correction to the open-loop base pulse. This pulse duration consists of a base pulse duration $\tau_b(n)$ based on $\dot{M}_a$ measurements (as in open loop) and a closed-loop correction factor ($C_L(n)$) in the representative form

$$\tau_F(n) = \tau_b(n)[1 + C_L(n)] \tag{6.17}$$

One example algorithm for computing this correction factor is a linear combination of a proportional-like term and a discrete-time integral-like term as given below:

$$C_L(n) = \alpha I(n) + \beta P(n) \tag{6.18}$$

where $I(n)$ is the integral term, $P(n)$ is the proportional term, α is the integral gain, and β is the proportional gain.

The integral-like term is determined in the digital control system as a function of the EGO sensor voltage v_o. As explained in Chapter 5, this voltage is a function of exhaust gas oxygen concentration. This voltage can also be characterized in terms of a variable called the exhaust equivalence ratio (λ_o). After a given engine cycle is complete, this exhaust gas equivalence ratio is given approximately by a time-delayed version of λ_i in the form

$$\lambda_o(t) \cong \lambda_i(t - T_e) \tag{6.19}$$

where T_e is the engine cycle time:

$$= \frac{120}{\text{RPM}}$$

With this notation, the EGO sensor voltage is given by

$$\begin{aligned} \text{v}_o(\lambda_o) &= V_H \quad \lambda_o < 1 (\text{rich mixture}) \\ &= V_L \quad \lambda_o > 1 (\text{lean mixture}) \end{aligned} \tag{6.20}$$

where V_H is the EGO sensor "high" level ≈ 1 V and V_L is the EGO sensor "low" level (≈ 0.1 V).

Using the above notation, the integral control algorithm at computation time t_k, $[I(k)]$ is given by

$$\begin{aligned} I(k+1) &= I(k) - 1 \quad \lambda_0(k) < 1 \\ &= I(k) + 1 \quad \lambda_0(k) > 1 \end{aligned} \tag{6.21}$$

In this algorithm, the computation time t_k is given by

$$\begin{aligned} t_k &= kT_S \quad k = 1,2,\ldots \\ T_s &= \text{sample time} \end{aligned}$$

In determining the value of $I(n)$ for the nth engine cycle, the most recent value for $I(t_k)$ is taken. During engine operation, $I(k)$ continuously increases or decreases stepwise with time t_k depending upon λ_0.

The "proportional" term for the nth engine cycle is the average over the K previous samples of the EGO sensor voltage:

$$P(n) = \frac{1}{K}\left[\sum_{k=1}^{K} \text{v}_o(t_n - t_k)\right] - \text{v}_{om} \tag{6.22}$$

where v_{om} is the EGO sensor midrange or time average value (corresponding to stoichiometry). The linear combination above for $C_L(n)$ is representative of closed-loop correction calculations used by a digital fuel control system to modify the base pulse duration.

A fundamental characteristic of a limit-cycle control system is the oscillatory behavior of its control variable. The $C_L(n)$ term continuously oscillates about a nominal value even for a steady engine load and RPM. In the case of the fuel control, the frequency of oscillation and the amplitude of the deviation vary inversely with T_e.

To illustrate the behavior of a limit-cycle controller, a MATLAB/SIMULINK simulation was constructed for the example block diagram of Fig. 6.4. For the purposes of simulation, the equation for $C_L(n)$ is more conveniently represented in terms of $\lambda_i(n)$.

If both sides of the equation can be divided by the measured mass of air during cycle n, a model for λ_i (C_L) can be derived as follows:

$$\frac{\tau_F(n)}{M_a(n)} = \frac{\tau_b(n)}{M_a(n)}(1 + C_L(n)) \tag{6.23}$$

The base pulse duration is derived from a stoichiometric mixture such that we have the following relationship:

$$\lambda_i = \frac{M_a(n)/\tau_F(n)}{M_a(n)/\tau_b(n)} = \frac{1}{1 + C_L(n)} \tag{6.24}$$

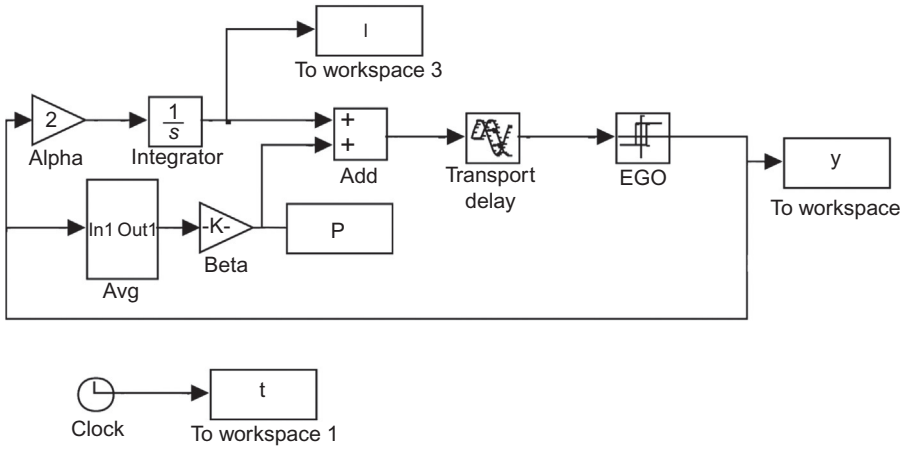


FIG. 6.5 Closed-loop fuel control system.

The simulation with appropriate coefficients α and β results in $C_L(n) \ll 1$ such that

$$\lambda_i \simeq 1 - C_L(n) \tag{6.25}$$

The simulation of the closed-loop fuel control was done with the MATLAB/SIMULINK model depicted in Fig. 6.5.

The simulation has been implemented for deviation in λ_i, which is denoted $\delta\lambda_i$ and is given by

$$\delta\lambda_i = \lambda_i - 1 = -C_L(n) \tag{6.26}$$

The deviation in air/fuel into the engine, which is denoted δA_F, is given by

$$\delta A_F = 14.7\,\delta\lambda_i$$

The sample period was $T_s = 0.01$ s, and the RPM was taken to be about 1000 RPM. The closed-loop control parameters were taken to be $\alpha = 2T_s$, $\beta = 0.025$, and the average for $P(n)$ is computed over $K = 25$ samples.

The simulation block diagram uses an ideal model for the EGO sensor (see Fig. 5.23), (in which the transition from V_H to V_L occurs at $\delta\lambda_i = 0.007$ and from V_L to V_H at $\delta\lambda_i = -0.007$) and implements the control logic of Eq. (6.26). Since the time steps are in multiples of T_s and since the integrator is integrating a constant magnitude with only a sign change, the actual stepwise function of Eq. (6.21) is very closely approximated using the continuous-time integrator (which is simpler to implement in the simulation than the discrete-time version). The hysteresis is ± 0.05 air/fuel ratio, for example, ideal sensor. The time delay is $T_e = 0.067$ s and is implemented in a transport delay SIMULINK block.

Fig. 6.6 is a sample of the waveform where the solid curve is the EGO sensor output voltage, and the dashed curve is the deviation of the air/fuel ratio. Note that this deviation is ± 0.1 air/fuel ratios and that the deviation is increasing during intervals in which $V_{\text{EGO}} = V_H$ corresponding to $\tau_F(n)$ decreasing. In addition, the air/fuel deviation is decreasing during intervals in which $V_{\text{EGO}} = V_L$ corresponding to an increase in $\tau_f(n)$.

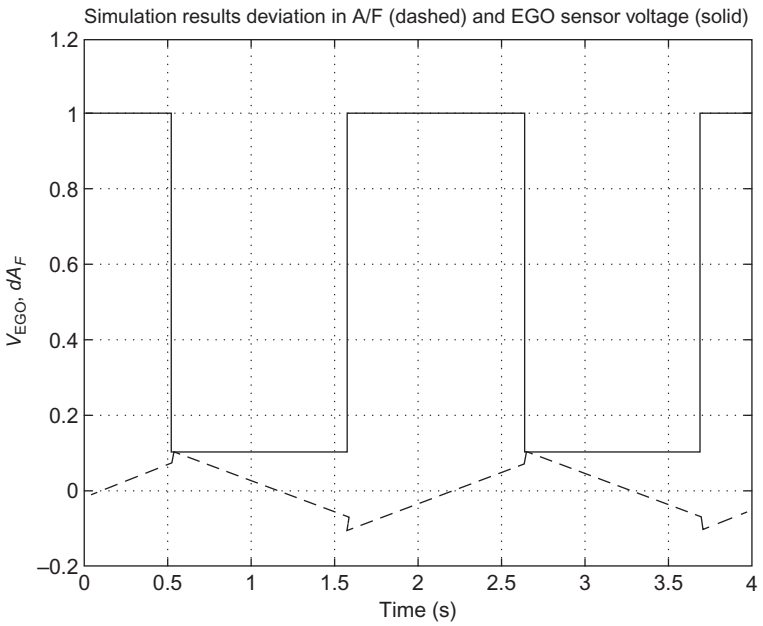


FIG. 6.6 Example of limit-cycle operation.

The time delay between the integral part of $C_L(n)$ and the EGO sensor output is too small to be evident from the figure. Only a short time interval of the waveforms is presented in order to show the detailed response. Also apparent in this figure is the relationship between the exhaust gas concentration and the slope of the integral part of $C_L(n)$. Whenever the EGO sensor voltage is high, corresponding to a rich mixture relative to stoichiometry, the integral component is decreasing, which decreases τ_F causing the mixture to become leaner. Conversely, a low EGO sensor voltage causes the integral part to increase, thereby enriching the mixture.

In Fig. 6.6, it can be seen that the deviation in air/fuel oscillates within ± 0.1 air/fuel ratio of stoichiometry (14.7) corresponding to $\delta A_F = 0$. This performance should be sufficient that the tailpipe gases after passing through the three-way converter should meet government-mandated limits.

ACCELERATION ENRICHMENT

During periods of heavy engine load such as during hard acceleration, fuel control is adjusted to provide an enriched air/fuel ratio to maximize engine torque and very briefly neglect fuel economy and emissions. This condition of enrichment is permitted within the regulations of the EPA as it is only a temporary condition. It is well recognized that hard acceleration is occasionally required for maneuvering in certain situations and is, in fact, related at times to safety. A relatively large increase in throttle angle corresponds to heavy engine load and is an indication that heavy acceleration is called for by the driver. In some vehicles, a switch is provided to detect wide open throttle. The fuel system controller responds by increasing the pulse duration of the fuel injector signal for the duration of the heavy load.

This enrichment enables the engine to operate with a torque greater than that allowed when emissions and fuel economy are controlled. Enrichment of the air/fuel ratio to about 12:1 is sometimes used and corresponds roughly to a maximum engine brake torque.

Alternatively, in an illustrative example, heavy acceleration can be detected from the time derivative of throttle angle θ_T. In discrete-time control systems, the rate of throttle change r_T is given by

$$r_T(k) = \frac{\theta_T(k) - \theta_T(k-1)}{T_S} \tag{6.27}$$

Enrichment is enabled whenever r_T exceeds a predetermined threshold value (r_{Tt}). For $r_T > r_{Tt}$, enrichment is accomplished by increasing τ_F from its normal closed-loop value. For example, τ_F for $r_T > r_{Tt}$ can include an extra term of the following form:

$$\tau_F(r_T) = \tau_b(1 + C_L + F(r_T)) \quad r_T > r_{Tt}$$

where $F(r_T)$ is often an empirically determined function for a given vehicle engine configuration.

DECELERATION LEANING

During periods of light engine load and high RPM such as coasting or deceleration, the engine may operate with a very lean air/fuel ratio to reduce excess emissions of HC and CO. Deceleration is indicated by a sudden decrease in throttle angle or by closure of a switch when the throttle is closed (depending on the particular vehicle configuration). When these conditions are detected by the control computer, it computes a decrease in the pulse duration of the fuel injector signal. The fuel may even be turned off completely for very heavy deceleration. This decrease can be represented by the equation for acceleration in which the function

$$F(r_T) = F_d(r_T) \quad r_T < r_{Td} \tag{6.28}$$

where r_{Td} is a threshold value for r_T below which enleanment is required and where $F_d(r_T)$ is the enleanment function.

IDLE SPEED CONTROL

The idle speed control mode is used to prevent engine stall during idle. The goal is to allow the engine to idle at as low an RPM as possible yet keep the engine from running rough and stalling when power-consuming accessories, such as air-conditioning compressors and alternators, turn on.

The control mode selection logic switches to idle speed control when the throttle angle reaches its zero (completely closed) position as detected by a switch on the throttle that is closed and engine RPM falls below a minimum value. This condition often occurs when the vehicle is stationary. Idle speed is controlled by using an electronically controlled throttle bypass valve, as seen in Fig. 6.6, which allows air to flow around the throttle plate and produces the same effect as if the throttle had been slightly opened such that sufficient $\dot{M}_a$ flows to maintain engine operation.

There are various schemes for operating a valve to introduce bypass air for idle control. One relatively common method for controlling the idle speed bypass air uses a special type of motor called a *stepper motor*. One stepper motor configuration consists of a rotor with permanent magnets and two sets of windings in the stator that is powered by separate driver circuits. The configuration of a stepper motor is similar to that of a brushless DC motor as explained in Chapter 5 (see Fig. 5.36).

Such a motor can be operated in either direction by supplying pulses in the proper phase to the windings as explained in Chapter 5. This is advantageous for idle speed control since the controller can very precisely position the idle bypass valve by sending the proper number of pulses of the correct phasing.

A digital engine control computer can precisely determine the position of the valve in a number of ways. In one way, the computer can send sufficient pulses to close completely the valve when the ignition is first switched on. Then, it can open pulses (phased to open the valve) to a specified (known) position. The physical configuration for the idle speed control is depicted in Fig. 6.7A. A block

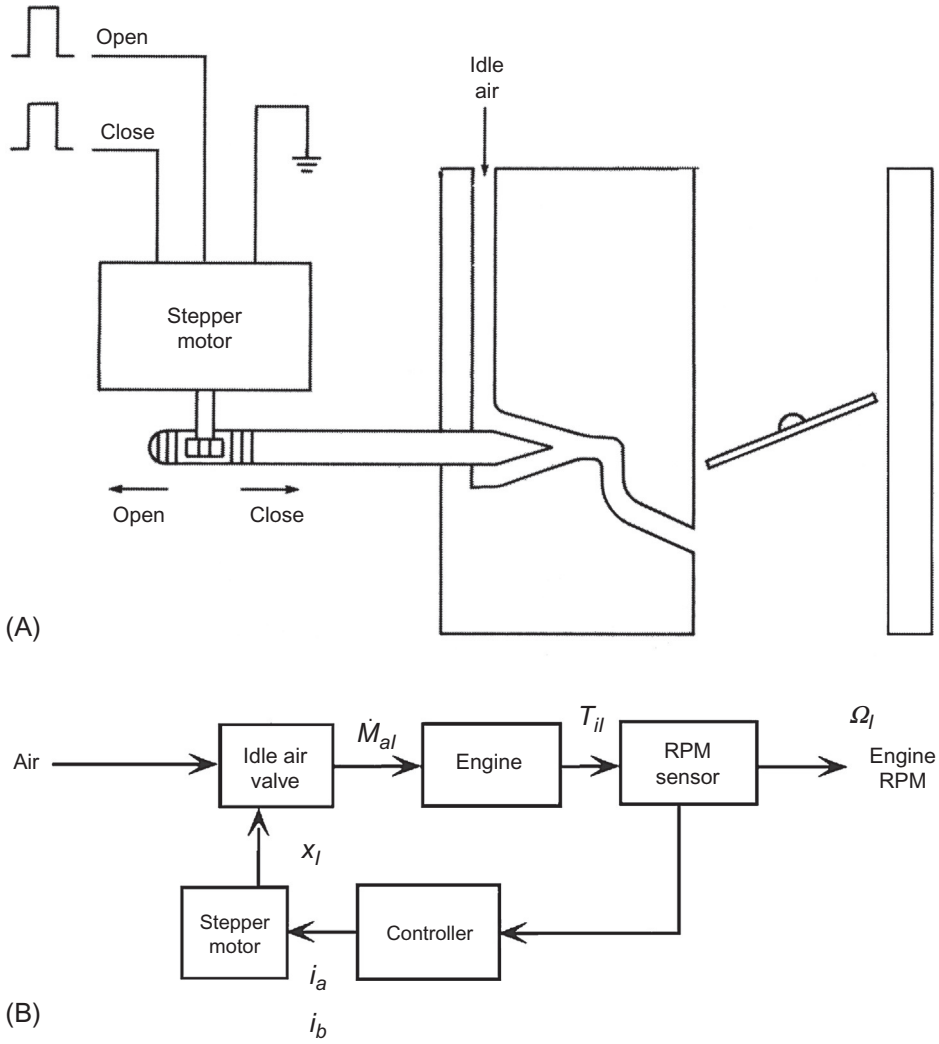


FIG. 6.7 Idle speed control system.

diagram for an exemplary idle speed control is depicting Fig. 6.7B. The variables have the same notation as given in Chapter 4.

In addition, the digital engine control system receives digital on/off status inputs from several power-consuming devices attached to the engine, such as the air-conditioner clutch switch, park-neutral switch, and the battery charge indicator. These inputs indicate the load that is applied to the engine during idle.

DISCRETE TIME IDLE SPEED CONTROL

In Chapter 4, an idle speed control system (ISC) was introduced based upon the continuous-time control theory of Appendix A. As explained in Chapter 4, the purpose of the ISC is to maintain the engine idle speed Ω at a constant (set point) value Ω_s. The ISC is one of the many control modes of the digital engine control system. Since this function is implemented digitally, the ISC is inherently a discrete-time system.

In this section, we consider a digital (i.e., discrete time) implementation of the same ISC that was presented in Chapter 4. Fig. 6.8 is a block diagram of this discrete-time system in which the control subsystem labeled H_c is implemented in the integrated digital electronic engine control system. To be consistent with the continuous-time idle speed control presented in Chapter 4, the stepper motor actuator is assumed to move in sufficiently small steps that the actuator input signal can be represented accurately enough that the output of the controller is modeled as a continuous-time signal $\bar{u}(t)$. However, in practical implementation, the controller outputs pulses to move the stepper motor.

The present discussion is an example of discrete-time control introduced in Appendix B. In this figure, the plant being controlled consists of the engine with the idle air bypass actuator. This plant is an analog system modeled by continuous-time equations. Using the Laplace transform methods of Appendix A, it was shown in Chapter 4 that, for the example, ISC, the plant transfer function $H_p(s)$ is given by

$$H_p(s) = \frac{5000}{s^3 + 35s^2 + 875s + 6250} \tag{6.29}$$

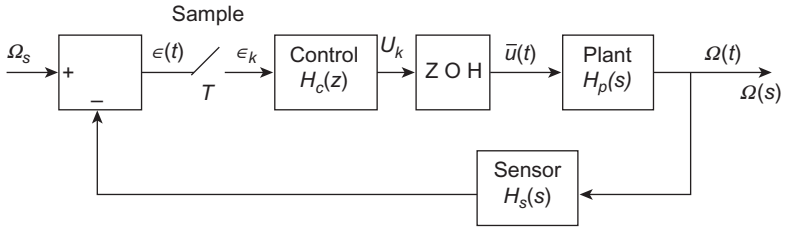


FIG. 6.8 Discrete-time idle speed control block diagram.

The desired idle angular speed (or set point for the controller) is denoted Ω_s in Fig. 6.8. Also depicted in the block diagram of this figure is the actual idle angular speed $\Omega(t)$ or $\Omega(s)$. A measurement of Ω made by the sensor is fed back to the system input forming an error $\in$:

$$\in(s) = \Omega_s - H_s(s)\Omega(s) \tag{6.30}$$

In the example of Chapter 4, it was assumed for computational simplicity that the sensor is ideal such that $H_s(s) = 1$. For the purposes of comparing the continuous-time idle speed control system with the present discrete-time, digital implementation, we make the same assumption here along with assuming the same plant model.

In the present discrete-time implementation, the error is sampled periodically with period T. In accordance with the discrete-time control theory of Appendix B, we assume an ideal sampler/quantizer (i.e., A/D converter) such that the input to the discrete-time control system is $\in_k$:

$$\in_k = \in(kT) \quad k = 1,2,3\ldots \tag{6.31}$$

We further assume that in keeping with the continuous-time system, the control is proportional-integral (PI). The continuous-time model for the control system is given by its operational transfer function

$$H_c(s) = \frac{u(s)}{\in(s)}$$

$$H_c(s) = K_p + \frac{K_I}{s} \tag{6.32}$$

In the time domain, the control variable $u(t)$ can be written as

$$u(t) = K_p \in(t) + K_I \int_o^t \in(t')dt'$$

The discrete-time model for $u(t)$ at sample time t_k (i.e., u_k) is given by

$$u(k) = K_p \in_k + K_I u_{kI}$$

where $\in_k = \in(t_k)$, $t_k = kT$, and $T =$ sample period.

In the PI model, u_{kI} is the discrete-time version of the integral term evaluated at time t_k. There are many ways of approximating the continuous-time integral with a discrete-time version. The trapezoidal integration rule is chosen here. In this method, the integral of $\in(t)$ at time t_k can be approximated by the following:

$$\int_o^{t_k} \in(t)dt \cong \int_o^{t_{k-1}} \in(t)dt + \frac{T}{2}(\in(t_k) + \in(t_{k-1})) \tag{6.33}$$

where the second term approximates the contributions to the integral at t_k by the integral evaluated at t_{k-1} + the area of a trapezoidal area under the function $\in(t)$ from t_{k-1} to t_k. Using this model, we obtain the following recursive equation:

$$u_k = u_{k-1} + \frac{K_I T}{2}(\in_k + \in_{k-1}) \tag{6.34}$$

Taking the z-transform of this equation yields the following expression:

$$u_I(z) = z^{-1}u_I(z) + \frac{K_1 T(1+z^{-1})}{2} \in(z) \tag{6.35}$$

This equation can be rewritten as

$$u_I(z) = \frac{K_I T}{2}\frac{(z+1)}{(z-1)}\in(z) \tag{6.36}$$

It can be shown that the z-transform operational transfer function $H_c(z)$ is given by Eq. (6.33)

$$H_c(z) = \frac{u(z)}{\in(z)}$$

$$H_c(z) = K_p + \frac{K_I T(z+1)}{2(z-1)} = \frac{(K_p + K_I T/2)z + (K_I T/2 - K_p)}{(z-1)} \tag{6.37}$$

The controller outputs a sequence $\{u_k\}$ control signal that is converted to a piecewise continuous-time control signal $\bar{u}(t)$ via the ZOH (see Appendix B) that operates the plant actuator.

It was also shown in Appendix B that the z-transform operational transfer function of the combination ZOH and plant, which is denoted $G(z)$, is given by

$$G(z) = (1 - z^{-1})\mathcal{Z}\left(\frac{H_p(s)}{s}\right) \tag{6.38}$$

As shown in Appendix B, the method of finding the z-transform of $H_p(s)/s$ is first to find the partial fraction expansion of this function and then using the table of Appendix B to find the individual z-transforms of each partial fraction. Then, the desired $G(z)$ is found by combining those terms into a ratio of polynomials in z. A simpler approach for calculating $G(z)$ is to use the Matlab function c2d in the form $G(z) =$ c2d(Hp, T, 'zoh'). This function places the ZOH at the input to Hp and computes the function given above. This method was used to obtain the following $G(z)$ with $T = 0.01$ s.

$$G(z) = \frac{10^{-3}(0.762z^2 + 2.788z + 0.6391)}{(z^3 - 2.629z^2 + 2.3381z - 0.7040)}$$

The z-transform operational transfer function for the forward path $H_F(z)$ is given by

$$\begin{aligned} H_F(z) &= H_c(z)G(z) \\ &= \frac{10^{-3}[0.9982z^3 + 2.8204z^2 - 2.201z - 0.6972]}{z^4 - 3.6286z^3 + 4.9672z^2 - 3.0432z + 0.704} \end{aligned} \tag{6.39}$$

The closed-loop transfer function $H_{CL}(z)$ (as explained in Appendix B) is given by

$$H_{CL}(z) = \frac{H_c(z)G(z)}{1 + H_c(z)G(z)} \tag{6.40}$$

It can be shown that using the parameters of the example of Chapter 4, $H_{CL}(z)$ is given by

$$H_{CL}(z) = \frac{10^{-3}(0.9982z^3 + 2.8204z^2 - 2.201z - 0.6972}{z^4 - 3.629z^3 + 4.9698z^2 - 3.0456z + 0.7040} \tag{6.41}$$

The four poles of $H_{CL}(z)$ are given by

$$\begin{aligned} z_1 &= 0.9250 + 0.1932i \\ z_2 &= 0.9250 - 0.1932i \\ z_3 &= 0.9293 \\ z_4 &= 0.8483 \end{aligned}$$

All four poles are inside the unit circle ($|z| = 1$) in the complex z-plane, so the system is stable.

In Chapter 4, the performance of the continuous-time ISC was examined by computing the step response in which the command speed was changed from 550 to 600 RPM at time $t = 0.5$ s. A similar step change can be determined for the discrete-time ISC by assuming a command input $\Omega_S(t)$ given by

$$\Omega_S(t) = 550 + 50u(t) \tag{6.42}$$

where $u(t) =$ unit step at $t = 0$. The z-transform of the ISC dynamic response to this input is given by

$$\begin{aligned}\Omega(z) &= 550 + 50\frac{z}{z-1}H_{CL}(z)\\ &= 550 + 50y(z)\\ &= 550 + \frac{0.05(0.9982z^4 + 2.8204z^3 - 2.201z^2 - 0.6972z)}{z^5 - 4.622z^4 + 8.5975z^3 - 8.0154z^2 + 3.7496z - 0.7040}\end{aligned} \tag{6.43}$$

The system output at times t_k can be found by writing the partial fraction expansion for the product $y(z)$:

$$y(z) = \frac{zH_{CL}(z)}{z-1} \tag{6.44}$$

As shown in Appendix B, this partial fraction is of the form

$$y(z) = \sum_{m=1}^{5} \frac{\alpha_m}{z - z_m}$$

where α_m is the residue of $y(z)$ at pole z_m and z_m denotes poles of $y(z)$ $m = 1,2,3,4,5$.

The response of the system at time t_k that is denoted y_k was shown in Appendix B (by equating coefficients of z^{-k} on both sides of the above equation) to be given by

$$y_k = \sum_{m=1}^{5} \alpha_m z_m^{k-1}$$

Fig. 6.9 is a plot of $\Omega(t_k)$ where

$$\Omega(t_k) = 550 + 50y_k$$

A comparison of Fig. 4.28 in Chapter 4 with Fig. 6.9 shows that the dynamic performance of the discrete-time digital version of ISC is nearly identical with the corresponding continuous-time system. When the engine is not idling, the idle speed control valve may be completely closed so that the throttle plate has total control of intake air.

EGR CONTROL

A second electronic engine control subsystem involves the control of exhaust gas that is recirculated back to the intake manifold, which is not required on all engines (e.g., see section of this chapter on variable valve timing), but is presented for completeness. Under normal operating conditions, engine cylinder temperatures can reach a point at which NOx is formed during combustion. The exhaust will have NOx emissions that increase with increasing combustion temperature. As explained in Chapter 4,

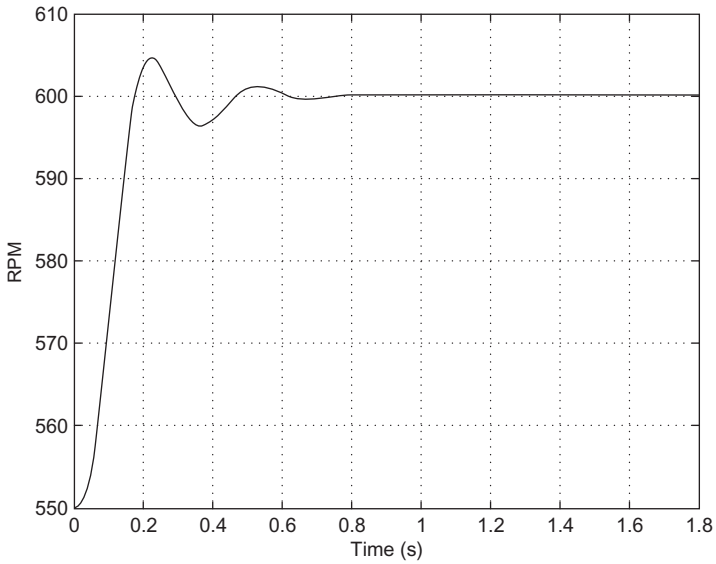


FIG. 6.9 Step response of discrete-time idle speed control.

a small amount of exhaust is introduced into the cylinder to replace some of the normal intake air. This results in lower combustion temperatures, which reduces NO*x* emissions.

The control mode selection logic determines when EGR is turned off or on. EGR is turned off during cranking, cold engine temperature (engine warm-up), idling, acceleration, or other conditions demanding high torque. Since exhaust gas recirculation was first introduced as a concept for reducing NO*x* exhaust emissions, its implementation has gone through considerable change. There are, in fact, many schemes and configurations for EGR realization. We discuss here one method of EGR implementation that incorporates enough features to be representative of all schemes in use today and in the near future.

Fundamental to all EGR schemes is a passageway or port connecting the exhaust and intake manifolds. A valve is positioned along this passageway whose position regulates EGR from zero to some maximum value. In one configuration, the valve is operated by a diaphragm connected to a variable vacuum source. The controller operates a solenoid in a periodic variable-duty-cycle mode. By varying this duty cycle, the control system has proportional control over the EGR valve opening and thereby over the amount of EGR. However, EGR activation also can be done using a motor such as a stepper motor as described in Chapter 5. The solenoid-based EGR actuator has cost advantages over a motor-based system, although manifold vacuum required to operate it varies with engine-operating conditions and is very low at wide open throttle.

In many EGR control systems, the controller monitors the differential pressure between the exhaust and intake manifold via a differential pressure sensor (DPS). With the signal from this sensor, the controller can calculate the valve opening for the desired EGR level. The amount of EGR required is a predetermined function of the load on the engine (i.e., power produced).

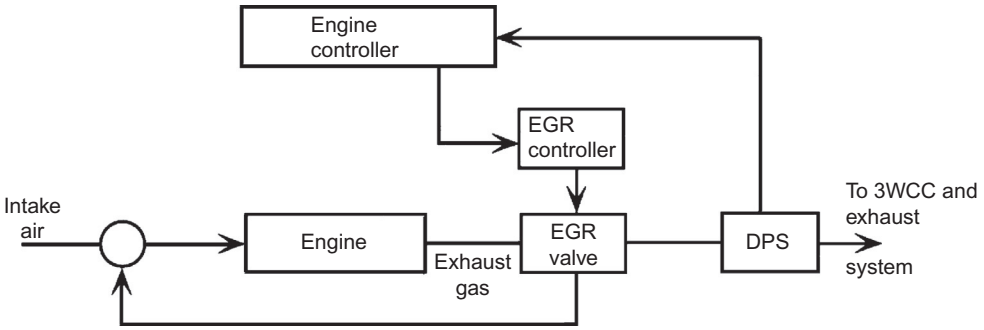


FIG. 6.10 EGR control block diagram.

A simplified block diagram for an exemplary EGR control system is depicted in Fig. 6.10. In this figure, the EGR valve is operated by a solenoid-regulated vacuum actuator (coming from the intake). An explanation of this proportional actuator is given in Chapter 5. The engine controller determines the required amount of EGR based on the engine-operating condition and the signal from the differential pressure sensor (DPS) between intake and exhaust manifolds. The controller then commands the correct EGR valve position to achieve the desired amount of EGR via a variable-duty-cycle actuator signal.

The optimum amount of EGR can be determined empirically as a function of engine-operating conditions. Ideally, closed-loop control of EGR would require, for example, a combustion temperature sensor. Although a cost-effective sensor for directly measuring combustion temperature has not been developed yet, there is a correlation between exhaust gas temperature and combustion temperature. The former is readily measurable with relatively inexpensive sensors. In principle, the amount of EGR could be based upon a closed-loop control system using exhaust gas temperature measurements for a feedback signal.

VARIABLE VALVE TIMING CONTROL

Chapter 4 introduced the concept and relative benefits of variable valve timing for improved volumetric efficiency. There, it was explained that performance improvement and emission reductions could be achieved if the opening and closing times (and ideally the valve lift) of both intake and exhaust valves could be controlled as a function of operating conditions. In Chapter 5, a representative mechanism was discussed for varying camshaft phasing that can be used for varying both intake and exhaust camshaft phasing. This system improves volumetric efficiency by varying valve overlap from exhaust closing to intake opening and absolute phase of valve opening and closing. In addition to improving volumetric efficiency, this variable valve phasing (VVP) can assist in achieving desired EGR fraction.

The amount of valve overlap is directly related to the relative exhaust-intake camshaft phasing. Generally, minimal overlap is desired at idle. The desired optimal amount of overlap is determined during engine development as a function of RPM and load (e.g., by engine mapping).

The desired exhaust and/or intake camshaft phasing is stored in memory (ROM) in the engine control system as a function of RPM and load. Then during engine operation, the correct camshaft phasing can be found via table lookup and interpolation based on measurements of RPM and load. The RPM measurement is achieved using a noncontacting angular speed sensor (see Chapter 5). Load is measured either using MAF and RPM or via an MAP sensor (see Chapter 5).

Once the desired camshaft phasing has been determined, the engine control system sends an appropriate electrical control signal to an actuator (e.g., a motor or a solenoid-operated valve). In Chapter 5, it was shown that for one configuration, camshaft phasing is regulated by the axial position of a helical spline gear. This axial position is determined by the pressure of (engine) oil action on one face of the helical spline gear acting against a spring. This oil pressure is regulated by the solenoid-operated valve.

In Chapter 5, an alternate mechanism for varying camshaft phasing is implemented using oil pressure-activated movable vanes in recesses in the camshaft drive gear. For either this latter mechanism or one based upon a helical gear axial position, closed-loop control enables the engine control system to optimize volumetric efficiency.

Since a VVP system is in fact a position control system, closed-loop control of a camshaft phase requires a measurement of camshaft position relative to the crankshaft. This angular-position measurement can be accomplished by measuring the angle between the camshaft and its drive gear. Numerous angular-position sensor configurations are discussed in Chapter 5. For the following discussion of VVP, it is assumed that such a sensor is part of the system. Fig. 6.11A depicts a physical configuration of a representative camshaft phasing control system.

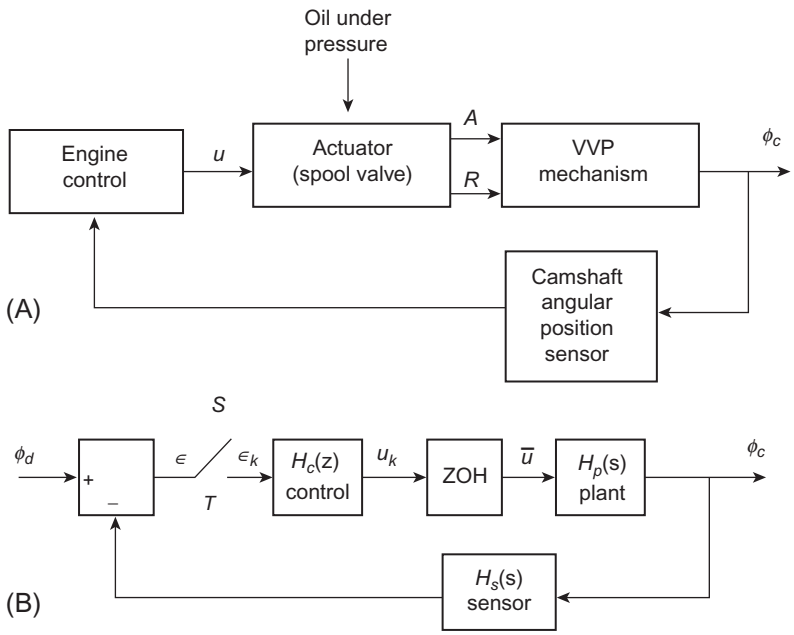


FIG. 6.11 Physical configuration and block diagram of VVP system.

Control of a VVP mechanism has a number of objectives and is subject to certain constraints based upon automotive engine-operating characteristics. Except for steady highway cruise, an automotive engine load and RPM vary over a relatively large range. Consequently, the VVP control must have the capability to follow relatively rapid changes in command. The response to step changes in command should have relatively low overshoot (e.g., $<10\%$) and should reach its command position without a steady-state offset. The control, of course, must be stable and should be robust with respect to parameter changes. The example VVP system presented here is based upon the actuation mechanism described in Chapter 5, which uses vanes attached to the camshaft that move within recesses in the camshaft drive gear. Recall that movement of the vanes relative to this gear results in the variation in camshaft phasing. Recall also that movement of the vanes within the gear recesses is in response to differential pressure on opposite sides of each vane, resulting from a spool valve actuator, which supplies engine oil under pressure to A or R chambers as shown in Fig. 5.47. The dynamic response of the VVP control system should be robust with respect to oil viscosity, which changes in engine temperature; that is, the closed-loop gain for the control system should have large gain and phase margins (see Appendix A).

The VVP control is one function of the digital control system. When operating in VVP mode, the block diagram of the VVP is shown in Fig. 6.11B. As explained in Chapter 5, the actuator for this exemplary VVP includes a variable-duty-cycle pulsed solenoid that can be modeled with a continuous-time control variable, which is denoted $\bar{u}$ (Fig. 6.11B). As explained in Appendix B, a discrete-time control system that regulates a continuous-time plant requires a sample and A/D converter and a zero-order hold (ZOH), both of which are incorporated in the block diagram of Fig. 6.11B. Sensor measurements for such a system are assumed to be ideal such that the sensor transfer function is taken to be

$$H_s(s) \cong 1$$

In Chapter 5, it was shown that the plant transfer function for this VVP configuration is given by

$$H_p(s) = \frac{K_a}{s(s+s_o)} \tag{6.45}$$

For the present example, the following parameters are chosen:

$$K_a = 2600$$
$$s_o = 17$$

A PID control law is selected to provide sufficient flexibility to meet design objectives. The continuous-time PID control law is given by

$$u = K_p \in + K_D \frac{d\in}{dt} + K_I \int \in dt \tag{6.46}$$

Using the root-locus techniques of Appendix A, the following gain parameters are given, which satisfy the overshoot and response time criteria:

$$K_p = 0.080$$
$$K_D/K_p = 0.020$$
$$K_I/K_p = 0.100$$

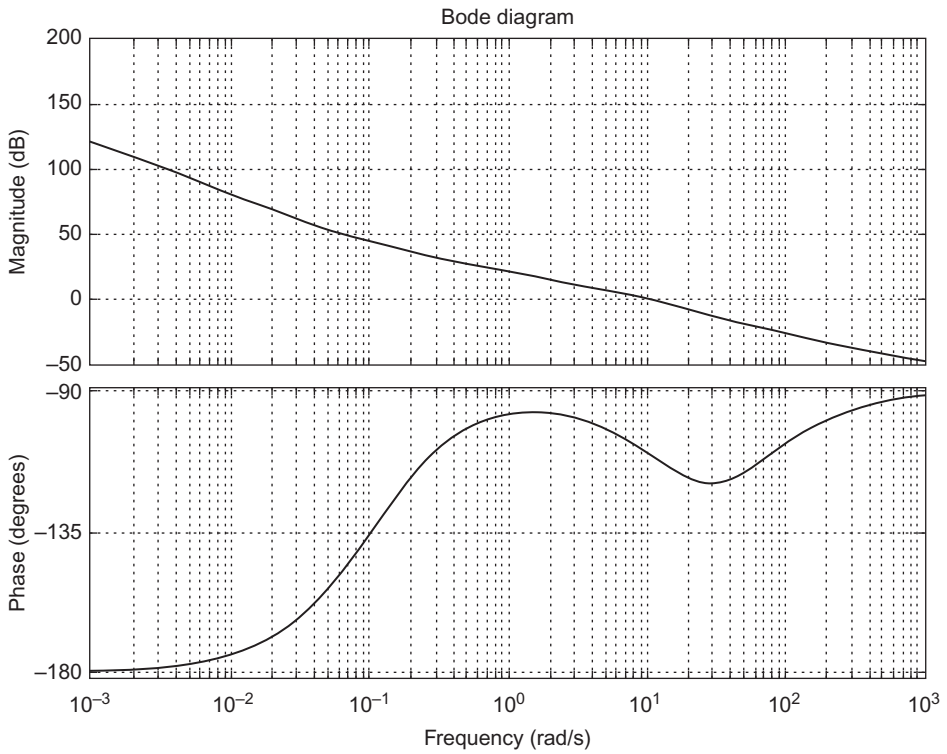


FIG. 6.12 Bode plot of HF(s) for VVP system.

One of the requirements for the VVP control system is robust stability; in Appendix A, it was shown that robustness is expressed meaningfully by gain and phase margins as determined by the Bode plot for the product $H_c(s)H_p(s)$. Fig. 6.12 is the Bode plot for this system.

The gain crossover frequency is at 10 rad/s, and the phase margin there is about 109 degrees. The phase crossover frequency is at about 0.02 rad/s where the gain margin is more than 100 dB. This system has very robust stability.

The gain crossover frequency is at 10 rad/s, and the phase margin there is about 109 degrees. The phase crossover frequency is at about 0.001 rad/s where the gain margin is more than 100 dB. This system has very robust stability.

The discrete-time model for the control system is given by

$$u_k = K_p \in_k + \frac{K_D}{T}(\in_k - \in_{k-1}) + K_I u_{kI} \tag{6.47}$$

where $\epsilon_k = \epsilon(t_k)$, $t_k = kT$, $k = 1, 2, \ldots$, and T = sample period.

In the section on idle speed control, it was shown that the z-transform of u_{kI} using trapezoidal integration rule is given by

$$u_{kI}(z) = K_I z\left[\int \in dt\right]$$
$$= \frac{K_I T(z+1)}{2(z-1)}$$

Combining all three terms in the control variable u_k of Eq. (6.47), the control system transfer function $H_c(z)$ is given by

$$H_c(z) = \frac{u(z)}{\in(z)}$$

$$H_c(z) = K_p + \frac{K_D(z-1)}{Tz} + \frac{K_I T}{2}\frac{(z+1)}{(z-1)} \tag{6.48}$$

$$= \frac{\left[\left(K_p + \frac{K_D}{T} + \frac{K_I T}{2}\right)z^2 - \left(K_p + \frac{2K_D}{T} - \frac{K_I T}{2}\right)z + \frac{K_D}{T}\right]}{z(z-1)}$$

The plant and ZOH z-transform operational transfer function $G(z)$ is found using the method given in Appendix B:

$$G(z) = \left(1 - z^{-1}\right)\mathcal{Z}\left(\frac{H_p(s)}{s}\right)$$

The z-transform in the above equation can be found by expanding $H_p(s)/s$ in a partial fraction. The function $H_p(s)/s$ is given by

$$\frac{H_p(s)}{s} = \frac{K_a}{s^2(s+s_o)} \tag{6.49}$$

This function has a double pole at $s=0$. Using the parameters for the plant given above, the partial fraction expansion is given by

$$\frac{H_p(s)}{s} = \frac{8.9965}{s+17} - \frac{8.9965}{s} + \frac{152.94}{s^2} \tag{6.50}$$

Using the tables of z-transforms from Appendix B and assuming a sample period $T=0.025$ s, the operational transfer function $G(z)$ is given by

$$G(z) = \left(1 - z^{-1}\right)\left[\frac{8.9965z}{z - z_1} - \frac{8.9965z}{z-1} + \frac{152.94zT}{(z-1)^2}\right] \tag{6.51}$$

where $z_1 = e^{-s_0 T}$

The poles of $G(z)$ are all on or in the unit circle, which assures a stable system with a combined transfer function:

$$G(z) = \frac{0.7087z + 0.6152}{z^2 - 1.6538z + 0.6538} \tag{6.52}$$

Using the gain parameters K_p, K_D, and K_I given above, the forward transfer function H_F (from $\in_k$ to the plant output $\phi_c(k)$) is given by

$$\begin{aligned} H_F(z) &= \frac{\phi_c(z)}{\in(z)} \\ H_F(z) &= H_c(z)G(z) \\ &= \frac{0.1021z^3 - 0.0587z^2 - 0.0825z + 0.0394}{z^4 - 2.6538z^3 + 2.3075z^2 - 0.6538z} \end{aligned} \tag{6.53}$$

The closed-loop z-transfer function $H_{cl}(z)$ is given by

$$H_{CL}(z) = \frac{\phi_c(z)}{\phi_d(z)} \tag{6.54}$$

$$\begin{aligned} &= \frac{H_F(z)}{1 + H_F(z)} \\ &= \frac{0.1021z^3 - 0.0587z^2 - 0.0825z + 0.0394}{z^4 - 2.5517z^3 + 2.2489z^2 - 0.7363z + 0.0394} \end{aligned} \tag{6.55}$$

The poles of the closed-loop transfer function are given by

$$\begin{aligned} z_1 &= 0.9975 \\ z_2 &= 0.7442 + 0.2166i \\ z_3 &= 0.7442 - 0.2166i \\ z_4 &= 0.0657 \end{aligned}$$

All poles are within the unit circle for which system stability is assured.

The dynamic response of the VVP system is illustrated by finding the output sequence $\phi_c(k)$ for 10° step command input, which is given by

$$\begin{aligned} \phi_d &= 0 \quad t < 0 \\ &= 10^\circ \quad t \geq 0 \end{aligned}$$

The z-transform of ϕ_d is given by

$$\Phi_d(z) = \frac{10z}{z - 1}$$

The camshaft phase (i.e., system output) is given by

$$\Phi_c(z) = H_{CL}(z)\Phi_d(z)$$

The output sequence $\phi_c(k)$ at time t_k is found using the method of finding the inverse z-transform explained in Appendix B. Recall that this method involves finding the partial fraction expansion of $\Phi_c(z)$ and writing each term as a power series in z^{-k}. The output camshaft phase $\phi_c(k)$ at time t_k is the sum of all coefficient of z^{-k} in the separate power series terms in the partial fraction expansion.

Fig. 6.13 is a plot of this sequence $\{\phi_c(k)\}$ versus time t_k: The transient response error has essentially decayed to zero in <0.5 s, and the overshoot is 6.5%. Thus, this digital variable camshaft phase control system meets the original objectives.

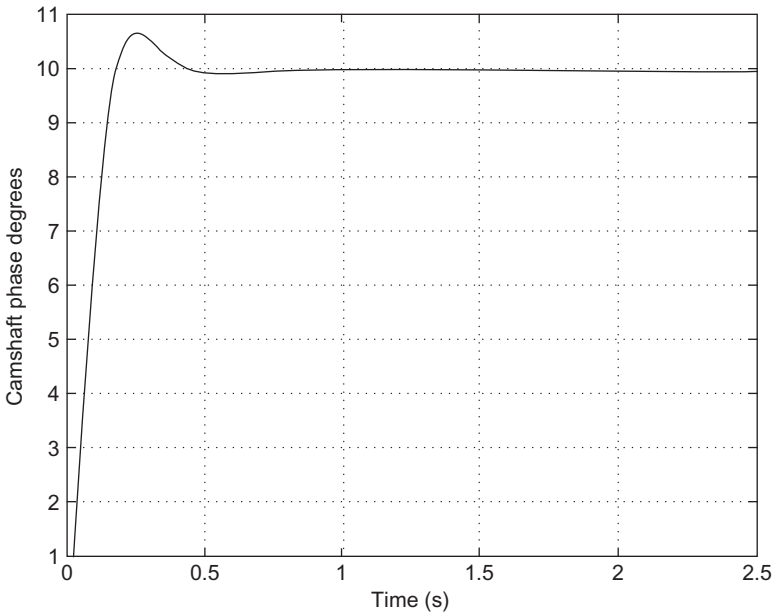


FIG. 6.13 VVP response to 10 degrees step command.

TURBOCHARGING

Some additional technologies that are electronically controlled and that have been added to gasoline-fueled engines include turbocharging and direct injection (DI) of fuel into cylinders. Turbocharging was incorporated with reciprocating aircraft engines to significantly increase engine power at high altitudes.

Turbocharging involves passing engine exhaust gases through a turbine. A turbine configuration consists of a set of aerodynamic blades mounted on a rotatable shaft contained in a housing. Fig. 6.14 depicts a single turbine blade mounted on the turbine shaft. Each blade has a cross section in the form of a highly cambered airfoil. The exhaust gas flows along the axis of the shaft at a velocity denoted V_g in Fig. 6.14B. The details of the gas dynamics involved in the production of rotary power on the shaft are beyond the scope of this book.

An approximate model of the force L_b acting on a turbine blade, however, perhaps is helpful for an understanding of turbine created rotary power. Fig. 6.14B depicts the angle of attack of V_g relative to the turbine blade chord α_T for a cross section of a turbine blade at a given radial distance from the axis of the shaft. This angle decreases with radial distance. The torque T_b produced on a turbine blade is proportional to the dynamic pressure q_b of the exhaust gas:

$$T_b = C_b q_b$$

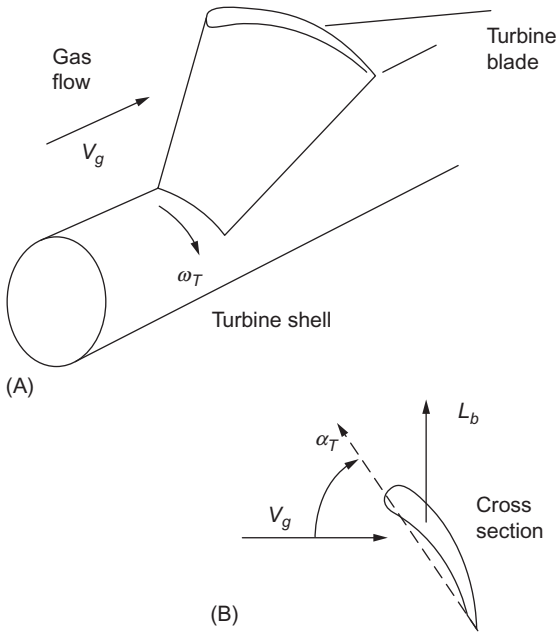


FIG. 6.14 Illustration of turbine configuration. (A) Turbine blade configuration; (B) Gas/blade geometry.

where

$$q_b = \frac{1}{2}\rho_g V_g^2$$

$\rho_g =$ exhaust gas density

$V_g =$ exhaust gas axial velocity component

In this model, C_b is a constant for the blade that is proportional to its area normal to gas flow and is also a function of the blade geometry and the variation of α_T with radial distance from the axis of rotation.

In a given turbine configuration, there are N blades attached in one or more regions that are orthogonal to the turbine axis. The net torque T_N of an N blade turbine is approximately given by

$$T_N = \sum_{n=1}^{N} T_b(n)$$

where $T_b(n) =$ torque on blade n $n = 1,2,\ldots,N$.

The turbine shaft rotational speed is denoted ω_T. The turbine shaft power P_T is given by

$$P_T = T_n \omega_T$$

The turbine shaft is connected to an air pump, sometimes called a super charger, that provides air at a controllable pressure and mass flow rate to the engine intake manifold. The combination of turbine and air pump (or super charger) is called a turbocharger (TC). There are multiple air-pump configurations, the details of which are not important for an understanding of the operation or control of the

turbocharger. One such air-pump configuration is similar to the turbine in that it has multiple aerodynamically shaped blades on the shaft that is connected to or part of the turbine shaft. Rotation of this type of air pump causes an axial flow of ambient air that can be directed to the engine intake.

The amount of air to be pumped into the engine and the intake manifold pressure must be controlled to be useful for improving engine performance and efficiency. For an understanding of the benefits of turbocharging the intake air, it is best compared with a so-called normally aspirated engine that is one for which the intake air is at ambient pressure and temperature. As explained in Chapter 4, the power produced by a reciprocating engine is proportional to the intake mass airflow rate. With a turbocharged engine, the mass airflow into the engine can be significantly increased for a given engine at a given RPM through the use of turbocharging relative to normally aspirated engines of the same displacement. This means that a vehicle of a given size and weight can be powered by a much smaller displacement turbocharged engine than by a normally aspirated engine. In addition, the range of power levels for a turbocharged engine is much larger than that for a normally aspirated engine of comparable displacement. Furthermore, with an increase in intake pressure coming from the turbocharger, the air-pumping losses are reduced. In addition, with fewer cylinders, the friction losses at a given RPM are reduced. In effect, a turbocharger recovers power that is lost to the exhaust. The combination of these factors means that a vehicle of a given size and weight can be powered by a physically smaller, more efficient engine than a normally aspirated engine at the same power levels.

On the other hand, there are restrictions on the maximum power produced by a given engine configuration. The power produced from a given engine can be increased by increasing mass air flow rate. The absolute upper bound is the power level at which an engine component fails due to excessive stress.

The operation of a turbocharged engine perhaps can best be understood with reference to the schematic mechanical system drawing of Fig. 6.15.

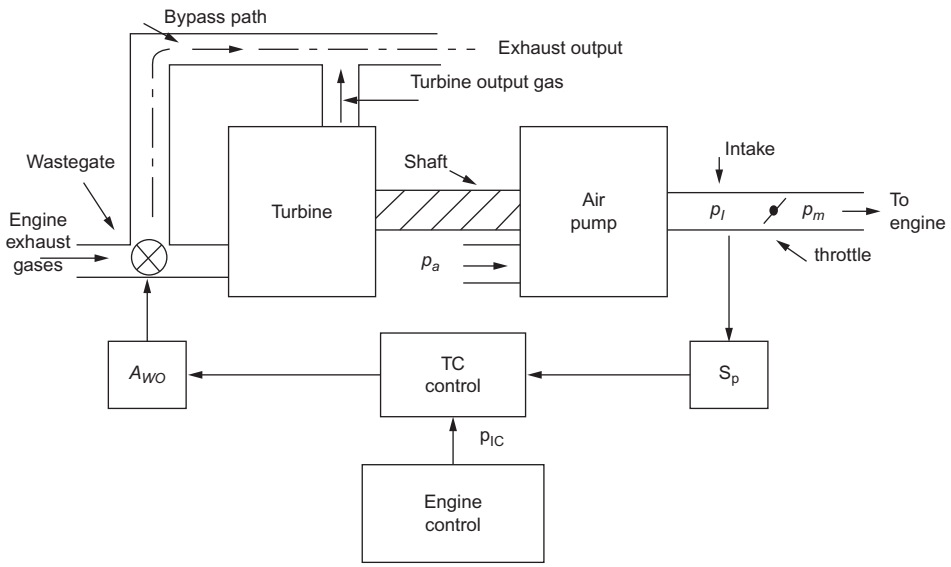


FIG. 6.15 Turbocharger/engine configuration.

In Fig. 6.15, the turbine and air pump are simply represented schematically to show the flow of exhaust gases and intake air along with the components associated with controlling engine performance. The exhaust gases coming from the engine pass into the turbine. However, not all of the exhaust gases pass through the turbine. The fraction that does go through the engine is regulated by a special type of valve called a "waste gate."

The air pump receives air at atmospheric pressure (p_a) and creates output air that is fed to the engine intake at pressure p_I. The increase in p_I relative to p_a is termed "boost." It is this boost in intake pressure that enables an increase in engine power relative to a normally aspirated engine that receives intake air at pressure p_a.

For any p_I, the throttle sets engine power level via regulation of manifold pressure p_m as explained in Chapter 4. There are numerous control strategies for turbocharged engines that, in part, regulate the setting of the waste gate as a means of controlling p_I. The control of the turbocharger operation (which regulates p_I) is explained by depicting the controller as a separate block (labeled TC control) in Fig. 6.15, even though this control is implemented in engine digital control system.

To explain this control, a specific representative example engine control is chosen as an example in which the engine control system determines a desired value for p_I, which is denoted p_{IC}, based upon engine-operating conditions and environmental parameters (e.g., p_a, Ta). A linear model for the turbocharger operation is developed below to simplify the explanation of its control, which is sufficiently accurate over a relatively narrow range of operating conditions to correctly explain turbocharger operation.

In this simplified model, the pressure p_I in the TC output is given by

$$p_I = K_b p_a \omega_T$$

where K_b = fixed parameter of the air pump and ω_T = turbocharger shaft angular speed. The turbine speed for a given exhaust gas flow rate is determined by the waste gate that is characterized by a variable θ_T. This variable is determined by the actuator that is denoted A_{wo} in Fig. 6.15. This variable is represented by the output of the TC control. The turbine dynamic response to waste gate variable can be represented by the following first-order model:

$$\dot{\omega}_T + \frac{\omega_T}{\tau_T} = K_\theta \theta_T$$

where τ_T = time constant for turbine response.

In this model, K_θ is parameter that is proportional to exhaust gas flow rate. The transfer function for the turbocharger $H_T(s)$ is given by

$$H_T(s) = p_I(s)/\theta_T(s) = \frac{K_b p_a K_\theta}{s + \dfrac{1}{\tau_T}}$$

A block diagram of a closed-loop turbocharger control system is depicted in Fig. 6.16.

The theory and analyses of a closed-loop control system such as is depicted in Fig. 6.16 is explained in detail in Appendix A. In this system, the control law is represented by the transfer function $H_c(s)$ of the control subblock. For the purposes of explaining the TC control system, it is assumed that a PI control law (see Appendix A) is used where

$$\theta_T(t) = K_P \in + K_I \int \in dt$$

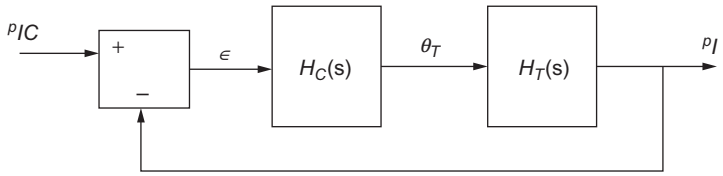


FIG. 6.16 Block diagram of TC control subsystem.

where

$$\epsilon = p_{IC} - p_I$$

K_P = proportional control parameter, p_{IC} = command intake pressure from the engine control, and K_I = integral control parameter.

In the above expression for ϵ, it is implicitly assumed that the pressure sensor model is a part of the engine control such that the true error ϵ is known. The transfer function $H_C(s)$ is given by

$$H_C(s) = \theta_T(s)/\epsilon(s)$$
$$= K_P + K_I/s = \frac{K_P s + K_I}{s}$$

The performance of the TC control system is represented by the closed-loop transfer function ($H_{C\ell}(s)$). It is further shown in Appendix A that this closed-loop transfer function is given by

$$H_{c\ell}(s) = \frac{p_I(s)}{p_{IC}(s)}$$

$$H_{C\ell}(s) = \frac{H_C(s)H_T(s)}{1 + H_C(s)H_T(s)}$$

Once the parameters for the turbocharger are known, it is simple to compute the dynamic response and evaluate time delays associated with p_I responding to any changes in the command pressure p_{IC} In addition, optimization of the controller/compensator and stability are readily determined (e.g., with computer simulation). In a practical engine configuration, the TC control is implemented via a digital controller.

DIRECT FUEL INJECTION

Another technology that has been introduced relatively recently in addition to VVT and turbocharging is direct fuel injection into the cylinders of gasoline-fueled engines. Direct injection of fuel into cylinders has been in use with diesel engines and was used in a number of very early gasoline-fueled engines. However, the more recent era of electronically controlled engines have involved fuel injected external to the cylinder during the intake stroke at relatively low pressure as explained in Chapter 4. For gasoline-fueled direct injection engines, the fuel is injected directly into the cylinder and requires relatively high fuel pressure. For direct fuel injection (DFI), the fuel injectors are mounted in the cylinder head and spray the gasoline into the combustion chamber from a fuel rail.

An engine incorporating VVT, turbocharging, and DFI has potential for improving fuel economy and emissions and performance relative to a comparably sized engine having fixed valve phasing, which is normally aspirated and uses multiport intake manifold fuel injection. However, to take advantage of DFI, it is necessary to operate in multiple control modes.

One of the DFI control strategies is set for very low required engine power including idle and some constant engine load conditions at low-to-moderate vehicle speeds. Another control strategy would be required when exhaust emissions require a stoichiometric air/fuel for optimal catalytic converter operation (see Chapter 4). Yet, another control mode is used under full or near-full-throttle operating conditions when the required engine power is at or near its maximum output power capability. This third control strategy is available for relatively short time intervals (e.g., climbing a relatively steep slope) since exhaust emissions briefly exceed regulated standards.

The low output control strategy involved relatively high air/fuel (e.g., A/F > 25:1). As explained in Chapter 4, air/fuel greater than stoichiometric ratio results in combustion temperatures exceeding those for stoichiometry and results in an increase in NOx emissions. Although the catalytic converter efficiency of conversion is less than optimal for NOx emissions, for sufficiently low engine power, it is still possible to meet government regulations.

For this relatively low required engine power control strategy, only air is dumped into the engine during the intake stroke. The fuel is injected during the last few degrees of crankshaft rotation (near TDC) on the compression stroke. The air/fuel mixture for this mode of control and strategy is not homogeneous (as is desired) for conventional multiport fuel injection during the intake stroke. When combustion occurs, the pressure in the combustion chamber rises such that torque/power are produced but at relatively low levels. For each engine configuration, the power levels at which leaner than stoichiometric air/fuel control strategy is used are determined during engine calibration. Each manufacturer must be capable of assuring that exhaust emissions meet government standards.

For any DFI engine, there is a limit to the engine power for which this leaner than stoichiometry control strategy can be used. When the power required reaches or exceeds this level, the control strategy returns to maintaining air/fuel at stoichiometry. For the stoichiometric control strategy, fuel is injected directly into the cylinder during the intake stroke. In this case, the air/fuel mixture is produced within the cylinder. The engine valve configuration is such that "swirl" of the entering air mixes with fuel forming an essentially homogeneous mixture. In fact, the resulting mixture is closer to being uniformly homogeneous than the traditional intake port fuel injection. This condition results in combustion with exhaust gas that maintain concentrations closer to the optimum for catalytic converter conversion efficiency (see Chapter 4).

The exception to the stoichiometric and lean mixture control strategies is the operation of the engine near wide open throttle as mentioned above. Except for race cars that need not meet emission regulations, the maximum power output for street vehicles is a somewhat rare operating mode. The control strategy for this operating mode involves direct injection of fuel into the cylinder during the intake stroke with air/fuel less than stoichiometric and, in fact, corresponding to maximum power for a given RPM. Although this air/fuel varies somewhat between engine models, it is in the general region of $\frac{\text{air mass}}{\text{fuel mass}} \simeq 12:1$.

In general, a DFI engine that also incorporates turbocharging and VVT has performance and emissions superior to a traditional normally aspirated, fixed valve phasing, and port fuel injection engine of the same displacement. The trend in contemporary vehicles is to incorporate these technologies.

FLEX FUEL

Except for diesel-engine vehicles, land vehicles have been fueled for decades with gasoline. The hydrocarbon composition of gasoline and additives vary with location and season. In addition, in more recent years, ethanol has been added to gasoline to a maximum of 85% (which is called E85 fuel). Ethanol is a biofuel that is produced typically from corn rather than petroleum from which gasoline is produced. It is a combustible fluid, but it has less energy density than gasoline.

Although ethanol itself is a fuel for an IC engine, it has certain limitations that affect its use. For example, in cold climates (e.g., the northern portion of the United States and Canada), it does not have acceptable cold starting capabilities in any engine that was designed for gasoline fuel. In addition, its lower energy density results in a reduction in fuel economy (i.e., MPG) that can vary from roughly 10%–30% depending upon the vehicle size and overall engine configuration. On the other hand, ethanol has an antiknock capability that is equivalent to a relatively high octane rating gasoline.

Flex fuel is generally not recommended in vehicles that have not been designed for its use. Among its drawbacks (particularly with older vehicles) is the possibility of damaging some materials that have been used in the fuel system of these older cars.

As explained below, the engine control system for a flex-fuel capable engine must have different fuel delivery and ignition control algorithms than those for gasoline only engines. However, these algorithms are not necessarily difficult to program into an engine control system. In addition, the hardware components of a flex-fuel vehicle engine are different than for a gasoline only engine. These hardware differences include a special fuel composition sensor along with fuel handling components (e.g., fuel injectors, fuel lines, and fuel pumps) that can safely operate with ethanol in the fuel.

In order to achieve the same performance as gasoline, a mixture of ethanol and gasoline must have a lower air/fuel than pure gasoline. In addition, the stoichiometric mixture requires air/fuel in the range

$$8.765 \leq \frac{m_a}{m_f} \leq 14.7$$

where m_a = mass of air pumped into cylinder and m_f = mass of fuel pumped with the air or injected directly into the cylinder.

The higher air/fuel is stoichiometric for gasoline only as explained in Chapter 4. The lowest air/fuel limit above is stoichiometric for E85. For any given flex-fuel composition, the air/fuel is a function of the composition as expressed by the fraction of the mixture that is ethanol, which is denoted η_{ef}.

For notational convenience, the air/fuel is expressed by R_{af} on an engine cycle basis, which is defined

$$R_{af} = \frac{m_a}{m_f}$$

The engine control maintains R_{af} at stoichiometric ratio for exhaust emission requirements for the majority of engine-operating periods. Exceptions to this control strategy have been explained in Chapter 4 and this chapter for port-injected engines and in this chapter for DFI engines. The fuel mass (m_f) delivered to a cylinder during any engine cycle is determined by the fuel rail pressure, by the fuel injector characteristics, and by the fuel injector open time, as explained in Chapters 4 and this chapter. The fuel injector operation and model are given in Chapter 5.

In a closed-loop control strategy, it is possible to maintain stoichiometry with the feedback from the EGO sensor (see Chapter 4). However, in any open-loop operation, the correct fuel delivery can be controlled with an input to the engine control system from a flex-fuel composition sensor (FFS) as explained and modeled in Chapter 5. Moreover, the FFS can assist in engine control adaptation to changes in various components. The discussion in Chapter 5 explains that η_{ef} is measured by measuring the capacitance (C_{FF}) of a sensor configuration that is a capacitor with a dielectric constant C_{FF} that is a function of η_{ef}. It is also explained in Chapter 5 that the normal means of measuring C_{FF} involves measurement of frequency or cycle time intervals of an oscillator that involves the FFS as an essential component. In addition, it is explained in Chapter 5 that flex-fuel temperature is measured to permit the engine control to calculate corrections to the calculated η_{ef} due to variations in the constituent fuel components dielectric constant (or permittivity) with temperature. The influence of the electrical conductivity on the FFS equivalent circuit and its corresponding affect on measurement of η_{ef} is also discussed in Chapter 5. Fuel additives that increase conductivity have the effect of minimizing sparking in fuel containers and some other fuel handling hardware.

The computation of η_{ef} from the capacitance C_{FF} can be done in the main engine control computer. However, it is also possible to incorporate a microprocessor-based subsystem as a part of the FFS that can compute η_{ef} from C_{FF} measurements when it is programmed with algorithms based upon the formulas derived in Chapter 5.

ELECTRONIC IGNITION CONTROL

As explained in Chapter 4, an engine must be provided with fuel and air in correct proportions and the means to ignite this mixture in the form of an electric spark. Before the development of contemporary electronic ignition, the traditional ignition system included the spark plugs, a distributor, and a high-voltage ignition coil. The distributor (which was a form of rotary switch) would sequentially connect the coil output high voltage to the correct spark plug. In addition, it would cause the coil to generate the spark by interrupting the primary current (via ignition points) in the coil circuit, thereby generating the required spark. The time of occurrence of this spark (i.e., the ignition timing) in relation of the piston to TDC that influences the torque generated was determined mechanically by distributor phasing relative to the engine cycle.

The distributor and single coil have been replaced by multiple coils and an electronic control system. Each coil supplies the spark to either one or two cylinders. In such a system, the controller selects the appropriate coil and delivers a trigger pulse to the ignition control circuitry at the correct time for each cylinder. (*Note* that in some cases, the coil is on the spark plug as an integral unit.)

Fig. 6.17 illustrates such a system, for example, four-cylinder engine. In this example, a pair of coils provides the spark for firing two cylinders for each coil. Cylinder pairs are selected such that one cylinder is on its compression stroke while the other is on exhaust. The cylinder on compression is the cylinder to be fired (at a time somewhat before it reaches TDC). The other cylinder is on exhaust. The coil fires the spark plugs for these two cylinders simultaneously. For the former cylinder, the mixture is ignited, and combustion begins for the power stroke that follows. For the other cylinder (on exhaust stroke), the combustion has already taken place, and the spark has no effect.

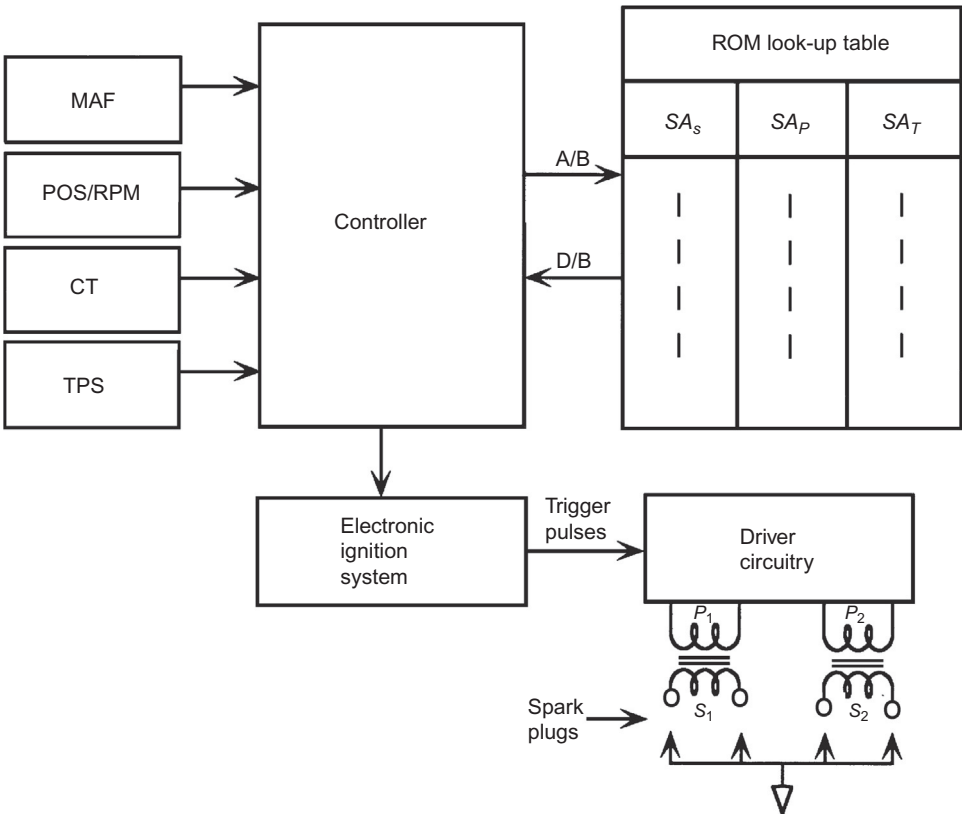


FIG. 6.17 Example of ignition circuit diagram.

Although the mixture for contemporary vehicle engines is constrained by emission regulations, the spark timing can be varied in order to achieve optimum performance within the exhaust emission constraint. For example, the ignition timing can be chosen to produce the best possible engine torque for any given operating condition. This optimum ignition timing is known for any given engine configuration from empirical studies of engine performance as measured on an engine dynamometer. As explained in Chapter 4, this optimum ignition timing is known as "spark advance for mean best torque" that is abbreviated MBT.

Ignition timing is normally represented quantitatively by the angular position of the crankshaft relative to TDC for each cylinder during its compression stroke. Spark occurs before TDC because of the time required for combustion to be completed such that power during the power stroke is optimized. Spark timing in degrees of crankshaft rotation is termed "spark advance" (SA).

In the example configuration of Fig. 6.17, the spark advance value is computed in the main engine control (i.e., the same controller that regulates fuel). This system receives data from the various sensors (as described above with respect to fuel control) and determines the correct spark advance for the instantaneous operating condition.

The variables that influence the optimum spark timing at any operating condition include RPM, manifold pressure (or mass airflow), barometric pressure, and coolant temperature. The correct ignition timing for each value of these variables is stored in an ROM lookup table. The engine control system obtains readings from the various sensors and generates an address to the lookup table (ROM). After reading the data from the lookup tables, the control system computes the correct spark advance (possibly including interpolation). An output signal is generated at the appropriate time to activate the spark.

In the configuration depicted in Fig. 6.17, the electronic ignition is implemented in a stand-alone ignition module. This solid-state module receives the correct spark advance data and generates electrical signals that operate the coil driver circuitry. These signals are produced in response to timing inputs coming from crankshaft and camshaft signals (POS/RPM).

The coil driver circuits generate the primary current in windings P_1 and P_2 of the coil packs depicted in Fig. 6.16. These primary currents build up during the so-called dwell period before the spark is to occur. The process of spark generation for ignition purposes was explained in Chapter 5. There, it was explained that the spark is produced by a short-duration very high voltage that is generated in the ignition coil. In the example depicted in Fig. 6.16, a pair of coil packs, each firing two spark plugs, is shown. Such a configuration would be appropriate for a four-cylinder engine. Normally, there would be one coil pack for each pair of cylinders or possibly for each cylinder.

In a typical electronic ignition control system, the total spark advance, SA (in degrees before TDC), is made up of several components that are added together:

$$SA = SA_S + SA_P + SA_T \tag{6.56}$$

The first component, SA_S, is the basic spark advance, which is a tabulated function of RPM and MAP or MAF. The control system reads RPM and MAP or MAF and calculates the address in ROM of the SA_S that corresponds to these values. Fig. 6.18 depicts a representative variation in SA_S versus RPM.

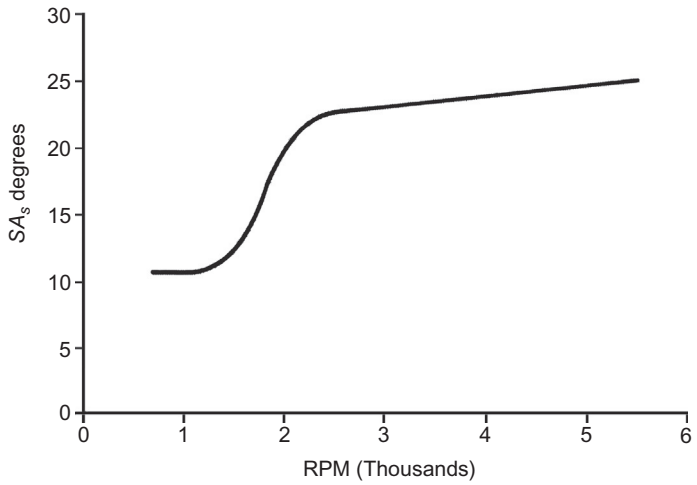


FIG. 6.18 Representative SA curve versus RPM.

In the example, the advance of RPM from idle to about 1200 RPM is relatively slow. Then, from about 1200 to about 2300 RPM, the slope of SA_s with respect to RPM is relatively steep. Beyond 2300 RPM, the increase in SA_s with respect to RPM is again relatively small. Each engine configuration has its own spark advance characteristic, which is normally a compromise between a number of conflicting factors (the details of which are beyond the scope of this book). The SA_s tabulated values that are placed in ROM are normally determined via engine mapping during development of an engine control system.

The second component, SA_P, is the contribution to spark advance due to mass airflow or manifold pressure. This value is obtained from ROM lookup tables with MAF or MAP as the independent variable. In general, the SA_P is reduced as intake manifold pressure increases, owing to an increase in combustion rate with pressure.

The final component, SA_T, is the contribution to spark advance due to temperature. Temperature effects on spark advance are relatively complex, including such effects as cold cranking, cold start, warm-up, and fully warmed-up conditions, the details of which are engine configuration-specific.

CLOSED-LOOP IGNITION TIMING

The ignition system described in the foregoing is an open-loop system. The major disadvantage of open-loop control is that it cannot automatically compensate for mechanical changes in the system. Closed-loop control of ignition timing is desirable from the standpoint of improving engine performance and maintaining that performance in spite of system changes.

One scheme for closed-loop ignition timing is based on the improvement in performance that is achieved by advancing the ignition timing relative to TDC. For a given RPM and manifold pressure, the variation in torque with spark advance is as depicted in Fig. 6.19.

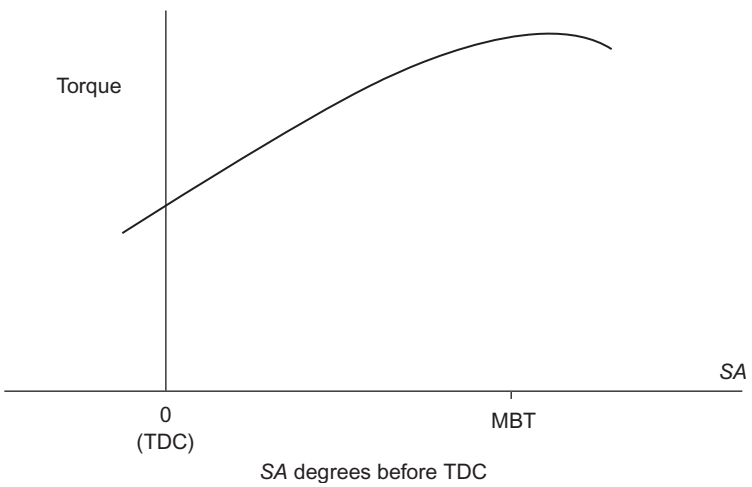


FIG. 6.19 Engine brake torque versus SA.

One can see that advancing the spark relative to TDC increases the torque until a point is reached at which best torque is produced. As introduced above and explained qualitatively in Chapter 3, this spark advance is known as the SA for *mean best torque* or MBT.

When the spark is advanced too far, an abnormal combustion phenomenon occurs that is known as *knocking*. Although the details of what causes knocking are beyond the scope of this book, it is generally a result of a portion of the air-fuel mixture abruptly igniting (autoigniting), as opposed to being normally ignited by the advancing flame front that occurs in normal combustion following spark ignition. Roughly speaking, the amplitude of knock is proportional to the fraction of the total air and fuel mixture that autoignites. It is characterized by an abnormally rapid rise in cylinder pressure during combustion, followed by very rapid oscillations in cylinder pressure. The frequency of these oscillations is specific to a given engine configuration and is typically in the range of a few kilohertz. Fig. 6.20 is a graph of a representative cylinder pressure versus time under knocking conditions. A relatively low level of knock is arguably beneficial to performance, although excessive knock is unquestionably damaging to the engine and must be avoided.

One control strategy for spark advance under closed-loop control is to advance the spark timing until the knock level becomes unacceptable. At this point, the control system reduces the spark advance (retarded spark) until acceptable levels of knock are achieved. Of course, a spark advance control scheme based on limiting the levels of knocking requires a knock sensor such as that explained in Chapter 5. This sensor responds to the acoustic energy in the spectrum of the rapid cylinder pressure oscillations, as shown in Fig. 6.20.

Fig. 6.21A is a diagram of an exemplary instrumentation system for measuring knock intensity. Output voltage V_E of the knock sensor is proportional to the acoustic energy in the engine block at the sensor mounting point. This voltage is sent to a narrow band-pass filter that is tuned to the knock frequency (for the particular engine configuration). The filter output voltage V_F is proportional to the amplitude of the knock oscillations and is thus a "knock signal." The envelope voltage of these oscillations, V_d, is obtained with a detector circuit which can, for example, be implemented with a

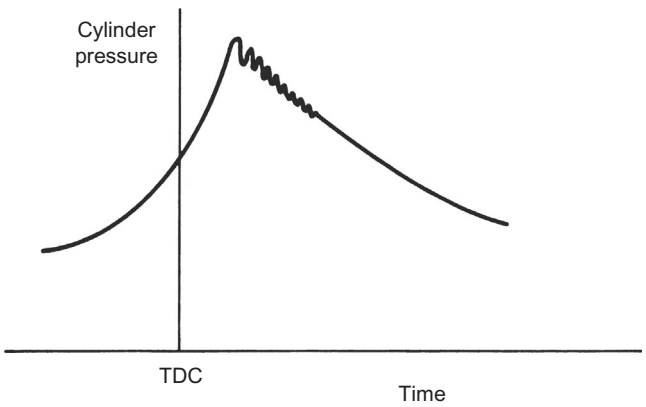


FIG. 6.20 Cylinder pressure under knock conditions.

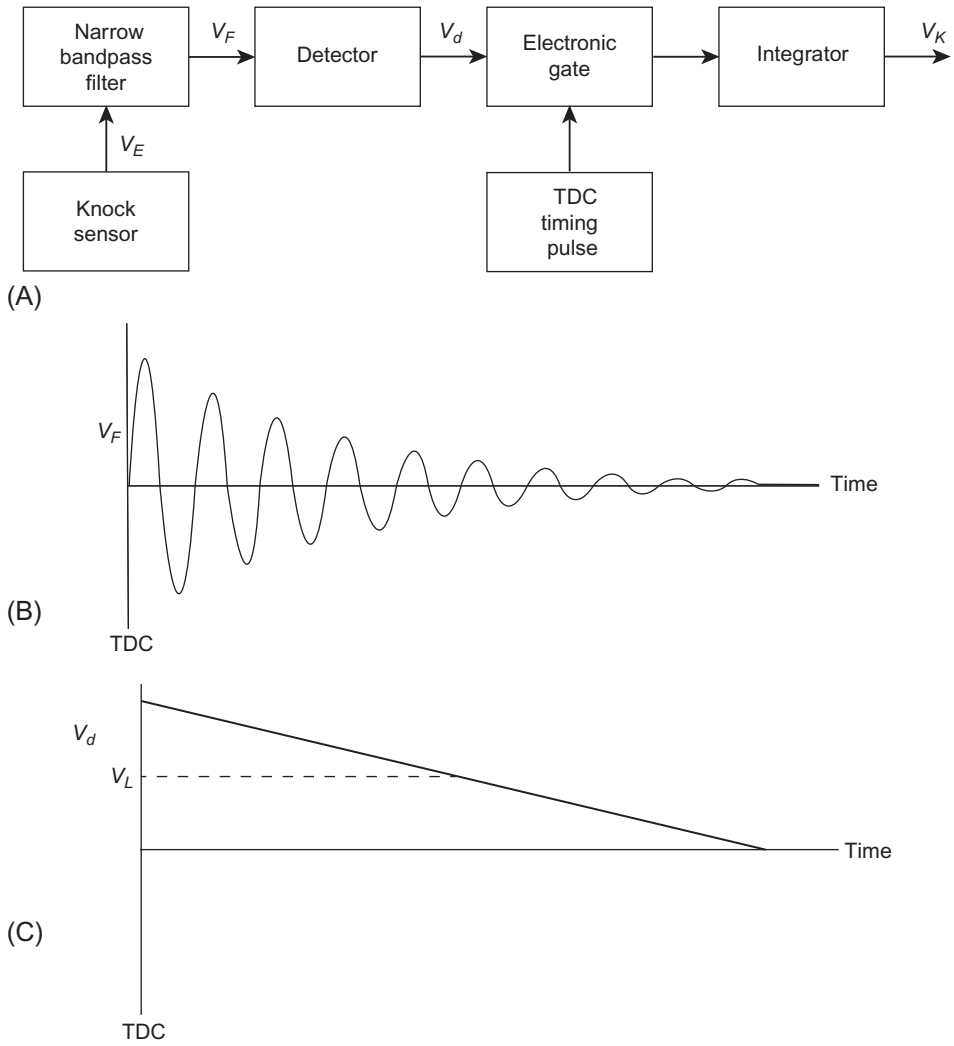


FIG. 6.21 Instrumentation for measuring knock intensity. (A) Knock control block diagram; (B) Exemplary filtered knock sensor voltage; (C) Exemplary detector output voltage.

rectifier-type circuit that includes a diode and a capacitor (see Chapter 2). An exemplary sample of some actual vehicular signals is presented in Fig. 6.21B.

Following the detector in the circuit of Fig. 6.21 of the example knock detection system is an electronic gate that normally blocks V_d for much of the engine cycle but passes it during the portion of the engine cycle for which the knock amplitude is largest (i.e., shortly after TDC). The gate is, in essence, an electronic switch that is normally open but is closed for a short interval (from 0 to T) following TDC. It is during this interval that the knock signal is largest in relationship to engine noise. The probability

of successfully detecting the knock signal is greatest during this interval. Similarly, the possibility of mistaking normal engine acoustic noise for true knock signal is smallest during this interval.

The final stage in the exemplary knock-measuring instrumentation is integration with respect to time. Integration can be accomplished numerically in the engine control or as a part of the knock sensor instrumentation using an operational amplifier circuit configured to perform analog integration. For example, the circuit of Fig. 6.22A could be used to integrate the gate output. In our example system, the electronic gate is implemented via a pair of switches, S_1 and S_2. Switch S_1 is normally open and S_2 closed, but S_1 is closed and S_2 opened at $t=0$ corresponding to the beginning of the period where knock can occur. The end of this period is $t=T$. This gate operation is repetitive and occurs following TDC for the power stroke of the associated cylinder. The output voltage V_K at the end of the gate interval T is given by

$$V_K = -(1/RC)\int_0^T V_d(t)\,dt \tag{6.57}$$

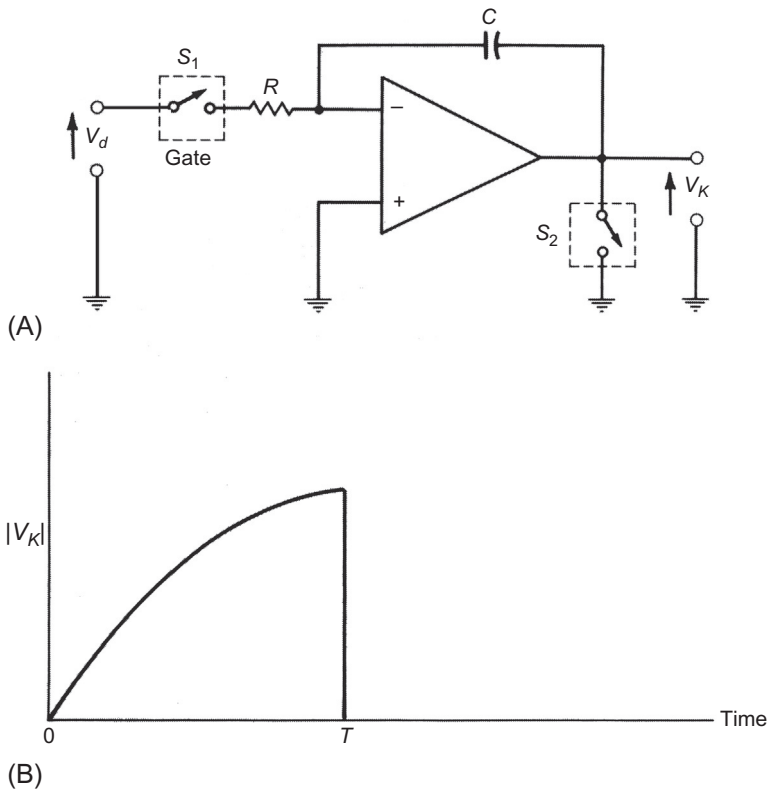


FIG. 6.22 Analog integrator for knock detection system. (A) Block diagram; (B) Exemplary output voltage.

This voltage increases sharply in magnitude but is negative for V_d as depicted in Fig. 6.22B because the input is connected to the op-amp inverting input. Fig. 6.22B is a plot of the absolute magnitude of V_k (i.e., $|V_k|$) in the absence of noise. This voltage reaches a maximum amplitude at the end of the gate interval, as shown in Fig. 6.22B, provided knock occurs. However, if there is no knock, V_K remains near zero.

The level of knock intensity is indicated by voltage $|V_K(T)|$ at the end of the gate interval. The spark control system compares this voltage with a threshold voltage to determine whether knock has or has not occurred.

This envelope-detected voltage is sent to the controller, where it is compared with a level corresponding to the knock intensity threshold. Whenever the knock level is less than the threshold, the spark is advanced. Whenever it exceeds the threshold, the spark is retarded. The comparator function is normally implemented in the digital control system by numerically comparing the integrated knock intensity signal with a threshold T_K (under program control; see Fig. 6.23).

In such an implementation, the controller generates a binary-valued variable (denoted K in Fig. 6.23) having the following algorithm:

$$\begin{aligned} K &= 0 \; |V_K(T)| < T_K \\ &= 1 \; |V_K(T)| > T_K \end{aligned} \tag{6.58}$$

In an illustrative closed loop spark control system, the spark is advanced by an engine specific amount for $K=0$ and retarded for $K=1$. Knock detection with the above algorithm has two types of error: (1) missed detection in which knock has occurred but the system output is $K=0$ and (2) false alarm in which there is normal combustion but the system output is $K=1$. The quantitative error analysis for the above knock detection method generally is covered in the field of statistical decision theory. The theory of this topic is outside the scope of this book. However, for those readers having a background in statistical analysis, we present the following brief models and analysis of the probability of error in the above knock detection system.

Essentially, the voltage of any point in the exemplary knock detection system is a random process. In this exemplary knock detection system, the detection of knock is based upon the voltage $V_K(T)$ and is, in effect, a form of statistical hypothesis testing. This method can perhaps best be explained from the histogram of Fig. 6.24 for voltage $V_K(T)$ for a large sample of engine cycles under the two hypotheses:

H_0—normal combustion
H_1—knocking conditions

For notational convenience, we let $x=|V_K(T)|$ in Fig. 6.24. In this figure, the number of occurrences of x at a particular value for hypothesis H_0 is denoted $n_{H_0}(x)$ and for hypothesis H_1 is denoted $n_{H_1}(x)$. For a sufficiently large sample space, these histograms approach the continuous probability density functions for the two hypotheses that are denoted $p_{H_0}(x)$ and $p_{H_1}(x)$, respectively.

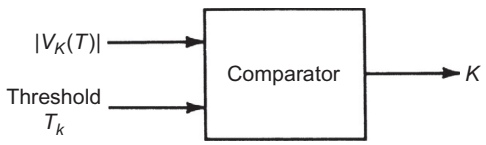


FIG. 6.23 Comparator for knock detector.

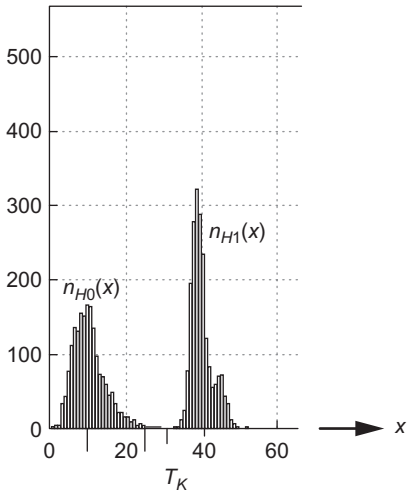


FIG. 6.24 Histogram for hypotheses H_0 and H_1.

The detection threshold T_K is depicted in Fig. 6.24. The total probability of error P_e for our example knock detection method is given by

$$P_e = \int_{T_K}^{\infty} p_{H_0}(x)dx + \int_{O}^{T_K} p_{H_1}(x)dx \tag{6.59}$$

where the first term corresponds to false-alarm errors and the second to missed detection errors. For any such knock detection method, an optimum threshold that minimizes the total probability of error can be determined empirically.

Although this scheme for knock detection has shown a constant threshold, there are some production applications that have a variable threshold. The threshold in such cases increases with RPM because the competing acoustic noises in the engine increase with RPM.

SPARK ADVANCE CORRECTION SCHEME

Although the details of spark advance control are vehicle model and manufacturer-specific, there are generally two classes of correction that are used: fast correction and slow correction. In the fast correction scheme, the spark advance is decreased for the next engine cycle by a fixed amount (e.g., 5 degrees) whenever knock is detected. Then, the spark advance is incremented in one-degrees increments every 5–20 crankshaft revolutions.

The fast correction ensures that minimum time is spent under heavy knocking conditions. Further, this scheme compensates for hysteresis (i.e., for one degrees of spark advance to cause knocking, more than one degrees must be removed to eliminate knocking). The fast correction scheme is depicted qualitatively by the waveform depicted in Fig. 6.25.

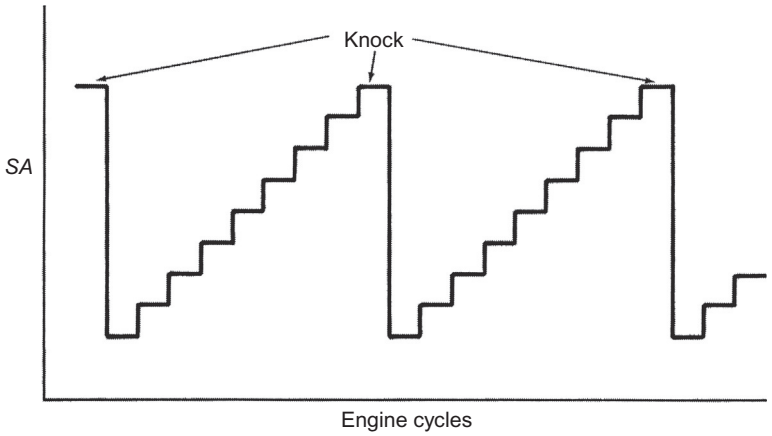


FIG. 6.25 Fast correction of SA.

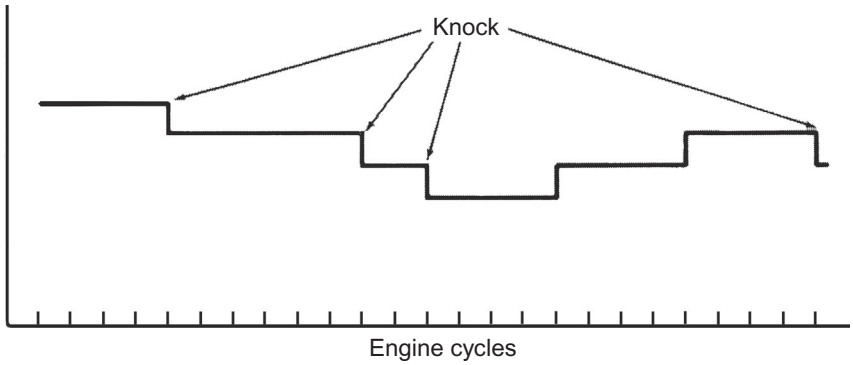


FIG. 6.26 Slow correction of SA.

In the slow correction scheme (Fig. 6.26), spark advance is decreased by one (or more) degree each time knock is detected, until no knocking is detected. The spark advance proceeds in one-degrees increments after many engine cycles.

The slow correction scheme is more of an adaptive closed-loop control than is the fast correction scheme. It primarily is employed to compensate for relatively slow changes in engine condition or fuel quality (i.e., octane rating).

INTEGRATED ENGINE CONTROL SYSTEM

Each control subsystem for fuel control, spark control, and EGR has been discussed separately. However, in a contemporary vehicle, an integrated electronic engine control system employs an open

architecture and can include these subsystems and provide additional functions. (Usually, the flexibility of the digital control system allows such expansion quite easily because the computer program can be changed to accomplish the expanded functions.) Several of these additional functions are discussed in the following.

SECONDARY AIR MANAGEMENT

Secondary air management is used to improve the performance of the catalytic converter by providing extra (oxygen-rich) air either to the converter itself or to the exhaust manifold. The catalyst temperature must be above about 200°C to efficiently oxidize HC and CO and reduce NO*x*. During engine warm-up when the catalytic converter could be cold, HC and CO are oxidized in the exhaust manifold by routing secondary air to the manifold. This creates extra heat to speed warm-up of the converter and EGO sensor, enabling the fuel controller to go to the closed-loop mode relatively quickly.

The converter can be damaged if too much heat is applied to it. This can occur if large amounts of HC and CO are oxidized in the manifold during periods of heavy loads, which call for fuel enrichment, or during severe deceleration. In such cases, the secondary air is directed to the air cleaner, where it has no direct effect on exhaust temperatures.

After warm-up, the main use of secondary air is to provide an oxygen-rich atmosphere in the second chamber of the three-way catalyst, dual-chamber converter system. In a dual-chamber converter, the first chamber contains rhodium, palladium, and platinum to reduce NO*x* and to oxidize HC and CO. The second chamber contains only platinum and palladium. The extra oxygen from the secondary air improves the latter converter's ability to oxidize HC and CO in the second converter chamber.

The computer program for the control mode selection logic can be modified to include the conditions for controlling secondary air. In one configuration, the engine controller regulates the secondary air by using two solenoid valves similar to the EGR valve. One valve switches airflow to the air cleaner or to the exhaust system. The other valve switches airflow to the exhaust manifold or to the converter. The air routing is based on engine coolant temperature and air/fuel ratio. The control system diagram for secondary air is shown in Fig. 6.27.

EVAPORATIVE EMISSIONS CANISTER PURGE

In preemission controlled vehicles, the fuel stored in the fuel system tended to evaporate and release hydrocarbons (HCs) into the atmosphere. In contemporary vehicles, to reduce these HC emissions, the fuel tank is sealed and evaporative gases are collected by a charcoal filter in a canister. The collected fuel is released into the intake through a solenoid valve controlled by the computer. This normally is done during closed-loop operation to reduce fuel calculation complications in the open-loop mode.

AUTOMATIC SYSTEM ADJUSTMENT

Another important feature of microcomputer engine control systems is their ability to be programmed to adapt to parameter changes. Many control systems use this feature to enable the computer to modify lookup table values for computing open-loop air/fuel ratios. While the computer is

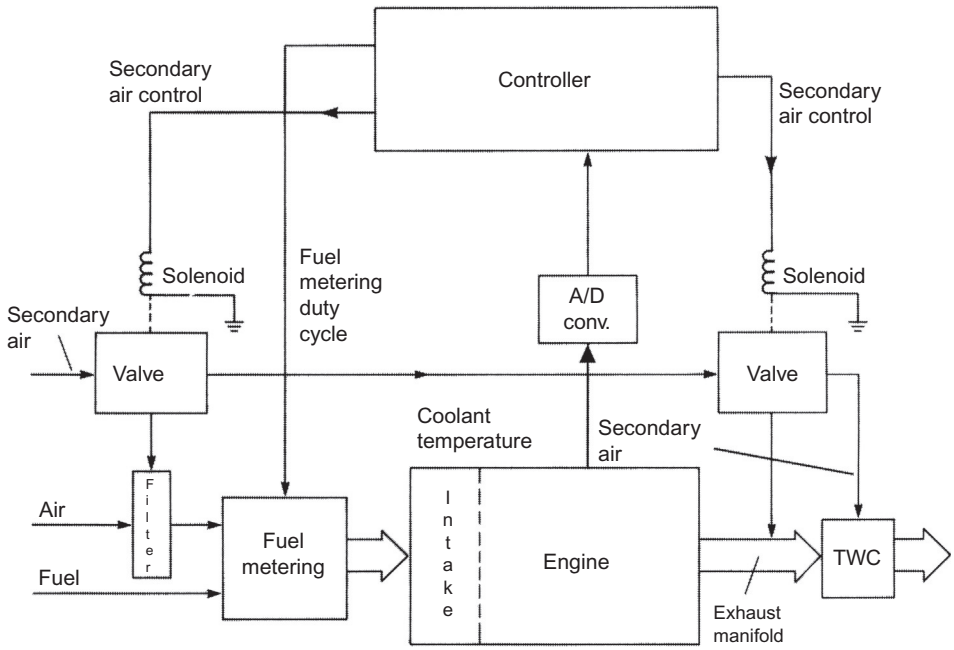


FIG. 6.27 Secondary air system.

in the closed-loop mode, the computer checks its open-loop calculated air/fuel ratios and compares them with the closed-loop average limit-cycle values. If they match closely, the open-loop lookup tables are unchanged. If the difference is large, the system controller corrects the lookup tables so that the open-loop values more closely match the closed-loop values. This updated open-loop lookup table is stored in separate memory (RAM), which is always powered directly by a car battery or a separate "keep alive" battery so that the new values are not lost, while the ignition key is turned off. The next time the engine is started, the new lookup table values will be used in the open-loop mode and will provide more accurate control of the air/fuel ratio than the unmodified values. This feature is very important because it allows the system controller to adjust to long-term changes in engine and fuel system conditions. This feature can be applied in individual subsystem control systems or in the fully integrated control system. If not available initially, it may be added to the system by modifying its control program.

SYSTEM DIAGNOSIS

Another important feature of microcomputer engine control systems is their ability to diagnose failures in their control systems or components and alert the operator. Sensor and actuator failures or misadjustments can be detected readily by the computer under certain operating conditions. For instance, the computer will detect a malfunctioning MAF sensor if the sensor's output goes above or below certain specified limits or fails to change for long periods of time. A prime example is the automatic adjustment

system just discussed. If the open-loop calculations consistently come up different from those indicated in closed-loop mode, the engine control computer may determine that one of the many sensors used in the open-loop calculations has experienced a calibration change or has failed completely.

If the computer detects the loss of a primary control sensor or actuator, it may switch to a different mode until the problem is repaired. The operator is notified of a failure by an indicator on the instrument panel (e.g., check engine indicator). Because of the flexibility of the microcomputer engine control system, additional diagnostic programs might be added to accommodate different engine models that contain more or fewer sensors. Chapter 11 discusses self-diagnosis in engine control systems. Keeping the system totally integrated gives the microcomputer controller access to more sensor inputs so they can be checked. Chapter 11 discusses system diagnosis in detail. Often, there is sufficient redundancy to permit suboptimal engine operation when a component has failed such that the vehicle can be driven to a repair facility in an operating mode that has been termed a "limp home mode."

SUMMARY OF CONTROL MODES

A summary of the control modes for a digital engine control system is presented below.

ENGINE CRANK (START)

The following list is a summary of the engine operations in the engine crank (starting) mode, wherein the primary control concern is rapid and reliable engine start:

1. Engine RPM at cranking speed
2. Engine coolant at relatively low temperature (cold start)
3. Air/fuel ratio low (cold start)
4. Spark retarded
5. EGR off
6. Secondary air to exhaust manifold
7. Fuel economy not closely controlled
8. Emissions not as closely controlled as during fully warmed engine

ENGINE WARM-UP

While the engine is warming up, the engine temperature is rising to its normal operating value. Here, the primary control concern is rapid and smooth engine warm-up. A summary of the engine operations during this period is as follows:

1. Engine RPM above cranking speed at command of driver
2. Engine coolant temperature rises to minimum threshold
3. Air/fuel ratio controlled versus engine temperature
4. Spark timing set by controller
5. EGR off
6. Heat supplied to HEGO
7. Secondary air to exhaust manifold

8. Fuel economy not as closely controlled as fully warmed engine
9. Emissions not as closely controlled as fully warmed engine

OPEN-LOOP CONTROL

The following list summarizes the engine operations when the engine is being controlled in an open-loop mode. This mode is used before the EGO sensor has reached the correct temperature for closed-loop operation. Fuel economy and emissions are closely controlled:

1. Engine RPM at command of driver (or idle speed control)
2. Engine temperature above warm-up threshold
3. Air/fuel ratio controlled by an open-loop system to 14.7
4. EGO sensor temperature less than minimum threshold
5. Heat supplied to HEGO
6. Spark timing set by controller
7. EGR controlled
8. Secondary air to catalytic converter
9. Fuel economy controlled
10. Emission controlled

CLOSED-LOOP CONTROL

For the closest control of emissions and fuel economy under various driving conditions, the electronic engine control system is in a closed loop. Fuel economy and emissions are controlled very tightly. The following is a summary of the engine operations during this period:

1. Engine RPM at command of driver (or idle speed control)
2. Engine temperature in normal range (above warm-up threshold)
3. Average air/fuel ratio controlled to 14.7, ± 0.05
4. EGO sensor's temperature above minimum threshold detected by a sensor output voltage indicating a rich mixture of air and fuel for a minimum amount of time
5. System returns to open loop if EGO sensor cools below minimum threshold or fails to indicate rich mixture for given length of time
6. EGR controlled
7. Secondary air to catalytic converter
8. Fuel economy tightly controlled
9. Emissions tightly controlled

HARD ACCELERATION

When the engine must be accelerated quickly or if the engine is under heavy load, it is in a special mode. Now, the engine controller is primarily concerned with providing maximum performance. Here is a summary of the operations under these conditions:

1. Driver asking for sharp increase in RPM or in engine power (via rapid throttle angle increase) and demanding maximum torque
2. Engine temperature in normal range

3. Air/fuel ratio rich mixture
4. EGO not in loop (very briefly)
5. EGR off
6. Secondary air to intake
7. Relatively poor fuel economy (relative to normal closed loop)
8. Relatively poor emission control (relative to normal closed loop)

DECELERATION AND IDLE

Slowing down, stopping, and idling are combined in another special mode. The engine controller is primarily concerned with reducing excess emissions during deceleration and keeping idle fuel consumption at a minimum. This engine operation is summarized in the following list:

1. RPM decreasing rapidly due to driver command or else held constant at idle
2. Engine temperature in normal range
3. Air/fuel ratio lean mixture
4. Special mode in deceleration to reduce emissions
5. Special mode in idle to keep RPM constant at idle as load varies due to air conditioner, automatic transmission engagement, etc.
6. EGR on
7. Secondary air to intake
8. Good fuel economy during deceleration
9. Possibly relatively poor fuel economy during idle but fuel consumption kept to minimum possible (except for HEV)

AUTOMATIC TRANSMISSION CONTROL

The vast majority of cars and light trucks sold in the United States are equipped with automatic transmissions. The majority of these transmissions are controlled electronically. The configuration of an automatic transmission consists of a torque converter and a sequence of planetary gear sets.

The transmission (whether automatic or manual) is a gear system that adjusts the ratio of engine speed to wheel speed. Essentially, the transmission enables the engine to operate within its optimal performance range regardless of the vehicle load or speed. It provides a gear ratio between the engine speed and vehicle speed such that the engine provides adequate power to drive the vehicle at any speed. Any gear system connecting a pair of shafts along which torque/power is transmitted is the mechanical equivalent of an electrical transformer. Just as a transformer can maximize the power transmitted from a source to a load, a gear system has the capability of maximizing the transfer of engine power to the load at the drive wheels while maintaining engine speed (under load) at acceptable values.

To accomplish optimal power transfer to the load with a manual transmission, the driver selects the correct gear ratio from a set of possible gear ratios (usually three to five for passenger cars). An automatic transmission selects the gear ratio by means of an automatic control system.

The configuration for an automatic transmission consists of a fluid coupling mechanism, known as a torque converter, and a system of planetary gear sets. The torque converter is formed from a pair of

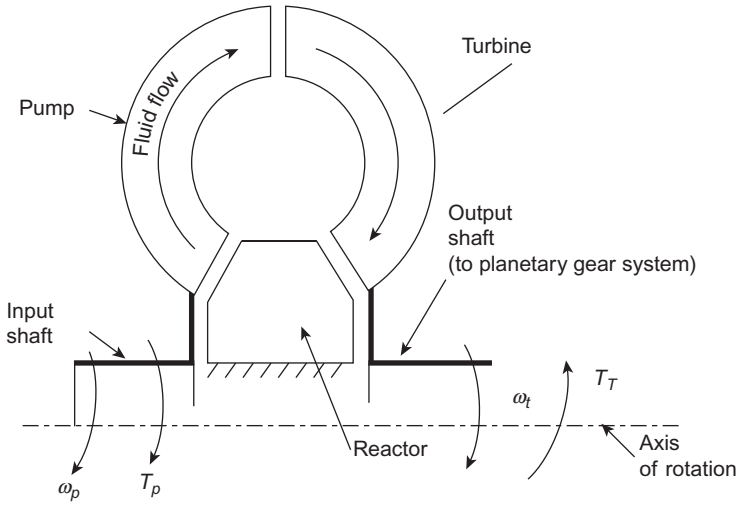


FIG. 6.28 Torque converter configuration.

structures of a semitoroidal shape (i.e., a donut-shaped object split along the plane of symmetry). Fig. 6.28 is a schematic sketch of a torque converter showing the two semitoroids.

One of the toroids is driven by the engine by the input shaft and is called the pump. The other is in close proximity and is called the turbine. Both the pump and the turbine have vanes that are nearly in axial planes. In addition, a series of vanes are fixed to the frame and are called the reactor. The entire structure is mounted in a fluid-tight chamber and is filled with a hydraulic fluid (i.e., transmission fluid). As the pump is rotated by the engine, the hydraulic fluid circulates as depicted by the arrows in Fig. 6.28. The fluid impinges on the turbine blades, imparting a torque to it. The torque converter provides a fluid coupling to transmit engine torque and power to the turbine from the engine. The torque that is applied to the pump portion of the torque converter is the engine brake torque (T_b). Denoting the torque applied to the output shaft by the turbine T_T, this latter torque is given by $T_T = T_R T_b$ where T_R is the torque multiplication factor of the torque converter. However, the properties of the torque converter are such that when the vehicle is stopped corresponding to a nonmoving turbine, the engine can continue to rotate (as is does when the vehicle is stopped with the engine running). Normally, with the vehicle stopped and the torque converter output shaft not rotating, the engine is at idle and producing minimal T_b. The turbine blades are in a stalled condition, and T_T is sufficiently low that only a small torque applied to the wheels by the brakes is capable of stopping the vehicle.

A detailed analytic model for a torque converter is given in a paper by Allen Kotwicki.[1] In this paper, it is explained that a torque converter is a form of fluid coupling device in which a reactor is added that is rigidly connected to the transmission housing and normally does not rotate. However, torque converter efficiency is improved whenever the torque reaction on the fluid is zero by allowing

[1] *Dynamic Models for Torque Converter Equipped Vehicles*, Allen Kotwicki, SAE paper # 820393, 1982.

the reactor to rotate freely. The torque converter is filled with transmission fluid that is caused to circulate through the pump-turbine-reactor by rotation of the pump by the engine crankshaft rotation. This fluid flows in an annular path as depicted in Fig. 6.28. The operating physical principle upon which a fluid coupling or a torque converter is based is that torque in any such system results from a time rate of change of angular momentum. In the reference cited above, it is shown that the torques of the pump T_p and turbine T_t are given by

$$\begin{aligned} T_p &= A\omega_p Q + BQ^2 \\ T_T &= A\omega_p Q - C\omega_t Q + DQ^2 \end{aligned} \tag{6.60}$$

where ω_p = the pump angular speed (rad/s), ω_t = the turbine angular speed (rad/s), and Q = the fluid volume flow rate

$$A = \rho R_{px}^2$$

$$B = \rho\left[\frac{R_{px}\tan\alpha_{px}}{A_{px}} - \frac{R_{rx}\tan\alpha_{rx}}{A_{rx}}\right]$$

$$C = \rho R_{tx}^2$$

and

$$D = \rho\left[\frac{R_{px}}{A_{px}}\tan\alpha_{px} - \frac{R_{rx}}{A_{tx}}\tan\alpha_{tx}\right]$$

where ρ is the transmission fluid density.

In these equations, a double subscript on a variable means first subscript $p \rightarrow$ pump, $r \rightarrow$ reactor, and $t \rightarrow$ turbine and the second subscript $e \rightarrow$ entrance and $x \rightarrow$ exit. The double-subscripted parameters have the following meaning:

A is the converter cross-sectional area normal to annular flow (p).
R is the radius from converter axis.
α is the element blade angle relative to axis.

It is further shown that the volume flow rate is given by

$$Q = -\frac{(H\omega_t - G\omega_p)}{2I} + \frac{\left[(H\omega_t - G\omega_p)^2 + 4I\left(E\omega_p^2 + F\omega_t^2\right)\right]^{\frac{1}{2}}}{2I} \tag{6.61}$$

where E, F, G, H, and I are constants given in the cited reference. In this reference, empirical evaluation of coefficients for a first-order linear regression-based polynomial for Q of the form is developed:

$$Q \approx \alpha_1\omega_p + \beta\omega_t$$

where $\omega_t \cong S\omega_p$ is assumed
where S is the speed ratio.

Using this approximation, it is shown in the reference that the torque ratio T_R is given by

$$T_R = \frac{T_T}{T_p} = \frac{(A + D\alpha_1)\omega_p + (D\beta - C)\omega_t}{(A + B\alpha_1)\omega_p + B\beta\omega_t} \tag{6.62}$$

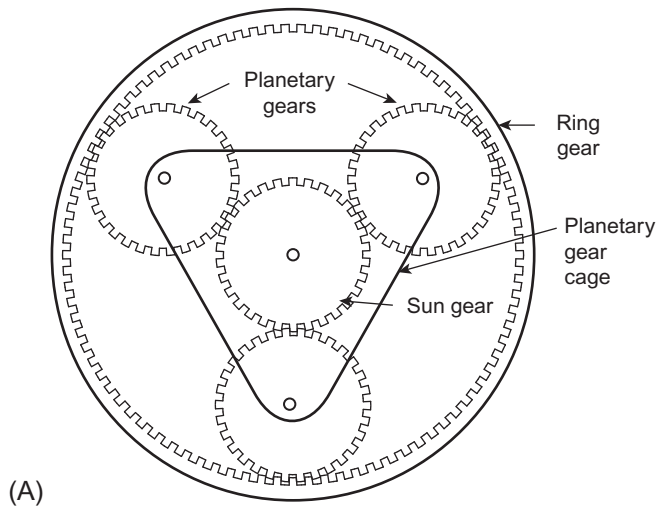
where

$$\alpha_1 = \frac{E}{\sqrt{I(E + FG^2/H^2)}} + \frac{G}{2I}$$

$$\beta = \frac{FG}{H\sqrt{I(E + FG^2/H^2)}} - \frac{H}{2I}$$

This simplified model is shown in the reference to correlate well with experimental data and is normally sufficient for the development of transmission controls.

The planetary gear system consists of a set of three types of gears connected together as depicted in Fig. 6.29A. The inner gear is known as the sun gear. There are three gears meshed with the same gear at equal angles, which are known as planetary gears. These three gears are tied together with a cage that



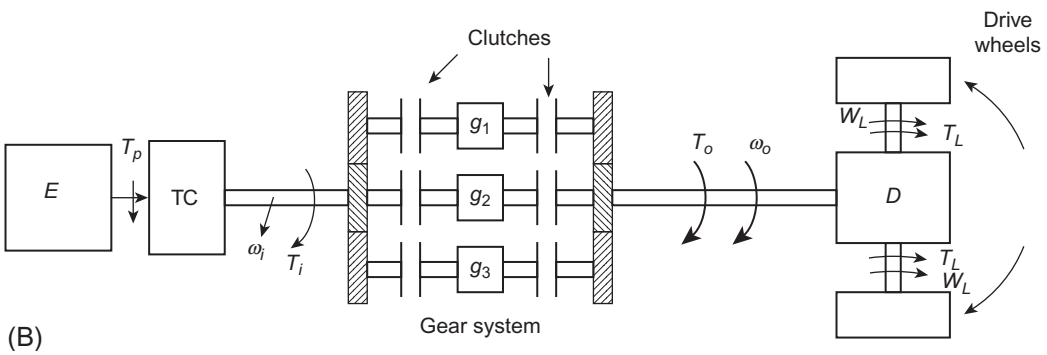


FIG. 6.29 Schematic automatic transmission configuration. (A) Planetary gear configuration; (B) Illustrative powertrain configuration.

supports their axles. The third gear, known as a ring gear, is a section of a cylinder with the gear teeth on the inside. The ring gear meshes with the three planetary gears.

In operation, one or more of these gear systems are held fixed to the transmission housing via a set of hydraulically actuated clutches. The action of the planetary gear system is determined by which set or sets of clutches are activated. For example, if the ring gear is held fixed and input power (torque) is applied to the sun gear, the planetary gears rotate in the same direction as the sun gear but at an increased torque. We denote the input torque applied to the sun gear and the angular speed of the shaft driving this gear system by T_i and ω_i, respectively. The output torque and its speed are denoted T_o and ω_o, respectively. A model for this gear system is given by

$$\begin{aligned} T_o &= gT_i \\ \omega_o &= \omega_i/g \end{aligned} \tag{6.63}$$

where g is the gear ratio:

$$g = N_p/N_s$$

N_s is the number of teeth on the sun gear, and N_p is the number of teeth on a planetary gear.

If the planetary gear cage is fixed, then the sun gear drives the ring gear in the opposite direction as is done when the transmission is in reverse. If all three sets of gears are held fixed to each other rather than the transmission housing, then direct drive (gear ratio = 1) is achieved.

A typical automatic transmission has a number of planetary gear systems (denoted g_1, g_2, and g_3 in Fig. 6.29B), each with its own set of hydraulically actuated clutches as depicted schematically in Fig. 6.29B. In an electronically controlled automatic transmission, the clutches are electrically or electrohydraulically actuated via solenoid-type actuators such as are described in Chapter 5.

Most automatic transmissions have three forward gear ratios, although a few have two and some have four or more and all have reverse. A properly used manual transmission normally has efficiency advantages over an automatic transmission (because of power losses in the torque converter), but the automatic transmission is the most commonly used transmission for passenger automobiles in the United States. In the past, automatic transmissions have been controlled by a hydraulic and pneumatic system, but it is common in contemporary vehicles to use electronic controls as part of an integrated powertrain control system. The control system must determine the correct gear ratio by sensing the driver-selected command, accelerator pedal position, engine load, and vehicle motion. Once again, as in the case of electronic engine control, the electronic transmission control can optimize transmission control. However, since the engine and transmission function together as a power-producing unit, it is sensible to control both components in a single electronic controller. The proper gear ratio is actually computed in the electronic transmission control portion of the powertrain control system.

Fig. 6.29B depicts schematically the powertrain denoting the engine E, the torque converter (TC), the gear system, the differential D (having gear ratio g_D), and the axles with the drive wheels (which could be front or rear). The configuration and operating principles of the differential are explained later in this chapter. For simplicity, it is convenient to assume that both right and left drive wheels (or all four drive wheels for four-wheel drive) are identical and present a combined load torque T_L to the drive axle. In this case, the transmission output torque T_o is given by

$$T_o = T_L/g_D$$

The gear system consists of a set of planetary gear units each having a gear ratio g_n ($n = 1,2,\ldots,N$). The appropriate gear is selected by the control system, which operates the correct set of clutches via an

electrohydraulic actuator (e.g., solenoid-operated valve supplying transmission fluid under pressure to a set of sprag clutches). For gear systems connected in series, the total gear ratio g from the torque converter output to the load is given by

$$g_T = g_D \prod_{n=1}^{N} g_n \tag{6.64}$$

Otherwise, for a parallel connected system of gears as shown in Fig. 6.29B, the gear ratio is given by

$$g_T = g_D g_n \tag{6.65}$$

Although there are many possible powertrain control modes depending upon vehicle-operating conditions and driver command, an illustrative example mode is maximizing the power delivered to the load (drive wheels) for a given engine brake power ($P_b = T_b \omega_e$). A simple approximate and artificial model for explaining maximum power transfer across a gear system is based upon an electrical equivalent circuit in which torque is analogous to voltage (V) and angular speed of the shaft along which the torque is applied is analogous to current I. As in the case of mechanical power, electric power P_e for purely resistive circuits is given by

$$P_e = VI$$

The impedance z is given by

$$z = V/I$$

For an AC electrical circuit, the power delivered to a load through a transformer of turns ratio $N_2/N_1 = r$ is maximized when

$$r = \sqrt{\frac{R_L}{R_s}}$$

R_s = source resistance and R_L = load resistance.

The mechanical equivalent of impedance Z_m is given by T/ω. For the sake of this artificial model, it is assumed that the engine available power is fixed by the throttle angle and that internal frictional losses are proportional to ω_e. Using this model, a gear system with gear ratio g is analogous to a transformer with turn ratio r.

Based on the transformer analogy to a gear train, the gear ratio, which maximizes this transfer of engine power to load power ($P_L = T_L \omega_L$) g^* is given approximately by

$$g^* = \sqrt{\frac{T_L/\omega_L}{T_b/\omega_e}}$$

In this simple powertrain model, the controller is programmed to select the nearest available gear ratio from the set of possible choices to g^*. However, in practice, the gear selection criteria are based on optimizing engine fuel efficiency, except under heavy acceleration conditions for which the gear selection would normally be such that the engine operates near peak torque.

Another control mode for the transmission is to maximize drive axle torque T_L, thereby maximizing vehicle acceleration whenever the driver command yields wide open throttle (WOT). This mode calls

for the maximum available gear ratio subject to the constraint that engine RPM remains near the point for maximum brake torque.

The relevant clutches are activated by the pressure of transmission fluid acting on pistonlike mechanisms. The pressure is switched on at the appropriate clutch via solenoid-activated valves that are supplied with automatic transmission fluid under pressure. The solenoids are actuators that receive an electrical signal from the powertrain control system as explained in Chapter 5.

During normal driving, the electronic transmission controller determines the desired gear ratio from measurements of engine load and RPM and transmission output shaft RPM. These RPM measurements are made using noncontacting angular speed sensors (usually magnetic in nature) as explained in Chapter 5. Once this desired gear ratio is determined, the set of clutches to be activated is uniquely determined, and control signals are sent to the appropriate clutches.

Normally, the highest gear ratio (i.e., ratio of input shaft speed to output shaft speed) is desired when the vehicle is at low speed such as in accelerating from a stop. As vehicle speed increases from a stop, a switching level will be reached at which the next lowest gear ratio is selected. This switching (gear-changing) threshold is an increasing function of load as measured by the MAF or MAP sensor.

At times (particularly under steady vehicle speed conditions), the driver demands increasing engine power (e.g., for heavy acceleration). In this case, the controller shifts to a higher gear ratio, resulting in higher acceleration than would be possible in the previous gear setting. At a steady-cruise condition, the transmission gear ratio is unity, and the total gear ratio from engine to drive wheels is g_D (i.e., differential gear ratio). The functional relationship between gear ratio and operating condition is often termed the "shift schedule," which is programmed into ROM.

TORQUE CONVERTER LOCK-UP CONTROL

As explained above, automatic transmissions use a hydraulic or fluid coupling to transmit engine power to the wheels. There is some relatively small power loss in the TC such that the fluid coupling is less efficient than the nonslip coupling of a pressure-plate manual clutch used with a manual transmission. Thus, fuel economy is usually lower with an automatic transmission than with a standard transmission. This problem has been partially remedied by placing a clutch functionally similar to a standard pressure-plate clutch inside the torque converter of the automatic transmission and engaging it during periods of steady cruise. This enables the automatic transmission to provide fuel economy near that of a manual transmission and still retain the automatic shifting convenience.

The torque-converter-locking clutch (TCC) is activated by a lockup solenoid controlled by the engine control system computer. The computer determines when a period of steady cruise exists from throttle position and vehicle speed changes. It pulls in the locking clutch and keeps it engaged until it senses conditions that call for disengagement. This condition is known as "torque converter lockup."

DIFFERENTIAL AND TRACTION CONTROL

The transmission output shaft is coupled to the drive axles via the differential or transaxle. The differential is a necessary component of the drivetrain because the left and right drive wheels turn at different speeds whenever the car moves along a curve (e.g., turning a corner). Whenever a car is executing a turn, the outside drive wheel rotates at a higher angular speed than the inside wheel. The differential achieves this function permitting both wheels to propel the vehicle. Fig. 6.30 depicts the configuration for a differential.

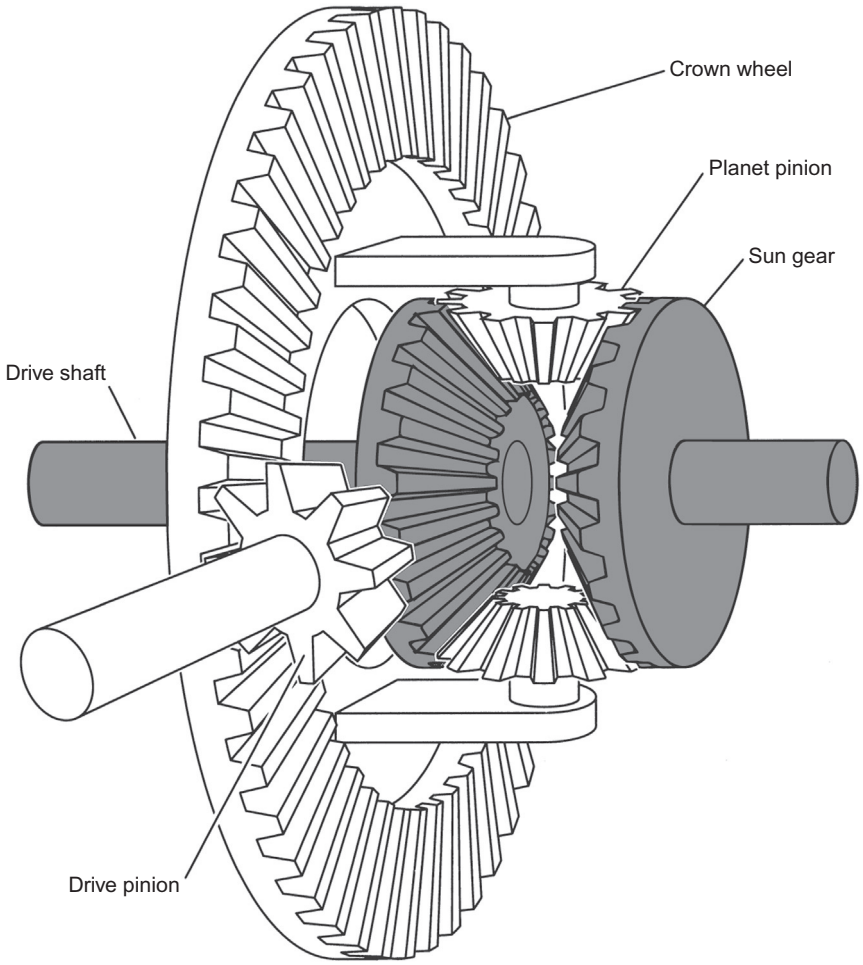


FIG. 6.30 Differential configuration.

A transaxle incorporates a gear system structure that is similar to the differential depicted in Fig. 6.30. Unfortunately, wherever there is a large difference between the tire/road friction from left to right, the differential will tend to spin the low-friction wheel. An extreme example of this occurs whenever one drive wheel is on ice and the other is on dry road. In this case, the tire on the ice side will spin, and the wheel on the dry side will not. Typically, the vehicle will not move in such circumstances.

A majority of contemporary vehicles are equipped with so-called traction control devices that can overcome this disadvantage of the differential. One method of achieving traction control involves differentials that incorporate electrohydraulic solenoid-activated clutches somewhat similar to those used in an automatic transmission that can "lock" the differential, permitting power to be delivered to both drive wheels. It is only desirable to activate these clutches in certain conditions and to disable them during normal driving, permitting the differential to perform its intended task.

An alternative traction control is available in vehicles having an automatic brake system. In this case, a brake is automatically applied to the spinning wheel, which causes drive torque to be applied to the nonspinning drive wheel. The details of this topic are explained in Chapter 10.

A traction control system incorporates sensors for measuring wheel speed and a controller that determines the wheel slip condition based on these relative speeds. Wherever a wheelspin condition is detected, the controller sends electrical signals to the solenoids, thereby activating the clutches to eliminate the wheel slip.

HYBRID ELECTRIC VEHICLE POWERTRAIN CONTROL

The concept of a HEV, in which propulsive power comes from an internal combustion engine (ICE) and an electric motor (EM), has emission and fuel advantages relative to a conventional vehicle powered only by an ICE. As explained in Chapter 4, the hybrid vehicle combines the low (ideally zero) emissions of an electric vehicle with the range and performance capabilities of IC-engine-powered cars. However, optimization of emission performance and/or fuel economy is a complex control problem.

There are different types of hybrid electric vehicles based upon the degree of hybridization. A vehicle that can operate on either the ICE or the electric propulsion or a combination of both is known as a full hybrid. In order to have any practical range for electric propulsion only, the vehicle must have a suitable very high-capacity battery pack. This battery pack is capable of storing far more energy than a conventional storage battery found in ICE only vehicles.

On the other hand, there are certain hybrids that are incapable of electric propulsion only. These vehicles, which are commonly called "mild hybrids," require the ICE for some of their propulsion. In one configuration, a mild hybrid has an ICE connected to a motor that serves several functions including starting the ICE, adding a power boost to the ICE, and regenerative braking to recover and store some energy during deceleration. In regenerative braking, the electric motor acts as a generator that receives its mechanical drive power from the vehicle momentum and delivering its output electric power to the battery pack. The discussion of induction motors in Chapter 5 explains this operation of a motor acting as a generator.

There are numerous issues and considerations involved in hybrid vehicle powertrain control, including the efficiencies of the IC engine and electric motor as a function of operating condition; the size of the vehicle and the power capacity of the IC engine and electric motor; the storage capacity and state of charge of the battery pack; accessory load characteristics of the vehicle; and, finally, the driving characteristics of the driver. With respect to this latter issue, it would be possible to optimize vehicle emissions and performance if the exact route, including vehicle speed, acceleration, deceleration, road inclination, and wind characteristics, could be programmed into the control memory before any trip was to begin. As explained in Chapter 12, certain levels of autonomous vehicles have some of the required capabilities to optimize the trip from a given starting point to destination. On the other hand, unknown variables (e.g., traffic congestion) make it somewhat impractical to achieve optimum performance by preprogramming only. However, by monitoring instantaneous vehicle operation, it is possible to achieve good, though suboptimal, vehicle performance and emissions.

Depending on operating conditions, the controller in a full hybrid can command pure electric vehicle operation, pure IC engine operation, or a combination. Whenever the IC engine is operating, the controller should attempt to keep it at its peak efficiency.

Certain special operating conditions should be noted. For example, the IC engine is stopped wherever the vehicle is stopped. Clearly, such stoppage benefits vehicle fuel economy and improves air quality when the vehicle is driven in dense traffic with long stoppages such as those that occur while driving in large urban areas.

There are two major types of hybrid electric vehicles depending on the mechanism for coupling the IC engine (ICE) and the electric motor (EM). Fig. 6.31 is a schematic representation of one hybrid vehicle configuration known as a series hybrid vehicle (SHV).

In this SHV, the ICE drives a generator (G) and has no direct mechanical connection to the drive axles. The vehicle is propelled by the electric motor (EM), which receives its input electric power from a high-voltage bus. This bus, in turn, receives its power either from the engine-driven generator (for ICE propulsion) or from the battery pack (for EM propulsion), or from a combination of the two. In this figure, mechanical power is denoted MP and electric power EP. The mechanical connection from the EM to the transaxle (T/A) provides propulsive power to the drive wheels (DWs). The term transaxle refers to the entire drive system from the EM to the drive wheels.

Fig. 6.32 is a schematic of a hybrid vehicle type known as a parallel hybrid. The parallel hybrid of Fig. 6.32 can operate with ICE alone by engaging both solenoid-operated clutches on either side of the

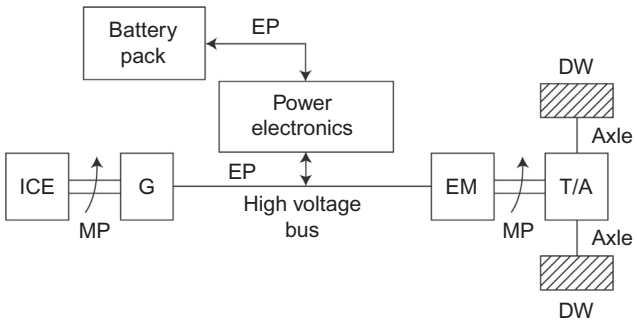


FIG. 6.31 Series hybrid electric vehicle (HEV) schematic.

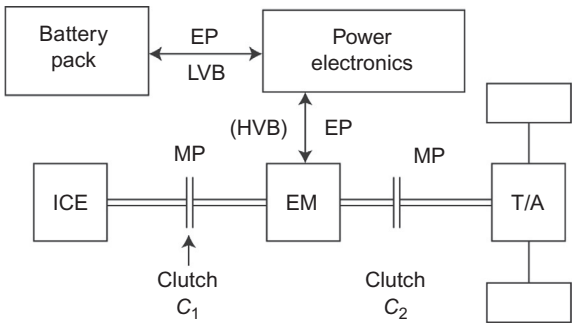


FIG. 6.32 Parallel hybrid schematic.

EM but with no electric power supplied to the EM. In this case, the MP supplied by the ICE directly drives the transaxle T/A, and the EM rotor spins essentially without any mechanical drag. This hybrid vehicle can also operate with the EM supplying propulsive power by switching off the ICE, disengaging clutch C_1, engaging clutch C_2, and providing electric power to the EM from the high-voltage bus (HVB). Of course, if both ICE and EM are to produce propulsive power, then both clutches are engaged. Not shown in Fig. 6.32 is a separate controller for the motor. Also not shown in this figure but discussed later in this section is the powertrain controller that optimizes performance and emissions for the overall vehicle and engages/disengages clutches as required.

The HEV of Fig. 6.33 operates similarly to that of Fig. 6.32 except that mechanical power from ICE and EM is combined in a mechanism denoted coupler. For the system of Fig. 6.33, pure ICE propulsion involves engaging clutch C_1, disengaging clutch C_2, and providing no electric power to the EM. Alternatively, pure EM propulsion involves disengaging clutch C_1, switching off the ICE, engaging clutch C_2, and providing electric power to the EM via the high-voltage bus (HVB). Simultaneous ICE and EM propulsion involves running the ICE, providing electric power to the EM, and engaging both clutches.

In principle, any type of electric motor could be used to provide the electric propulsion in a hybrid vehicle. However, for the purpose of explanation, two main types are presented in this chapter: the brushless DC motor and the induction motor. Both are explained and modeled in Chapter 5. It should be recalled that the brushless DC motor incorporates a permanent magnet rotor normally with multiple poles. The stator has multiple windings that are excited by AC currents. Typically, the stator windings are arranged for three-phase operation.

However, the stored electric power in a hybrid vehicle is DC (from the battery pack). The frequency condition for this type of motor requires that the rotational frequency ω_m be identical to the stator excitation frequency ω_s since the rotor excitation is at $\omega_r = 0$.

Operation of the brushless DC motor in the exemplary hybrid vehicle during electric propulsion requires that an electrical system converts the stored DC electric power to poly-phase (e.g., three-phase) AC power. This conversion is accomplished in a motor control system that creates an electric control signal at frequency ω_s in addition to power switching circuits (normally implemented via

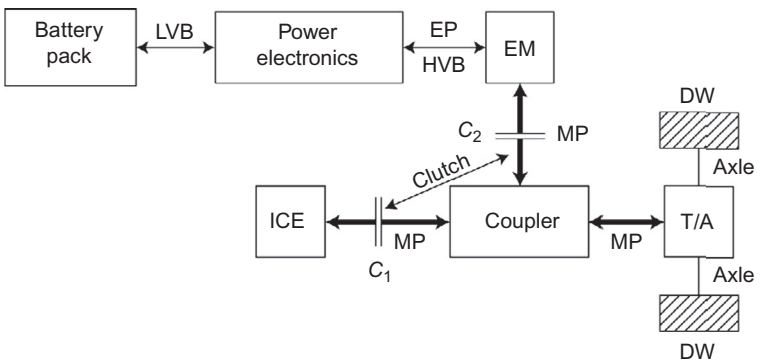


FIG. 6.33 HEV with mechanical coupler.

high-power switching transistors). Ideally, the stator excitation should be three sinusoidal voltages of equal amplitude, which in phasor notation are given by

$$\begin{aligned} V_A &= Ve^{j\omega_s t} \\ V_B &= Ve^{j(\omega_s t + 2\pi/3)} \\ V_C &= Ve^{j(\omega_s t + 4\pi/3)} \end{aligned} \tag{6.66}$$

However, in practice, the excitation waveforms are not sinusoidal. Rather, they are more often of a form of square or trapezoidal waveform. Motor control requires correct phasing relative to the orientation of the rotor. Such phasing requires a noncontacting rotor position sensor (e.g., Hall effect; see Chapter 5).

In order to provide torque and power levels required for hybrid vehicle operation, a brushless DC motor is made using powerful magnets having so-called rare-earth elements. A typical magnet for a hybrid vehicle brushless DC motor is made of an alloy of iron, boron, and, the relatively expensive rare-earth element, neodymium.

A brushless DC motor can also function as an alternator. The motion of the rotor creates a time-varying flux linking the stator turns Φ_A, Φ_B, and Φ_C. This time-varying flux linkage, in turn, creates a voltage given by V_A, V_B, and V_C in each winding:

$$\begin{aligned} V_A &= \frac{d\Phi_A}{dt} \\ V_B &= V_A e^{j(2\pi/3)} \\ V_C &= V_A e^{j(4\pi/3)} \end{aligned}$$

The zero phase corresponds to the rotor rotation angle for which Φ_A is a maximum.

These voltages can be converted to DC using a set of transformers (to achieve correct voltage levels) and rectifier circuits (see Chapter 2). The corresponding DC power can be supplied to the battery pack to increase its state of charge. In this way, the motor acting as a generator can provide braking torque to decelerate the vehicle and recover some of the vehicle kinetic energy that would otherwise be dissipated in brakes. Such generator action is known as regenerative braking.

Other electric motor types also have application in hybrid vehicle propulsion. In Chapter 5, the induction motor was explained. Induction motors of high torque/power output and high efficiency can be built without requiring rare-earth magnetic material. A model for an induction motor was presented in Chapter 5 where it was shown that the frequency condition for average torque generation at any given motor RPM is automatically satisfied.

Induction motors for hybrid vehicle use are normally three phase, meaning three separate windings (one for each phase) are required for both stator and rotor. In Chapter 5, it was shown that the torque produced by the induction machine (with current excitation amplitude I_s) is given by

$$T_e = \frac{(\omega_S - \omega_m)M^2 R_r^2 I_S^2}{R_r^2 + (\omega_S - \omega_m)^2 L_r^2} \tag{6.67}$$

where ω_s is the excitation frequency and ω_m is the motor rotational frequency.

All parameters in this model for T_e are defined in Chapter 5. It is also shown in Chapter 5 that the steady-state motor speed for a given excitation is the motor angular speed ω_o at which the motor torque $T_e(\omega_o)$ balances the load torque T_L:

$$T_e(\omega_o) = T_L(\omega_o) \tag{6.68}$$

This point is illustrated for a hypothetical hybrid vehicle being propelled solely by an induction motor. The load torque at the motor output is proportional to the force F_V required to move the vehicle at the commanded speed.

We consider first a hybrid vehicle moving along at a steady speed on a straight, level road. There are two primary contributions to F_V, tire rolling resistance F_{rr} and aerodynamic drag D. The rolling resistance is essentially independent of vehicle speed but is proportional to vehicle weight and varies as a decreasing function of tire pressure. If we assume that all tires are equally inflated, then the total rolling resistance force is given by

$$F_{rr} = \mu_{rr} W_V$$

where W_V is the vehicle weight and μ_{rr} is the coefficient of rolling resistance of tires. The coefficient μ_{rr} is generally in the range $0.02 \leq \mu_{\text{rr}} \leq 0.04$.

The aerodynamic drag D is given by

$$D = \frac{\rho}{2} C_D S_{\text{ref}} V^2$$

where ρ is the local air density (kg/m^3 or slug/ft^3), C_D is the drag coefficient, S_{ref} is the reference area (m^2 or ft^2), and V is the vehicle speed (m/s or ft/s).

The reference area is an arbitrary choice that ultimately determines the value for C_D. It is common practice to choose S_{ref} as the vehicle projected area on a vertical plane normal to the vehicle plane of symmetry. The force necessary to move the vehicle along a straight, level road at a constant speed V is given by

$$F_V = \mu_{rr} W_V + \tfrac{1}{2} \rho C_D S_{\text{ref}} V^2$$

The above expression for F_V is valid for a level road. Whenever the vehicle encounters a nonzero slope (i.e., along a hill), this force includes a term that is proportional to the vehicle weight and the slope of the hill. For a vehicle traveling along a road with a slope (relative to horizontal) of angle θ, the total force F_V is given by

$$F_V = \mu_{rr} W_V + \frac{1}{2} \rho C_D S_{\text{ref}} V^2 + W_V \sin\theta$$

Thus, a road with nonzero slope can shift load torque on the motor (T_L) up or down depending upon whether sign (θ) is + or −, respectively.

In the hypothetical example, the induction motor drives the vehicle wheels through a transmission and differential such that ω_m is proportional to V. The load torque at the motor output T_V is proportional to the force F_V (6.67):

$$T_V = r_T F_V / g_v \tag{6.69}$$

where g_V is the gear ratio from motor to drive wheels and r_T is the tire effective radius.

Fig. 6.34 is a plot of normalized motor torque T_m and load torque T_L (normalized to the maximum motor torque T_{max}) versus the ratio ω_m/ω_s where

$$T_m = \frac{T_e}{T_{\text{max}}}$$

$$T_L = \frac{T_V}{T_{\text{max}}}$$

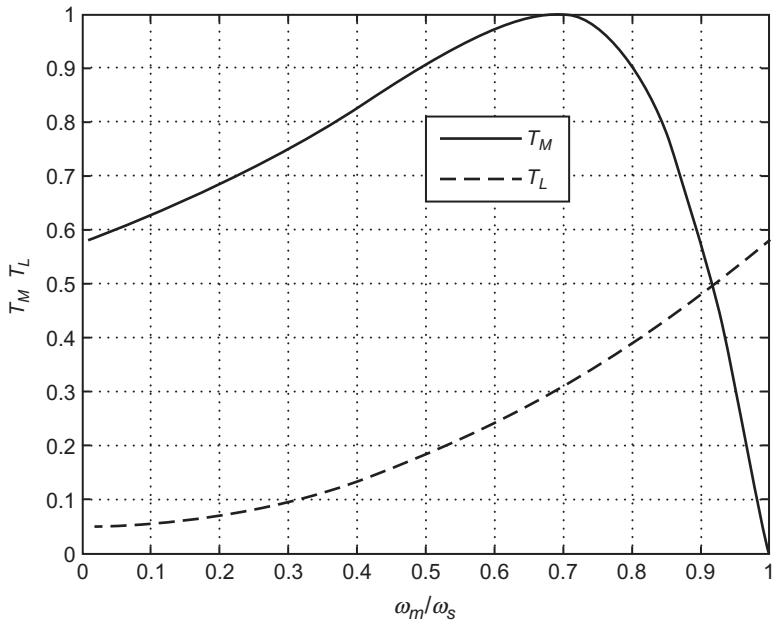


FIG. 6.34 Normalized motor torque T_M versus normalized load torque T_L.

where, for a given excitation, $T_{\max}$ is defined as

$$T_{\max} = \max_{\omega_m}(T_e)$$

The steady-state operating motor speed is at the intersection of these two curves (i.e., at $\omega_m/\omega_s \simeq 0.92$). A change in load torque (e.g., due to a nonzero road slope) causes the load curve to shift to a new motor operating point. The system is stable as long as

$$T_L/T_{\max} < 1$$

The efficiency of the induction motor is influenced, in part, by the steady-state operating point. In general, as long as the steady-state operating point (i.e., $\omega_m = \omega_o$) is in the negative slope region of $T_e(\omega_m)$ (and operation is stable), the motor produces torque that varies in proportion to slip s. However, motor efficiency varies inversely with slip as explained in Chapter 5.

The induction motor controller can regulate $T_e(\omega_m)$ via the excitation frequency (ω_s) and current amplitude I_s or motor voltage V_s. One hypothetical control strategy would vary the excitation and synchronous excitation frequency (ω_s) to optimize the motor efficiency. However, there are many other factors that influence the overall vehicle efficiency including the choice of ICE and/or electric propulsion, battery status, and vehicle-operating conditions and driving patterns (e.g., urban or highway).

The current that provides the induction motor excitation I_s is determined by the source voltage V_s and motor impedance. Normally, motor control is preferably done via regulation of V_s directly rather than via I_s. We consider next the model for the motor torque based upon the excitation voltage V_s.

The stator current magnitude I_s is related to the complex terminal voltage amplitude V_s. For sinusoidal excitation and using the parameter notation for induction motors from Chapter 5, the relationship between V_s and I_s is given by Eq. (6.68)

$$V_s = j\omega_s L_s I_s + \frac{\omega_s^2 M^2 I_s (R_r/s)}{(R_r/s)^2 + \omega_s^2 L_r^2} - j\frac{\omega_s^3 M^2 L_r I_s}{(R_r/s)^2 + \omega_s^2 L_s^2} \tag{6.70}$$

This expression gives the voltage/current relationships for each phase. See Chapter 5 for the definitions of all parameters. Solving the above equation for I_s and substituting it into the equation for motor torque yield

$$T_e = \frac{(M^2/\omega_S L_s L_r)(L_r/L_s)(R_r/s)V_s^2}{\left[\omega_s\left(1 - \frac{M^2}{L_r L_s}\right)L_r\right]^2 + (R_r/s)^2} \tag{6.71}$$

The above equation provides a basis for motor torque control in hybrid vehicle applications.

For an induction motor at constant supply voltage amplitude V_s, the slip s will vary until the motor torque is the same as load torque T_L:

$$T_e(s) = T_L$$

There is a family of curves of $T_e(s)$ for each excitation voltage that is similar in form to that given for current excitation. Fig. 6.34 presents normalized versions of motor and load torque. Normal operation of an induction motor is in a region in which

$$\frac{dT_e}{d\omega_s} < 0 \tag{6.72}$$

and s is relatively small (i.e., $\omega_m \lesssim \omega_s$). In this region, the motor torque is given approximately by

$$T_e \cong \left(\frac{M^2}{\omega_s L_s^2}\right)\frac{s}{R_r}V_s^2 \tag{6.73}$$

On the other hand, when slip is relatively large, the torque can be shown to be given approximately by

$$T_e \cong \frac{M^2 R_r V_s^2}{(L_r L_s)^2 \omega_s^3 \left(1 - \frac{M^2}{L_r L_s}\right)^2 s} \tag{6.74}$$

The above approximate expressions can be used to control motor torque for the two distinct regions of operation. In any event, the motor control can regulate torque by controlling excitation voltage and frequency ω_s as explained later in this chapter.

For either series or parallel hybrid vehicle, dynamic braking is possible during vehicle deceleration, with the EM acting as a generator. The EM/generator supplies power to the high-voltage bus, which is converted to the low-voltage bus (LVB) voltage level by the power electronic subsystem. In this deceleration circumstance, the energy that began as vehicle kinetic energy is recovered with the motor acting as a generator and is stored in the battery pack. This storage of energy occurs as an increase in the state of charge (SOC) of the battery pack. This process (regenerative braking) was discussed above with respect to the brushless DC motor but applies equally well with an induction motor drive system.

In addition to the lead-acid battery in common use today, there are new energy storage means including nickel-metal hydride (NiMH), lithium ion, and even special capacitors called ultracaps. Each of these electrical energy storage technologies has advantages and disadvantages for hybrid vehicle application.

The battery pack has a maximum SOC that is fixed by its capacity. Dynamic braking is available as an energy recovery strategy as long as SOC is below its maximum value. Nevertheless, dynamic braking is an important part of hybrid vehicle fuel efficiency. It is the only way some of the energy supplied by the ICE and/or EM can be recovered when the vehicle is traveling along a road with a negative slope or is decelerating instead of being dissipated in the vehicle brakes.

For each battery type, there is a maximum rated stored charge Q_r that is determined by construction. The SOC for the battery is normally expressed by the instantaneous Q expressed as a fraction of Q_r. A storage battery is, in effect, a type of nonlinear capacitor (with a nonlinear source resistance) in which the open-circuit voltage V_{oc} is a function of stored charge Q:

$$V_{oc} = f(Q)$$

The design and/or analysis of electrical systems or components that are powered by the battery requires a model for the vehicle storage battery. This model is best represented by a so-called equivalent circuit. A somewhat simplified illustrative battery equivalent circuit is given near the end of Appendix A. The storage of the energy recovered during dynamic braking requires that the corresponding electrical energy be direct current and at a voltage compatible with the battery pack. Since most automotive systems apart from the motor operate in the range of 12–14 V, for convenience they are termed 12 V batteries. A common battery pack might consist of a connection of multiple 12 V batteries.

Conversion of electric power from one voltage level V_1 to a second V_2 is straightforward using a transformer as long as this power is alternating current. Fig. 6.35 schematically illustrates transformer structure and the conversion of voltages from one level to another.

A transformer consists of a core of highly magnetically permeable material (a ferromagnetic material) around which a pair of closely wrapped coils are formed. One coil (termed the primary) consists of N_1 turns, and the other (termed the secondary) consists of N_2 turns. In a well-designed transformer, essentially all of the magnetic flux in the core links all turns in both coils.

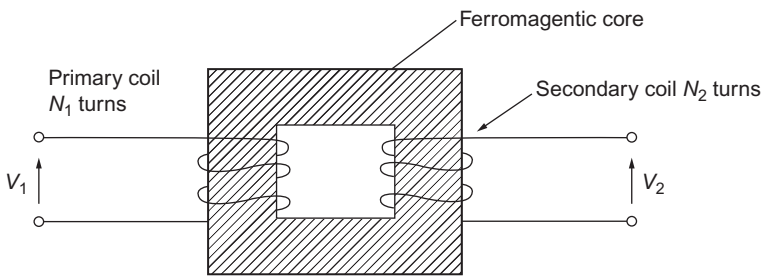


FIG. 6.35 Transformer configuration.

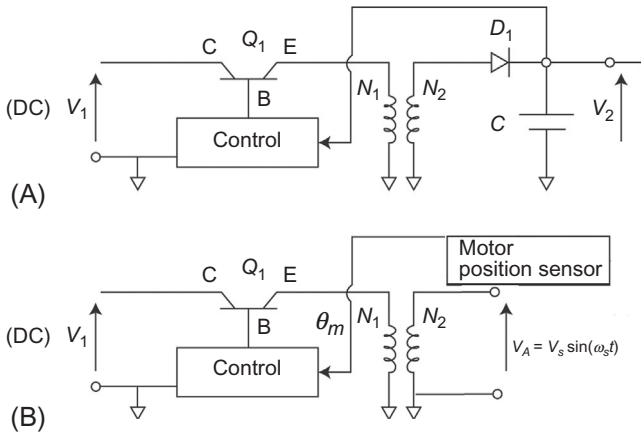


FIG. 6.36 Voltage conversion circuit. (A) Schematic for DC voltage conversion; (B) Circuit for generating AC from DC voltage.

Assuming (arbitrarily) that AC electric power comes from a source (e.g., an AC generator) at peak voltage V_1, then the power flowing from the transformer secondary to a load will be at a peak voltage V_2 where

$$V_2 = (N_2/N_1)V_1$$

Conversion of DC electric power from one voltage to another can be accomplished using a transformer only if the DC power is first converted to AC and then converted back to DC as explained below. Fig. 6.36A is a greatly simplified schematic of a DC-to-DC converter in which a transistor is used to convert an input DC signal to AC that is sent to a transformer for conversion to a different voltage.

The control electronics supply a pulsating signal to the base B of transistor Q_1, alternately switching it on and off. When Q_1 is on (i.e., conducting), voltage V_1 is applied to the transformer primary (i.e., N_1). When Q_1 is off (i.e., nonconducting), transformer primary voltage is zero. In this case, the pulsating AC voltage that is alternately V_1 and 0 is applied to the primary results in an AC voltage in the secondary that is essentially N_2/N_1 times the primary voltage. This secondary voltage is converted to DC by rectification using diode D_1 and filtering via capacitor C (see Chapter 2). The secondary voltage is fed back to the control electronics, which varies the relative ON and OFF times to maintain V_2 at the desired level.

A variation of the circuit of Fig. 6.36A appears in the power electronics module for conversion between the battery pack or from an ICE-driven generator and the hybrid vehicle motor driver. Regardless of the type of motor used, the generation of the voltages that provide the motor excitation (i.e., V_A, V_B, and V_C for a three-phase motor) can be accomplished using circuits of the configuration shown in Fig. 6.36B. Although this figure depicts a single phase (i.e., V_A), a separate driver transistor such as Q_1 along with a transformer (of N_1 primary and N_2 secondary turns) is required for each phase. The control electronics internally compute the signals that control the phases (i.e., 0, $2\pi/3$, and $4\pi/3$) of the remaining three phases. In order to achieve correct phasing for the motor, it is necessary to measure motor instantaneous angular position (θ_m) using an angle-measuring sensor (see Chapter 5). This control

is normally implemented in the powertrain control system. Of course, the specific details of the relevant power electronics depend on the hybrid vehicle manufacturer.

Powertrain control for a hybrid vehicle is achieved using a multimode digital control system. It is somewhat more complicated than the digital engine control system discussed earlier in this chapter in that it must control an IC engine and an EM motor. In addition, it must achieve the balance between ICE and EM power, and it must engage or disengage the solenoid-operated clutches (if present).

The inputs to this controller come from sensors that measure the following:

- Power demand from driver (accelerator pedal)
- State of charge of battery pack
- Vehicle speed
- ICE RPM and load
- EM voltage and current
- EM angular position
- Regulation of electric power flow and voltage

The system outputs include control signals to

- ICE throttle position,
- EM motor control inputs (e.g., V_A, V_B, and V_C),
- clutch engage/disengage,
- switch ICE ignition on/off.

Depending upon the HEV configuration, there may be no direct mechanical link from the accelerator pedal to the throttle. Rather, the throttle position (as measured by a sensor) is set by the control system via an electrical signal sent to an actuator (motor) that moves the throttle in a system called drive-by-wire.

The control system itself is a digital controller using the inputs and outputs listed above and has the capability of controlling the hybrid powertrain in many different modes. These modes include starting from a standing stop, steady cruise, regenerative braking, recharging battery pack, and many others that are specific to a particular vehicle configuration.

In almost all circumstances, it is desirable for the IC engine to be off at all vehicle stops. Clearly, it is a waste of fuel and an unnecessary contribution to exhaust emissions for an IC engine to run in a stopped vehicle. Exceptions to this rule involve cold weather operations in which it is desirable or even necessary to have some limited engine operations with a stopped vehicle in order to maintain engine and catalytic converter at proper temperature. In addition, a low-battery SOC might call for ICE operation at certain vehicle stops in order to provide charge to the battery pack.

When starting from a standing start, normally, the EM propulsion is used to accelerate the car to desired speed, assuming the battery has sufficient charge. If charge is low, then the controller can engage the clutch to the ICE such that the EM can begin acceleration and at the same time crank the ICE to start it. Then, depending on the time that the vehicle is in motion, the ICE can provide propulsive power and/or battery charge. Should the vehicle go to a steady cruise at low-battery SOC for engine operation near its optimum, then the control strategy normally is to switch off the electric power to the EM and power the vehicle solely and recharge the battery pack with the ICE (parallel hybrid only). In other cruise conditions, the controller can balance power between ICE and EM in a way that maximizes total fuel economy (subject to emission constraints).

For urban driving with frequent stops, the control strategy favors EM operation as long as SOC is sufficient. In this operating mode, regenerative braking may be used (in which energy is absorbed by vehicle deceleration), and the recovered energy appears as increased SOC.

The various operating modes and control strategies for an HEV depend on many factors, including vehicle weight, relative size and power capacity of ICE/EM, and exhaust emissions and fuel economy of the ICE (as installed in the particular vehicle). It is beyond the scope of this book to attempt to cover all possible operating modes for all HEV configurations. However, the above discussion has provided background within which specific HEV configurations' operating modes and control strategies can be understood.

In addition to the HEV, there is also the pure electric vehicle (EV) that has no ICE for powering the vehicle. This vehicle incorporates many of the components of an HEV including an electric motor, a battery pack for storing electric energy, and an electronic controller that provides the motor excitation. As any EV is driven, the battery SOC decreases.

Control of the EM in an EV is accomplished in a way that is similar to that described above for EM motor control in an HEV. This control is done by regulating the excitation voltage or current and the excitation frequency (which must satisfy the frequency condition for any motor). At some point, the battery pack requires recharging. The power for this recharging comes from the electric power grid. It is worth remembering that although an EV has essentially zero vehicle-out emissions, the creation of the electric power to recharge the batteries is done at some electric utility. Depending on the type of power generation at the electric utility, there may be increased emissions from that plant to meet the power requirements to recharge EV battery packs except for nuclear electric power generators. In this sense, the EV is not always a pure zero-emission vehicle.

CHAPTER

VEHICLE MOTION CONTROLS 7

CHAPTER OUTLINE

Representative Cruise Control System 344
Digital Cruise Control 351
Hardware Implementation Issues 354
Throttle Actuator 356
Cruise Control Electronics 359
Stepper Motor-Based Actuator Electronics 360
Vacuum-Operated Actuator 362
Advanced Cruise Control 364
Antilock Braking System 368
Tire Slip Controller 377
Electronic Suspension System 377
Variable Damping via Variable Strut Fluid Viscosity 395
Variable Spring Rate 396
Electronic Suspension Control System 397
Electronic Steering Control 398
Four-Wheel Steering Car 401
Summary 408

The term *vehicle motion* refers to the translation along and rotation about all three axes (i.e., longitudinal, lateral, and vertical) for a vehicle. By the term *longitudinal axis*, we mean the axis that is parallel to the ground (vehicle at rest) on a horizontal plane along the length of the car. The lateral axis is orthogonal to the longitudinal axis and is also parallel to the ground (vehicle at rest). The vertical axis is orthogonal to both the longitudinal and lateral axes.

Rotations of the vehicle around these three axes correspond to angular displacement of the car body in roll, yaw, and pitch. *Roll* refers to angular displacement about the longitudinal axis; *yaw* refers to angular displacement about the vertical axis; and *pitch* refers to angular displacement about the lateral axis.

In characterizing the vehicle dynamic motion, it is common practice to define a body-centered Cartesian coordinate system in which the x-axis is the longitudinal axis with positive forward. The y-axis is the lateral axis and is taken as the lateral axis with the positive sense to the right-hand side (RHS). The vertical axis is taken as the z-axis with the positive sense up.

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-12-810434-7.00007-7

Copyright © 2017 Elsevier Inc. All rights reserved.

The vehicle dynamic motion is represented as displacement, velocity, and acceleration of the vehicle relative to an earth-centered, earth-fixed (ECEF) inertial coordinate system (as will be explained later in this chapter) in response to forces acting on it. Although strictly speaking, the ECEF coordinate system is not truly an inertial reference, with respect to the types of motion of interest in most vehicle dynamics, it is essentially an inertial reference system.

Electronic controls have been recently developed with the capability of regulating the motion along and about all three axes. Individual car models employ various selected combinations of these controls. This chapter discusses motion control electronics beginning with control of motion along the longitudinal axis in the form of a cruise control system.

The forces and moments/torque that influence vehicle motion along the longitudinal axis include those due to the power train (including, in selected models and traction control), the brakes, the aerodynamic drag, and the tire-rolling resistance, as well as the influence of gravity when the car is moving on a road with a nonzero inclination (or grade). In a traditional cruise control system, the tractive force due to the power train is balanced against all resisting forces to maintain a constant speed. In an advanced cruise control (ACC) system, brakes are also automatically applied as required to maintain speed when going down a hill of sufficiently steep grade. Longitudinal vehicle motion refers to translation of the vehicle in an ECEF y-z plane.

REPRESENTATIVE CRUISE CONTROL SYSTEM

Automotive cruise control is an excellent example of the type of electronic feedback control system that is discussed in general terms in Appendix A. It is explained in Appendix A that the components of a closed-loop control system include the plant or system being controlled and a sensor for measuring the plant variable being regulated. It also includes an electronic control system that receives inputs in the form of the desired value of the regulated variable and the measured value of that variable from the sensor. The control system generates an error signal constituting the difference between the desired and actual values of this variable. It then generates an output from this error signal that drives an electromechanical actuator. The actuator controls the input to the plant in such a way that the regulated plant variable is moved toward the desired value.

We begin with a simplified traditional cruise control for a vehicle traveling along a straight road (along the x-axis in our ECEF coordinate system). An ACC is explained in a later section of this chapter. In the case of a traditional cruise control, the variable being regulated is the vehicle speed:

$$\mathrm{V} = \frac{dx}{dt}$$

where x is the translation of the vehicle in the ECEF frame.

The driver manually sets the car speed at the desired value via the accelerator pedal. Upon reaching the desired speed (V_d), the driver activates a momentary contact switch that sets that speed as the command input to the control system. From that point on, the cruise control system maintains the desired speed automatically by operating the throttle via a throttle actuator.

Under normal driving circumstances, the total external forces acting on the vehicle are such that a net positive traction force (from the power train) is required to maintain a constant vehicle speed. The total external forces acting on the vehicle include rolling resistance of the tires, aerodynamic drag, and a component of vehicle weight whenever the vehicle is traveling on a road with a slope relative to level.

However, when the car is on a downward sloping road of sufficient grade, drag, and tire-rolling resistance are insufficient to prevent vehicle acceleration (i.e., $\dot{V} > 0$) and maintaining a constant vehicle speed requires a negative tractive force that the power train cannot deliver. In this case, the car will accelerate unless brakes are applied. For our initial discussion, we assume this latter condition does not occur and that no braking is required. It is further assumed that the power train has sufficient power capability of maintaining constant vehicle speed on an up-sloping grade.

The plant being controlled consists of the power train (i.e., engine and drivetrain), which propels the vehicle through the drive axles and wheels. As described above, the load on this plant includes friction and aerodynamic drag as well as a portion of the vehicle weight when the car is going up- and downhills.

For an understanding of the dynamic performance of a cruise control, it is helpful to develop a model for vehicle motion along a road. The basic performance of a cruise control can be presented with a few simplifying assumptions. In the interest of safety, a typical traditional cruise control cannot be activated below a certain speed (e.g., 40 mph). For the purposes of presenting the present somewhat simplified model, it is assumed that the vehicle is traveling along a straight road at a cruise speed with the automatic transmission in torque converter lockup mode (see Chapter 6). This assumption removes some power train dynamics from the model. It is further assumed that the transmission is in direct drive such that its gear ratio is 1. The total gear ratio is given by the differential/transaxle gear ratio g_A where typically $2.8 \leq g_A \leq 4.0$. Under this assumption, the torque applied to the drive wheels T_w is given by

$$T_w = g_A T_b \tag{7.1}$$

where T_b is the engine brake torque.

The cruise control system employs an actuator that moves the throttle in response to the control signal. Of course whenever the cruise control is disconnected (e.g., by brake application), this actuator must release control of the throttle such that the driver controls throttle angular position via the accelerator pedal and associated linkage. Except for roads with relatively steep grades, normally, once cruise control is activated relatively small, changes in throttle position are required to maintain selected vehicle speed. For our simplified model, we assume that T_b varies linearly with cruise control output electrical signal u:

$$T_b = K_a u \tag{7.2}$$

where K_a is a constant for the engine/throttle actuator. This assumption, though not strictly valid, permits a system performance analysis using the discussion of linear control theory of Appendix A without any serious loss of generality.

A vehicle traveling along a straight road at speed V experiences forces due to the wheel torque T_w, aerodynamic drag D tire-rolling resistance F_{rr}, and inertial forces. A dynamic model for the vehicle longitudinal speed in m/s or ft/s (i.e., along the direction of travel and vehicle fore/aft axis) is given by

$$M\dot{V} + D + F_{rr} = \frac{g_A T_b}{r_w} - W_V \sin\theta \tag{7.3}$$

where

M = vehicle mass
W_V = vehicle weight (Mg)
g = gravitational constant (e.g., 9.81 m/s^2 or 32 ft/s^2)
r_w = drive wheel effective radius

$F_{rr} = \mu_{rr} W_V$
μ_{rr} = coefficient of tire-rolling resistance
$0.02 \leq \mu_{rr} \leq 0.04$ typically
θ = angle of the road surface relative to a horizontal plane

$$\text{Drag force } D = \frac{\rho}{2} C_D S_{\text{ref}} (V + V_w)^2$$

ρ = air density
C_D = drag coefficient
S_{ref} = reference area
V_w = the component of wind along vehicle longitudinal axis (positive for head wind negative for tail wind).

In specifying a drag coefficient for a car, it is necessary to specify a reference area. Although the choice of S_{ref} is somewhat arbitrary, conventional practice takes the largest vehicle cross-sectional area projected in a body y-z plane. In the above nonlinear differential Eq. (7.3), the first term on the RHS is the force acting on the vehicle due to the applied road torque acting at the tire/road interface due to the power train, and g_A is the combined gear ratio from the engine to the drive axle. The second term on the RHS is the component of force along the vehicle axis due to its weight and any road slope expressed by θ.

For a car traveling at constant cruise speed V_C (i.e., $\dot{V} = 0$) along a level, horizontal road (i.e., $\theta = 0$) with zero wind, the differential equation above reduces to an algebraic expression in terms of the engine brake torque and speed V_C:

$$\rho \frac{C_D S_{\text{ref}}}{2} V_C^2 + \mu_{rr} W_V = g_A \frac{T_b}{r_w} \tag{7.4}$$

This equation permits a determination of engine brake torque versus cruise speed for a level road.

If the vehicle is traveling at a steady speed along a hill with slope angle θ, then the T_b is determined from the following equation:

$$g_A \frac{T_b}{r_w} = \rho \frac{C_D S_{\text{ref}} V_C^2}{2} + \mu_{rr} W_V + Mg \sin\theta \tag{7.5}$$

For the operation of the cruise control system, it is normally sufficient to model vehicle dynamics with a linearized version of the nonlinear differential equation. The drag term can be linearized by representing vehicle instantaneous speed ($V(t)$) with the approximate model assuming for simplicity that $V_w = 0$:

$$\begin{aligned} D &= D_C + \delta D \\ V(t) &= V_C + \delta V \end{aligned} \tag{7.6}$$

where D_C is the drag at speed V_C:

$$\begin{aligned} \delta D &= \frac{dD}{dV}\Big|_{V_C} \delta V \\ &= \rho C_D S_{\text{ref}} V_C \delta V \\ &= K_D \delta V \end{aligned}$$

where K_D is a constant for a given initial steady cruise speed V_C and constant ρ and δV is the variation in speed about V_C.

In modeling the cruise control system, it is helpful to consider the influence of road grade (θ) as a disturbance. This disturbance can be linearized to a close approximation by the substitution (provided that the slope of the hill is sufficiently small):

$$\sin\theta \approx \theta$$

The linearized equation of motion is given by

$$M\delta\dot{V} + \rho C_D S_{\text{ref}} V_C \delta V - Mg\theta = g_A \frac{\delta T_b}{r_w} \tag{7.7}$$

The operational transfer function $H_p(s)$ for the "plant" for zero disturbance (i.e., $\theta = 0$) is given by

$$\begin{aligned} H_p(s) &= \frac{\delta V(s)}{\delta T_b(s)} \\ &= \frac{g_A/(Mr_w)}{s + \rho \dfrac{C_D S_{\text{ref}} V_C}{M}} \end{aligned} \tag{7.8}$$

The configuration for a representative automotive cruise control is shown in Fig. 7.1.

When the vehicle reaches the desired speed V_d under normal driver accelerator pedal regulation of the throttle, to activate cruise control at that speed the driver pushes a momentary contact switch S_1 in Fig. 7.1, thereby setting the command speed in the controller. At this point, control of the throttle position is via the cruise control actuator. The momentary contact (push-button) switch that sets the command speed (V_d) is denoted S_1 in Fig. 7.1.

Also shown in this figure is a disengage switch that completely disengages the cruise control system from the power supply such that throttle control reverts back to the accelerator pedal. This switch is denoted S_2 in Fig. 7.1 and is a safety feature. In an actual cruise control system, the disable function can be activated in a variety of ways, including the master power switch for the cruise control system and a brake pedal-activated switch that disengages the cruise control any time that the brake pedal is moved from its rest position.

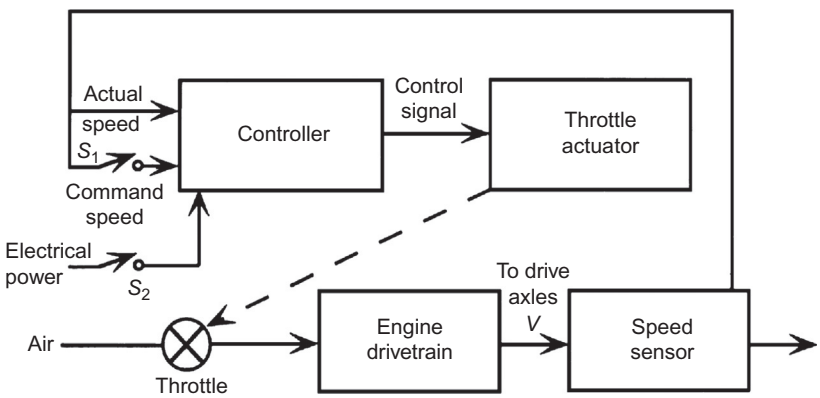


FIG. 7.1 Cruise control configuration.

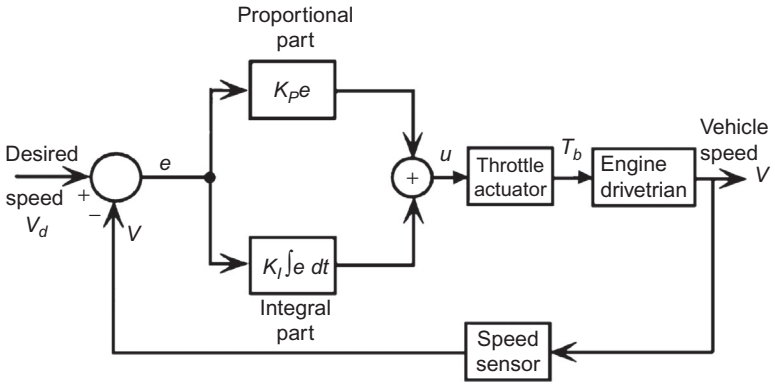


FIG. 7.2 Cruise control block diagram.

During normal cruise control operation, the throttle actuator moves the throttle to open or close the throttle in response to the error between the desired and actual speed. Whenever the actual speed is less than the desired speed, the throttle opening is increased by the actuator, which increases vehicle speed, until the error is zero at which point the throttle opening remains fixed until either a disturbance occurs or the driver calls for a new desired speed.

A block diagram of a cruise control system is shown in Fig. 7.2. In the cruise control depicted in this figure, a proportional integral (PI) control strategy has been assumed. Before the advent of digital cruise control, there were a variety of analog systems that had a proportional-only (P) control law. Nevertheless, the PI controller is representative of good design for such a control system since it can reduce steady-state speed errors to zero (as explained in Appendix A). In this strategy, an error e is formed by subtracting (electronically) the actual speed V from the desired speed V_d:

$$e = V_d - V \tag{7.9}$$

It should be noted that the speed differential from V_d is the negative of the error (i.e., $e = -\delta V$). The controller then electronically generates the actuator signal by combining a term proportional to the error ($K_p e$) and a term proportional to the integral of the error:

$$K_I \int e dt \tag{7.10}$$

The actuator signal u is given by

$$u = K_p e + K_I \int e dt \tag{7.11}$$

Operation of the system can be understood by considering the operation of a PI controller. We assume that the driver has reached the desired speed (say, 60 mph) and activated the speed set switch. The car is initially traveling on a level road at the desired speed (i.e., $V_c = V_d$). Then, at some point, it encounters a long hill with a steady positive slope (i.e., a hill going up).

The control signal at the output of the PI controller u is given by Eq. (7.11). It is consistent with the linearized approximation to model the change in brake torque δT_b due to actuator change in throttle position in response to the control signal u as linear in the control signal (as presented earlier):

$$\delta T_b = K_a u \tag{7.12}$$

where K_a is a constant for the throttle actuator-engine combination. With the above models and notation, the vehicle dynamic equation of motion becomes

$$\begin{aligned} M\delta\dot{V} + K_D\delta V + Mg\theta &= \frac{g_A K_a u}{r_w} \\ &= g_A \frac{K_a}{r_w}\left[K_p e + K_I \int e dt\right] \end{aligned} \tag{7.13}$$

Taking the Laplace transform of the above equation and solving for the speed differential yield

$$\delta V(s) = -\frac{\text{sg}\theta(s)}{\left[s^2 + \left(\frac{K_D}{M} + \frac{g_A K_A K_p}{M r_w}\right)s + \frac{g_A K_A K_I}{M r_w}\right]} \tag{7.14}$$

A computer simulation of this simplified cruise control was done for a step change in grade of 5% starting at 2 s into the simulation for the following parameters in English units:

$W_V = 3100$ lb
$C_D = 0.3$
$S_{\text{ref}} = 18$ ft^2
$\rho = 0.0024$ slug/ft^3 (i.e., sea level on a standard day)

$$K_D = \rho C_D S_{\text{ref}} V_C$$

$V_d = 88$ ft/s (i.e., 60 mph)
$K_A = 10$
$K_p = 10$
$K_I = 50$
$r_w = 1$ ft
$g_A = 3.0$

The simulation was done for the PI control but for reference purposes was also run for $K_I = 0$ (i.e., a proportional-only control). Fig. 7.3 is a plot of $V(t)$ (converted to mph) for the car initially traveling under cruise control at 60 mph (88 ft/s). At time $t = 2$ s, a hill of steady 5% (i.e., $\theta = 0.05$) grade occurs (for the particular gains chosen). The dashed curve is the response of proportional-only control. Note that the speed drops down to a steady 53 mph for the controller. The solid curve depicts the vehicle speed for the preferred PI control. Except for a brief overshoot, this control returns the vehicle speed to the set point of 60 mph in a few seconds. It should be noted that the P-only control performance can be improved by increasing K_p (provided the system satisfies stability robustness criteria (see Appendix A)).

The response characteristics of a PI controller depend strongly on the choice of the gain parameters K_p and K_I. It is possible to select values for these parameters to increase the rate at which the system

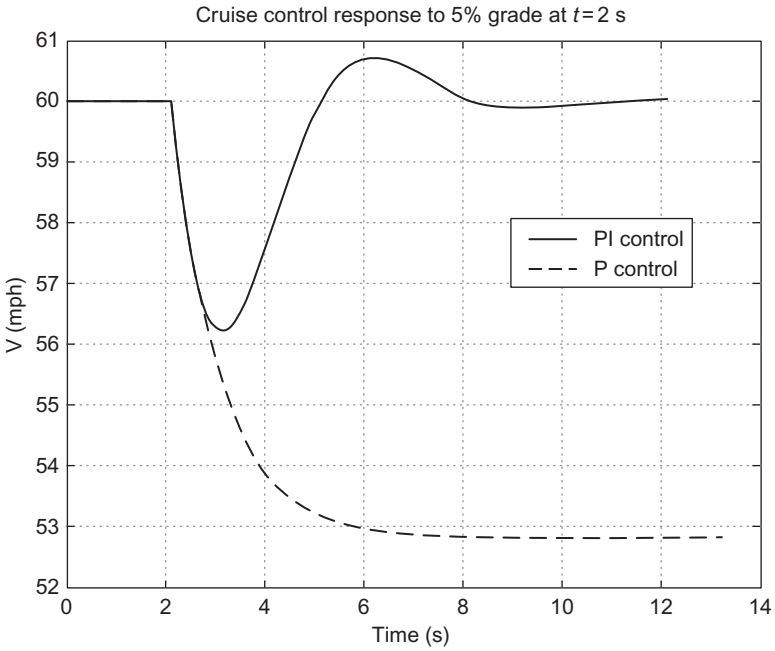


FIG. 7.3 Cruise control speed performance.

responds to disturbance. If this rate is increased too much, however, overshoot will increase, and stability robustness (e.g., gain/phase margins) generally is reduced. As explained in Appendix A, the amplitude of the speed error oscillations decreases by an amount determined by a parameter called the *damping ratio*. The damping ratio that produces the fastest response without overshoot is called *critical damping*.

The importance of these performance curves of Fig. 7.3 is that they demonstrate how the performance of a cruise control system is affected by the controller gains. These gains are simply parameters that are contained in the control system. They determine the relationship between the error, the integral of the error, and the actuator control signal.

Usually a control system designer attempts to balance the proportional and integral control gains so that the system is optimally damped. However, because of system characteristics, in many cases, it is impossible, impractical, or inefficient to achieve the optimal time response, and therefore, another response is chosen. The control system should cause T_b to respond quickly and accurately to the command speed, but should not overtax the engine in the process. Therefore, the system designer chooses the control electronics that provide the following system qualities:

1. Quick response
2. Stable system
3. Small steady-state error
4. Optimization of the control effort required

DIGITAL CRUISE CONTROL

The explanation of the operation of cruise control thus far has been based on a continuous-time formulation of the problem. This formulation correctly describes the concept for cruise control regardless of whether the implementation is by analog or digital electronics. Cruise control is now mostly implemented digitally using a microprocessor-based controller. For such a system, proportional and integral control computations are performed numerically in the computer. The digital cruise control is inherently a discrete-time system with samples of the vehicle speed taken at integer multiples of the sample period T_s.

The block diagram for a representative digital cruise control is depicted in Fig. 7.4.

The plant variable being controlled is its forward speed V. The desired speed or set point for the controller is denoted V_d. The model for the plant as represented by its transfer function $H_p(s)$ is taken to be the same as that developed above for the analog version of the cruise control. However, the actuator signal that is the output of the zero-order hold (ZOH) (see Chapter 2) circuit $\bar{u}(t)$ is a piecewise continuous signal (see Appendix B):

$$\begin{aligned} H_p(s) &= \frac{V(s)}{\bar{u}(s)} \\ &= \frac{g_A K_a}{M r_w \left(s + {}^{K_D}/_{M}\right)} \\ &= \frac{K}{s + s_0} \end{aligned} \tag{7.15}$$

where

$$K = \frac{g_A K_a}{M r_w}$$

$$s_o = K_D / M$$

Using some of the parameters as were used for the analog version of the cruise control, except for $K_A = 2$ this model is given numerically by the following transfer function:

$$H_p(s) = \frac{0.4129}{(s + 0.0118)} \tag{7.16}$$

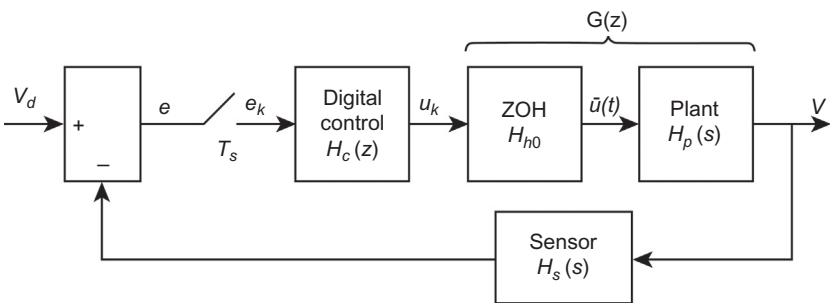


FIG. 7.4 Digital speed control block diagram.

As explained in Appendix B, the z-transfer function for the combination of ZOH and plant ($G(z)$) is given by

$$G(z) = \left(1 - z^{-1}\right)\mathcal{Z}\left(\frac{H_p(s)}{s}\right) \tag{7.17}$$

From the methods of Appendix B, the z-transform above can be found by expanding $H_p(s)$/s in a partial fraction series and then using the tables of Appendix B. Then, it is left as an exercise to show that for sample period $T_s = 0.01$ s, $G(z)$ is given by

$$G(z) = \frac{K}{s_o}\left[\frac{(1 - z_0)}{(z - z_0)}\right] \tag{7.18}$$

where $K = 0.4129$ and $s_o = 0.0118$

$$z_o = e^{-s_o T}$$

The continuous-time PI control law is given by

$$u(t) = K_p e(t) + K_I \int e dt \tag{7.19}$$

In Chapter 6 under the section discussing control of variable valve phasing, it was shown that one discrete-time z-transform of the integral term (using the trapezoidal integration rule) is given by

$$\mathcal{Z}\left[K_I \int e dt\right] = \frac{K_I T_s (z + 1)}{2(z - 1)}$$

The z-operational transfer function for the controller $H_c(z)$ is given by

$$H_c(z) = \frac{u(z)}{e(z)} \tag{7.20}$$

$$H_c(z) = K_p + \frac{K_I T (z + 1)}{2(z - 1)}$$

$$H_c(z) = \frac{\left(K_p + \frac{K_I T}{2}\right) z - \left(K_p - \frac{K_I T}{2}\right)}{(z - 1)} \tag{7.21}$$

Using the same gains ($K_p = 10$ and $K_I = 50$) as for the continuous-time control, one obtains

$$H_c(z) = \frac{10.25z - 9.75}{(z - 1)} \tag{7.22}$$

Appendix B also showed that the forward path z-transfer function $H_F(z)$ for a discrete-time control system as shown in Fig. 7.4 is given by

$$\begin{aligned} H_F(z) &= \frac{\delta V(z)}{e(z)} \\ &= H_c(z) G(z) \\ &= \frac{0.0423z - 0.0403}{z^2 - 1.9998z + 0.9998} \end{aligned} \tag{7.23}$$

Assuming an ideal sensor for which $H_s(s)=1$, the closed-loop gain z-transform function $H_{CL}(z)$ is given by

$$\begin{aligned} H_{CL}(z) &= \frac{H_F(z)}{1+H_F(z)} \\ &= \frac{0.0423z-0.0403}{z^2-1.9576z+0.9595} \end{aligned} \tag{7.24}$$

The poles of this closed-loop transfer function are

$$\begin{aligned} z_1 &= 0.9788+0.0394i \\ z_2 &= 0.9788-0.0394i \end{aligned}$$

Since all poles are inside the unit circle ($|z|<1$), the closed-loop cruise control system is stable as explained in Appendix B.

The dynamic response for this discrete-time cruise control system can be found by evaluating its response to a step change in the input. Assume that the vehicle is cruising at a steady 60 mph. Then, at $t=2$ s (i.e., at sample k_1 where $k_1=200$), the cruise control set point is changed by a step increase of 10–70 mph. This system set point is given by

$$\begin{aligned} V_d &= 60 \quad t<2 \\ &= 70 \quad t\geq 2 \\ V_d &= 60+10U_s(2) \end{aligned} \tag{7.25}$$

where $U_s(2)=$ unit step at $t=2$

The z-transform for this system input is given by

$$V_d(z) = 60+\frac{10z}{z-1} \tag{7.26}$$

The output z-transform $V(z)$ is given by

$$V(z) = H_{CL}(z)V_d(z) \tag{7.27}$$

The vehicle speed V_k at times t_k is found by taking the inverse z-transform of $V(z)$. Using the partial fraction expansion method of Appendix B, the time response at $t=t_k$ is shown in Fig. 7.5 in which $t_k=kT_s$ and $T_s=5$ ms. The speed is constant until $k=k_1$ where $t(k_1)=2$ s and then increases with a relatively small overshoot approaching the final set point value of 70 mph.

We consider next the implementation of the digital cruise control system in actual hardware. The vehicle speed sensor and the actuator are analog and can be modeled as either continuous- or discrete-time devices (examples of each are discussed below), and the control system is digital. When the car reaches the desired speed, V_d, the driver activates the speed set switch. At this time, the output of the vehicle speed sensor is sampled, converted to a digital value, and transferred to a storage register. This is the set point for the controller.

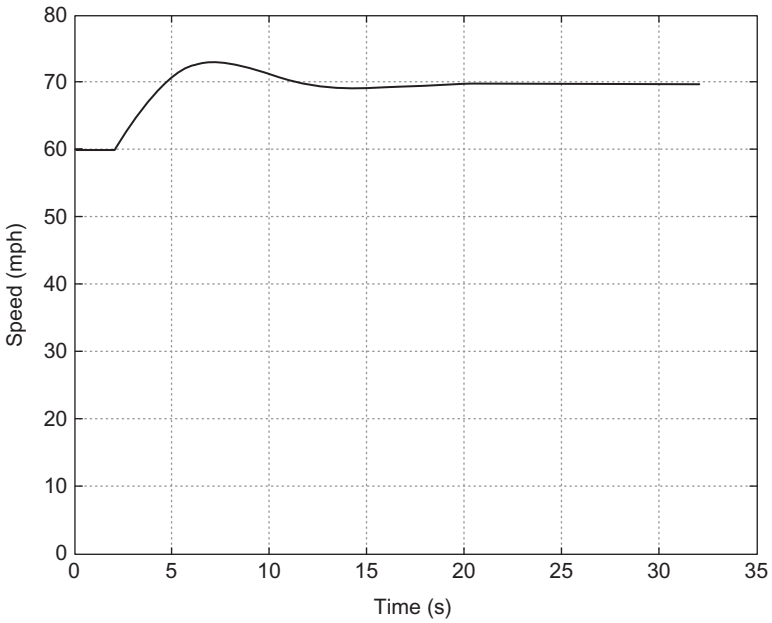


FIG. 7.5 Response of digital cruise control to step change in set speed.

HARDWARE IMPLEMENTATION ISSUES

The computer continuously reads the actual vehicle speed, V, and generates an error, e_n, at the sample time, t_n:

$$e_n = V_d - V(t_n)$$

A control signal, u_n, is computed that has the following form:

$$u_n = K_p e_n + K_1 \sum_{m=1}^{M} e_{n-m} \tag{7.28}$$

This sum, which is computed in the cruise control computer, is then multiplied by the integral gain K_I and added to the most recent error multiplied by the proportional gain K_p to form the control signal. The computed discrete-time control signal u_n then must be converted to a piecewise continuous form $\bar{u}(t)$ suitable to operate the actuator (via a ZOH). It should be noted that $\bar{u}(t)$ corresponds to the control signal u for the continuous-time linear cruise control above. The correct form for this signal is discussed below in conjunction with the throttle actuator configuration.

The operation of the cruise control system can be further understood by examining the vehicle speed sensor and the actuator in detail. Fig. 7.6A is a sketch of a sensor configuration suitable for vehicle speed measurement.

In a representative vehicle speed measurement system, the vehicle speed information is mechanically coupled to the speed sensor by a flexible cable coming from the driveshaft, which rotates at an

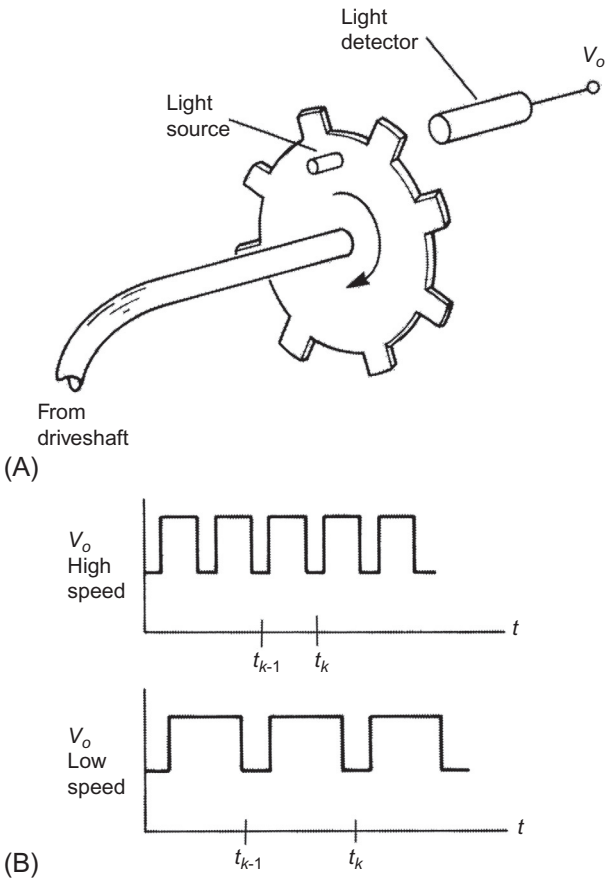


FIG. 7.6 Example speed sensor configuration. (A) Speed sensor configuration; (B) Illustrative sensor output voltages.

angular speed proportional to vehicle speed. A speed sensor driven by this cable generates a pulsed electrical signal (Fig. 7.6B) that is processed by the computer to obtain a digital measurement of speed.

A speed sensor can be implemented magnetically or optically. The magnetic speed sensor was discussed in Chapter 5, so we hypothesize an optical sensor for the purposes of this discussion. For the hypothetical optical sensor, a flexible cable drives a slotted disk that rotates between a light source and a light detector. The placement of the source, disk, and detector is such that the slotted disk interrupts or passes the light from source to detector, depending on whether a slot is in the line of sight from source to detector. The light detector produces an output voltage whenever a pulse of light from the light source passes through a slot to the detector. The number of pulses generated per second is proportional to the number of slots in the disk and the vehicle speed:

$$f = NVK$$

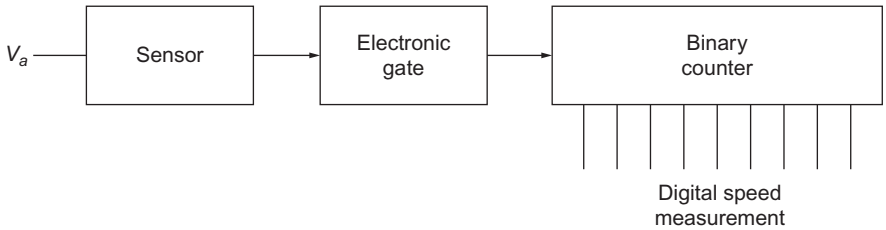


FIG. 7.7 Digital speed measurement system.

where f is the frequency in pulses per second, N is the number of slots in the sensor disk, V is the vehicle speed, and K is the proportionality constant that accounts for differential gear ratio and wheel size.

The sampled pulse frequency f_k is computed from measurements of the time of each low to high transition denoted t_k in Fig. 7.6B:

$$f_k = \frac{1}{t_k - t_{k-1}}$$

The output pulses are passed through a sample gate to a binary counter (Fig. 7.7).

The gate is an electronic switch that either passes the pulses to the counter or blocks their passage depending on whether the switch is closed or open. The time interval during which the gate is closed is precisely controlled by the computer. The digital counter counts the number of pulses from the light detector during time $T_g(n)$ that the gate is closed and pulses from the sensor are sent to the counter during the nth speed measurement cycle. The number of pulses $P(n)$ that is counted by the digital counter is given by

$$P(n) = T_g(n)\text{NVK} \tag{7.29}$$

That is, the number $P(n)$ is proportional to vehicle speed V at speed sample n. The electrical signal in the binary counter is in a digital format that is suitable for reading by the cruise control computer (as explained in Chapter 2).

THROTTLE ACTUATOR

The throttle actuator is an electromechanical device that, in response to an electrical input from the controller (u), moves the throttle through some appropriate mechanical linkage. Two relatively common throttle actuators operate either from manifold vacuum or with a stepper motor. The stepper motor implementation operates similarly to the idle speed control actuator described in Chapter 6 and is essentially a digital device. The throttle opening is either increased or decreased by the stepper motor in response to the sequences of pulses sent to the two windings depending on the relative phase of the two sets of pulses.

For a stepper motor-type actuator, the control signal (u) is converted to a pair of pulse sequences to drive the A and B coils (see Chapter 5). The stepper motor displacement causes a change in throttle plate angle $\delta\theta_t(n)$ (see Chapter 4) corresponding to u_n. Let f_p be the pulse frequency for the stepper motor pulse pairs. Normally, the pulse signal is generated in the digital control system as part of its timing circuitry. The controller regulates throttle angle changes by setting the time interval T_a during

which pulses are sent to the stepper motor. The total number of pulse pairs sent to the stepper motor actuator ($N_p(n)$) during a time interval T_a is given by

$$N_p(n) = f_p T_a(n) \tag{7.30}$$

where $T_a(n)$ is the actuator time during actuation cycle.

The actuation time interval is proportional to u_n:

$$T_a(n) = K_T u_n \tag{7.31}$$

where K_T is a constant for the control system.

The throttle plate angular displacement $\delta\theta_t(n)$ is proportional to $N_p(n)$:

$$\delta\theta_t(n) = K_\theta N_p(n) \tag{7.32}$$

where K_θ is the angular displacement for each pair of stepper motor pulses.

The time interval for throttle actuation must be sufficiently long to permit the full actuation of $\delta\theta_t(n)$ to occur but should be less than the discrete-time sample period.

For the linearized vehicle model, the change in brake torque $\delta T_b(n)$ is approximated linearly proportional to $\delta\theta_t(n)$ (for relatively small $\delta\theta_t$ at cruise condition):

$$\begin{aligned} \delta T_b(n) &= K_b \delta\theta_t(n) \\ &= K_b K_\theta K_T f_p u_n \end{aligned} \tag{7.33}$$

A dynamic performance of the digital cruise control is as explained for the discrete-time model given above where $\delta T_b(n)$ is a discrete-time version of $\delta T_b(t)$ as explained in the section on analog cruise control. An example of the electronics for generating the stepper motor actuator is discussed later in this chapter.

We consider next an exemplary analog (continuous-time) throttle actuator. This throttle actuator is operated by manifold vacuum through a solenoid valve, which is similar to that used for the exhaust gas recirculation (EGR) valve described in Chapter 6 and further explained later in this chapter. During cruise control operation, the throttle position is set automatically by the throttle actuator in response to the actuator signal generated in the control system. This type of manifold-vacuum-operated actuator is illustrated in Fig. 7.8.

A pneumatic piston arrangement is driven from the intake manifold vacuum. The piston-connecting rod assembly is attached to the throttle lever. There is also a spring attached to the lever. If there is no force applied by the piston, the spring pulls the throttle closed. When an actuator input signal energizes the electromagnet in the control solenoid, the pressure control valve (CV) is pulled down and changes the actuator cylinder pressure p by providing a path to manifold pressure p_m. Manifold pressure is lower than atmospheric pressure p_a, so the actuator cylinder pressure quickly drops, causing the piston to pull against the throttle lever to open the throttle.

Although the actuation signal is a binary-valued voltage, the actuator can be considered an analog device with actuation proportional to the pulse duty cycle (see Chapter 5). The force exerted by the piston is varied by changing the average pressure p_{av} in the cylinder chamber. This is done by rapidly switching the pressure control valve between the outside air port, which provides atmospheric pressure, and the manifold pressure port, the pressure of which is lower than atmospheric pressure. In one implementation of a throttle actuator, the actuator control signal V_c is a variable-duty-cycle type of signal like that discussed for the fuel injector actuator. A high V_c signal energizes the electromagnet;

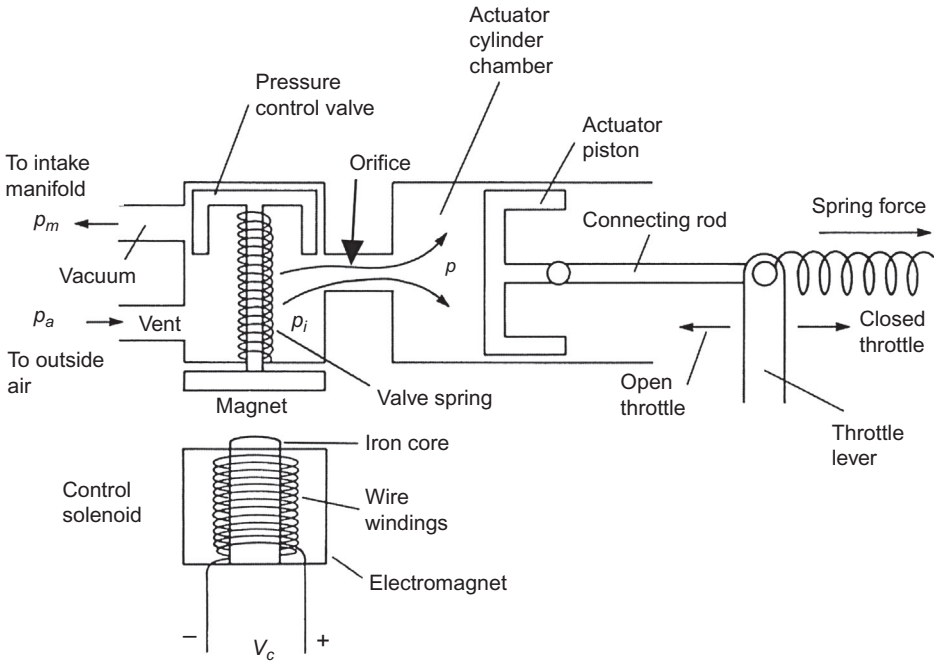


FIG. 7.8 Vacuum-operated throttle actuator.

whenever $V_c = 0$ the electromagnet is de-energized. Switching back and forth between the two pressure sources causes the average pressure in the chamber to be somewhere between the low manifold pressure and outside atmospheric pressure.

For the exemplary solenoid-operated actuator, the pressure applied to the valve side of the orifice p_i in Fig. 7.8 is given by

$$\begin{aligned} p_i &= p_m \quad V_c = V_H \\ &= p_a \quad V_c = 0 \end{aligned} \tag{7.34}$$

where p_m is the manifold pressure and p_a the atmospheric pressure.

The cruise control computer generates actuator control signal:

$$\begin{aligned} V_c(t) &= V_H \quad t_k \le t \le t_k + \tau \\ &= 0 \quad\;\; t_k + \tau < t < t_{k+1} \end{aligned}$$

The duty cycle δ_p is given by

$$\delta_p = \frac{\tau}{(t_{k+1} - t_k)} \tag{7.35}$$

where t_k is the periodic cycle time for speed control in the cruise control computer. This duty cycle (δ_p) is proportional to control signal u_n.

The average pressure (p_{av}) in the actuator cylinder chamber (averaged over a period (T_{av}) corresponding to several cycles) is given by

$$p_{av}(t) = \frac{1}{T_{av}} \int_{t-T_{av}}^{t} p_i(t')dt' = p_a + (p_m - p_a)\delta_p \tag{7.36}$$

Since p_m is a function of engine operating conditions, the control system continuously adjusts δ_p to maintain cruise speed at the desired value V_d. This average pressure and, consequently, the piston force are proportional to the duty cycle of the valve control signal V_c. The duty cycle is in turn proportional to the control signal u_n (explained above) that is computed from the sampled error signal e_n.

This type of duty-cycle-controlled throttle actuator is ideally suited for use in digital control systems. If used in an analog control system, the analog control signal must first be converted to a duty-cycle control signal. The same frequency response considerations apply to the throttle actuator as to the speed sensor. In fact, with both in the closed-loop control system, each contributes to the total system phase shift and gain and must be considered during system design.

CRUISE CONTROL ELECTRONICS

Cruise control can be implemented electronically in various ways, including with a microcontroller, with special-purpose digital electronics (or traditionally with analog electronics). It could, theoretically, also be implemented (in proportional control strategy alone) with an electromechanical speed governor, although such technology is obsolete.

The physical configuration for a digital, microprocessor-based cruise control is depicted in Fig. 7.9.

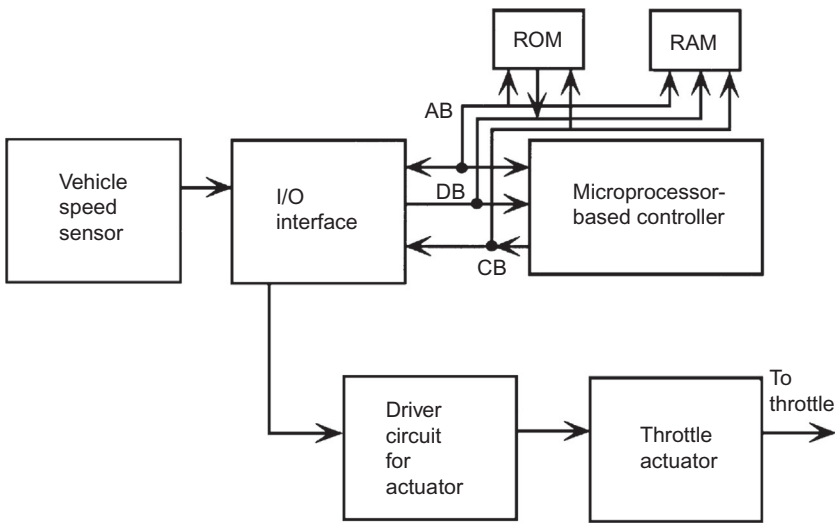


FIG. 7.9 Digital cruise control configuration.

A system such as is depicted in Fig. 7.9 has a digital controller that is often called a *microcontroller* since it is implemented with a microprocessor operating under program control that is a part of the system design. The actual program that causes the various calculations to be performed is stored in read-only memory (ROM). Typically, the ROM also stores parameters that are critical to the correct calculations. In addition, the system uses RAM memory to store the command speed and to store any temporary calculation results. Input from the speed sensor and output to the throttle actuator are handled by the I/O interface (normally an integrated circuit that is a companion to the microprocessor). The output from the controller (i.e., the control signal) is sent via the I/O (on one of its output ports) to so-called driver electronics. The latter electronics receives this control signal and generates a signal of the correct format and power level to operate the actuator (as explained below).

A microprocessor-based cruise control system performs all of the required control law computations digitally under program control. For example, a PI control strategy is implemented as explained above, with a proportional term and an integral term that is formed by a summation. In performing this task, the controller continuously receives samples of the speed error e_n. This sampling occurs at a sufficiently high rate to be able to adjust the control signal to the actuator in time to compensate for changes in operating condition or to disturbances. At each sample, the controller reads the most recent error and then performs the control law computations necessary to generate an actuator signal u_n. As explained earlier that error is multiplied by the proportional gain K_P, yielding the proportional term in the control law. It also computes the sum of a number of M previous error samples (the exact sum is chosen by the control system designer in accordance with the allowable steady-state error and the available computation time). Then, this sum is multiplied by a constant K_I and added to the proportional term, yielding the control signal.

The control signal u_n at this point is simply a number that is stored in a memory location in the digital controller. The use of this number by the electronic circuitry that drives the throttle actuator to regulate vehicle speed depends on the configuration of the particular control system and on the actuator used by that system.

STEPPER MOTOR-BASED ACTUATOR ELECTRONICS

For example, in the case of a stepper motor actuator, the actuator driver electronics reads the control variable u_n and then generates a sequence of pulses to the pair of windings on the stepper motor (with the correct relative phasing) at frequency f_p as explained in Chapter 5 to cause the stepper motor to either advance or retard the throttle setting as required to bring the error toward zero. An illustrative example of driver circuitry for a stepper motor actuator is shown in Fig. 7.10.

The basic idea for this circuitry is to drive the stepper motor in such a way as to advance or retard the throttle in accordance with the control signal u_n that is stored in memory. Just as the controller periodically updates the actuator control signal, the stepper motor driver electronics continually adjusts the throttle by an amount determined by this actuator signal. This signal is, in effect, a signed number (i.e., a positive or negative numerical value). A sign bit indicates the direction of the throttle movement (advance or retard). The numerical value determines the amount of advance or retard.

The magnitude of the actuator signal (in binary format) is loaded into a parallel load serial downcount binary counter. The direction of movement is in the form of the sign bit (SB of Fig. 7.10). The stepper motor is activated by a pair of quadrature phase signals (i.e., signals that are out of phase by $\pi/2$) coming from a pair of oscillators. To advance the throttle, phase A signal is applied to coil 1 and phase

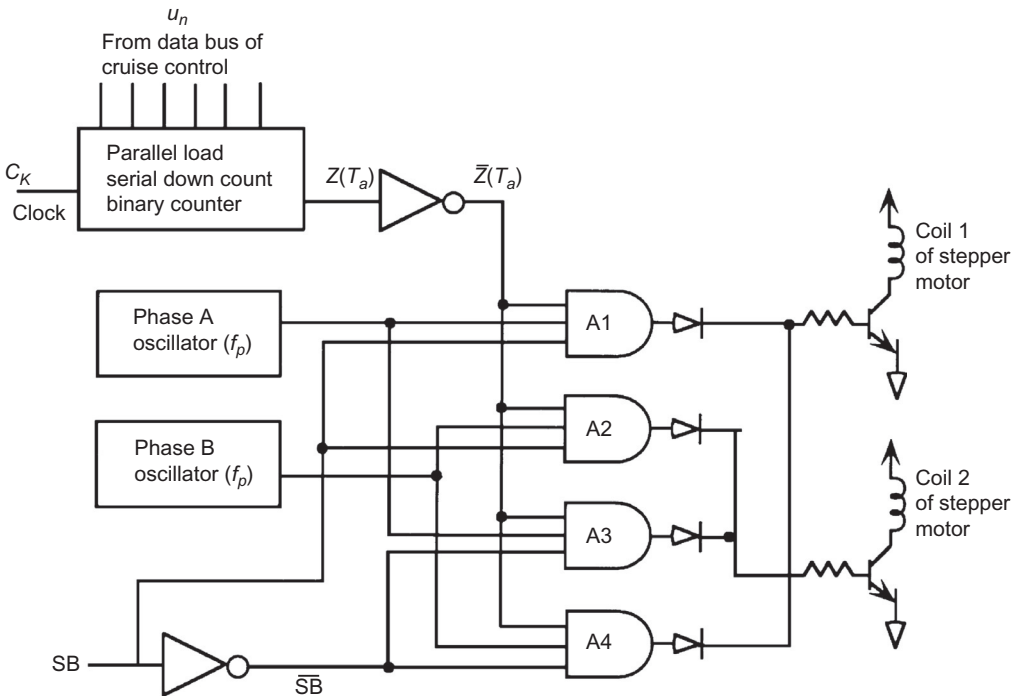


FIG. 7.10 Stepper motor actuator electronics for cruise control.

B signal to coil 2. To retard the throttle, these phases are each switched to the opposite coil. The amount of movement in either direction is determined by the number of cycles $N_p(n)$ of A and B, one step for each cycle.

The number of cycles of these two phases is controlled by a logical signal ($Z(T_a)$) in Fig. 7.10. This logical signal is switched low such that $\bar{Z}(T_a)$ is high for period T_a, enabling a pair of AND gates (from the set A1, A2, A3, and A4). The length of time that $\bar{Z}$ is switched high (T_a) determines the number of cycles and corresponds to the number of steps of the motor.

The logical variable Z corresponds to the contents of the binary counter being zero. As long as the logical inverse of Z (i.e., $\bar{Z}$) is high, a pair of AND gates (A1 and A3 or A2 and A4) is enabled, permitting phase A and phase B signals to be sent to the stepper motor. The pair of gates enabled is determined by the sign bit. When the sign bit is high, A1 and A2 are enabled and the stepper motor advances the throttle position as long as Z is not high. Similarly, when the sign bit is low, A3 and A4 are enabled, and the stepper motor retards the throttle position. The diodes in the AND gate outputs isolate the inactive from the active AND gates.

To control the number of steps, the controller loads a binary value into the binary counter. With the contents not being zero, the appropriate pair of AND gates is enabled. When loaded with data, the binary counter counts down at the frequency of a clock (C_K in Fig. 7.10). When the countdown reaches zero, logical variable Z switches high (and $\bar{Z}$ switches low) and the gates are disabled, and the stepper motor stops moving.

The time required to countdown to zero is determined by the numerical value loaded into the binary counter. By loading signed binary numbers into the binary counter, the cruise controller regulates the amount and direction of movement of the stepper motor and thereby the corresponding movement of the throttle.

VACUUM-OPERATED ACTUATOR

The driver electronics for a cruise control based on a vacuum-operated system generates a variable-duty-cycle signal as described above. In this type of system, the duty cycle at any time is proportional to the control signal as explained above. For example, if at any given instant a large positive error exists between the command and actual signal, then a relatively large control signal will be generated. This control signal will cause the driver electronics to produce a large duty-cycle signal to operate the solenoid so that most of the time the actuator cylinder chamber is nearly at manifold vacuum level. Consequently, the piston will move against the restoring spring and cause the throttle opening to increase. As a result, the engine will produce more power and will accelerate the vehicle until its speed matches the command speed.

It should be emphasized that, regardless of the actuator type used, a microprocessor-based cruise control system will

1. read the command speed;
2. measure actual vehicle speed;
3. compute an error (error = command − actual);
4. compute a control signal using P, PI, or PID control law;
5. send the control signal to the driver electronics;
6. cause driver electronics to send a signal to the throttle actuator such that the error will be reduced.

Although analog electronics are obsolete in contemporary vehicles, we include the following example of a pure analog system to illustrate principles introduced in Chapter 2 and because there remain some older vehicles with such systems on the road. A pure analog speed sensor in the form of a d-c generator is assumed. Its output voltage V_o is linearly proportional to vehicle speed V:

$$V_o = K_g V \tag{7.37}$$

where K_g is the constant for the sensor. An example of electronics for a cruise control system that is basically analog is shown in Fig. 7.11.

The vehicle speed sensor of Fig. 7.11A generates the output V_o, which is sent to the driver-operated switch for setting a voltage corresponding to desired speed (V_d) in a hold circuit such as was described in Chapter 2. This voltage value will remain until reset by the driver to a new value. The sensor voltage also provides the feedback signal to the error amplifier of this PI control system. Notice that the system uses four operational amplifiers (op-amps) as described in Chapter 2 and that each op-amp is used for a specific purpose. Op-amp 1 is used as an error amplifier. The output of op-amp 1 (V_e) is proportional to the difference between the command speed and the actual speed. The error signal is then used as an input to op-amps 2 and 3. Op-amp 2 is a proportional amplifier with a gain of $K_P = -R_2/R_1$ with an output voltage $V_p = K_p V_e$. Notice that R_1 is variable so that the proportional amplifier gain can be adjusted. Op-amp 3 is an integrator with a gain of $K_I = -1/R_3C$, which generates output voltage V_I, that is given by

$$V_I = -\frac{1}{R_3C}\int V_e dt \tag{7.38}$$

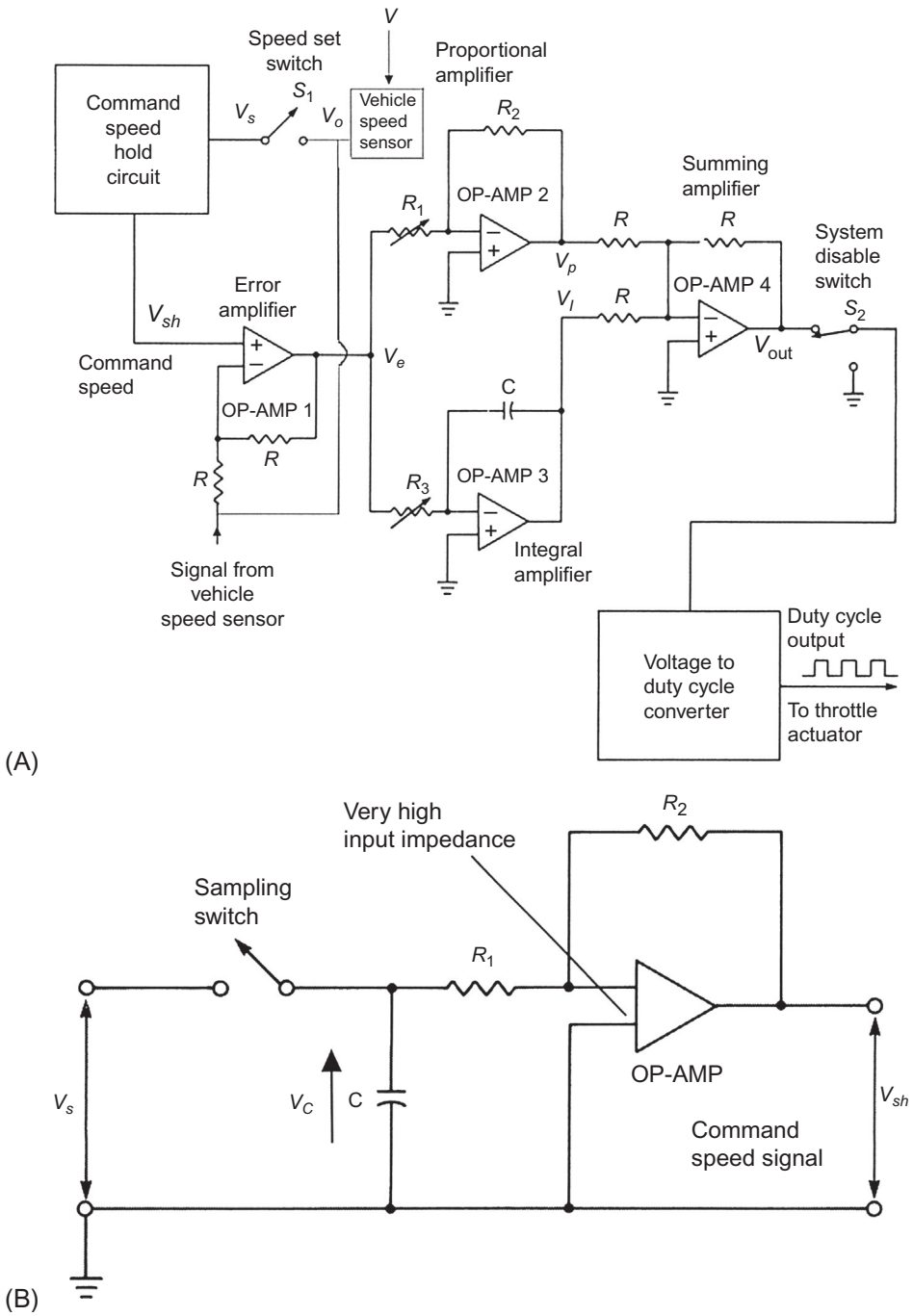


FIG. 7.11 Analog cruise control configuration. (A) PI control circuit; (B) Command speed sample and hold circuit.

The outputs of the proportional and integral amplifiers are added using a summing amplifier, op-amp 4. The summing amplifier adds voltages V_p and V_I and inverts the resulting sum. The inversion is necessary because both the proportional and integral amplifiers invert their input signals while providing amplification. Inverting the sum restores the correct sense, or polarity, to the control signal.

The summing amplifier op amp produces an analog voltage, V_{out}, that must be converted to a duty-cycle signal before it can drive the throttle actuator. A voltage-to-duty-cycle converter is used whose output directly drives the throttle actuator solenoid. The voltage-to-duty-cycle converter is a voltage-controlled oscillator that generates an output wave form at frequency f_p with duty cycle that is proportional to V_{out}.

Two switches, S_1 and S_2, are shown in Fig. 7.11A. Switch S_1 is operated by the driver to set the desired speed. It signals the sample-and-hold electronics (Fig. 7.11B) to sample the present vehicle speed at the time S_1 is activated and hold that value until the next switch operation by the driver. Voltage V_c, representing the vehicle speed at which the driver wishes to set the cruise controller, is sampled, and it charges capacitor C. A very-high-input-impedance amplifier detects the voltage on the capacitor without causing the charge on the capacitor to "leak" off. The output from this amplifier is a voltage, V_{sh}, proportional to the command speed that is sent to the error amplifier:

$$V_{sh}(t) = V_s(t_a) \tag{7.39}$$

where t_a is the time driver activates S_1.

Switch S_2 (Fig. 7.11A) is used to disable the speed controller by interrupting the control signal to the throttle actuator. Switch S_2 disables the system whenever the ignition is turned off, the controller is turned off, or the brake pedal is pressed. The controller is switched on when the driver presses the speed set switch S_1.

For safety reasons, the brake turnoff is often performed in two ways. As just mentioned, pressing the brake pedal turns off or disables the electronic control. In certain cruise control configurations that use a vacuum-operated throttle actuator, the brake pedal also mechanically opens a separate valve that is located in a hose connected to the throttle actuator cylinder. When the valve is opened by depression of the brake pedal, it allows outside air to flow into the throttle actuator cylinder so that the throttle plate is rapidly closed. The valve is shut off whenever the brake pedal is in its inactive position. This ensures a fast and complete shutdown of the speed control system whenever the driver presses the brake pedal.

ADVANCED CRUISE CONTROL

The cruise control system previously described is adequate for maintaining constant speed, provided that any required deceleration can be achieved by a throttle reduction (i.e., reduced engine power). The engine has limited braking capability with a closed throttle, and this braking in combination with aerodynamic drag and tire-rolling resistance may not provide sufficient deceleration to maintain the set speed. For example, a car entering a long, relatively steep downgrade in a mountainous region may accelerate due to gravity even with the throttle closed.

For this driving condition, vehicle speed can be maintained only by application of the brakes. For cars equipped with a conventional cruise control system, the driver has to apply braking to hold speed.

An ACC system has a means of automatic brake application whenever deceleration with throttle input alone is inadequate. A somewhat simplified block diagram of an ACC is shown in Fig. 7.12, emphasizing the automatic braking portion. This system consists of a conventional brake system with

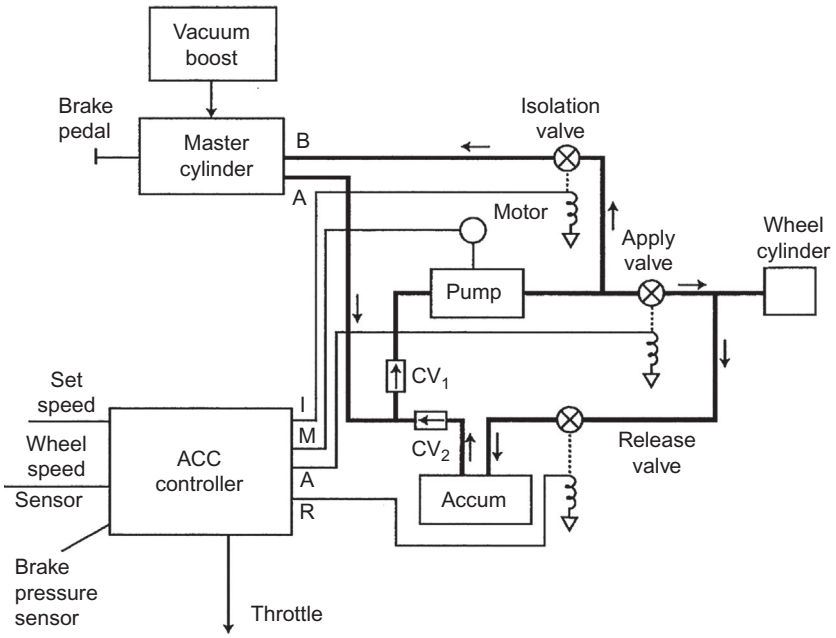


FIG. 7.12 ACC system configuration.

master cylinder wheel cylinders, vacuum boost (power brakes), and various brake lines. Fig. 7.12 shows only a single-wheel cylinder, although there are four in actual practice. In addition, proportioning valves are present to regulate the front/rear brake force ratio. Some of the components depicted in Fig. 7.12 are also a part of a system that optimizes braking under relatively low tire/road friction. This subject is explained in the following section of this chapter under the heading of antilock braking systems.

In normal driving, the system functions like a conventional brake system. As the driver applies braking force through the brake pedal to the master cylinder, brake fluid (under pressure) flows out of port A and through a brake line to the junction of check valves CV_1 and CV_2. Check valve CV_2 blocks brake fluid, whereas CV_1 permits flow through a pump assembly P and then through the apply valve (which is open) to the wheel cylinder(s), thereby applying brakes.

In cruise control mode, the ACC controller regulates the throttle (as explained above for a conventional cruise control) and the brake system via electrical output signals and in response to inputs, including the vehicle speed sensor and set cruise speed switch. The ACC system functions as described above until the maximum available deceleration with closed throttle is inadequate. Whenever there is greater deceleration required than this maximum value, the ACC applies brakes automatically. In this automatic brake mode, an electrical signal is sent from the M (i.e., motor) output of the controller to the motor, causing the pump to send more brake fluid (under pressure) through the apply valve (maintained open) to the wheel cylinder. At the same time, the release valve remains closed such that brakes are applied.

The braking pressure can be regulated by varying the isolation valve, thereby bleeding some brake fluid back to the master cylinder. By activating isolation valves separately to the four wheels, brake proportioning can be achieved. Brake release can be accomplished by sending signals from the ACC to close the apply valve and open the release valve. We present next a continuous-time model for the ACC.

The vehicle model under ACC mode is given by

$$M\dot{V} + D + Mg\sin\theta = \frac{g_A T_{bo}}{r_w} - \frac{T_B}{r_w} \tag{7.40}$$

where T_{bo} is the engine torque at closed throttle and T_B the braking torque. This braking torque is normally zero under steady cruise. It is only increased from zero in the ACC mode when required to maintain cruise speed.

Under normal circumstances, for a sufficiently steep downgrade (i.e., $\theta < 0$), T_{bo} is negligible. For simplification purposes, it is assumed that the braking torque is linearly proportional to brake pressure p_B:

$$T_B = K_B p_B \tag{7.41}$$

where K_B is a constant for the brake configuration. A linearized model for the vehicle traveling on a straight road with vehicle speed $V = V_d + \delta V$ is given by

$$\begin{aligned} M\delta\dot{V} + K_D\delta V + Mg\theta &= -K_B p_B / r_w \\ &= K_B K_A u / r_w \end{aligned} \tag{7.42}$$

where K_A is the brake pressure actuator constant, V_d is the cruise speed set point, and u is the ACC control signal.

If a PI control law is assumed for this ACC automatic braking mode, the control signal is given by

$$u = K_p e + K_I \int e dt \tag{7.43}$$

where $e = V_d - V =$ error signal $= -\delta V$, where V is the actual vehicle speed.

Substituting the control signal model into the linearized vehicle mode and taking the Laplace transform of the resulting equation yield the following:

$$\left(s + \frac{K_D}{M}\right)\delta V(s) + g\theta = -\frac{K_B K_A}{r_w M}\left[K_p + \frac{K_I}{s}\right]\delta V(s) \tag{7.44}$$

Solving for $\delta V(s)$ yields

$$\delta V(s) = \frac{gs|\theta|}{s^2 + \left(\dfrac{K_D}{M} + \dfrac{K_B K_A K_p}{r_w M}\right)s + \dfrac{K_B K_A K_I}{r_w M}} \tag{7.45}$$

Note the similarity to the model for cruise control developed earlier in which the actuator drives the throttle plate angle. In the above equation, the negative sign of the θ for a downgrade is accounted for by replacing $-\theta$ with $|\theta|$. The dynamic response of a car with ACC traveling along a straight horizontal road and encountering a steep downgrade (with slope $\theta = -|\theta|$) is similar to that for an ordinary cruise

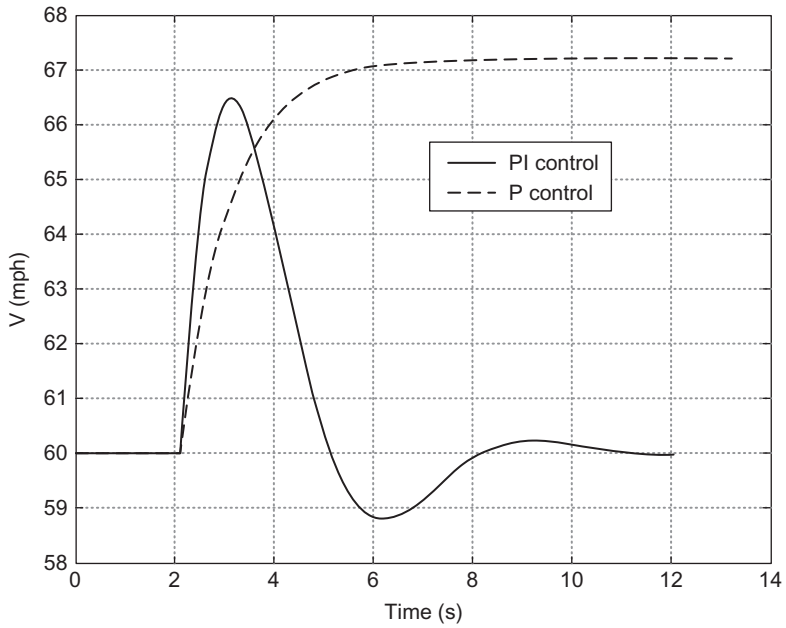


FIG. 7.13 Vehicle speed with ACC on hill with long downgrade.

control encountering a sudden change in slope except that the speed initially increases and then comes to an asymptotic value.

A simulation of this ACC was run for the same vehicle parameters of the earlier example. Here, it is assumed that the vehicle encounters the steep downgrade at $t=2$ s. It is further assumed for simplicity that the ACC switches instantly to automatic braking mode (when the throttle closed switch signals the controller). Fig. 7.13 is a plot of vehicle speed for P-only control and PI control. The same coefficients are assumed for the controller, and K_B is taken to be 4.

Fig. 7.13 is a plot of the ACC speed response to a long steep downgrade of -7% encountered at $t=2$ s for a vehicle with ACC that is initially in a steady 60 mph cruise. Note that for P-only control, the speed increases to an asymptotic value of about 67 mph. During the asymptotic range, this speed is maintained with a steady brake pressure. However, for PI control, the speed initially increases and then with applied brakes decreases with small undershoot reaching the desired cruise speed of 60 mph. The action of various control laws is described in Appendix A. The present simulation confirms the predicted behavior.

In addition to maintaining a vehicle speed on ACC in contemporary vehicles can compensate automatically for other vehicle traffic, for road obstructions, and for unintentional lane deviation. This adjustment of the control of an ACC can only be made automatically in combination with a surveillance system that detects and measures relative positions of other vehicles or obstacles. The detailed description of vehicle environmental surveillance system is explained in Chapter 10, which is devoted to safety-related systems. The sensor system components for certain optical surveillance systems are

explained in detail in Chapter 5. The information provided to the ACC by the surveillance system offers the potential for the ACC to set the speed command to a level consistent with traffic, which in a heavy traffic or road construction environment involves a reduction in the speed set point. Other options are available to vehicles equipped with automatic steering as discussed briefly later in this chapter and explained in detail in Chapter 12, which is devoted to autonomous vehicles.

Another potential application for automatic braking involves separate brake pressure applied individually to all four wheels. This independent brake application can be employed for improved handling when both braking and steering are active (e.g., braking on curves). Later in this chapter, an application of automatic braking to enhance the lateral stability of the vehicle is discussed. The theory of enhanced lateral stability is presented in detail in Chapter 10, which is devoted to safety-related systems.

ANTILOCK BRAKING SYSTEM

One of the most readily accepted applications of electronics in automobiles has been the antilock brake system (ABS). ABS is a safety-related feature that assists the driver in deceleration of the vehicle in poor or marginal braking conditions (e.g., wet or icy roads). In such conditions, panic braking by the driver (in non-ABS-equipped cars) results in reduced braking effectiveness and, typically, loss of directional control due to the tendency of the wheels to lock (i.e., to stop rolling and to be held firmly against rotation by the brakes).

In ABS-equipped cars, the wheel is prevented from locking by a mechanism that automatically regulates the force applied to the wheels by the brakes to an optimum for any given low-friction condition. The physical configuration for an ABS is shown in Fig. 7.14.

In addition to the normal brake components, including brake pedal, master cylinder, vacuum boost, wheel cylinders, calipers/disks, and brake lines, this system has a set of angular speed sensors at each

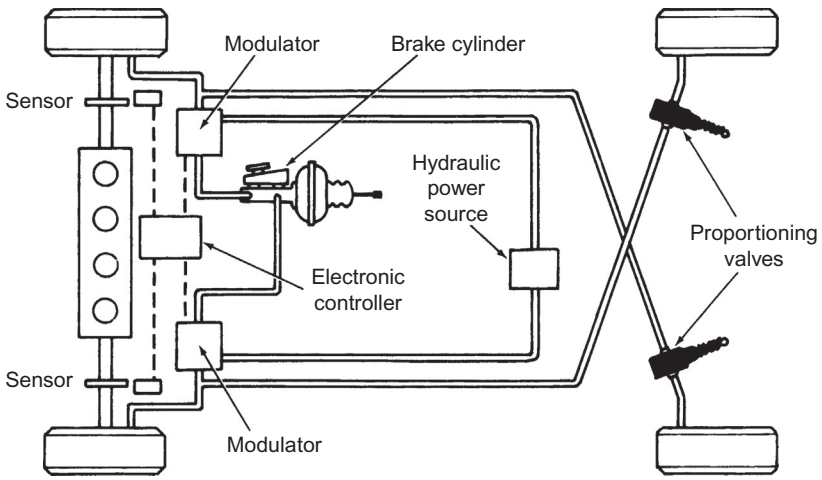


FIG. 7.14 Antilock braking system.

wheel, an electronic control module and a hydraulic brake pressure modulator (regulator). For simplicity in the drawing, only a pair of brake pressure modulators is shown. However, in practice, there is a separate modulator for each brake. A more detailed description of the configuration of the ABS (including the modulator) is presented later in this section of the chapter.

In order to understand the ABS operation, it is first necessary to understand the physical mechanism of wheel lock and vehicle skid that can occur during braking. The car is traveling at a speed U, and the wheels are rotating at an angular speed ω_w where

$$\omega_w = \frac{\pi \mathrm{RPM}_w}{30} \tag{7.46}$$

and where RPM_w is the RPM of the wheel in revolutions per minute. When the wheel is rolling (no applied brakes and no drivetrain torque), U and ω_w are linearly proportional

$$U = r_w \omega_w \tag{7.47}$$

where r_w is the tire effective radius.

When the brake pedal is depressed, the pads are forced by hydraulic pressure against the disk, as depicted schematically in Fig. 7.15A. Fig. 7.15B illustrates the forces applied to the wheel by the road during braking. This pressure causes a force that acts as a torque T_b in opposition to the wheel rotation. The actual force that decelerates the car is shown as F_b in Fig. 7.15B. The lateral force that maintains directional control of the car is shown as F_L in Fig. 7.15B.

The wheel angular speed begins to decrease, causing a difference between the vehicle speed U and the tire speed over the road (i.e., $\omega_w r_w$). In effect, the tire slips relative to the road surface. The amount of slip s determines the braking force and lateral force. The slip, as a fraction of car speed, is given by

$$s = \frac{U - \omega_w r_w}{U}$$

Note: A rolling tire has slip $s = 0$, and a fully locked tire has $s = 1$.

The braking and lateral forces are proportional to the normal force (from the weight of the car and from inertial forces due to deceleration) acting on the tire/road interface (N in Fig. 7.15B) and the friction coefficients for braking force (F_b) and lateral force (F_L):

$$\begin{aligned} F_b &= N\mu_b \\ F_L &= N\mu_L \end{aligned} \tag{7.48}$$

where μ_b is the braking friction coefficient and μ_L is the lateral friction coefficient.

These coefficients depend markedly on slip, as shown qualitatively in Fig. 7.16. The solid curves are for a dry road and the dashed curves for a wet or icy road. As brake pedal force is increased from zero, slip increases from zero. For increasing slip, μ_b increases to $s = s_o$. Further increase in slip actually decreases μ_b, thereby reducing braking effectiveness. The curves presented in Fig. 7.16 are only depicting general relationships between friction and slip and are not representative of actual curves in precise detail.

On the other hand, μ_L decreases steadily with increasing s such that for fully locked wheels the lateral force has its lowest value. For wet or icy roads, μ_L at $s = 1$ is so low that the lateral force often is insufficient to maintain directional control of the vehicle. However, directional control can often be maintained even in poor braking conditions if slip is optimally controlled. This is essentially the function of the ABS, which performs an operation equivalent to pumping the brakes (as done by

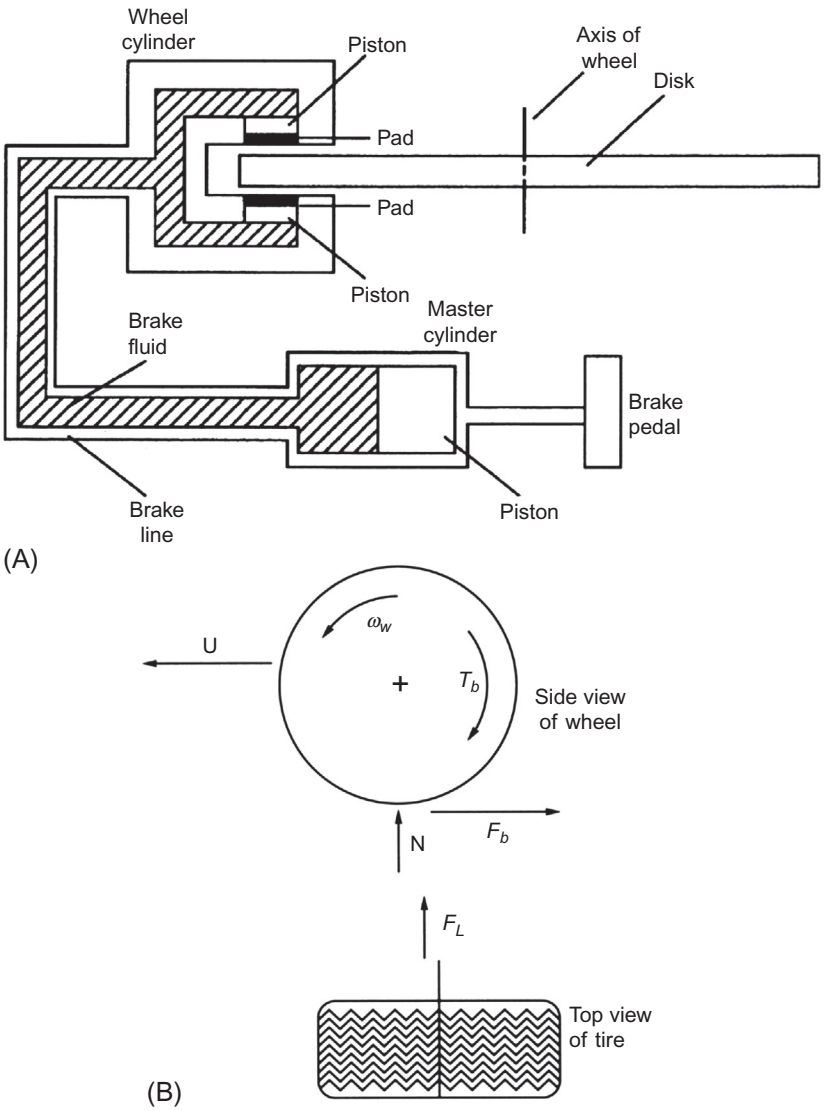


FIG. 7.15 Brake configuration and forces acting on wheel. (A) Simplified disk brake configuration. (B) Forces on wheel.

experienced drivers before the development of ABS). In ABS-equipped cars under marginal or poor braking conditions, the driver simply applies a steady brake force, and the system adjusts tire slip dynamically to achieve near-optimum value (on average) automatically.

In an exemplary ABS configuration, control over slip is affected by regulating the brake line pressure under electronic control. The configuration for ABS is shown in Fig. 7.14. This ABS regulates or

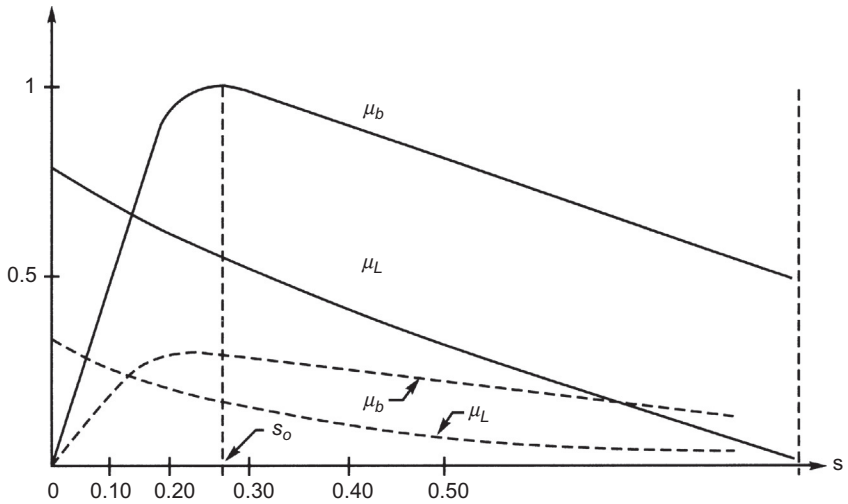


FIG. 7.16 Exemplary variation in friction coefficients with slip.

modulates brake pressure to maintain slip as near to optimum for as much time as possible (e.g., at s_o in Fig. 7.16). The operation of this ABS is based on estimating the torque T_w applied to the wheel at the road surface by the braking force F_b:

$$T_w = r_w F_b \tag{7.49}$$

The braking torque T_b is applied to the disk by the brake pads in response to brake pressure p_b and is a function of p_b:

$$T_b = f(p_b) \tag{7.50}$$

Although it is not necessary for ABS application, for the purposes of explaining ABS operation, it is convenient to simplify the model for T_b to the following:

$$T_b \cong k_b p_b \tag{7.51}$$

where k_b is a constant for the given brakes.

The difference between these two torques acts to decelerate the wheel. In accordance with basic Newtonian mechanics, the wheel torque T_w is related to braking torque and wheel deceleration by the following equation:

$$T_w = T_b + I_w \dot{\omega}_w$$

where I_w is the wheel moment of inertia about its rotational axis and $\dot{\omega}_w$ is the wheel deceleration $d\omega_w/dt$, that is, the rate of change of wheel speed.

During heavy braking under marginal conditions, sufficient braking force is applied to cause wheel lockup (in the absence of ABS control). We assume such heavy braking for the following discussion of the ABS. As brake pressure is applied, T_b increases and ω_w decreases, causing slip to increase. The wheel torque is proportional to μ_b, which reaches a peak at slip s_o. Consequently, the wheel torque

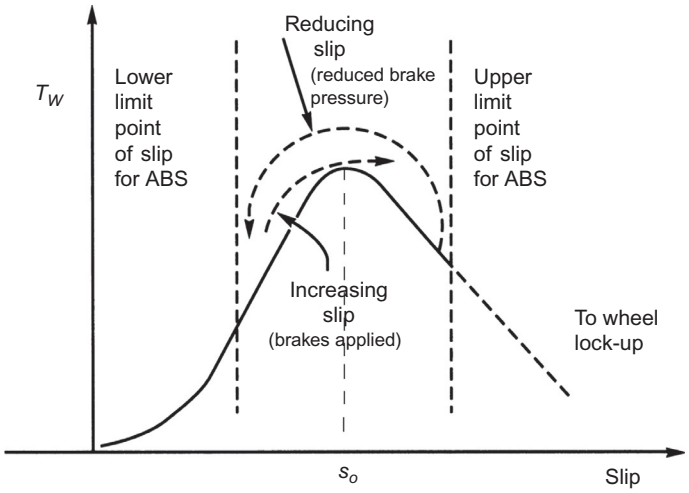


FIG. 7.17 Wheel torque versus slip under ABS action.

reaches a maximum value (assuming sufficient brake force is applied) at this level of slip and decreases for $s > s_o$. For this region of slip, the slope of μ_b is negative (i.e., $\frac{d\mu_b}{ds} < 0$), and wheel deceleration is unstable causing $\omega_w \rightarrow 0$ resulting in wheel lock condition. It is the function of the ABS to regulate T_b to maintain slip near-optimum as explained below.

Fig. 7.17 is a sketch of wheel torque versus slip during ABS action illustrating the peak T_w. After the peak wheel torque is sensed electronically, the electronic control system commands that brake pressure be reduced (via the brake pressure modulator). This point is indicated in Fig. 7.17 as the limit point of slip for the ABS. As the brake pressure is reduced, slip is reduced, and the wheel torque again passes through a maximum.

The wheel torque reaches a value below the peak on the low slip side denoted lower limit point of slip, and at this point, brake pressure is again increased. The system will continue to cycle, maintaining slip near the optimal value as long as the brakes are applied, and the braking conditions lead to wheel lockup.

The ABS control laws and algorithms are, naturally, proprietary for each manufacturer. Rather than dealing with such proprietary issues here, an ABS control concept is presented here based upon a paper by the author of this book and has demonstrated successful ABS operation in laboratory (wheel dynamometer) tests. This discussion can be considered exemplary of much of the mechanical dynamics and control algorithms.

An ideal ABS control would maintain braking force/torque such that slip would remain at exactly the optimum slip (i.e., s_o) for any given tire/road condition. However, a suboptimal control system having very near-optimal performance can be achieved by cycling brake pressure such that slip cycles up and down about the optimum as depicted qualitatively in Fig. 7.17. The cycling should be such that the average of the time varying s and μ_b, μ_L are very close to optimum.

The present exemplary ABS control is based upon the use of a so-called sliding mode observer (SMO). The SMO is a robust state vector estimator that has the capability of estimating very closely the state vector of a dynamic system (see Appendix A for the definition of a state vector). The SMO for the present discussion estimates a single-dimensional state vector, the differential torque applied to the wheel, (δT_b), where

$$\begin{aligned} \delta T_b &= T_w - T_b \\ &= I_w \dot{\omega}_w \end{aligned} \tag{7.53}$$

Rewriting Eq. (7.53) yields a form from which the SMO can be readily derived:

$$\dot{\omega} = -\frac{\delta T_b}{I_w} \tag{7.54}$$

The goal for the SMO for this application is to calculate an estimate $\left(\delta \hat{T}_b\right)$ of the differential torque. It obtains $\delta \hat{T}_b$ by solving the following differential equation for the estimate $(\hat{\omega}_w)$ of wheel angular speed:

$$\dot{\hat{\omega}}_w = -m \operatorname{sgn}(\hat{\omega}_w - \omega_w) \tag{7.55}$$

where sign $(\hat{\omega}_w - \omega_w)$ is the sign of the argument and where m is the SMO gain that must satisfy the following inequality:

$$m \geq \max |\delta T_b| \tag{7.56}$$

The SMO requires an accurate, precise measurement of wheel angular speed (ω_w). As shown in Drakonov (1997)[1], the desired estimate $\left(\delta \hat{T}_b\right)$ is the solution to the following first-order differential equation:

$$\tau \frac{d\delta \hat{T}_b}{dt} + \delta \hat{T}_b = -m \operatorname{sgn}(\hat{\omega}_w - \omega_w) \tag{7.57}$$

Effectively, $\left(\delta \hat{T}_b\right)$ is a first-order low-pass-filtered version of the RHS of the above equation. The low-pass filter (LPF) bandwidth (i.e., $1/\tau$) must be sufficiently large to accommodate the relatively large fluctuations in wheel angular speed. It is possible to use a higher-order than first-order LPF. Experiments and simulations have been run with second-order LPF with good braking performance. The SMO generates a very close estimate of δT_b such that the control logic can detect that extremal values for the actual differential torque have occurred by detecting extremal values of the SMO estimate $\left(\delta \hat{T}_b\right)$. This estimate is the input to the control algorithm for regulating brake pressure.

The actual control algorithm for applying or releasing brakes is based upon the estimate of δT_b. Whenever the slip passes the optimal value (s_o), either increasing or decreasing the $\delta \hat{T}_b$, has an extremal value. One control scheme incorporates an extremal value detector applied to $\delta \hat{T}_b$. Whenever an extremum is detected with brakes applied, this indicates s has crossed s_o while increasing. Upon detection of this extremum, the control generates a command signal to release brake pressure (using a mechanism described below). Conversely, whenever an extremal value of $\delta \hat{T}_b$ is detected with brakes not being applied (or at reduced brake pressure), this indicates that s has crossed s_o while decreasing. Upon detecting this condition, the control system generates a signal that causes brake pressure to be reapplied.

[1]Drakonov S. Sliding Mode Observer Based on Equivalent Control Methods. 31st CDC Conference, Tucson, AZ, Dec. 1997.

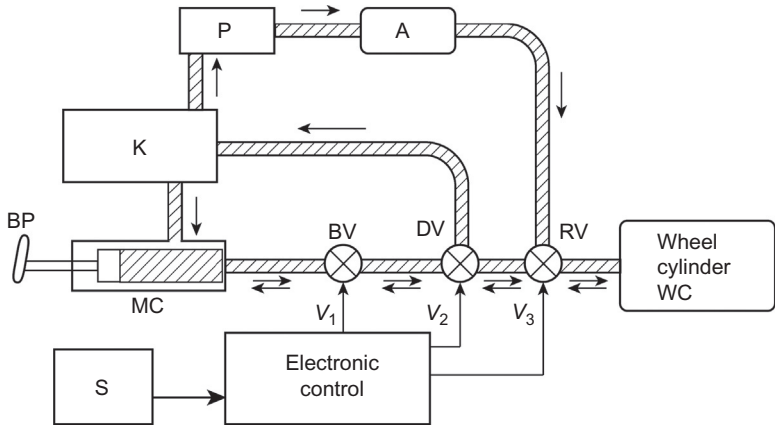


FIG. 7.18 Schematic illustration of ABS.

During ABS operation, the control logic essentially detects that slip has increased beyond $s_{o,}$ and at some point between s_o and the upper limit point of slip for ABS (as shown in Fig. 7.17), this logic detects an impending wheel lock condition and generates control signals that cause brake pressure to rapidly decrease. With brake pressure reduced, the wheel tends toward a rolling condition, and slip decreases as depicted in Fig. 7.17. As the slip crosses s_o while decreasing, μ_b increases to its maximum value at s_o and then decreases. The corresponding δT_b has an extremum as s crosses s_o. The SMO detects the extremal value of $\delta\hat{T}_b$, thereby creating a logic condition that brakes are to be reapplied.

In an actual ABS, the brakes are individually controlled at each wheel. Separate control of each wheel is required because during braking, the inertial forces can result in different normal force (N) at each wheel. In addition, the friction coefficient may well be different for each tire/road interface.

There are two major benefits to ABS. One of these is achieving optimal friction coefficient at each wheel. The other is to maintain sufficient lateral friction coefficient (μ_L) for good directional control of the vehicle during stopping.

The mechanism for modulating brake pressure is illustrated in Fig. 7.18.

In Fig. 7.18, the notation is as follows:

BP	Brake pedal
MC	Master cylinder
K	Brake fluid reservoir
BV	Blocking valve
DV	Pressure dump valve
RV	Repressurization valve
P	Pump
A	Accumulator
S	Wheel speed sensor
WC	Wheel cylinder
V_1,V_2,V_3	Actuator control signals

During braking with ABS control, the driver is assumed to apply brake pressure to the line connecting MC and WC. The driver is assumed to maintain a relatively high pressure. Although Fig. 7.18 depicts ABS for a single wheel, it is assumed that a separate set of valves are supplied for each of the four wheel cylinders.

Each of the valves depicted in Fig. 7.18 are two-position solenoid-operated valves, each having two separate functions. The blocking valve in the inactive position for $V_1 = 0$ passes brake fluid under pressure from its input line to its output line. Under normal (non-ABS) braking, the dump valve ($V_2 = 0$) passes this fluid from its input to its output line that leads to the repressurization valve. This latter valve passes the pressurized brake fluid to the wheel cylinder that thereby applies brake torque to the corresponding wheel.

Whenever the ABS control detects a potential wheel lockup owing to slip $s > \mathrm{s_o}$ (due to the negative $d\mu_b/ds$), it generates nonzero control signals V_1, V_2, and V_3 in a precise sequence. In the exemplary ABS, potential wheel lock is detected by an extremum in $\delta\hat{T}_b$ with brakes applied. The control sends a voltage V_1 to BV that causes it to switch to a brake pressure-blocked position. In this position, the master cylinder is isolated from the wheel cylinder by the BV. Only the input line to BV is under driver-applied brake pressure. A few milliseconds after the BV is activated, the control generates a voltage V_2 that activates the DV that switches it to its second position. In this position, the line to the RV and wheel cylinder is connected to the reservoir, and the WC pressure drops rapidly toward 0.

During all times, a pump (P) maintains a supply of brake fluid under pressure in accumulator A. In its deactivated state (i.e., $V_3 = 0$), the RV isolates the accumulator from the line leading to the WC and provides a stop in the A output line. This A pressure is the pressure that is used to repressure the WC at the appropriate time. This appropriate time is the time at which the control system detects an extremum in $\delta\hat{T}_b$ for brakes "off" (or low T_b). When the controller detects this condition, it initially sets control voltage $V_2 = 0$, thereby deactivating DV. A few milliseconds after V_2 is set to zero, the controller generates voltage V_3 that activates the repressurization valve. When activated, the RV connects the A with its pressurized brake fluid to the WC. It simultaneously applies the pressure to the output line of the DV that also pressurizes the BV output line. The pressurized WC applies the force required to apply brake torque T_b to the wheel.

Assuming that a low μ_b condition is maintained, the process of increasing slip with s passing s_o and a new extremal valve in $\delta\hat{T}_b$ is detected. The entire process of pressure dump followed by repressurization is repeated. The cycling of the ABS normally continues until the wheel speed with brakes "off" is below a preset value (e.g., 1–5 mph) or until the driver releases the brake pedals.

Fig. 7.19 illustrates the braking during an ABS action in simulation of an experimental system. In this illustration, the vehicle is initially traveling at 55 mph, and the brakes are applied as indicated by decreasing speed of Fig. 7.19A. The solid curve of Fig. 7.19A depicts vehicle speed over the ground and the dashed curve the instantaneous wheel speed ($r_w\omega_w$). The wheel speed begins to drop until the control detects incipient wheel lock (e.g., for an extremum of $\delta\hat{T}_b$). At this point, the ABS reduces brake pressure, and the wheel speed increases until the control reaches the condition to reapply brake pressure. With the high applied brake pressure, the wheels again tend toward lockup, and ABS reduces brake pressure. The cycle continues until the vehicle is slowed sufficiently.

Fig. 7.19B depicts the instantaneous friction coefficient $\mu_b(t)$. It can be seen that the ABS action of releasing and then reapplying brake pressure causes this μ_b to cycle back and forth about its peak value ($\mu_b(s_o)$). Similar results to those of Fig. 7.19 were achieved in laboratory tests with suitable instrumentation.

It should be noted that by maintaining slip near s_o, the maximum deceleration is achieved for a given set of conditions. Some reduction in lateral force occurs from its maximum value by maintaining

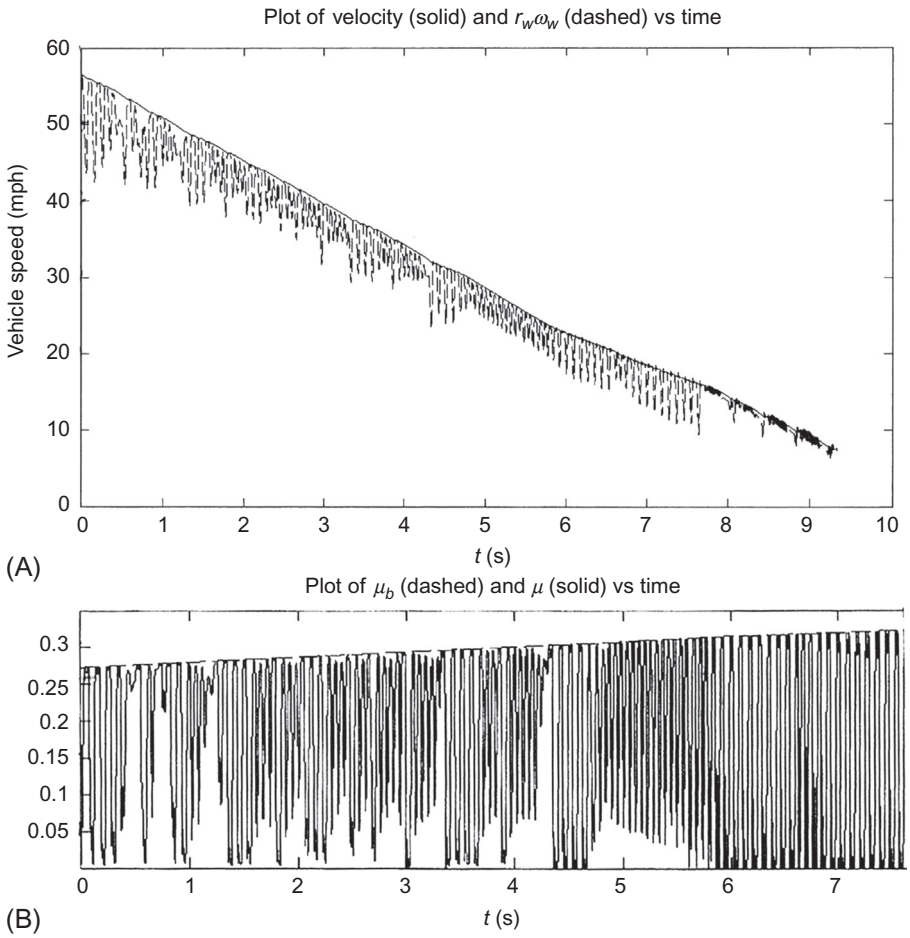


FIG. 7.19 Illustration of ABS action. (A) Tire angular and vehicle ground speed. (B) Actual (dashed) and optimal (solid) brake friction coefficient.

slip near s_o. However, in most cases, the lateral force is large enough to maintain directional control, thereby permitting the driver to steer the vehicle.

In some ABSs, the mean value of the slip oscillations is shifted below s_o, sacrificing some braking effectiveness to enhance directional control. This can be accomplished by adjusting the upper and lower slip limits.

The components of the ABS depicted in Fig. 7.18 can be combined with an electronic control system and system capable of monitoring the vehicle, traffic, and obstacle environment surrounding the vehicle. These combined components can yield an automatic braking system to prevent a collision in the event the driver has been determined by the system to not have reacted to a potential collision or in an autonomous vehicle. However, since this application of ABS components is part of a safety-related system, the detailed explanation of automatic braking is contained in Chapter 10, which is devoted exclusively to safety-related vehicular electronic systems.

TIRE SLIP CONTROLLER

Another benefit of the ABS is that the brake pressure modulator can be used for ACC as explained earlier and for tire slip control. Tire slip is effective in moving the car forward just as it is in braking. Under normal driving circumstances with power train torque applied to the drive wheels, the slip that was defined previously for braking is negative. That is, the tire is actually moving at a speed that is greater than for a purely rolling tire (i.e., $r_w\omega_w > U$). In fact, the traction force is proportional to slip.

For wet or icy roads, the friction coefficient can become very low and excessive slip can develop. In extreme cases, one of the driving wheels may be on ice or in snow, while the other is on a dry (or drier) surface. Because of the action of the differential (see Chapter 6 and Fig. 6.30), the low-friction tire will spin, and relatively little torque will be applied to the dry-wheel side. In such circumstances, it may be difficult for the driver to move the car even though one wheel is on a relatively good friction surface.

The difficulty can be overcome by applying a braking force to the free spinning wheel. In this case, the differential action is such that torque is applied to the relatively dry-wheel surface, and the car can be moved. In the example ABS, such braking force can be applied to the free spinning wheel by the hydraulic brake pressure modulator (assuming a separate modulator for each drive wheel). Control of this modulator is based on measurements of the speed of the two drive wheels. Of course, the ABS already incorporates wheel speed measurements, as discussed previously. The ABS electronics have the capability of performing comparisons of these two wheel speeds and of determining that braking is required of one drive wheel to prevent wheel spin.

ABS components have another important application in relationship to vehicle safety. This application of ABS technology is in a vehicular electronic system that is called enhanced stability system (ESS). Although major components of the EVS are part of ABS, the primary purpose is to improve the directional stability of vehicles during maneuvers involving steering inputs. The EVS is discussed in Chapter 10, which is devoted to electronic safety-related vehicle systems because the end goal of EVS is to improve vehicle safety. An entire section of Chapter 10 is devoted solely to EVS with multiple references to relevant portions of this chapter.

Still another safety-related application of ABS or its components is automatic braking. This topic is also covered in detail in Chapter 10. Although major components of ABS are involved in automatic braking, there are sensor inputs to automatic braking that are beyond those discussed in this chapter for ABS application. These involve sensing the environment surrounding the given vehicle that is explained in Chapter 10. Also explained in Chapter 10 is the application of automatic braking to a collision avoidance system.

Antilock braking can also be achieved with electrohydraulic brakes. An electrohydraulic brake system was described in the section of this chapter devoted to ACC.

Recall that for ACC a motor-driven pump supplied brake fluid through a solenoid-operated "brakes" apply valve to the wheel cylinder. For ACC application of the brakes, the apply and isolation valves operate separately to regulate the braking to each of the four wheels.

ELECTRONIC SUSPENSION SYSTEM

An automotive suspension system consists of springs, shock absorbers, and various linkages to connect the wheel assembly to the car body. The purpose of the suspension system is to isolate the car body motion as much as possible from wheel vertical motion due to rough-road input. Fig. 7.20 depicts,

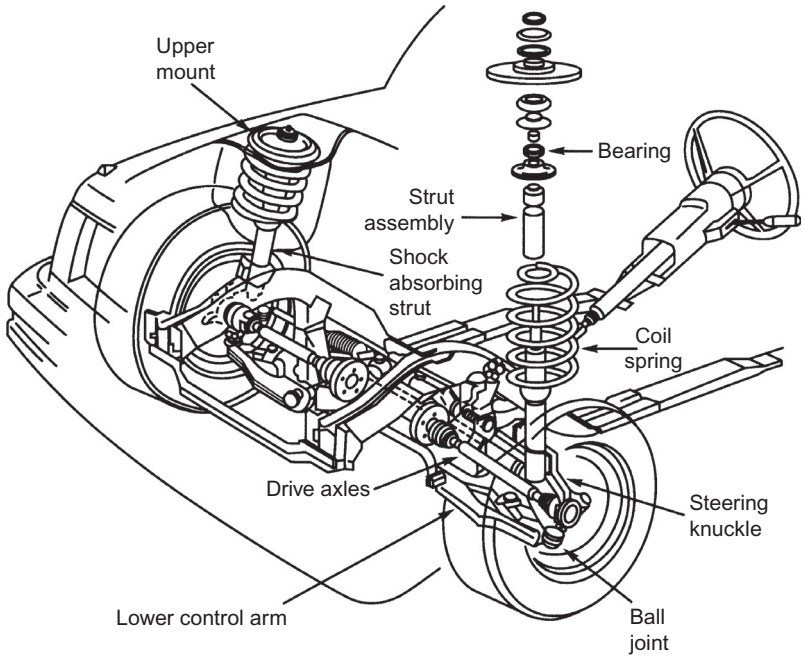


FIG. 7.20 Illustration of front suspension system.

schematically, the suspension system for the front wheels of a front-wheel-drive car. In essence, a suspension system is a mass, spring, damping assembly that connects the car body (whose mass is called the "sprung" mass to the wheel/axle, brake, and other linkages connected to them, which are called the "unsprung" mass).

The two primary subjective performance measures from a driver/passenger standpoint are ride and handling. *Ride* refers to the motion of the car body in response to road bumps or irregularities. *Handling* refers to how well the car body responds to dynamic vehicle motion such as cornering or hard braking.

Damping in the suspension system is provided by the shock absorber portion of the strut assemble. Viscous damping is provided by fluid motion through orifices in a piston portion of the strut. The structure and details of a strut are given later in this chapter, but the interested reader can look ahead to Fig. 7.23. For the present, attention is focused on the influence of strut damping on ride and handling. Generally speaking, ride is improved by lowering the shock absorber damping, whereas handling is improved by increasing this damping. In traditional suspension design, the damping parameter is fixed and is chosen to achieve a compromise between ride and handling (i.e., an intermediate value for shock absorber damping is chosen).

In electronically controlled suspension systems, this damping can be varied depending on driving conditions and road roughness characteristics. That is, the suspension system adapts to inputs to maintain the best possible ride, subject to handling constraints that are associated with safety.

There are two major classes of electronic suspension control systems: active and semiactive. The semiactive suspension system is purely dissipative (i.e., power is absorbed by the shock absorber under

control of a microcontroller). In this system, the shock absorber damping is regulated to absorb the power of the wheel motion in accordance with the driving conditions.

In an active suspension system, power is added to the suspension system via a hydraulic or pneumatic power source. At the time of the writing of this book, electronic control of commercial suspension systems is primarily semiactive. In this chapter, we explain the semiactive system first, then the active one.

The primary purpose of the semiactive suspension system is to provide a good ride for as much of the time as possible without sacrificing handling. Good ride is achieved if the car's body is isolated as much as possible from the road surface variations. The vertical input to the unsprung mass motion is the road surface profile. For a car traveling at a steady speed, this input is a random process. Depending upon the nature of the road surface (i.e., newly paved road vs. ungraded gravel dirt road), this random process may be either a stationary or a nonstationary process. For the following discussion, we assume a stationary random process. A semiactive suspension controls the shock absorber damping to achieve the best possible ride without sacrificing handling performance.

In addition to providing isolation of the sprung mass (i.e., car body and contents), the suspension system has another major function. It must also dynamically maintain the tire normal force as the unsprung mass (wheel assembly) travels up and down due to road roughness. Recall from the discussion of antilock braking that braking and lateral forces depend on normal tire force. Of course, in the long-term time average, the normal forces will total the vehicle weight plus any inertial forces due to acceleration, deceleration, or cornering.

However, as the car travels over the road, the unsprung mass moves up and down in response to road input. This motion causes a variation in normal force, with a corresponding variation in potential cornering or braking forces. For example, while driving on a rough curved road, there is a potential loss of steering or braking effectiveness if the suspension system does not have good damping characteristics. We consider next certain aspects of vehicle dynamics to understand the role played by electronically controlled suspension.

The geometry for describing the vehicle motion relative to the suspension is depicted in Fig. 7.21A and B. In this figure, three major axes are defined for the vehicle: (1) longitudinal, (2) lateral, and (3) vertical. The ECEF inertial coordinate system axes are denoted (x',y',z'). The vehicle body axes are denoted (x,y,z).

The longitudinal axis is a line in the plane of symmetry through the center of gravity (CG) parallel to a ground reference plane. The ground plane is the plane through the wheel axles when the vehicle is sitting on an exactly horizontal plane. In this configuration, the deflection of the front and rear springs due to vehicle weight depends upon the location of the CG along the longitudinal axis. Fig. 7.21A is a side view of the vehicle depicting the body longitudinal axis x (fixed to the vehicle).

This figure also depicts the x-axis for the vehicle at rest with the x'-axis that constitutes an inertial (e.g., ECEF) reference. In this figure, the x-axis is deflected by a "pitch angle" α_p relative to the x'-axis. The vertical displacement of the CG is denoted δz_{cg} in the figure and is called heave. The front and rear springs are assumed to be identical right (r) and left (l). The front suspension spring rate is denoted K_F and the rear K_R. Viscous damping is also assumed to be symmetrical right and left and has linear damping coefficients D_F and D_R for front and rear, respectively (in the present, simplified model).

Fig. 7.21B depicts the vehicle in a front view for which the body lateral axis (y) is shown in the rest position by the dashed line y' and in the deflected position by the solid line. The angle ϕ_R is the "roll" angle about the longitudinal axis. The z-axis is orthogonal to the x–y plane through the CG. The y'- and z'-axis are part of the inertial reference for the following discussion on vehicle dynamic motion.

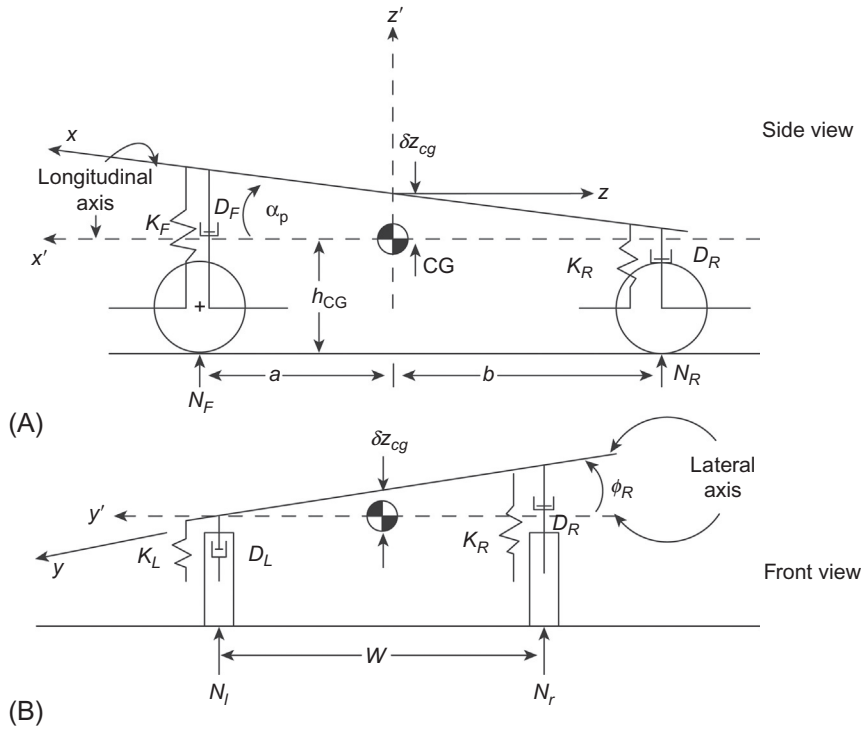


FIG. 7.21 Schematic illustration of suspension. (A) Side view of vehicle and ECEF coordinates. (B) Front view of vehicle and ECEF coordinates.

It is beyond the scope of this book to present a full discussion of vehicle dynamics that involves sets of coupled nonlinear differential equation models. Rather, the goal here is to focus on electronic control of the suspension and to illustrate the corresponding aspect of vehicle dynamics for a few representative maneuvers. For this purpose, a set of simplified linear dynamic models are presented. In such simplified models, there are many forces acting on the vehicle sprung mass including: drivetrain and braking torques/forces, inertial forces, and normal forces coupled from the unsprung mass acting on the tires to the sprung mass.

The normal forces acting on the tires are different for all four wheels whenever the vehicle is maneuvering. These forces include components due to the vehicle weight and reaction forces to inertial forces due to vehicle dynamics. These four forces have the following notations:

N_{Fr}	Front right
$N_{F\ell}$	Front left
N_{Rr}	Rear right
$N_{R\ell}$	Rear left

We begin with a relatively simple example vehicle maneuver consisting of braking on a straight and level road. The front and rear forces acting on the car from the tires are denoted F_F and F_R, respectively. These forces are positive for acceleration and negative for braking, which is assumed here, and are given by

$$\begin{aligned} F_F &= (N_{Fr} + N_{F\ell})\mu_F \\ F_R &= (N_{Rr} + N_{R\ell})\mu_R \end{aligned} \tag{7.58}$$

where μ_F is the friction coefficient for front tires and μ_R is the friction coefficient for rear tires.

The combination of these braking forces produces a moment about the CG T_b given by

$$T_b = (F_F + F_R)h_{\text{CG}} \tag{7.59}$$

Countering this moment is a moment (T_n) about the CG due to the tire normal forces given by

$$T_n = N_F a - N_R b$$

where

$$\begin{aligned} N_F &= N_{Fr} + N_{Fr} \\ N_R &= N_{Rr} + N_{R\ell} \end{aligned} \tag{7.60}$$

and where a and b are the distances along the longitudinal axis of the vehicle from the CG to the front and rear axles, respectively (i.e., see Fig. 7.21A).

The normal forces acting on the tires are transmitted through the tires to the spring/damper system of the suspension. For the present, the tire dynamics are neglected although they are included in a later example. The forces N_F and N_R produce a deflection in the suspension springs from the unloaded positions such that N_F and N_R are given by

$$\begin{aligned} N_F &= -[K_F \delta z_F + D_F \delta \dot{z}_F] \\ N_R &= -[K_R \delta z_R + D_R \delta \dot{z}_R] \end{aligned} \tag{7.61}$$

where δz_F is the deflection of front spring and δz_R the deflection of rear spring.

$$\begin{aligned} K_R &= K_{Rr} + K_{R\ell} = \text{rear spring rate} \\ K_F &= K_{Fr} + K_{F\ell} = \text{front spring rate} \\ D_F &= D_{Fr} + D_{F\ell} = \text{front damping coefficient} \\ D_R &= R_{Rr} + D_{R\ell} = \text{rear damping coefficient} \end{aligned} \tag{7.62}$$

Note that in the absence of any vertical motion of the CG (i.e., it is assumed here that $\delta \ddot{z}_{cg} = 0$), the normal forces sum to the vehicle weight (W_V):

$$N_F + N_R = W_V \tag{7.63}$$

Furthermore, it is reasonable to assume that front and rear tires have identical friction coefficient

$$\mu_R = \mu_F$$

The total force acting on the vehicle due to braking is given by

$$\begin{aligned} F &= F_F + F_R \\ &= -\mu W_V \end{aligned}$$

The moment acting around the CG due braking (T_b) is given by

$$T_b = -W_V \mu h_{\text{CG}}$$

The sum of the moments of all forces acting on the sprung mass results in an angular acceleration of the pitch angle $(\ddot{\alpha}_p)$ about the lateral (y) axis, yielding the following model:

$$I_{yy}\ddot{\alpha}_p = -\mu W_V h_{\mathrm{CG}} + T_n \tag{7.64}$$

where I_{yy} = moment of inertia of the sprung mass about the lateral axis through the CG.

$$T_n = aN_F - bN_R$$

For sufficiently small pitch angle changes, the front and rear displacement and vertical velocity are given by

$$\begin{aligned}
\delta z_F &= a\alpha_p (\text{front displacement}) \\
\delta z_R &= -b\alpha_p (\text{rear displacement}) \\
\delta \dot{z}_F &= a\dot{\alpha}_p (\text{front vertical velocity}) \\
\delta \dot{z}_R &= -b\dot{\alpha}_p (\text{rear vertical velocity})
\end{aligned} \tag{7.65}$$

Substituting these relationships into the pitch dynamic Eq. (7.64) yields

$$I_{yy}\ddot{\alpha}_p = (F_F + F_R)h_{\mathrm{CG}} - \left[a\left(K_F a\alpha_p + D_F a\dot{\alpha}_p\right) + b\left(K_R b\alpha_p + D_R b\dot{\alpha}_p\right)\right] \tag{7.66}$$

Simplifying and rearranging terms in this equation yield the following second-order differential equation in α_p:

$$I_{yy}\ddot{\alpha}_p + D\dot{\alpha}_p + K\alpha_p = Fh_{\mathrm{CG}} \tag{7.67}$$

where

$$\begin{aligned}
D &= a^2 D_F + b^2 D_R \\
K &= a^2 K_F + b^2 K_R
\end{aligned}$$

The operational transfer function ($H_\alpha(s)$) relating braking force to pitch angle is given by

$$\begin{aligned}
H_\alpha(s) &= \frac{\alpha_p(s)}{F(s)} \\
&= \frac{h_{\mathrm{CG}}}{I_{yy}s^2 + Ds + K} = \frac{h_{\mathrm{CG}}}{I_{yy}}\left[\frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2}\right]
\end{aligned} \tag{7.68}$$

where $F(s) = F_F(s) + F_R(s)$

and $\omega_n = \sqrt{\dfrac{K}{I_{yy}}}$

$$\zeta = D/\left(2I_{yy}\omega_n\right) = \text{damping ratio}$$

Solution to this equation for the pitch dynamics due to an arbitrary braking force function $F(t)$ is found using the methods of Appendix A or for any given vehicle via simulation. For example, the pitch angle response to a step of amplitude change in braking force of magnitude F_o increases from $\alpha_p = 0$ with $\dot{\alpha}_p = 0$ rising toward an asymptotic value (α_{pss}) of

$$\alpha_{\mathrm{pss}} = -\frac{h_{\mathrm{CG}}F_0}{K} \tag{7.69}$$

Depending on the damping ratio ζ, there may be overshoot in α_p before settling to α_{pss} where, with the sign convention of Fig. 7.21A, $\alpha_{\mathrm{pss}} < 0$.

In addition to the operational transfer function, the pitch dynamics due to braking are given by the sinusoidal frequency response $H_\alpha(j\omega)$, which is given by

$$H_\alpha(j\omega) = \frac{h_{CG}/I_{yy}}{(\omega_n^2 - \omega^2) + 2j\zeta\omega_n\omega} \tag{7.70}$$

The peak response occurs at $\omega = \omega_n$ and has magnitude

$$|H_\alpha(j\omega_n)| = \frac{h_{CG}}{2\zeta K} \tag{7.71}$$

and a 90 degree phase shift from $F(j\omega)$ to $\alpha_p(j\omega)$. The importance of damping in determining the resonant response of pitch dynamics is clear from this frequency response.

Recall from the discussion of ABS that the braking force during periods in which ABS is active is time varying and is often essentially periodic. The pitch dynamic response to ABS cycling is potentially a concern in ride dynamics, although the excitation frequency is normally far from pitch dynamic resonance. Nevertheless, electronically damping, as discussed later, could potentially improve ride quality.

The pitch dynamic sinusoidal frequency response (although greatly simplified) has been developed and shown to be determined by suspension spring rate and damping. A similar set of equations describe the vertical displacement (i.e., heave) dynamics. A similar sinusoidal frequency response can be derived for heave. However, this discussion is deferred to a later section in which the vertical dynamic models include wheel and tire dynamics. Later in this chapter, a model is developed with these dynamics included. For the moment, we consider these dynamics and the associated frequency response qualitatively for an exemplary vehicle.

Fig. 7.22 illustrates qualitatively a representative tire normal force variation as a function of frequency of excitation for a fixed-amplitude, variable-frequency sinusoidal excitation (see Appendix A

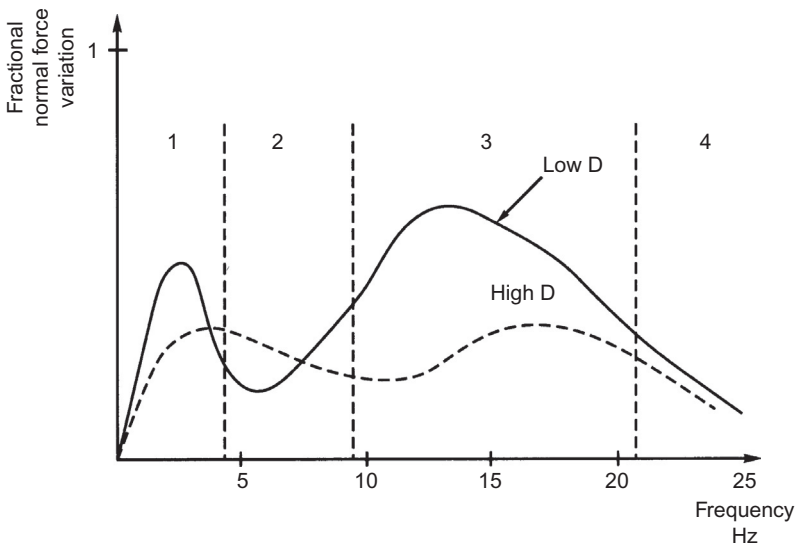


FIG. 7.22 Normal force variation due to sinusoidal excitation versus frequency.

for a discussion of sinusoidal frequency response) for an actual vehicle. The solid curve is the response for a relatively low-damping-coefficient shock absorber, and the dashed curve is the response for a relatively high damping coefficient.

The ordinate of the plot in Fig. 7.22 is the ratio of amplitude of force variation to the average normal load (i.e., due to weight). There are two relative peaks in this response. The lower peak is ~1–2 Hz and is generally associated with spring/sprung mass oscillation. The second peak, which is in the general region of 12–15 Hz, is resonance of the spring/unsprung mass combination.

Generally speaking, for any given fixed suspension system, ride and handling cannot both be optimized simultaneously. A car with a good ride is one in which the sprung mass motion/acceleration due to rough-road input is minimized. In particular, the sprung mass motion in the frequency region from about 2 to 8 Hz has often been found to be the most important for good subjective ride. Good ride is achieved for relatively low damping (low D in Fig. 7.22).

For low damping, the unsprung mass moves relatively freely due to road input, while the sprung mass motion remains relatively low. Note from Fig. 7.22 that this low damping results in relatively high variation in normal force, particularly near the two peak frequencies. That is, low damping results in relatively poor handling characteristics.

With respect to the four frequency regions of Fig. 7.22, the following generally desired suspension damping characteristics can be identified:

Region	Frequency (Hz)	Damping
1. Sprung mass mode	1–2	High
2. Intermediate ride	2–8	Low
3. Unsprung mass resonance	8–20	High
4. Harshness	>20	Low

Another major input to the vehicle that affects handling is steering input that causes maneuvers out of the ECEF inertial reference vertical plane (e.g., cornering). Whenever the car is executing such maneuvers, there is a lateral acceleration. This acceleration acting through the CG causes the vehicle to roll in a direction opposite to the maneuver.

Another relatively simple example of vehicle dynamics involves the vehicle encountering a curve in a level road. For convenience, assume that the car is traveling a straight road for $t<0$ and then encounters the curve at $t=0$. This example illustrates the influence of such a maneuver on roll dynamics (i.e., $\phi_R(t)$). For this example, it is necessary to include the variable ψ (which was introduced earlier in the chapter and is called yaw) in the dynamic model. It is the change in direction of the vehicle longitudinal axis relative to its direction on the straight level road. Because the road for $t<0$ is straight, the initial direction forms the ECEF inertial reference frame for this example. The notation for the time rate of change of ψ is r:

$$r=\dot{\psi} \tag{7.72}$$

Similarly, the notation for $\dot{\phi}_R$ is taken to be p:

$$p=\dot{\phi}_R \tag{7.73}$$

The lateral velocity component of the CG is denoted v:

$$v=\dot{y} \tag{7.74}$$

The inertial forces due to the motion of the car along the curve create a rolling moment T_R about the CG given by

$$T_R = -Mh_{\mathrm{CG}}(\dot{v} + ru_0) \tag{7.75}$$

where u_o is the vehicle speed (assumed constant) and M the vehicle sprung mass.

The sum of the moments about the CG for this maneuver yields the following approximate differential equation:

$$I_{xx}\ddot{\phi} + Mh_{\mathrm{CG}}(\dot{v} + ru_0) = -\left(L_\phi \phi_R + L_p p\right) \tag{7.76}$$

where I_{xx} is the moment of inertia of the sprung mass structure about the body longitudinal axis and where L_ϕ and L_p are given by:

$$\begin{aligned} L_\phi &= (K_F + K_R)w^2 \\ L_p &= (D_F + D_R)w^2 \end{aligned} \tag{7.77}$$

where w is the distance between right and left tire planes of symmetry (Fig. 7.21B).

In this equation, a term proportional to the cross product of inertia $I_{xz}\dot{r}$ has been neglected without serious loss of generality as it is usually small except for relatively high $\dot{r}$. The above equation can be rewritten in terms of the inertial (rolling) moment (T_R) in the form

$$I_{xx}\ddot{\phi}_R + L_p\dot{\phi}_R + L_\phi\phi = T_R \tag{7.78}$$

where

$$T_R = -Mh_{\mathrm{CG}}(\dot{v} + ru_o)$$

If the curve is a segment of a constant radius circle, then during the constant turn maneuver the moment T can be given as

$$\begin{aligned} T_R(t) &= 0 \quad t < 0 \\ &= T_0 \quad t \geq 0 \end{aligned} \tag{7.79}$$

It can be shown that for a vehicle traveling along a curve of constant radius R at a constant speed u_o, the lateral acceleration $a_y = u_o^2/R$ and T_0 is given by

$$T_0 = -Mh_{\mathrm{CG}}u_o^2/R$$

The operational transfer function for the roll dynamics $H_\phi(s)$ is given by

$$\begin{aligned} H_\phi(s) &= \frac{\phi_R(s)}{T_R(s)} \\ &= \frac{1}{I_{xx}\left(s^2 + \dfrac{sL_p}{I_{xx}} + \dfrac{L_\phi}{I_{xx}}\right)} \end{aligned} \tag{7.80}$$

The dynamic response $\phi_R(t)$ in roll to a step encounter with the curve at $t = 0$ has the same qualitative shape as that found for the pitch response to a step of applied brakes. The roll damping coefficient L_p that is proportional to the strut damping coefficient has the same influence on $\phi_R(t)$ as it does on $\alpha_p(t)$. The steady-state roll angle (ϕ_{RSS}) after the transient response has decayed is given by

$$\Phi_{\mathrm{RSS}} = \frac{T_0}{L_\phi} \tag{7.81}$$

That is, the suspension spring rate L_ϕ determines the roll for a given steady turn rate. For passenger cars under normal driving conditions, the sinusoidal frequency response in roll is typically of less interest than that for pitch or heave dynamics. A sinusoidal roll moment input might come, for example, from an oscillatory steering wheel input. This is not encountered in normal passenger car operation.

Car handling generally improves if the amount of roll for any given maneuver is reduced. The rolling rate for a given car and maneuver is improved if spring rate and shock absorber damping are increased.

In Appendix A, we discussed the dynamics of a spring/mass/damping system, identifying resonant frequency and unity damping D_c (i.e., $\zeta = 1$):

$$D_c = 2\sqrt{\mathrm{KM}}$$

For good ride, the damping should be as low as possible. However, from practical design considerations, the minimum damping is generally in the region of $0.1 < D/D_c < 0.2$. For optimum handling, the damping is in the region of $0.6 < D/D_c < 0.8$.

Technology has been developed permitting the damping characteristics of shock absorber/strut assembly to be varied electrically, which in turn permits the ride/handling characteristics to be varied, while the car is in motion. For an understanding of the operation of electronic suspension control, it is helpful to review the operation of a strut (shock absorber) with reference to Fig. 7.23. Physically, this strut consists of a closed cylinder with a movable piston. Opposite ends of this strut are attached to the vehicle body (sprung mass) and the wheel axle assembly (unsprung mass). The strut is filled with oil that can pass through relatively small apertures in the piston, thereby allowing relative motion between the attachment points. The strut provides viscous damping force whenever the piston is moving in the cylinder that is an increasing function of the relative piston/cylinder velocity, the size of the apertures, and the fluid viscosity. Although the force-velocity relationship is nonlinear, in the following analysis, this relationship was modeled as approximately linear.

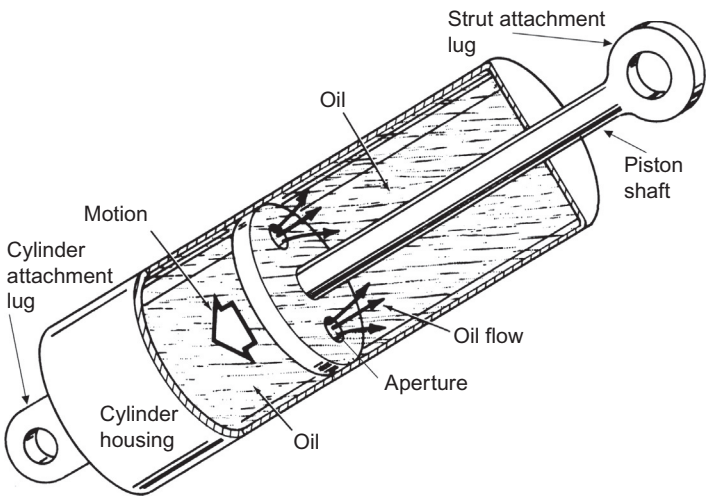


FIG. 7.23 Strut physical configuration.

Under normal steady cruise conditions, damping is electrically set low (e.g., with relatively large aperture) yielding a good ride. However, under dynamic maneuvering conditions (e.g., cornering), the damping is set high (relatively small aperture) to yield good handling. Generally speaking, as shown in the above, simplified example, high damping reduces vehicle roll in response to cornering or turning maneuvers, and it tends to maintain tire force on the road for increased cornering forces. Variable damping suspension systems can improve safety, particularly for vehicles with a relatively high CG (e.g., SUVs). Before proceeding with a discussion of electronically controlled strut damping, it is necessary to include tire dynamics in our vehicle dynamic model.

The tire dynamics in the vehicle dynamic models can be introduced adequately for the purposes of reviewing electronically controlled suspension by considering a single-strut configuration. This configuration and the model being developed apply to all four suspension assemblies. The model is often called "the quarter car model" (QCM). It is in effect a unicycle model. The configuration to be considered for this QCM is depicted in Fig. 7.24.

In this figure, the following notation is used:

y_0 = road height above a horizontal inertial reference (e.g., ECEF)
y_1 = height of unsprung mass above datum
y_2 = height of sprung mass above datum
M_s = sprung mass
M_u = unsprung mass
K_s = strut spring rate
D_s = strut damping coefficient
K_t = tire spring rate
D_t = tire damping coefficient

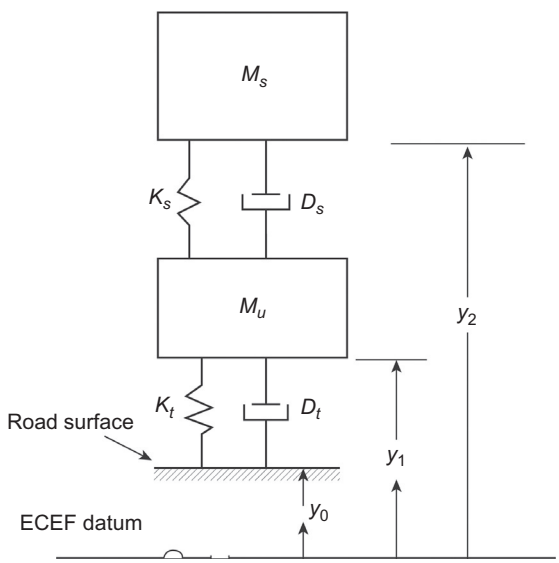


FIG. 7.24 QCM car suspension configuration.

Typically, tire damping is very small in comparison with strut damping so it is assumed to be negligible here.

A pair of differential equations can be written separately by summing forces acting on the sprung mass and on the unsprung mass. For the unsprung and sprung mass, respectively, the dynamic models are given by Eq. (7.82)

$$M_u\ddot{y}_1 + D_s(\dot{y}_1 - \dot{y}_2) + K_t(y_1 - y_0) + K_s(y_1 - y_2) = 0 \tag{7.82}$$

$$M_s\ddot{y}_2 + D_s(\dot{y}_2 - \dot{y}_1) + K_s(y_2 - y_1) = 0 \tag{7.83}$$

Solution of Eq. (7.84) can be found in two ways. The first way we consider leads to a closed-form analytic solution. Alternatively, the solution method that is best suited for numerical evaluation is to write the above equations in terms of a set of four state variable equations with state vector x given by

$$x = [v_1, v_2, y_1, y_2]^T$$

where

$$v_1 = \dot{y}_1$$
$$v_2 = \dot{y}_2$$

Taking the Laplace transform of Eqs. (7.82), (7.83) (with zero initial conditions) yields a pair of coupled algebraic equations in complex frequency s:

$$\begin{aligned}\left[M_u s^2 + D_s s + (K_t + K_s)\right] y_1(s) - (D_s s + K_s) y_2 &= K_t y_0(s) \\ \left(M_s s^2 + D_s s + K_s\right) y_2(s) - (D_s s + K_s) y_1(s) &= 0\end{aligned} \tag{7.84}$$

In matrix form, this pair of equations can be written in the form

$$A\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = K_t \begin{bmatrix} y_0 \\ 0 \end{bmatrix} \tag{7.85}$$

where

$$A = \begin{bmatrix} M_u s^2 + D_s s + (K_s + K_t) & -(D_s s + K_s) \\ -(D_s s + K_s) & M_s s^2 + D_s s + K_s \end{bmatrix} \tag{7.86}$$

The two-dimensional state vector $[y_1, y_2]^T$ is found using matrix methods yielding

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = K_t A^{-1} \begin{bmatrix} y_0 \\ 0 \end{bmatrix} \tag{7.87}$$

The 2×2 matrix A is readily inverted using standard methods from matrix algebra yielding an analytic solution for y_1 or y_2. The time response for an arbitrary $y_o(t)$ can be found using the inverse Laplace methods of Appendix A. However, for evaluating ride and handling, the frequency response characteristics are the most meaningful quantitative representation.

Our primary interest here is in finding the sprung mass motion since this directly affects "ride" quality. Ride is best characterized by the sprung mass acceleration (a_s) for any given road profile $y_o(x)$ where

$$a_s = \ddot{y}_2 \tag{7.88}$$

The operational transfer function relating $a_s(s)$ to $y_o(s)$ can be shown to be given by

$$\begin{aligned} H_a(s) &= \frac{a_s(s)}{y_o(s)} \\ &= \frac{s^2 y_2(s)}{y_o(s)} \\ &= \frac{2\zeta\omega_1^2\omega_2 s^3 + \omega_1^2\omega_2^2 s^2}{s^4 + 2\mu\zeta\omega_2 s^3 + (\omega_1^2 + \mu\omega_2^2)s^2 + 2\zeta\omega_1^2\omega_2 s + \omega_1^2\omega_2^2} \end{aligned} \tag{7.89}$$

where

$$\omega_1^2 = \frac{K_t}{M_u}$$

$$\omega_2^2 = \frac{K_s}{M_s}$$

$$\mu = \frac{M_s + M_u}{M_u}$$

$$\zeta = \frac{D_s}{2\sqrt{M_s K_s}}$$

Handling is strongly influenced by the variation in tire normal force δN. This normal force is proportional to the relative displacement $d = y_0 - y_1$:

$$\delta N = K_t d$$

The transfer function $H_H(s)$ is defined as

$$\begin{aligned} H_H(s) &= \frac{d(s)}{y_0(s)} \\ &= 1 - \frac{y_1(s)}{y_0(s)} \end{aligned} \tag{7.90}$$

The solution for $y_1(s)$ from the matrix equation yields the following:

$$H_H(s) = \frac{s^4 + 2\mu\zeta\omega_2 s^3 + \mu\omega_2^2 s^2}{s^4 + 2\mu\zeta\omega_2 s^3 + (\omega_1^2 + \mu\omega_2^2)s^2 + 2\zeta\omega_1^2\omega_2 s + \omega_1^2\omega_2^2} \tag{7.91}$$

As an illustration of the variation in relative displacement d versus road displacement y_o versus frequency, a plot of the sinusoidal frequency response for $H_H(j\omega)$ is given in Fig. 7.25.

For this figure, a representative QCM was used with the following parameters in English units:

$K_t = 1700$ lb/ft
$K_s = 8000$ lb/ft
$M_u = 2.34$ slugs
$M_s = 100$ slugs
$\zeta = 0.8$

Fig. 7.25 presents the magnitude of this frequency response (in dB) as 20 log $|H_H(j\omega)|$ and the phase of $H_H(j\omega)$ versus $\log(\omega)$. It can be seen that this QCM has a relatively sharp resonance at $\omega \cong 7$ rad/s or about 1.1 Hz. This resonance is primarily due to the dynamic response of the unsprung mass.

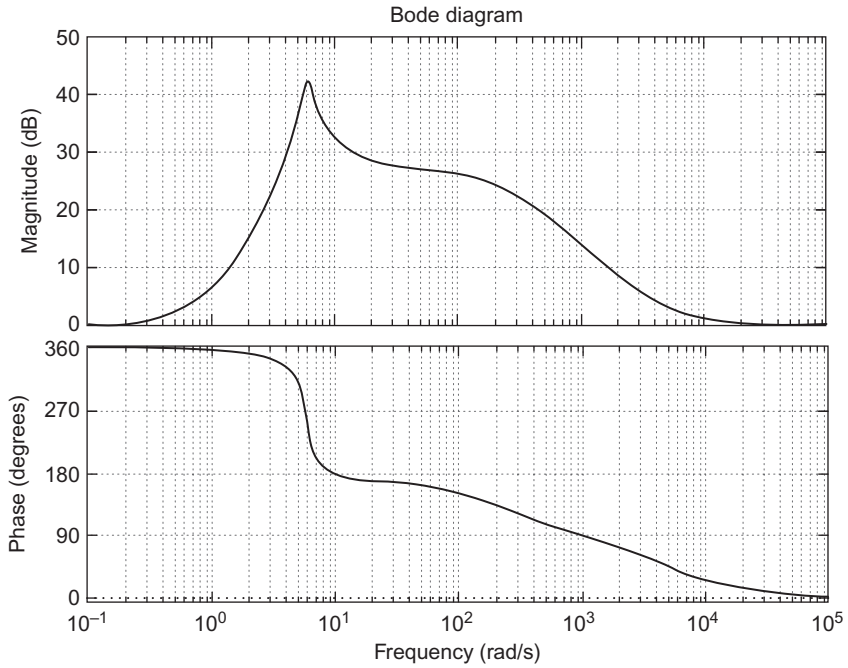


FIG. 7.25 Frequency response for $H_H(s)$.

In evaluating the influence of suspension system parameters on ride and handling, the road profile must be modeled. It is widely known that the road profile is a random process. A random process is quantitatively represented by its amplitude and spectral statistics. The amplitude statistics are given by its probability distribution function $P_y(Y) = p(y \leq Y)$. Its spectral statistics are represented by the power spectral density for y_o that is denoted $W_o(f)$ and is given by

$$W_o = |Y_o(j\omega)|^2 \tag{7.92}$$

where

$$Y_o(j\omega) = \underset{T\to\infty}{Lim} \int_{-T}^{T} y_o(t) e^{j\omega t} dt \tag{7.93}$$

However, road profiles are functions of distance along the road surface (i.e., $y_o(x)$). This road profile random process can be converted to a time function by considering motion at a constant speed u_o where

$$u_0 = \frac{dx}{dt}$$

The time function $y_o(t)$ is given by

$$y_o(t) = y_o\left(\frac{x}{u_o}\right)$$

The International Standards Organization (ISO) has established a standard model for $W_o(f)$, which is roughly an inverse function of frequency f.

Ride is best characterized quantitatively by the RMS (root-mean-squared) value of the sprung mass acceleration $\tilde{a}_s$:

$$\tilde{a}_s = \left[\underset{T\to\infty}{Lim} \left(\frac{1}{T} \int_0^T a_s^2(t)dt \right) \right]^{\frac{1}{2}}$$

This RMS value can also be found from the power spectral density ($W_a(f)$) for a_s. Assuming that the road profile is a stationary random process (or a quasi-stationary process, i.e., stationary over large segments), the RMS value for a_s is given by

$$\tilde{a}_s^2 = \int_0^\infty W_a(f)df$$

For a stationary random process, the power spectral density $W_a(f)$ is given by

$$W_a(f) = |H_a(j2\pi f)|^2 W_o(f) \tag{7.94}$$

Thus, the ride quality can be represented by the integral

$$\tilde{a}_s = \left[\int_0^\infty |H_a(j2\pi f)|^2 W_o(f)df \right]^{\frac{1}{2}} \tag{7.95}$$

The above equation for $\tilde{a}_s$ illustrates the significance of the sinusoidal frequency response of the sprung mass to road excitation. The important suspension parameters in $H_a(j\omega)$ are the sprung and unsprung mass and their ratio (M_s/M_c), the strut and tire spring rates, and the strut damping parameters.

Similarly, handling is quantitatively represented by the RMS value of tire deflection (d). The RMS value of d (i.e., $\tilde{d}$) is given by

$$\tilde{d} = \left[\int_0^\infty |H_H(j2\pi f)| W_o(f)df \right]^{\frac{1}{2}} \tag{7.96}$$

Clearly, both ride and handling are influenced by suspension parameters and M_s and M_u.

Considerable research and development has gone into determining optimum strut damping over the years. Table 7.1 is a summary of some of the results of those studies versus running condition, control objective, optimum condition, and optimum ζ and representative compact car value.

The benefits of variable strut damping in terms of improved ride and or handling have been demonstrated. We consider next actuator schemes for varying this damping. The damping of a suspension system is determined by the viscosity of the fluid in the shock absorber/strut and by the size of the aperture through which the fluid flows (see Fig. 7.23) as the wheel moves relative to the car body. For normal strut damping, the viscosity of the fluid in the strut is determined by the choice of fluid and its temperature. The damping force for a given viscosity varies as an inverse function of the aperture area. Thus, variable damping can, in principle, be varied either by varying the strut aperture mechanically or by somehow varying the strut fluid viscosity. We consider the mechanical approach first.

Although there are various mechanisms employed to vary the aperture, we illustrate with a hypothetical configuration (to avoid discussing proprietary information). In this configuration, a relatively thin tube that is coaxial with the piston shaft on its outside extends from the piston to the outside of the

Table 7.1 Summary of Optimum Suspension System Parameters

Running Condition	Control Objective	Optimum Condition	Optimum Damping Ratio	
			Theoretical Value ζ	For Compact Car
Ordinary driving	Ride improvement	To minimize sprung overall acceleration	$\frac{D_2}{2\sqrt{M_s K_s}} = \frac{1}{2}\sqrt{\frac{\mu(K_s/K_t)}{\mu - 1}}$	0.16
Roll	Roll reduction when turning	To suppress dynamic roll angle to a level below static roll angle	$\frac{(L_\phi/I_{xx})D}{4\sqrt{KI_{xx}}} = \frac{1}{\sqrt{2}}$	0.71
Pitch	Pitch reduction when accelerating, decelerating, and braking	To suppress dynamic pitch angle to a level below static pitch angle	$\frac{a^2 D_F + b^2 D_R}{\sqrt{2I_{yy}(a^2 K_F + b^2 K_R)}} = \frac{1}{\sqrt{2}}$	0.71
Bouncing	Reduction of bouncy feeling and ride improvement	To suppress light bouncy vibrations within a range where ride quality does not deteriorate	$\frac{D_s}{2\sqrt{M_s K_s}} = \frac{\sqrt{2}}{\mu} + \frac{1}{\mu}\sqrt{\frac{\mu(K_s/K_t)}{\mu - 1}}$	0.43
Rough-road	Road holdability improvement	To minimize the root-mean-square value of unsprung relative displacement	$\frac{D_s}{2\sqrt{M_s K_s}} = \frac{1}{2}\left[\frac{\mu^3 r_K^2 - 2\mu(\mu-1)r_K + (\mu-1)^2}{\mu^2(\mu-1)r_K}\right]^{1/2}$ where $r_K = K_s/K_t \quad D = D_F + D_R$	0.44

strut. This assembly is sealed where it protrudes from the cylinder to prevent any loss of strut fluid. This shaft connects with a plate that has apertures similar to the piston and that is part of the piston assembly. Rotation of this sleeve varies the overlap of the apertures in the piston and plate and effectively regulates the combined aperture through which the strut fluid flows in response to piston axial motion. The sleeve extends the full length of the piston shaft. At the end of the sleeve near the attachment lug and mechanically linked to it is a gear. This gear meshes with another gear that is driven by a motor (e.g., stepper motor) that functions as a strut aperture regulating actuator. The motor assembly is mounted on the structure to which the strut attaches. The strut aperture size is determined by the angular position of the plate relative to the piston. An electrical signal from the suspension control system operates the actuator that determines the strut aperture. This hypothetical electronically controlled strut provides the mechanism by which suspension damping is regulated. This mechanism can be either switched between two positions via a solenoid or varied continuously using, for example, a stepper motor such as has already been discussed. In order to be effective in electronically regulated strut damping, there must be an electronic control system that generates the actuator electrical signal.

Although there are many potential control strategies for regulating shock absorber damping, we consider first switched damping as in our example. In such a system, the shock absorber damping is switched to the higher value whenever lateral acceleration exceeds a predetermined threshold.

The vehicle analytic models from which the relationship between lateral acceleration, a_y vehicle speed u_o and steering angle δ_F is derived are presented in the section of this chapter devoted to electronic steering control. From these equations, it is possible to develop the following relationship between the variables

$$\frac{a_y}{\delta_F} = \frac{u_o^2/\ell}{(1+\eta u_o^2/g\ell} \tag{7.97}$$

where g = acceleration of gravity (9.81 m/s^2) and $\ell = a+b$ where a and b are given in Fig. 7.32.

The parameter η is called the understeer coefficient for the vehicle and is given by

$$\eta = -\frac{Mg}{\ell}\left(\frac{2aC_F - 2bC_R}{4C_FC_R}\right)$$

where M = vehicle mass, C_F = front tire cornering stiffness, and C_R = rear tire cornering stiffness. This understeer coefficient is a parameter that is further discussed in the section of Chapter 10 that is devoted to enhanced vehicle stability.

The latter two parameters are defined and explained in the section on electronic steering control along with representative numerical values for all parameters in the equation for η.

It can be shown from Eq. 7.97 above that for maneuvers involving a constant $\mathbf{a}_y$ for a certain speed range the front-wheel-steering angle δ_w (degrees) versus speed is given by

$$\delta_w = \frac{a_y(1+\eta u_o^2/(g\ell)}{u_o^2/\ell}$$

where for the exemplary vehicle parameters of the section on electronic steering control $\eta = 0.0423$. Fig. 7.26 is a plot of δ_w versus u_o for the representative vehicle for $a_y = 0.3$ g.

A separate curve of similar shape as that depicted in Fig. 7.26 corresponds to each value of a_y. The steering wheel angular deflection is denoted δ_w and is given by

$$\delta_w = g_s\delta_F$$

where g_s is the steering gear ratio with conversion from rad to degrees. In this example, the damping coefficients are switched from low damping D_ℓ for a relatively "soft" ride to a high value D_h whenever $a_y \geq 0.3$ g. With respect to the QCM, the control law for switched damping in the present example is given by

$$\begin{aligned} D_s &= D_\ell \quad a_y < 0.3g \\ &= D_h \quad a_y \geq 0.3g \end{aligned}$$

The specific numerical values for D_ℓ and D_h depend on the vehicle strut configurations and normally are different for front/rear locations.

The sensor for the present example is an accelerometer that must be mounted as close to the nominal vehicle CG as is practical. In this example, a sensor for measuring the lateral acceleration (called an accelerometer) is commercially available at relatively low cost. An analytic model and explanation for an acceleration sensor is given in Chapter 5. The accelerometer sensitive axis is along the vehicle

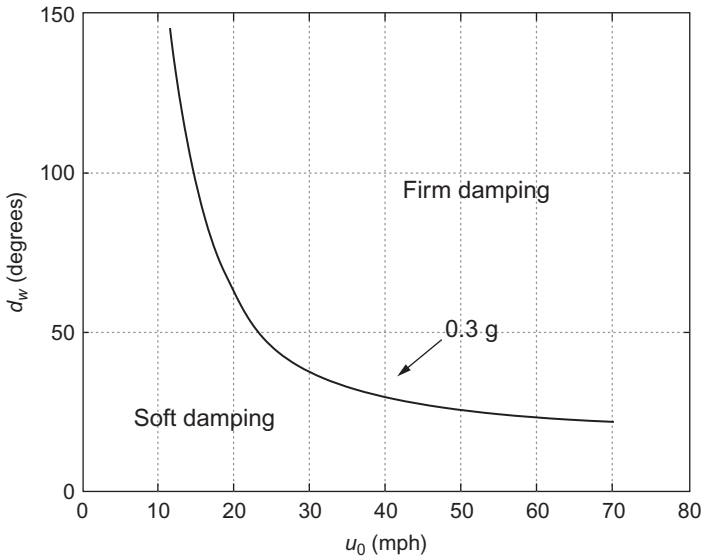


FIG. 7.26 Illustration of switching threshold for switched type variable strut damping.

lateral axis as depicted in Fig. 7.32. In this illustrative switched damping example, the output of the sensor is compared with a threshold value corresponding to 0.3 g for the present example to select which of the two strut apertures are to be selected and to generate the appropriate actuator signal. In a predigital vehicle control system, the comparison could readily be accomplished with an analog comparator as explained in Chapter 2.

However, a more practical system for variable damping involves a continuously variable aperture in the strut. For such a configuration, the variation in damping is controlled via a digital electronic control. The algorithms for selecting the desired damping coefficients are specific to vehicle configuration and handling/ride performance requirements. The damping coefficient for a strut with continuously variable aperture is represented by the force versus the relative velocity of the piston/cylinder assembly. Fig. 7.27 is an illustration of the force/relative velocity characteristics of a shock absorber having an electrically variable aperture. The figure illustrates these characteristics at the extreme limits of the variable aperture. A similar family of force-velocity profiles between these limits represents the strut characteristics for aperture sizes between these two limits.

Strut damping can also be varied continuously using the hypothetical mechanism above by means of a motor actuator. In this configuration, the force/velocity relationship will be a curve between the solid and dashed curves of Fig. 7.27. One control scheme that is potentially approachable to a continuously variable strut damping is based upon monitoring vehicle operational conditions. In this scheme, sensors are provided, which continuously monitor vehicle operating conditions. In addition to the lateral acceleration sensor, a solid-state accelerometer is available that can be placed at a convenient location on the car body to measure sprung mass acceleration ($a_s(t)$). Calculation of the RMS value $\tilde{a}_s$ yields an indication of ride. Whenever $\tilde{a}_s$ exceeds a given level (possibly driver adjusted), the control system can generate a signal to operate strut apertures to lower this acceleration. Another accelerometer

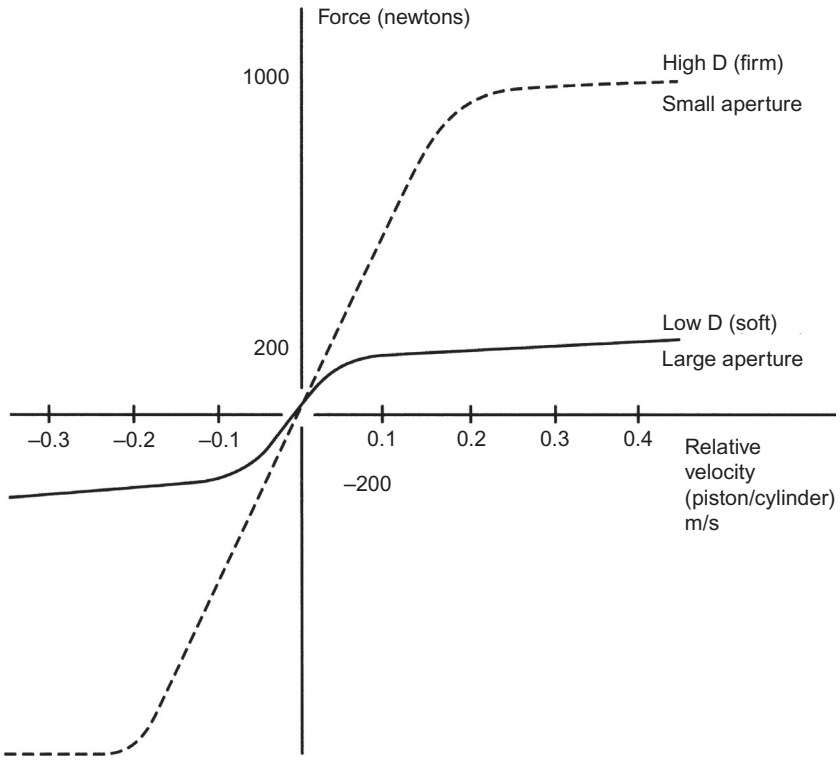


FIG. 7.27 Strut force-velocity relationship for variable aperture strut.

could be mounted on the unsprung masses (e.g., wheel axle assembly to monitor its acceleration ($a_u(t)$). Integration twice with respect to time can give a running measure of displacement:

$$d = \int_0^t \int_0^\tau a_u(t')dt'd\tau' \tag{7.98}$$

Whenever the RMS value of d indicates a potential handling problem, the strut damping could be commanded to optimize handling. Various algorithms are potentially available to set suspension damping to an optimum value with handling probably taking a higher priority over ride in the interest of safety. On the other hand as long as safety is not compromised, ride can be optimized.

VARIABLE DAMPING VIA VARIABLE STRUT FLUID VISCOSITY

Variable suspension damping is also achieved with a fixed aperture and variable fluid viscosity. The fluid for such a system consists of a synthetic hydrocarbon with suspended iron particles and is called a magnetorheological (MR) fluid. An electromagnet is positioned such that a magnetic field is created whose strength is proportional to current through the coil. This magnetic field passes through the MR

fluid. In the absence of the magnetic field, the iron particles are randomly distributed, and the MR fluid has relatively low viscosity corresponding to low damping. As the magnetic field is increased from zero, the iron particles begin to align with the field, and the viscosity increases in proportion to the strength of the field (which is proportional to the current through the electromagnet coil). That is, the damping of the associated shock absorber/strut, which incorporates MR fluid, varies continuously with the electromagnet coil current. Since damping is dependent on both viscosity and the strut aperture, this variable viscosity can be used to optimize damping either alone or in combination with variable aperture. However, in practice, the magnetic fields involved in varying the strut damping over a useful range tend to be large. The entire strut structure must be configured to permit such fields to be generated with practically achievable current levels.

VARIABLE SPRING RATE

It was shown above that the frequency response characteristics of a suspension system are influenced by the springs and the shock absorber damping. Conventional steel springs (i.e., coil or leaf) have a fixed spring rate (i.e., force deflection characteristics). For any given set of suspension springs, the vehicle height above the ground is determined by vehicle weight, which in turn depends on loading (i.e., passengers, cargo, and fuel). Some vehicles, having electronically controlled suspension, are also equipped with pneumatic springs as a replacement for steel springs. A pneumatic spring consists of a rubber bladder mounted in an assembly and filled with a gas under pressure. This mechanism is commonly called an air suspension system.

Unlike metallic springs, however, pneumatic springs have nonlinear force deflection relationship. A pneumatic spring consists of a cylinder/piston assembly with a gas (e.g., nitrogen or air between the end of the piston and the sealed cylinder). A gas under pressure p varies with the volume V of the chamber that contains it in accordance with the adiabatic gas law:

$$pV^{\gamma} = K \tag{7.99}$$

where K is the constant and γ the ratio of specific heat at constant pressure to that at constant volume ($\gamma = 1.4$ for air).

The pneumatic spring volume V is given by

$$V = A_p(\ell - x) \tag{7.100}$$

where A_p is the piston cross-sectional area, ℓ is the distance of the piston top surface to the cylinder end at its maximum extension, and x is the displacement due to external force $(0 \leq x < \ell)$.

The force versus displacement function is given by

$$\begin{aligned} F &= pA_p \\ &= \frac{KA_p}{\left[A_p(\ell - x)\right]^{\gamma}} \end{aligned} \tag{7.101}$$

As the piston moves toward the end of the cylinder (i.e., increasing x) due to increased normal force on the wheel assembly, the strut force increases nonlinearly with x. This type of gas spring has long been employed in aircraft landing gear structures where the nonlinear force/displacement is beneficial for absorbing vertical loads imparted during landings.

The force versus displacement rate for such pneumatic springs is proportional to the pressure in the bladder. In automotive suspension springs, a motor-driven pump is normally provided that varies the

pressure in the bladder, yielding a variable spring rate suspension. In conjunction with a suitable control system, the pneumatic springs can automatically adjust the vehicle height to accommodate various vehicle loadings and to increase spring "stiffness."

ELECTRONIC SUSPENSION CONTROL SYSTEM

The control system for an exemplar electronic suspension system is depicted in the block diagram of Fig. 7.28. The control system configuration in Fig. 7.28 is generic and not necessarily representative of the system for any production car. This system includes sensors for measuring vehicle speed, steering input (i.e., angular deflection of steered wheels), relative displacement of the wheel assembly and car body/chassis, lateral acceleration, and yaw rate. The outputs are electrical signals to the shock absorber/strut actuators and to the motor/compressor that pressurizes the pneumatic springs (if applicable). The actuators can be solenoid-operated (switched) orifices or motor-driven variable orifices or electromagnets for RH fluid-type variable viscosity struts. Certain vehicles may also be equipped with automatic electrically operated brakes (such as explained in the discussion of ACC and in Chapter 10) for stability-enhancement purposes.

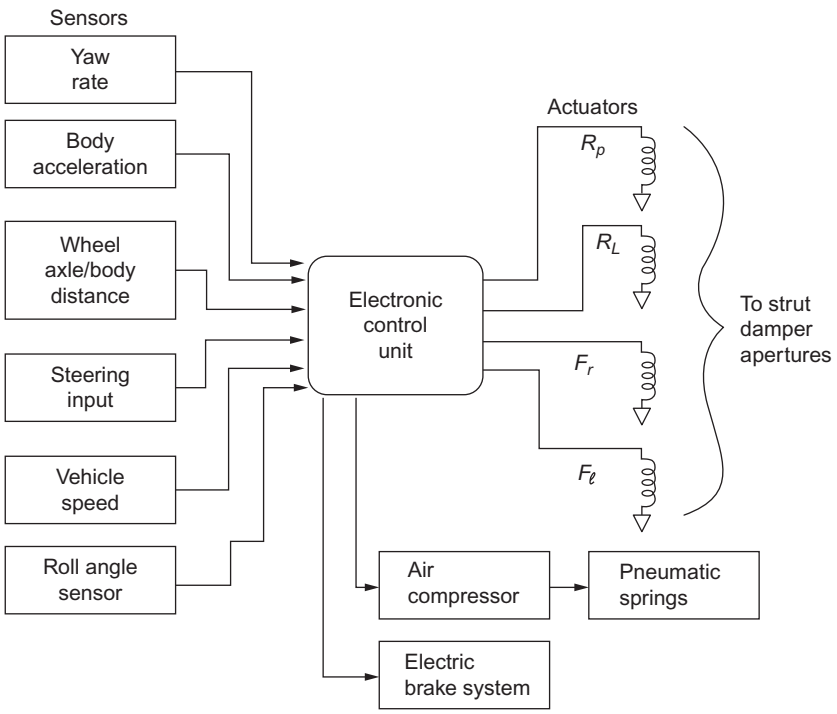


FIG. 7.28 Example electronic suspension system configuration.

The control system typically is in the form of a microcontroller or microprocessor-based digital controller. The inputs from each sensor are sampled, converted to digital format, and stored in memory. As explained above, the body acceleration measurement can be used to evaluate ride quality. The controller makes this evaluation based upon $\tilde{a}_s$ or similar metrics for body motion. The relative road/wheel axle displacement d can be used to estimate tire normal force, and damping is then adjusted to try to optimize this normal force.

Body roll angle (ϕ_R) or the yaw rate sensor (r) provides data that in relationship to vehicle speed and steering input measurements can be used to evaluate cornering performance. In certain vehicles, these measurements combine in an algorithm that is used to activate the electrohydraulic brakes for enhanced stability during extreme maneuvers. The details of automotive stability-enhancement are explained in Chapter 10 along with analytic models.

Under program control in accordance with the control strategy, the electronic control system generates output electrical signals to the various actuators. The variable damping actuators vary either the oil passage orifice or the RH fluid viscosity independently at each wheel to obtain the desired damping for that wheel.

There are many possible control strategies, and many of these are actually used in production vehicles. For the purposes of this book, it is perhaps most beneficial to present a representative control strategy that typifies features of a number of actual production systems.

The important inputs to the vehicle suspension control system come from road roughness-induced forces and inertial forces (due, for example, to cornering or maneuvering), steering inputs, and vehicle speed. In our hypothetical simplified control strategy, these inputs are considered separately. When driving along a nominally straight road with small steering inputs, the road input is dominant. In this case, the control is based on the spectral content (frequency region) of the relative motion. The controller (under program control) calculates such variables as $\tilde{a}_s$ or $\tilde{d}$ (from the corresponding sensor's data). Whenever the amplitude of the spectrum near the peak frequencies exceeds a threshold, damping is increased, yielding a firmer ride and improved handling. Otherwise, damping is kept low (soft suspension).

If, in addition, the vehicle is equipped with an accelerometer (usually located in the car body near the CG) and with motor-driven variable aperture shock absorbers, then an additional control strategy is possible. In this latter control strategy, the shock absorber apertures are adjusted to minimize sprung mass acceleration in the 2–8 Hz frequency region, thereby providing optimum ride control. However, at all times, the damping is adjusted to control unsprung mass motion to maintain wheel normal force variation at acceptably low levels for safety reasons. Whenever a relatively large steering input is sensed (sometimes in conjunction with body roll angle and/or yaw rate measurement), such as during a cornering maneuver, then the control strategy switches to the smaller aperture, yielding a "stiffer" suspension and improved handling. In particular, the combination of cornering on a relatively rough-road calls for damping that optimizes tire normal force, thereby maximizing cornering forces.

ELECTRONIC STEERING CONTROL

The steering system of a car consists of a mechanism for rotating the front wheels of the car about an axis that is nearly vertical in response to steering wheel angle changes. The basic mechanism is shown schematically in Fig. 7.29.

The force/torque of the steering wheel is influenced by the actual orientation of the pivot axes relative to the car body vertical axis. The fore/aft angle is known as camber angle. An increase in this angle relative to vertical increases steering torque; it also increases restoring torque (also called aligning torque),

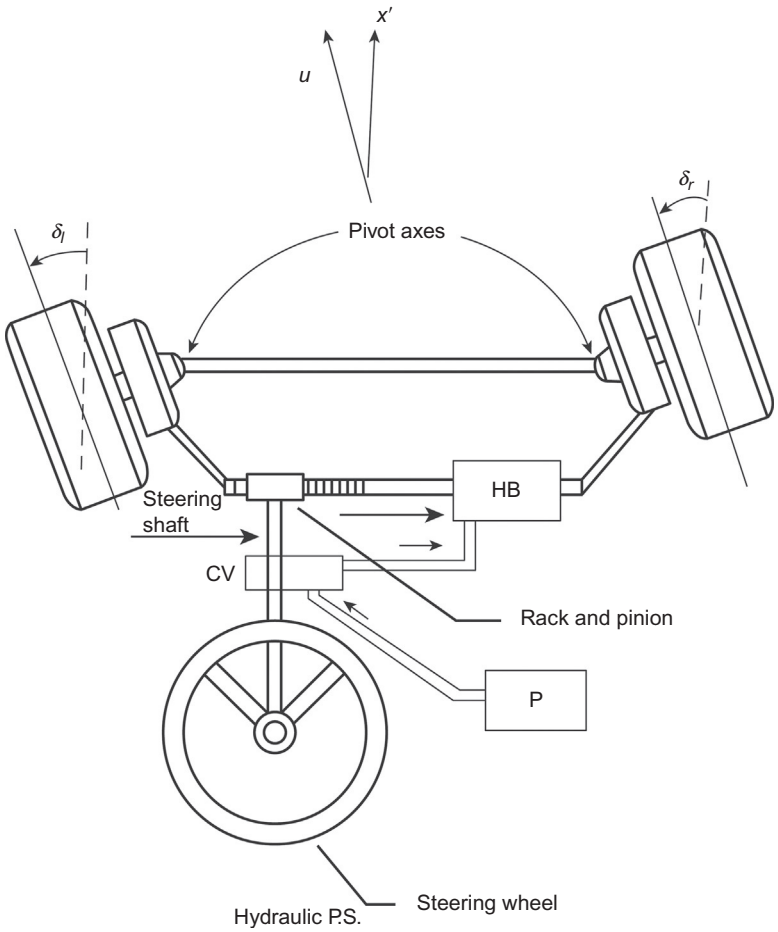


FIG. 7.29 Basic steering mechanism with power assist.

which tends to rotate the wheels toward the body symmetry plane after a turn has been completed. Typically, the camber angle is only a few degrees, but this angle in combination with the lateral orientation of the pivot axis (known as caster angle) is beneficial in steering stability. Proper alignment of these angles assists the vehicle in tracking a straight heading for neutral steering torque. In addition to the geometry of the steering mechanism and wheel angles, the aligning torque is a function of tire properties and the relative velocity of the wheels and road surface as explained later in this section.

The adverse effect of this wheel alignment torque is an increase in steering effort for the driver in proportion to alignment torque, which is undesirable. Rather than compromise on alignment to achieve lower steering effort, car manufacturers traditionally have found it desirable to provide a power assist via a hydraulic system as depicted in Fig. 7.29. An engine-driven pump *P* provides hydraulic fluid (power steering fluid) under pressure. This pressurized fluid is sent via hydraulic lines to a CV mounted

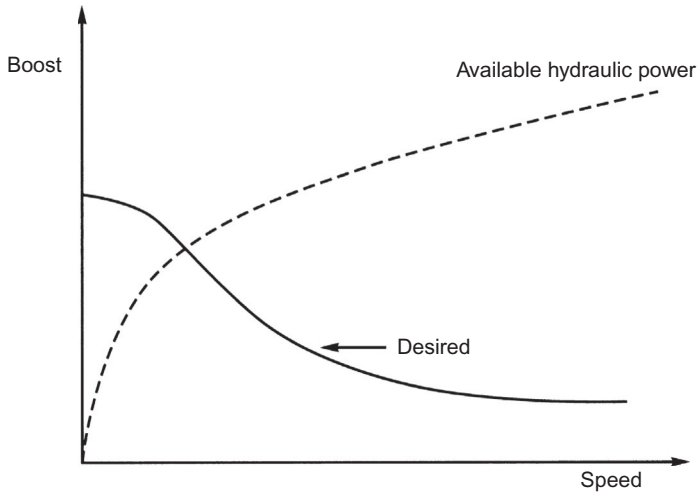


FIG. 7.30 Power steering boost versus speed.

at some point on the steering shaft. This control valve directs the pressurized fluid to a hydraulic cylinder mechanism that applies torque in the same direction as the steering wheel.

A basic problem with such a system comes from matching the desired boost with that available from the power steering pump. Fig. 7.30 shows qualitatively the desired boost that decreases with vehicle speed.

Unfortunately, in early power steering systems, the available boost was an increasing function of engine speed (owing to the increase in pump speed with engine speed) that is a function of vehicle speed and the transmission gear ratio. Although it is possible to obtain a constant boost with respect to engine RPM via pressure regulating valves yielding a constant boost with respect to vehicle speed, obtaining desired boost was not readily achievable with purely mechanical systems. On the other hand, electronic controls provide a relatively straightforward means of regulating boost to obtain desired results. Moreover, a digital power steering control system allows for the possibility of changing the boost versus speed profile via software changes. A control that adapts automatically to driving conditions is also achievable cost effectively. In an electrohydraulic power steering system, the hydraulic pressure to the boost cylinder can be varied via an adjustable pressure relief valve. An actuator for such a system can be a motor (e.g., stepper motor) or a solenoid, possibly driven by a variable-duty-cycle control signal (see Chapter 5).

An alternative power steering scheme uses a special electrical motor to provide the boost required instead of the hydraulic boost as depicted in Fig. 7.31. In this figure, a motor gear system is coupled to the steering mechanism in such a way as to provide the torque boost. A digital control system C receives vehicle speed measurements via speed sensors and generates a motor control signal to achieve the desired speed/boost profile. Electric boost power steering has several advantages over traditional hydraulic power steering. Electronic control of electric boost systems is straightforward and can be accomplished without any energy conversion from electrical power to mechanical actuation. Moreover, electronic control offers very sophisticated adaptive control in which the system can adapt to the driving environment. A basic problem with a direct electric motor steering boost is that a standard

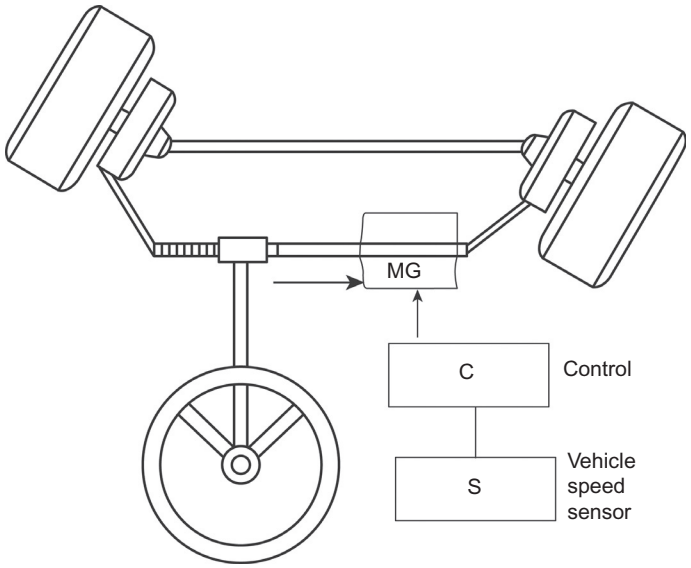


FIG. 7.31 Electric power steering boost.

electric motor must be rotating to produce meaningful torque (see Chapter 5). For any driving situation in which a constant steering angle input is required (e.g., for vehicle moving along an arc of constant radius of curvature), the motor would have to generate the torque boost while not rotating. An alternative electric boost scheme involves an electric motor directly driving a hydraulic pump that is part of an electrohydraulic power steering system.

FOUR-WHEEL STEERING CAR

Electronically controlled power steering also has the capability for four-wheel steering (4WS). As will be shown later, 4WS not only can be highly useful during vehicle curb parking but also has potential for improved road maneuverability. An example of an electronically controlled steering system that has had commercial production is for 4WS systems. In the 4WS-equipped vehicles, the front wheels are directly linked mechanically to the steering wheel, as in traditional vehicles. There is a power steering boost for the front wheels as in a standard two-wheel steering system. The rear wheels are steered under the control of a microcontroller via an actuator. Fig. 7.32 is an illustration of the 4WS configuration.

In Fig. 7.32, the notation is as follows:

x' = vehicle longitudinal axis
x = inertial (ECEF) reference axis (i.e., initial direction of x')
ψ = angle between x and x'
δ_F = angle between x' and the front tire plane of symmetry
δ_R = angle between x' and the rear tire plane of symmetry
δ_u = angle between x' and u_o

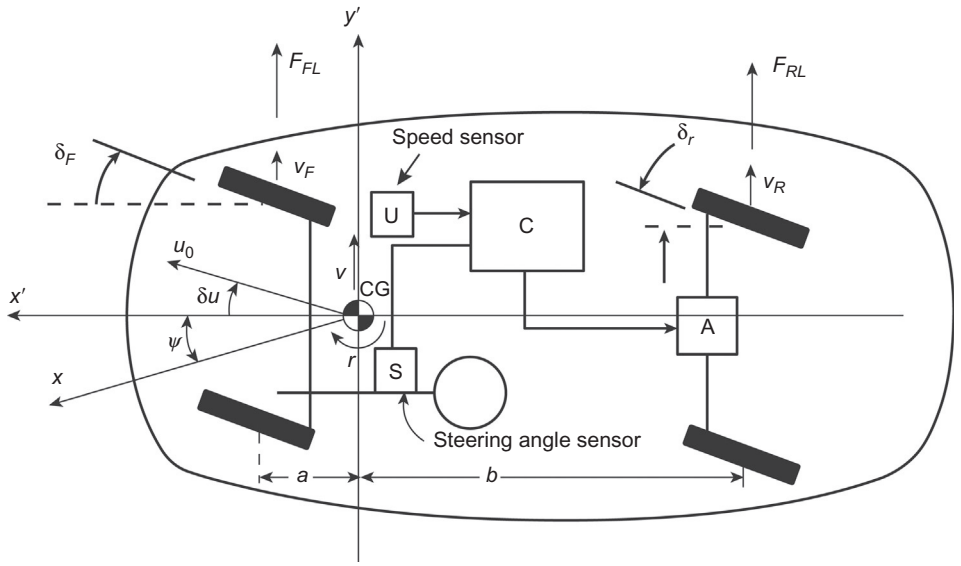


FIG. 7.32 4WS basic configuration.

u_o = car instantaneous velocity vector
a = longitudinal distance CG to front wheel axis
b = longitudinal distance CG to rear wheel axis
v = car lateral velocity
$r = \dot{\psi}$
v_F = lateral velocity of front wheel relative to road surface
v_R = lateral velocity of rear wheel relative to road surface
C = control system
A = actuator

During ordinary driving of a passenger car, the angle between the car longitudinal axis and the instantaneous velocity vector (δ_u) is small such that $\cos(\delta_u) \cong 1$, $\sin(\delta_u) \cong \delta_u$. Under these conditions, the lateral velocities of the front and rear tires, respectively, are given by

$$\begin{aligned} v_F &= v + ra \\ v_R &= v - br \end{aligned} \tag{7.102}$$

The models for the tire lateral forces at the front and rear tires F_{FL} and F_{RL} are based on the so-called tire slip angles α_F and α_R, respectively. Neglecting the small angle δu, these are the angles between the vehicle longitudinal axis x' and the instantaneous velocity vector of the tire contact point with the road and are given by

$$\begin{aligned} \alpha_F &= \delta_F - \tan^{-1}\left(\frac{v + ra}{u_o}\right) \\ \alpha_R &= \delta_R - \tan^{-1}\left(\frac{v - br}{u_o}\right) \end{aligned} \tag{7.103}$$

In these equations, right-and-left symmetry is assumed, which is valid for relatively small δ_F and δ_R such as is the case while driving on the highway. It is consistent with the small-angle assumptions that these angles are given approximately by

$$\begin{aligned}\alpha_F &\cong \delta_F - \left(\frac{v+ar}{u_o}\right)\\ \alpha_R &\cong \delta_R - \frac{v-br}{u_0}\end{aligned} \tag{7.104}$$

For a conventional front-wheel-steering car, $\delta_R = 0$. The tire lateral or so-called cornering forces (for small angles) F_{FL} (front) and F_{RL} (rear) and front wheel steering are given by

$$\begin{aligned}F_{FL} &= 2C_F\left[\delta_F - \left(\frac{v+ar}{u_o}\right)\right]\\ F_{RL} &= 2C_R\left[-\left(\frac{v-br}{u_o}\right)\right]\end{aligned} \tag{7.105}$$

where C_F is the front tire concerning stiffness, C_R is the rear tire concerning stiffness, and where right/left symmetry is assumed.

The cornering stiffness is a steering parameter, which is a function of the tire characteristics and road surface. It is also a function of the instantaneous tire normal force (i.e., $N_{F,}\, N_R$). However, for the present discussion, it is assumed to be a constant for the following steering maneuver.

The model for lateral translational motion is found by summing forces acting in the y'-direction:

$$M(\dot{v} + u_o r) = 2C_F\left[\delta_F - \left(\frac{v+ar}{u_o}\right)\right] - 2C_R\left(\frac{v-br}{u_o}\right) \tag{7.106}$$

where M is the vehicle mass.

Similarly, the model for the rotational motion about the vertical axis through the CG is found by summing all moments about the CG and is given by

$$I_{zz}\dot{r} = 2aC_F\left[\delta_F - \left(\frac{v+ar}{u_o}\right)\right] + 2bC_R\left(\frac{v-br}{u_o}\right) \tag{7.107}$$

where I_{zz} is the vehicle moment of inertia about the vertical axis through the CG.

The motion of the car in response to a steering input $\delta_F(t)$ is found by solving the above equations. The solution for any set of coupled linear first-order equations is facilitated using state variable approach as explained in Appendix A. In this case, the independent variables v,r are put in state vector (x) form where the state vector is given by

$$x = \begin{bmatrix} v \\ r \end{bmatrix} \tag{7.108}$$

The state variable model for the pair of equations is in the form

$$\dot{x} = Ax + Bu \tag{7.109}$$

where the state transition matrix A is given by

$$A = \begin{bmatrix} -2\dfrac{(C_F + C_R)}{Mu_o} & -2\dfrac{(aC_F - bC_R)}{Mu_o} - u_o \\ -2\dfrac{(aC_F - bC_R)}{I_{zz}u_o} & -2\dfrac{(a^2C_F + b^2C_R)}{I_{zz}u_o} \end{bmatrix}$$

and the input matrix B is given by

$$B = \begin{bmatrix} \dfrac{2C_F}{M} \\ \dfrac{2aC_F}{I_{zz}} \end{bmatrix} \tag{7.110}$$

The input u for front wheel steering is given by

$$u = \delta_F \tag{7.111}$$

Taking the Laplace transform of the state variable Eq. (7.109) and solving for $x(s)$ yields

$$x(s) = (sI - A)^{-1} Bu(s) \tag{7.112}$$

where I is an identity matrix that, for a two-dimensional vector, is given by

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Solution of the equation for $x(s)$ is straightforward once the parameters of the A and B matrix are determined for any given vehicle, set of tires, and steering command. As shown in Appendix A, the time domain state vector can be readily found using inverse Laplace (i.e., residue) methods.

The simplified steering model above for front wheel steering has ignored vehicle roll dynamics during a steering maneuver. We next develop a model for a 4WS (with electronic controls and for which δ_R can be nonzero) that includes roll dynamics. In writing a set of equations that includes roll, it is necessary to distinguish sprung mass (M_s) from unsprung mass (M_u) and total vehicle mass (M) where

$$M = M_s + M_u$$

Summing the forces acting in the y'-direction through the CG yields the following equation:

$$M(\dot{v} + u_o r) + M_s h_{\mathrm{CG}} \dot{p} = 2C_F \left[\delta_F - \left(\frac{v + ar}{u_o} \right) \right] + 2C_R \left[\delta_R - \left(\frac{v - br}{u_o} \right) \right] \tag{7.113}$$

where the possibility of nonzero δ_R is explicitly taken. Summing moments about the vertical axis through the CG yields the following equation:

$$I_{zz} \dot{r} - I_{zx} \dot{p} = 2aC_F \left[\delta_F - \left(\frac{v + ar}{u_o} \right) \right] - 2bC_R \left[\delta_R - \left(\frac{v - br}{u_o} \right) \right] \tag{7.114}$$

where the cross moment of inertia I_{zx} has not been neglected. Finally, summing moments about the longitudinal axis through the CG yields the following equation:

$$I_{xx} \dot{p} - I_{xz} \dot{r} + M_s h_{\mathrm{CG}} (\dot{v} + u_o r) = -(L_{\mathrm{PF}} + L_{\mathrm{PR}}) p - (L_{\phi F} + L_{\phi R}) \phi_R \tag{7.115}$$

where

$$p = \dot{\phi}_R$$

ϕ_R = angle between vehicle z-axis and inertial z-axis
I_{xx} = moment of inertia about the x-axis.
$I_{xz} = I_{zx}$ = product of inertia in x and z
$L_{\mathrm{PF}}, L_{\mathrm{PR}}$ = front, rear roll damping coefficient
$L_{\phi F} + L_{\phi R}$ = front, rear roll spring rate coefficient

The above set of coupled linear differential equations can be written in state variable form with a four-dimensional state vector

$$x = \begin{bmatrix} v \\ r \\ p \\ \phi_R \end{bmatrix} \tag{7.116}$$

and input vector

$$u = \begin{bmatrix} \delta_F \\ \delta_R \end{bmatrix}$$

This equation is four-dimensional requiring four coupled differential equations. In addition to the three given above, the fourth differential equation is given by

$$p = \dot{\phi}_R$$

The state vector equation is given by

$$G\dot{x} = Hx + Eu \tag{7.117}$$

The matrix G is given by

$$\begin{bmatrix} M & 0 & M_s h_{\mathrm{CG}} & 0 \\ 0 & I_{zz} & -I_{zx} & 0 \\ M_s h_{\mathrm{CG}} & -I_{xz} & I_{xx} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{7.118}$$

The matrix H is given by

$$\begin{bmatrix} -2(C_F + C_R)/u_o & -2(C_F a - C_R b)/u_o - M u_o & 0 & 0 \\ -2(C_F a - C_R b)/u_o & -2\left(a^2 C_F + b^2 C_R\right)/u_o & 0 & 0 \\ 0 & -M_s h_{\mathrm{CG}} u_o & -(L_{\mathrm{PF}} + L_{\mathrm{PR}}) & -\left(L_{\phi F} + L_{\phi R}\right) \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{7.119}$$

and matrix E is given by

$$E = \begin{bmatrix} 2C_F & 2C_R \\ 2aC_F & -2bC_R \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

The above state variable equation can be put in standard form by multiplying Eq. (7.117) by the inverse of G yielding

$$\begin{aligned} \dot{x} &= G^{-1}Hx + G^{-1}Eu \\ &= Ax + Bu \end{aligned} \tag{7.120}$$

where

$$\begin{aligned} A &= G^{-1}H \\ B &= G^{-1}E \end{aligned} \tag{7.121}$$

In the illustration of Fig. 7.32, the front wheels are steered to a steering angle δ_F by the driver's steering wheel input. A sensor (S) measures the steering angle and another sensor (U) gives the vehicle speed. The microcontroller (C) determines the desired rear steering angle δ_R under program control as a function of speed and front steering angle via actuator A.

In an exemplary 4WS control strategy for speeds below 10 mph, the rear steering angle is in the opposite direction to the front steering angle. This control strategy has the effect of decreasing the car's turning radius by as much as 30% from the value it has for front wheel steering only. Consequently, the maneuvering ability of the car at low speeds is enhanced (e.g., for parking).

At intermediate speeds (e.g., 11 mph $< U <$ 30 mph), the steering might be front wheel only. At higher speeds (including highway cruise), the front and rear wheels are steered in the same direction. At least one automaker has an interesting strategy for higher speeds (e.g., at highway cruise speed). In this strategy, the rear wheels turn in the opposite direction to the front wheels for a very short period (on the order of 1 s) and then turn in the same direction as the front wheels. This strategy has a beneficial effect on maneuvers such as lane changes on the highway.

As an illustration of the influence of electronic 4WS on vehicle maneuvers, a simulation has been run on a hypothetical vehicle having the following metric system parameters:

$$
\begin{aligned}
&M = 1350\,\text{kg}\\
&M_s = 1010\,\text{kg}\\
&I_{xx} = 300\,\text{kg}\,\text{m}^2\\
&I_{zz} = 1200\,\text{kg}\,\text{m}^2\\
&I_{xz} = I_{zx} = -11.25\,\text{kg}\,\text{m}^2\\
&a = 1.38\,\text{m}\\
&b = 1.64\,\text{m}\\
&h_{\text{CG}} = 0.6\,\text{m}\\
&L_{\text{PF}} = L_{\text{PR}} = 1045\,\text{N}\,\text{m}\,\text{s/rad}\\
&L_{\phi F} = L_{\phi R} = 15450\,\text{N}\,\text{m}\,\text{s}^2\\
&C_F = 2 \times 10^4\,\text{N/rad}\\
&C_R = 2.2 \times 10^4\,\text{N/rad}\\
&u_o = 30\,\text{m/s}
\end{aligned}
\tag{7.122}
$$

Fig. 7.33 qualitatively depicts the car position during alone change maneuver for 2WS and for 4WS.

From the simulation, Fig. 7.34 plots the steering wheel angle in radians for this lane change-type maneuver and the vehicle lateral motion $y(t)$ of the CG. The control strategy in this simulation is for the rear wheel deflection $\delta_R = -0.1\delta_F$. The solid curve represents 4WS response and the dashed curve the response for 2WS. Note that the lane change amount is about 60% more for 4WS than for 2WS during the steering input time interval. Alternatively, a given lateral displacement can be achieved in about 64% the time with 4WS compared with 2WS.

This simulation of a lane change maneuver with rear wheels steered in the opposite direction is only intended to illustrate the significant differences in maneuvering for 4WS compared with 2WS. In fact, such a control strategy is not necessarily desirable for passenger car electronically controlled 4WS. Rather, it might be more appropriate for certain race car applications.

For normal passenger cars, it is more likely that at highway cruise speeds the rear wheel steering would be in the same direction as the front wheels but at a somewhat smaller peak deflection. Another passenger car control strategy might be to steer the rear wheels opposite to the front wheels for a short

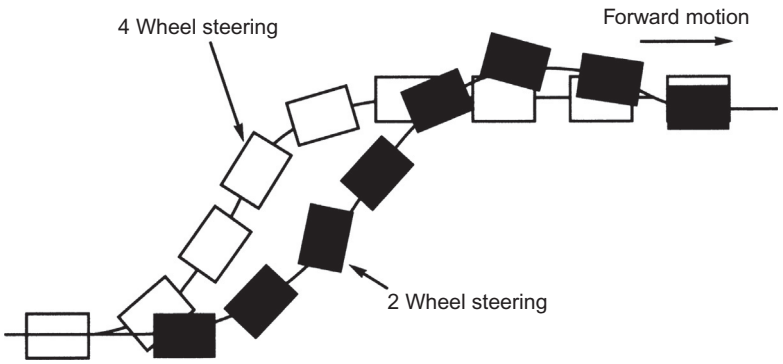


FIG. 7.33 Lane change maneuver (qualitative sketch).

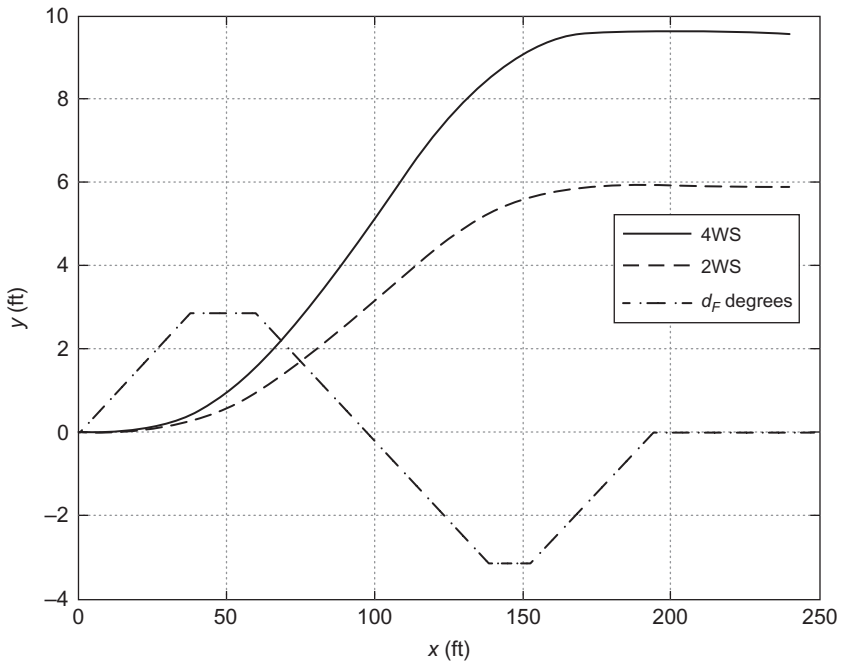


FIG. 7.34 Lane change maneuver 4WS versus 2WS from simulation.

period and then to steer them in the same direction (although at a smaller angle). Many control strategies can be evaluated in simulation using models such as are presented above or (preferably) more accurate models than above with respect to nonlinearities and unmodeled dynamics.

Turning the wheels in the same direction at cruising speeds has another benefit for a vehicle towing a trailer. When front and rear wheels turn in the same direction, the angle between the car and trailer

axes is less than it is for front wheel steering only. The reduction in this angle means that the lateral force applied to the rear wheels by the trailer in curves is less than that for front wheel only steering. This lateral force reduction improves the stability of the car or truck/trailer combination relative to front steering only.

The automatic steering applied to the rear wheels of a 4WS vehicle can also be applied to a very special 2WS vehicle. As explained in Chapter 12, automatic steering is available in vehicles with automatic parallel parking capability. This automatic steering puts driving under the control of a computer in an autonomous vehicle. The final chapter of this book is devoted to autonomous vehicles that incorporate automatic steering of the front wheels. In the extreme, such automatic steering would be required in driverless levels of autonomous vehicles. Several references to the steering portion of this chapter are made in the final chapter on autonomous vehicles.

SUMMARY

This chapter has reviewed some basic theory for vehicle motion control. In practice and in production vehicles, the models presented here would not be adequate for development of actual control systems. However, the relevant models involve very complicated nonlinear, coupled differential equations that extend beyond the intended scope of this book. On the other hand, the simplified models presented here illustrate the theory of such complex electronic systems. There is abundant literature available through the Society of Automotive Engineers (SAE) and its publication services for the reader who is interested in pursuing the advanced theory of vehicle motion control. It is hoped that the discussions in this chapter have prepared the reader well enough to be able to understand these publications.

CHAPTER

AUTOMOTIVE INSTRUMENTATION 8

CHAPTER OUTLINE

Modern Automotive Instrumentation 410
Input and Output Signal Conversion 413
Multiplexing 415
Multirate Sampling 416
Advantages of Computer-Based Instrumentation 419
Display Devices 419
Galvanometer-Type Display 420
Electro Optic Displays 423
Light-Emitting Diode 424
Liquid-Crystal Display 426
Transmissive LCD 428
Vacuum-Fluorescent Display 429
Alpha-Numeric Display 431
Flat Panel Display Instrument Clusters 434
Pictorial Display Capability of FPD 441
Digital Maps 442
Touch Screen 443
Measurement Examples 447
Fuel Quantity Measurement 447
Coolant Temperature Measurement 452
Oil Pressure Measurement 454
Vehicle Speed Measurement 456
Trip Information Function of the System 457

This chapter describes electronic instrumentation. By the term *instrumentation*, we mean the equipment and devices that measure engine and other vehicle variables and parameters for control or to display their status to the driver.

From about the late 1920s until the late 1950s, the standard automotive instrumentation included the speedometer, oil pressure gauge, coolant temperature gauge, battery charging rate gauge, and fuel quantity gauge. Strictly speaking, only the latter two are electrical instruments. In fact, this electrical instrumentation was generally regarded as a minor part of the automotive electrical system. By the late

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-12-810434-7.00008-9
Copyright © 2017 Elsevier Inc. All rights reserved.

1950s, however, in many vehicles, the gauges for oil pressure, coolant temperature, and battery charging rate were replaced by warning lights that were turned on only if specified limits were exceeded. This was done primarily to reduce vehicle cost and because of the presumption that many people did not necessarily regularly monitor these instruments.

Automotive instrumentation was not really electronic until the 1970s. At that time, the availability of relatively low-cost solid-state electronics brought about a major change in automotive instrumentation; the use of low-cost electronics has increased with each new model year. This chapter presents a general overview of typical automotive electronic instrumentation.

In addition to providing measurements for display, modern automotive instrumentation performs limited diagnosis of problems with various subsystems. Whenever a problem is detected, a warning indicator alerts the driver of a problem and indicates the appropriate subsystem. For example, whenever self-diagnosis of the engine control system detects a problem, such as an estimated error in a signal from a sensor, a lamp illuminates the "Check Engine" message on the instrument panel (IP). Such warning messages alert the driver to seek repairs from authorized technicians who have the expertise and special equipment to perform necessary maintenance. The incorporation of display devices with pictorial capability (e.g., similar to a laptop computer) with touch-screen capability is explained later in this chapter. Vehicular displays of this type are termed instrument clusters. However, the term flat-panel display (FPD) is a convenient euphemism with the abbreviation FPD for the present chapter. Before presenting this advanced FPD technology, it is desirable to review some of the basic concepts of vehicular instrumentation.

MODERN AUTOMOTIVE INSTRUMENTATION

The evolution of instrumentation in automobiles has been influenced by electronic technological advances in much the same way as the engine control system, which has already been discussed. Of particular importance has been the advent of the microprocessor, solid-state display devices, and solid-state sensors. In order to put these developments into perspective, recall the general concept of and the block diagram for modern electronic instrumentation (see Appendix A). The block diagram of measurement instrumentation is repeated here as Fig. 8.1. There, it was explained that measurement instruments consist of three functional components: sensor, signal processing, and display.

In electronic instrumentation, a sensor is required to convert any nonelectric signal to an equivalent voltage or current. Electronic signal processing is then performed on the sensor output to produce an

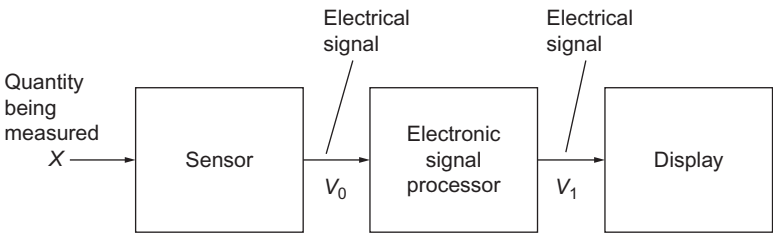


FIG. 8.1 General instrumentation block diagram.

electrical signal that is capable of driving the display device. The display device is read by the vehicle driver. If a quantity to be measured is already in electrical form (e.g., the battery charging current), this signal can be used directly, and no sensor is required.

As explained in Appendix A, the role of signal processing is to perform any required transformation of the sensor output voltage to generate a signal that is sent to the display such that the display presents the desired measurement in the correct format. In general, the sensor output voltage for the measurement of a physical variable (x) is in the following form

$$v_0 = f(x) \tag{8.1}$$

As explained in Appendix A, $f(x)$ can take many forms. The simplest and often the most desirable form is a linear transformation for which the sensor model is given by

$$v_0 = K_s x \tag{8.2}$$

where K_s = constant for the sensor.

However, other functional forms both linear and nonlinear are commonly encountered in practice. For example, a sensor can have a model that is given by

$$v_0 = K_s \frac{dx}{dt} \tag{8.3}$$

Signal processing might include the integration of the sensor voltage to obtain $x(t)$:

$$x(t) = \frac{1}{K_s} \int_0^t v_0(\tau) d\tau \tag{8.4}$$

Each measurement in any instrumentation system will have a specific sensor function requiring a particular signal processing transformation to yield the desired display. If the sensor and display are linear analog devices, then the signal processing operation can be given by the operation transfer function ($H_{sp}(s)$) where

$$H_{sp}(s) = \frac{v_1(s)}{v_0(s)} \tag{8.5}$$

where v_0 = sensor output and v_1 = input signal to the display (see Fig. 8.1).

Signal processing in contemporary vehicle instrumentation is performed in a digital system under program control. In this case, the sensor input is sampled at discrete times t_k, and the output of the signal processor is a discrete time sequence $\{y_n\}$. A representative linear signal processing operation can be written as a recursive algorithm as explained in Appendix B:

$$y_n = \sum_{k=0}^{K} a_k v_0(t_{n-k}) - \sum_{i=1}^{L} b_i y_{n-i} \tag{8.6}$$

The digital sequence $\{y_n\}$ is then converted to a signal (v_1) of the correct format to drive the display. Examples of digital signal processing (DSP) for specific measurements are presented throughout this chapter. The hardware for various types of display is discussed later in this chapter. Although modern automotive display devices are digital, it is still possible to have an analog display (e.g., galvanometer), in which case a D/A converter will provide the correct signal to drive the display.

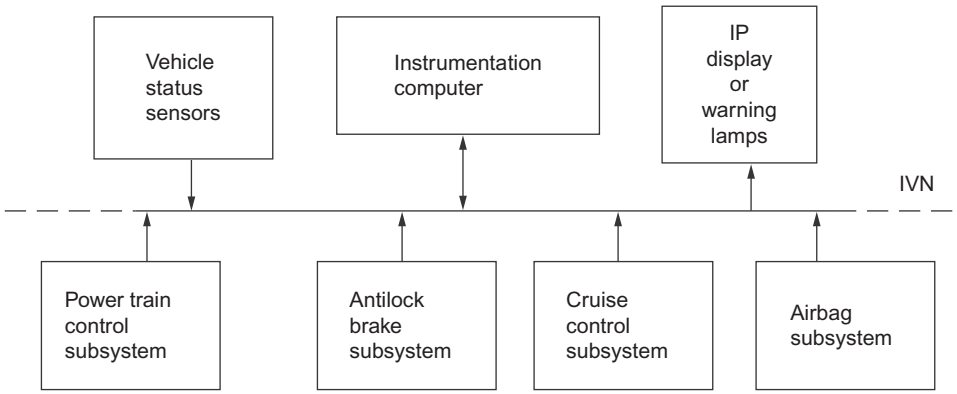


FIG. 8.2 Computer-based instrumentation system.

In contemporary automotive instrumentation, a microcomputer (or related digital subsystem) performs all signal processing operations for several measurements. The primary motivation for computer-based instrumentation is the great flexibility offered in the design of the instrumental panel (IP). A representative block diagram for such an instrumentation system is shown in Fig. 8.2.

The architecture for such a system is vehicle-model-specific, but Fig. 8.2 is exemplary. The system depicted in Fig. 8.2 incorporates a digital communication link in the form of a so-called in-vehicle network (IVN). Several example IVNs are discussed in detail in Chapter 9.

All measurements from the various sensors and switches are processed in a special-purpose digital computer, that is, the instrumentation computer. The processed signals are routed to the appropriate display or warning message. It is a common practice in modern automotive instrumentation to integrate the display or warning in a single module that may include solid-state alphanumeric display, lamps for illuminating specific messages, and traditional electromechanical indicators. For convenience, this display system will be termed the *instrument panel* (IP).

The inputs to the instrumentation computer include sensors (or switches) for measuring (or sensing) various vehicle variables and diagnostic inputs from the other critical electronic subsystems. The vehicle status sensors may include any of the following:

1. Fuel quantity
2. Fuel pump pressure
3. Fuel flow rate
4. Vehicle speed
5. Oil pressure
6. Oil quantity
7. Coolant temperature
8. Outside ambient temperature
9. Windshield washer fluid quantity
10. Brake fluid quantity
11. Wheel slip

In addition to these variables, the input may include switches for determining gear selector position, brake activation, and detecting open doors and trunk, as well as IP selection switches for multifunction displays that permit the driver to select from various display modes or measurement units. For example, the driver may be able to select vehicle speed in miles per hour (mph) or kilometers per hour (kph).

An important function of modern instrumentation systems is to receive diagnostic information from certain subsystems and to display appropriate warning messages to the driver. The power train control system, for example, continuously performs self-diagnosis operations. If a problem has been detected, a fault code is set indicating the nature and location of the fault. This code is transmitted to the instrumentation system via an IVN, for example, CAN, as explained in Chapter 9. This code is interpreted in the instrumentation computer and a "Check Engine" warning message is displayed. Similar diagnostic data are sent to the instrumentation system from each of the subsystems for which driver warning messages are deemed necessary (e.g., ABS, airbag, and cruise control). The way in which a fault is detected is explained in greater detail in Chapter 11.

INPUT AND OUTPUT SIGNAL CONVERSION

It should be emphasized that any single input can be digital, switched, or analog depending on the technology used for the sensor. A typical instrumentation computer is an integrated subsystem that is designed to accept all of these input formats. A typical system is designed with a separate input from each sensor or switch. An example of an analog input is the fuel quantity sensor, which can be a potentiometer attached to a float, as described in detail later in this chapter. The measurement of vehicle speed as discussed in Chapter 7 that uses a sensor described in Chapter 5 is an example of a measurement that is already in digital format. The theory of and model for most sensors employed in electronic instrumentation are covered in Chapter 5. This chapter discusses some of the associated signal processing.

The analog inputs must all be converted to digital format using an analog to digital (A/D) converter as explained in Chapter 3 and illustrated in Fig. 8.3. In the example of Fig. 8.3, a quantity x being measured uses an analog sensor with output voltage $v_o(x)$. The instrumentation computer causes a sample of

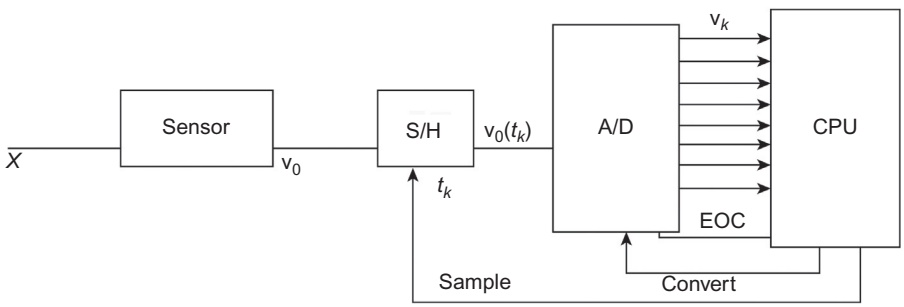


FIG. 8.3 Digital instrumentation input system.

v_o to be taken at time t_k via a sample and hold (S/H) circuit, an example of which is presented in Chapter 2 (see section "Zero-Order Hold Circuit"). The sampled voltage $v_o(t_k)$ is, then, converted to a digital input v_k by the A/D converter (see Chapter 3) and is input to the central processing unit (CPU) portion of the instrumentation computer (which performs DSP) in digital format.

The digital inputs are, of course, already in the desired format. The conversion process requires an amount of time that depends primarily on the A/D converter. After the conversion is complete, the digital output generated by the A/D converter is the closest possible approximation to the equivalent analog voltage, using an M-bit binary number (where M is chosen by the designer as determined by required precision of measurement). The A/D converter then sends a signal to the computer by changing the logic state on a separate lead (labeled EOC indicating end of conversion in Fig. 8.3) that is connected to the computer. (Recall the use of interrupts for this purpose, as discussed in Chapter 3.) The output voltage of each analog sensor for which the computer performs signal processing must be converted in this way. Once the conversion and any required DSP are complete, the digital output is transferred into a register in the computer. If the output is to drive a digital display, this output can be used directly. However, if an analog display is used, the binary number must be converted to the appropriate analog signal by using a digital-to-analog (D/A) converter (see Chapter 3).

Fig. 8.4 illustrates a typical D/A converter used to transform digital computer output to an analog signal. The N-digital output leads transfer the results of the signal processing to a D/A converter. When the transfer is complete, the computer sends a signal to the D/A converter to start converting. The D/A output generates a voltage that is proportional to the binary number in the computer output. As explained in Appendix B, the D/A conversion often includes a zero-order-hold (ZOH) circuit. A low-pass filter (LPF) (which could be as simple as a capacitor) is often connected across the D/A output to smooth the analog output between samples. The sampling of the sensor output, A/D conversion, DSP, and D/A conversion normally take place during the time slot allotted for the measurement of the variable in a sampling time sequence, (although time delays are possible) to be discussed later in this chapter.

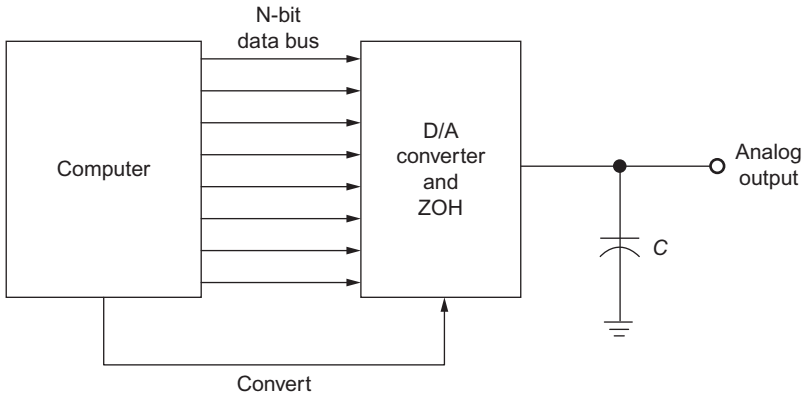


FIG. 8.4 Digital instrumentation analog output.

MULTIPLEXING

The instrumentation computer in our example system can only deal with the measurement of a single quantity at any one time. Therefore, the computer input must be connected to only one sensor at a time, and the computer output must be connected only to the corresponding display. The computer performs any necessary signal processing on a particular sensor signal and then generates an output signal to the appropriate display device.

The process of selectively and sequentially sending multiple inputs to a DSP system is known as multiplexing. We consider an instrumentation system in which a set of signals from N analog sensors is connected to the digital system. A means for accomplishing this process in time sequence is known as time-domain multiplexing (TDM). One configuration for TDM of N signals is shown schematically in Fig. 8.5.

In the configuration of Fig. 8.5, a set of N analog sensors generates output voltages $v_n(t)$. Each of these is connected to an electronic switch (S_n), which, for example, can be implemented using a transistor as described in Chapter 2 and called "an analog multiplexer/demultiplexer" located within an electronic module denoted as MUX. The MUX performs the multiplexing function and a sampling function. Not shown in this figure are the electrical connections that activate (i.e., close) the normally open switches. In the configuration of Fig. 8.5, the digital system activates each switch by sending digital data to a decoder (1 of N). When the data corresponding to switch S_n are transmitted to the decoder, it generates a signal that activates that switch effectively connecting voltage v_n to the A/D converter (see "MUX" section of Chapter 2). At the end of the conversion time, the A/D generates a signal on the EOC line, which causes the digital system to read the A/D output. The A/D converter holds v_n until EOC. Thus, the MUX in this configuration performs a sampling operation in addition to multiplexing (see Chapter 3 and Appendix B for a discussion of sampling).

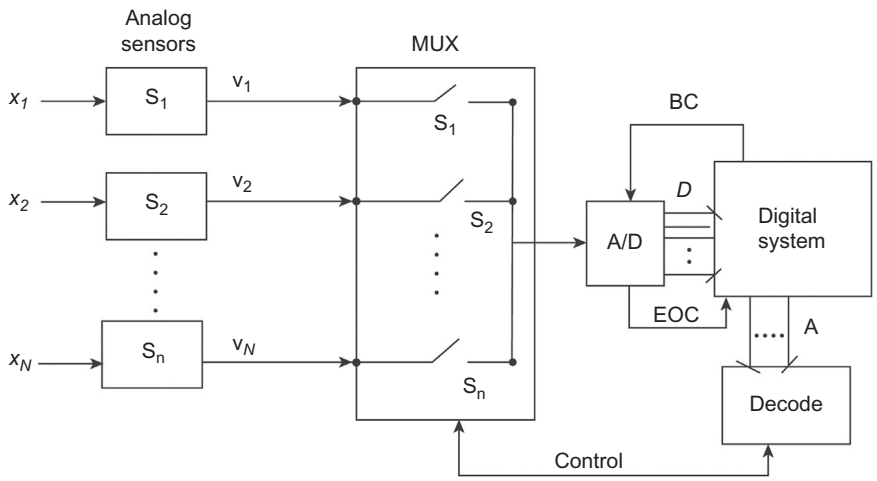


FIG. 8.5 Analog multiplexing system.

In the configuration of Fig. 8.5, it is assumed here that each sensor signal is assigned a time slot within a larger period. In this case, the sample time t_{nk} for sensor n during the kth MUX cycle is given by

$$\begin{aligned} t_{nk} &= T_k + n\delta T \qquad n = 1,2,\cdots,N \\ \delta T &= \frac{T_c}{N} \\ T_c &= T_{k+1} - T_k = \text{cycle period} \end{aligned} \tag{8.7}$$

N = number of inputs sampled during T_c.

This configuration is one of many such for performing the MUX function.

Multiplexing can also be done with digital signals. Such signals can come either from a digital sensor (e.g., a speed sensor as in Chapter 7) or from an analog sensor with its own dedicated A/D converter. Fig. 8.6 illustrates a digital MUX configuration.

Here, it is assumed for illustrative purposes that there are four inputs to the MUX (corresponding to digital data from four sensors). It is further presumed that the data are available in 8-bit digital format. In practice, however, contemporary vehicle instrumentation uses a higher number of bits for each measurement variable. Each of the multiplexers selects a single bit from each of the four inputs. There must be eight such MUX circuits, each supplying one data bit. The output lines from each MUX are connected to a corresponding data bus (DB) line in the digital computer (see Chapter 3). The digital system controls sequencing by generating data select line signals as shown in Fig. 8.6. This selection is done in a sequence corresponding to the sample time t_{nk} as explained for the analog MUX for the nth input and kth sample.

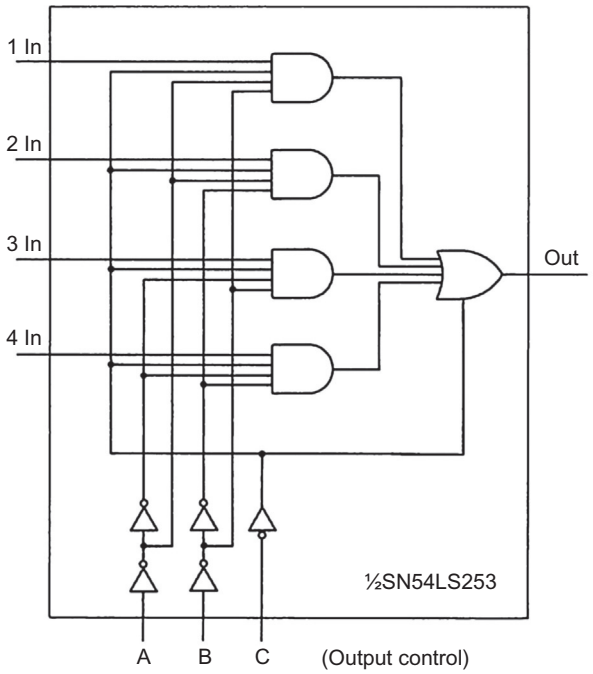
Once the required signal processing has been completed in the digital system and output signal y_n has been computed for sensor n corresponding to sample time t_{nk}, the correct signal must be sent to the display for that variable. Although the sensors for the configuration of Fig. 8.5 have been taken to be analog, it is assumed that the displays are digital. It is assumed that the digital output comes along a single set of output data lines in a time sequence similar to that shown for the input data. That is, each pair of data select bits (e.g., A and B of Fig. 8.7) has a time associated with it such that the data y_n are sent to the correct 8-bit digital display, one bit from each of the DEMUX circuits. Each display converts the 8-bit data to an alphanumeric character in the digital display as explained later.

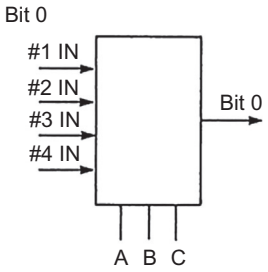
MULTIRATE SAMPLING

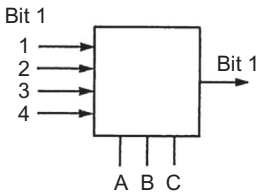
As explained above with respect to the configuration of Fig. 8.5, one possible scheme for measuring several variables by this process is to sample each quantity sequentially, giving each measurement a fixed time slot, t_{nk}, out of the total cycle period, T_c, as illustrated in Fig. 8.8.

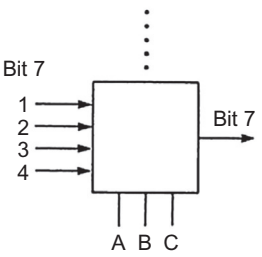
This method is satisfactory as long as the sample period is small compared with the time in which any quantity changes appreciably. Certain quantities, such as coolant temperature and fuel quantity, change very slowly with time. For such variables, a sample period of a few seconds or longer is often adequate.

On the other hand, variables such as vehicle speed, battery charge, and fuel consumption rate change relatively quickly and require a much shorter sample period, perhaps every second or every







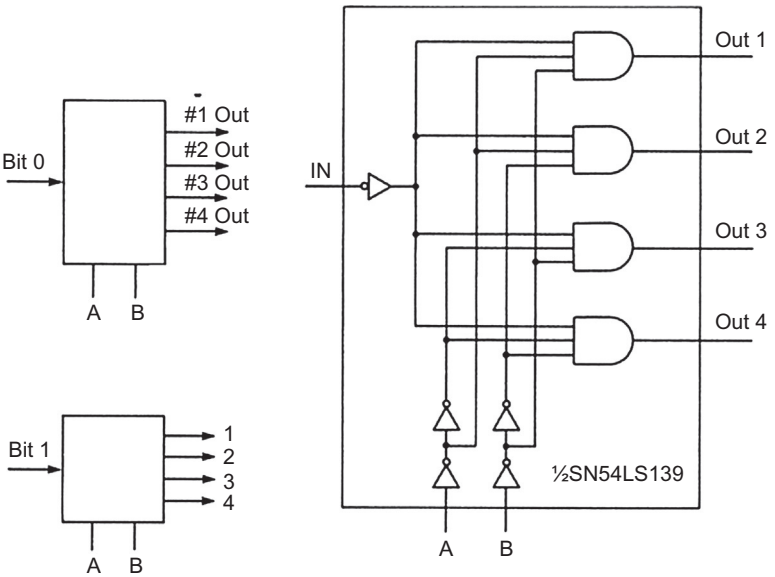


Data select bits			Data input selected
A	B	C	
X	X	1	High Z
0	0	0	1
0	1	0	2
1	0	0	3
1	1	0	4

FIG. 8.6 Digital data multiplexer.

few tenths of a second. To accommodate the various rates of change of the automotive variables being measured, the sample period varies from one quantity to another. This process of having a different sample period for different subsets of variables is known as multirate sampling. The most rapidly changing quantities are sampled with a very short sample period, whereas those that change slowly are sampled with a long sample period.

Multirate sampling can be accomplished by having different configurations such as shown in Fig. 8.5 for each subset of variables at a given sample rate. However, it is also possible to achieve



Data select bits		Data output selected
A	B	
0	0	1
0	1	2
1	0	3
1	1	4

FIG. 8.7 Digital data demultiplexer.

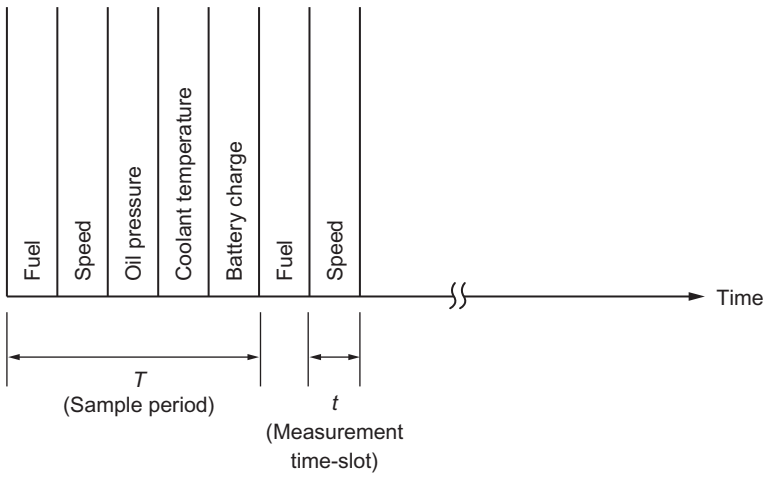


FIG. 8.8 Sequential sampling.

multirate sampling with a single MUX, which sample signal sensor at the highest rate required. Then, rather than store and process every sample for the low rate variable, the low rate variable sampling rate can be accomplished by a process called decimation. In this process, only one sample of data is stored in memory for every M cycle periods of duration T_c resulting in a decimation by M of the data. Effectively, this process reduces the sample rate by a factor of $1/M$ of the highest sample rate.

ADVANTAGES OF COMPUTER-BASED INSTRUMENTATION

One of the big advantages of computer-based instrumentation is its great flexibility. The signal processing algorithm required for each variable is typically unique for that variable and for the sensor and display. Each algorithm is implemented by a separate program that is stored in memory and called during the time period for the given measurement. This point is illustrated with a selected set of measurement examples presented at the end of this chapter. To change from the instrumentation for one vehicle or one model to another often requires only a change of computer program. This change can often be implemented by replacing one read-only memory (ROM) with another. Remember that the program is permanently stored in a ROM that is typically packaged in a single integrated circuit package (see Chapter 3).

Another advantage of computer-based instrumentation is common to all computer-based systems. This particular benefit is associated with changes that occur during the development of the system that involve essentially fixed hardware. The evolution of a digital system often involves refinements in the signal processing or analysis algorithms. Any such change is accomplished via the system software creation and modification. The details of software preparation via programming are specific to the software being used to create the programs (e.g., AUTOSAR). This topic is sufficiently broad that it is covered in other books or documentation and is beyond the scope of this book, as explained in Chapter 1. However, in this book, example algorithms are presented through analytic models and the associated analysis of the system being discussed.

Another benefit of microcomputer-based electronic automotive instrumentation is its improved performance compared with conventional instrumentation. For example, the traditional electromechanical fuel gauge system has errors that are associated with (1) nonlinearities in the mechanical and geometric characteristics of the tank relative to the sender unit, (2) the instrument voltage regulator, and (3) the display dynamic response. The electronic instrumentation system eliminates the error that results from imperfect voltage regulation. Generally speaking, the electronic fuel quantity measurement maintains calibration over essentially the entire range of fuel quantity in the tank. Moreover, it significantly improves the display accuracy by replacing the electromechanical galvanometer display with an all-electronic digital display.

DISPLAY DEVICES

One of the most important components of any measuring instrument is the display device. In automotive instrumentation, the display device must present the results of the measurement to the driver in a form that is easy to read and understand. The speedometer, ammeter, and fuel quantity gauge were originally electromechanical devices. Then, automotive manufacturers began using warning lamps

for certain variables (e.g., oil pressure) instead of gauges to cut cost. A warning lamp can be considered as a type of electro-optical display. In addition, electro-optical alphanumeric and pictorial display devices are in common use in contemporary vehicles as explained later in this chapter.

GALVANOMETER-TYPE DISPLAY

Even in certain models of contemporary vehicles, analog display devices are sometimes used (e.g., to display vehicle speed fuel quantity, coolant temperature, oil pressure, and engine RPM). The most common analog electromechanical display is the galvanometer. The basic physical configuration for a galvanometer is shown in Fig. 8.9.

This display device uses a movable pointer to indicate the numerical value of the displayed quantity along a scale. The scale consists of short segments of lines that are directed radially from the pointer pivot. Normally, numerical values are only given along the longer lines.

The pointer is attached via a long, thin rod that is supported at either end by small bearings of the type used to support shafts in mechanical watches or clocks. For the present discussion, it is assumed that the pointer system is perfectly mass balanced about the pivot. Any mass imbalance will result in an unintended and undesirable torque about the pivot axis in response to certain vehicle motions (e.g., cornering), which will result in erroneous display readings. The rod to which the pivot is attached is also attached to a coil of wire having N turns and one or more springs. A permanent magnet is the source of a magnetic field that flows through the ferromagnetic pole pieces and a fixed cylindrical core. The coil is separated from the pole pieces and cylindrical core by a small gap and can rotate about the pivot axis within this gap.

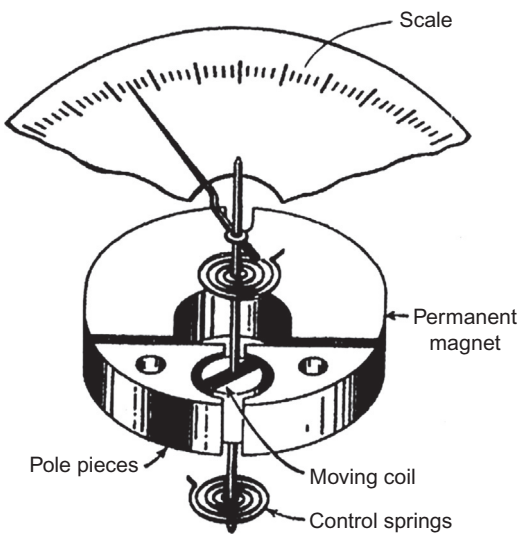


FIG. 8.9 Galvanometer configuration.

The analytic model for galvanometer dynamic response to an electric input (i.e., voltage or current) is derived, in part, from magnetic field theory as introduced in Chapter 5. The pole pieces and cylindrical core are designed such that the magnetic flux density vector $\bar{B}$ is directed radially from the pivot axis, for example, inward on the left pole piece and outward on the right. The magnitude of this magnetic flux density B_r is ideally constant over the entire region of coil movement as shown in Fig. 8.10. For the present discussion, the galvanometer input is the output voltage (v) of the signal processing or instrumentation computer. A properly calibrated instrumentation will generate an output voltage that, when applied to the galvanometer input, will cause the pointer to display the correct value of the quantity being measured.

From basic magnetic field theory (as introduced in Chapter 5), it is known that whenever a current flows through the movable coil, a torque T_c acts on the coil, which is given by

$$T_c = K_c N B_r i \tag{8.8}$$

where K_c = constant for the configuration, N = number of turns on the coil, and i = current through the coil.

The spring produces a torque on the pointer shaft in a direction opposite to that of the magnetic torque and such that it tends to move the pointer to $\theta = 0$.

The dynamic equation of motion for the galvanometer is given below:

$$\begin{aligned} J\ddot{\theta} + D\dot{\theta} + K\theta &= T_c \\ &= K_c N B_r i \end{aligned} \tag{8.9}$$

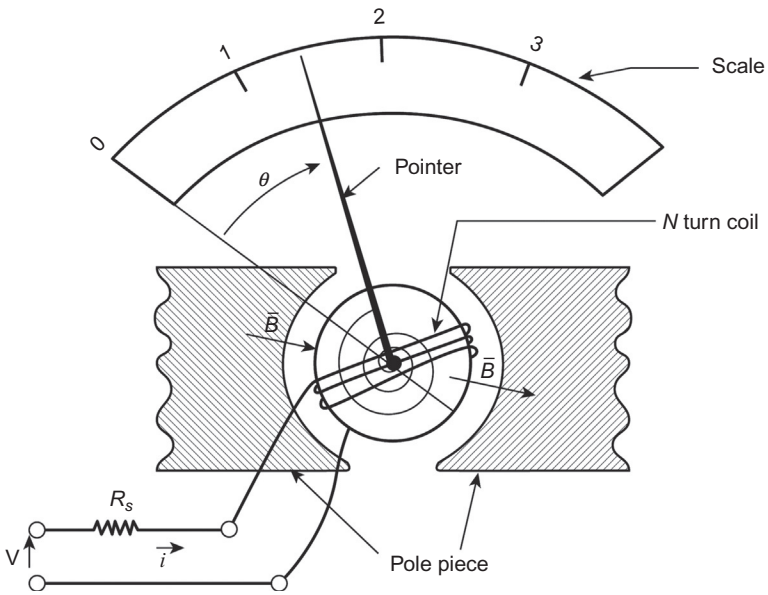


FIG. 8.10 Galvanometer magnetic field configuration.

where J = moment of inertia of the moving assembly about the pivot of axes, D = viscous damping coefficient for the movable elements, and K = spring rate of the torsional spring.

From Appendix A, it can be shown that the operational transfer function for the above model ($H_i(s)$) is given by

$$H_i(s) = \frac{\theta(s)}{i(s)} = \frac{K_c N B_r}{Js^2 + Ds + K} \tag{8.10}$$

It is also shown in Appendix A that Eq. (8.10), which is a second-order transfer function, has a standard form given by

$$H_i(s) = \frac{K_D}{s^2 + 2\zeta\omega_0 s + \omega_0^2}$$

where

$$K_D = K_c N B_r / J = \text{gain factor},\ \zeta = \frac{D}{2\omega_0 J} = \text{damping ratio, and}\ \omega_0 = \sqrt{K/J} = \text{natural frequency}$$

The dynamic response of the pointer deflection due to an input current is characterized fully by the above parameters.

The electrical input to the galvanometer of the circuit driving the display can be expressed by the terminal voltage and source impedance R_s (assumed to be purely resistive). This combination is the equivalent circuit of the electronic system that is driving the display (e.g., the D/A converter output of the digital instrumentation computer). A circuit diagram for the galvanometer is shown in Fig. 8.11.

The coil has a circuit model consisting of resistance R_c and inductance L_c. The dynamic model for the coil current is given by

$$v_s = (R_s + R_c)i + L_c\frac{di}{dt} \tag{8.11}$$

Using the Laplace transform methods of Appendix A, the operational transfer function for the galvanometer $H_g(s)$ can be shown to be

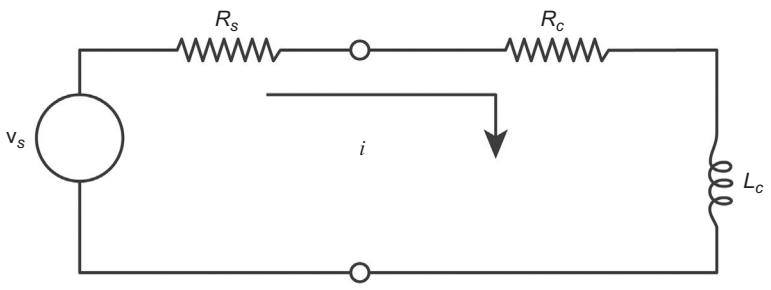


FIG. 8.11 Galvanometer circuit diagram.

$$H_g(s) = \frac{\theta(s)}{v_s(s)}$$

$$H_g(s) = H_i(s)\frac{i(s)}{v_s(s)}$$

where

$$\frac{i(s)}{v_s(s)} = \frac{1}{[sL_c + (R_s + R_c)]}$$

Thus,

$$H_g(S) = \frac{K_c B_r N}{(Js^2 + Ds + K)[(R_s + R_c) + sL_c]} \tag{8.12}$$

Eq. (8.12) can also be given in standard format as seen below:

$$H_g(s) = \frac{K_D \Big/ L_c}{(s^2 + 2\zeta\omega_0 s + \omega_0^2)(s + \omega_1)}$$

where

$$\omega_1 = (R_s + R_c)/L_c$$

One of the important issues for the performance of a galvanometer for automotive display applications is its dynamic response. For displaying relatively slowly changing variables (e.g., fuel quantity), its response should be slow. That is, it should have a relatively low bandwidth (e.g., 0.01 rad/s). With such low bandwidth, the fuel quantity display will indicate effectively the time average of the sensor signal. In this case, the relatively rapid fluctuations in sensor output (due, e.g., to fuel sloshing) is suppressed. The low bandwidth for fuel quantity display is achieved by choice of R_s. On the other hand, galvanometer display of relatively rapidly changing quantities (e.g., vehicle speed or engine RPM) requires a larger bandwidth than for fuel quantity. The optimum bandwidth for any galvanometer automotive display is determined by the designer through the choice of parameters in $H_g(s)$.

ELECTRO OPTIC DISPLAYS

Recent developments in solid-state technology in the field called optoelectronics have led to sophisticated electro-optical display devices that are capable of indicating alphanumeric or pictorial data. A display capable of presenting video-type pictorial displays that is called FPD (or cluster) is discussed in detail following the discussion of alphanumeric displays. An alphanumeric display means that both numeric and alphabetic information can be used to display the results of measurements of automotive variables or parameters. This capability allows messages in English or other languages to be given to the driver and numerical displays. The input for these devices is an electronic digital signal, which makes these devices compatible with computer-based instrumentation, whereas electromechanical displays require a D/A converter and are only capable of indicating a value along the scale.

Automobile manufacturers have considered many different types of electro-optical technology for automotive instrumentation, but only three have been really practical: light-emitting diode (LED), liquid-crystal display (LCD), and vacuum-fluorescent display (VFD). For the advanced high-definition

FPD/cluster, the active elements most frequently used a thin-film-transistor LCD (TFT-LCD), a large number of which are incorporated for high-definition (HD) FPD. Each of the first three technologies can be employed to display alphanumeric characters by placing them in a suitable geometric arrangement such that when illuminated in specific patterns, they appear as the desired alphanumeric characters as depicted in the section of this chapter devoted to VFD. The ultimate application of these electro-optical devices is in the form of a rectangular arrangement of individual devices yielding the so-called FPD (using TFT-LCD). This sophisticated display and its operation are explained after the following explanation of the electro-optical technologies. We consider these technologies separately in the following discussion. Each of these types is discussed briefly to explain their uses in automotive applications. We discuss the physics of the various devices first then explain the required interface electronics to convert the digital processor output to create alphanumeric characters.

LIGHT-EMITTING DIODE

The LED is a semiconductor diode that is constructed in a manner and of a material so that light is emitted when an electric current is passed through it. The semiconductor material most often used for an LED that emits red light is gallium arsenide phosphide (GaAsP). Light is emitted at the diode's PN junction when the positive carriers combine with the negative carriers at the junction (see Chapter 2 for a discussion of PN junctions). The diode is constructed so that the light generated at the junction can escape from the diode and be seen. The light emitted by such a junction has a relatively narrow spectral bandwidth such that it has a specific color. This spectrum is associated with the energy band gap of the carriers in the junction. In addition to providing a light source for an instrumentation system display, an LED also has application in creating light pulses that pass through a fiber-optic channel in creating vehicle communication systems (see "Vehicular Communication," Chapter 9).

Physically, an LED consists of a chip of semiconducting material doped with impurities to create a PN junction as depicted in Fig. 8.12.

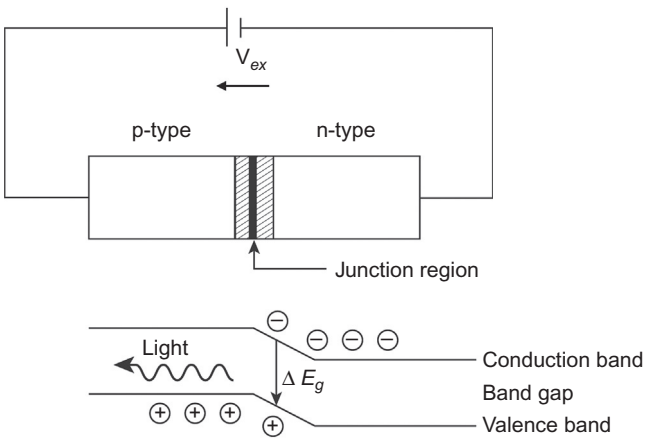


FIG. 8.12 LED configuration and energy bands.

The LED configuration and excitation voltage (V_{ex}) source are depicted in Fig. 8.12. The polarity for this voltage source forward biases the junction. The majority charge carriers flow readily across the junction region that is of such a size as to have a high probability of a free election combining with a hole. In doing so, the electron energy drops by an amount approximately equal to the band gap energy ΔE_g. This drop in energy causes a photon of frequency ν to be released. The photon frequency is proportional to the energy change of the electron:

$$\nu = \frac{\Delta E_g}{h} \tag{8.13}$$

where h = Planck's constant. The wavelength (λ) of the photon is given by

$$\lambda = \frac{c}{\nu} = \frac{ch}{\Delta E_g} \tag{8.14}$$

where c = speed of light.

The color of the emitted light is determined by its wavelength that, in turn, depends upon ΔE_g. The energy band gap (and hence the color of the emitted light) is determined by the semiconductor material and by doping and fabrication.

The light is emitted from an LED within a very narrow cone-shaped region whose axis is orthogonal to the output side surface of the semiconductor chip. The angle of this cone is only a few degrees from the normal to the surface because the semiconductor material has a very high index of refraction relative to air (e.g., the index of refraction of GaAsP $3.2 \leq n \leq 3.4$ in the visible range of wavelengths). The index of refraction for any material n is given by

$$n = \frac{c_o}{c}$$

where c = speed of light in the material and c_o = vacuum speed of light.

Light leaving a transparent medium can only escape the surface and be emitted when the angle of incidence to the surface is less than the critical angle Θ_c (measured from the normal to the surface). The critical angle is the maximum angle of incidence of light leaving the GaAsP material at which it can leave the material. Angles of incidence greater than Θ_c result in total internal reflection and no light leaving the material. For the GaAsP LED, Θ_c is given by

$$\theta_c = \sin^{-1}(1/n) \quad 17 \lesssim \theta_c \leq 18.3 \text{ degrees}$$

Any photon reaching the surface at an angle greater than the critical angle is internally reflected. Often, LED chip surfaces are convoluted with angled facets to increase light output and reduce internal reflections.

An LED display is normally made of small dots or rectangular segments arranged so that numbers and letters can be formed when selected dots or segments are turned on. The configuration for these segments is described in greater detail later in this chapter in the section on VFD. In the early stages of development, a single LED was not well suited for automotive display use because of its low brightness. Although it could be seen easily in darkness, it was difficult to impossible to see in bright sunlight. However, LED technology has evolved such that it is presently a technology capable of significant illumination.

LIQUID-CRYSTAL DISPLAY

The LCD display is commonly used in electronic digital watch displays because of its extremely low electrical power and relatively low voltage requirements. The heart of an LCD is a special liquid that is called a *twisted nematic liquid crystal*. This liquid has the capability of rotating the polarization of linearly polarized light.

The configuration of an LCD can be understood from the schematic drawings of Fig. 8.13. The liquid crystal is sandwiched between a pair of glass plates that have transparent, electrically conductive coatings. The transparent conductor is deposited on the front glass plate in the form of the character or segment of a character that is to be displayed. Next, a layer of dielectric (insulating) material is coated on the glass plate to produce the desired alignment of the liquid-crystal molecules. The polarization of the molecules is vertical at the front, and they gradually rotate through the liquid-crystal structure until the molecules at the back are horizontally polarized. Thus, the molecules of the liquid crystal rotate 90 degrees from the front plate to the back plate so that their polarization matches that of the front and back polarizers with no voltage applied. The operation of an LCD display depends fundamentally upon polarization of light. Before proceeding with an explanation of the LCD operation, it is helpful to review optical polarization.

Polarization of an electromagnetic wave (including light) is associated with the orientation of the electric (E) and magnetic (H) fields, which describe its propagation. At great distances from the source of an electromagnetic wave (e.g., radio wave) at a single frequency $\omega = 2\pi v$ can be represented locally

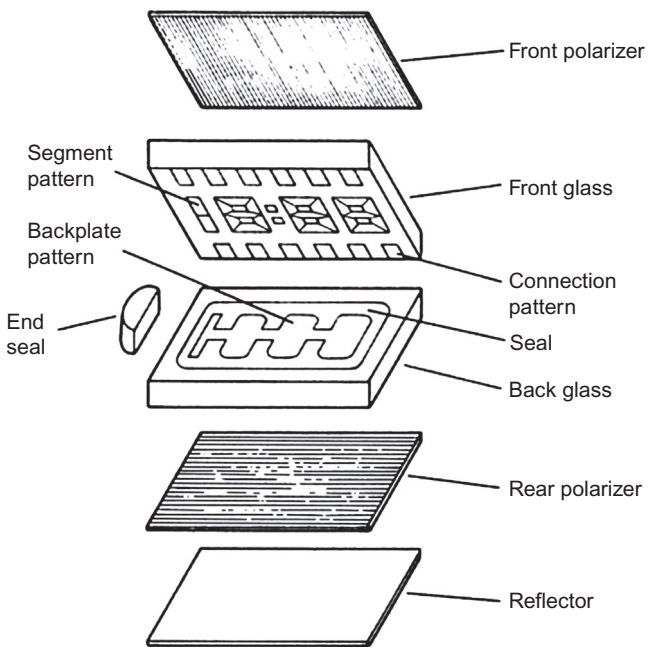


FIG. 8.13 Typical LCD construction.

by a so-called plane wave in which the surfaces of constant phase lie in planes orthogonal to the direction of propagation (here taken to be the z-direction). The electric field intensity vector $\bar{E}$ is assumed to be x-directed and is given by

$$\bar{E}(z,t) = E_x \hat{x} e^{j(\omega t - kz)} \tag{8.15}$$

where $k = \dfrac{2\pi}{\lambda}$ and $\hat{x}$ = unit vector in x direction and where λ = wavelength:

$$= \frac{c}{\mathrm{v}}$$

where c = speed of light in the medium of propagation:

$$= \frac{c_o}{n}$$

where c_o = vacuum speed of light and n = index of refraction of the medium.

The magnetic field intensity vector $\bar{H}$ is given by

$$\bar{H}(z,t) = H \hat{y} e^{j(wt - kz - \pi/2)} \tag{8.16}$$

where $\hat{y}$ = unit vector in the y direction.

The above electromagnetic wave is said to be linearly polarized because the field intensities $\bar{E}$,$\bar{H}$ have the directions $\hat{x}$ and $\hat{y}$, respectively. Light from the sun and from most artificial light sources is not polarized, and the field intensity vectors are randomly directed.

Nonpolarized light can be made to be linearly polarized by passing the light through a polarizing material. For example, light can be polarized by passing it at an angle through a so-called birefringent material. Calcite is an example of a crystalline birefringent material that has the property of having two different indexes of refraction for orthogonal light polarizations relative to the crystal axes. At the exit surface, light exiting at an angle within certain limits will pass the polarized component with the lower index of refraction. The polarization component having the larger index of refraction will be reflected at the surface and will not leave the exit surface. Thus, the light exiting this material is linearly polarized. There are other physical means of polarizing light as well.

If a second polarizer is placed behind the first (in the direction of propagation) with its polarization axis orthogonal to the first, any light exiting the first polarizer will not pass through the second. Such an orientation is termed "cross polarized" polarizers.

The operation of the LCD in the absence of applied voltage can be understood with reference to Fig. 8.14A.

Ambient light enters through the front polarizer so that the light entering the front plate is vertically polarized. As it passes through the liquid crystal, the light polarization is changed by the orientation of the molecules. When the light reaches the back of the crystal, its polarization has been rotated 90 degrees so that it is horizontally polarized and passes through the rear horizontal polarizer. The light is reflected from the reflector at the rear. It passes back through the liquid-crystal structure, the polarization again being rotated, and passes out of the front polarizer. Thus, a viewer sees reflected ambient light and does not see the segment.

The effect of an applied voltage to the transmission of light through this device can be understood from Fig. 8.14B. A voltage applied to any of the segments of the display causes the liquid-crystal molecules under those segments only to be aligned in a straight line rather than twisted. In this case, the light that enters the liquid crystal in the vicinity of the segments passes through the crystal structure

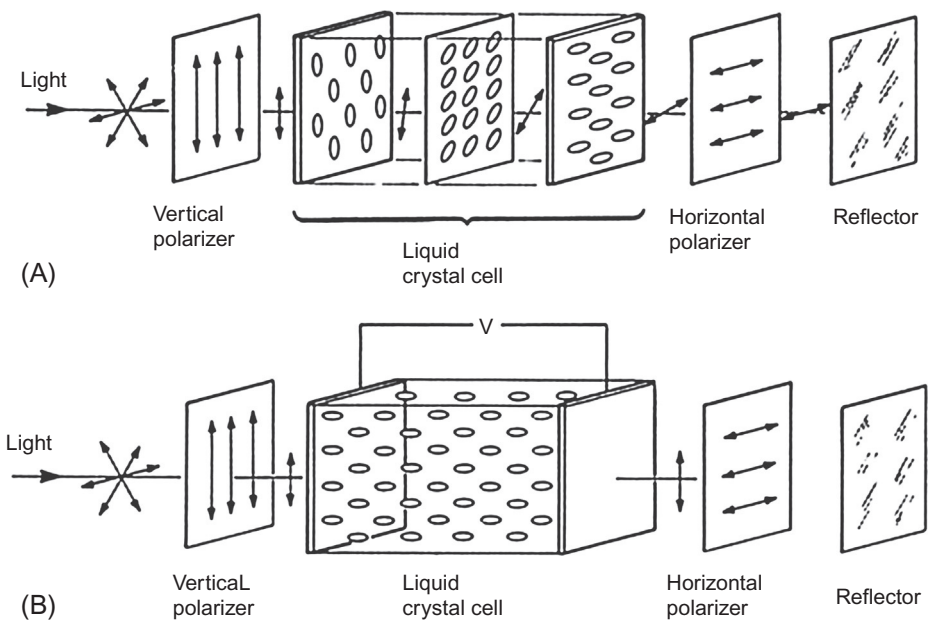


FIG. 8.14 Liquid-crystal polarization. (A) Cross-section showing light polarization with no voltage applied. (B) Cross-section showing light polarization with voltage applied.

without the polarization being rotated. Since the light has been vertically polarized by the front vertical polarizing plate, the light is blocked by the horizontal polarizer so it cannot reach the reflector. Thus, light that enters the cell in the vicinity of energized segments is not returned to the front face. These segments will appear dark to the viewer, the surrounding area will be light, and the segments will be visible in the presence of ambient light. Thus, a voltage of sufficient amplitude applied to any segment of an LCD will darken it relative to the surrounding region. Selective application of voltage to a multi-segment LCD display gives it the capability of displaying alphanumeric characters.

The LCD is an excellent display device because of its low-power requirement and relatively low cost. However, a potential disadvantage of the LCD for automotive application is the need for an external light source for viewing in the dark. Its characteristic is just the opposite of the LED; that is, the LCD is readable in the daytime, but not at night. For night driving, the display must be illuminated by small lamps inside the display. Another disadvantage is that the display does not work well at the low temperatures that are encountered during winter driving in some areas. These characteristics of the LCD have limited its use in automotive instrumentation.

TRANSMISSIVE LCD

An LCD display can also function as an optical transmission device from a light source at the rear of the structure to the front face. A configuration such as this permits an LCD to display messages in low ambient light conditions (e.g., night time). The intensity of the backlight for a transmission to type

LCD is automatically adjusted to produce optimum illumination as a function of the signal from an ambient light level sensor located inside the passenger compartment.

Some display manufacturers produce an LCD that combines reflective and transmissive structures in a so-called transflexive LCD structure. The combination of these two basic LCD types in a package permits optimal readability to be achieved for automotive displays over the entire range of ambient light conditions from bright sunny days to the darkest night conditions.

Another evolution of LCD technology has permitted automotive displays to be available in multiple colors. The LCD configuration described above is a black and white display. A suitable color filter placed in front of the mirror in a reflective LCD or in front of the backlight in a transmissive LCD yields a color display, with the color being determined by the optical filter.

Still another evolution in LCD technology is the development of a very large array of programmable multicolor display including instrument clusters (or FPDs). Such displays are capable of presenting complex programmable alphanumeric messages to the driver and can also present graphic data or pictorial displays (e.g., electronic maps). Since the array structure LCD is functionally similar to the flat-panel type, a detailed discussion of this array type is deferred to the section of this chapter devoted to the discussion of the flat-panel solid-state display.

The electro-optical display technology used for instrument cluster or FPD involves a combination of LCD with thin-film-transistor technology, which is abbreviated TFT-LCD. This type of display involves a relatively complex matrix of individual transmissive LCD elements (or pixels). The individual pixel sizes for each primary color RGB must be sufficiently small to achieve the resolution needed for the most advanced displays (e.g., electronic maps) as discussed in the section of this chapter that explains the FPD-type display.

The light-controlling physical mechanism of a TFT-LCD is essentially that of a transmissive LCD. However, each pixel LCD is controlled by a thin-film type of transistor (of the MOS-type technology, as explained in Chapter 2). The technology for fabrication of TFT-LCD elements is derived from the technology used to make ICs. However, the semiconductor material used to form the TFT is often amorphous silicon (Si) rather than single-crystal or polycrystalline Si. A thin film of amorphous Si is deposited on a glass panel that forms a part of each LCD pixel.

The TFT-LCD display requires backlighting to produce the optical display via transmission of light with RGB color components through each LCD pixel. Backlighting is readily accomplished using RGB LEDs. The actual display at any instant consists of selective activation of each RGB pixel as explained in the later section of this chapter, which is devoted to FPD operation.

VACUUM-FLUORESCENT DISPLAY

The VFD display has been widely used in automotive instrumentation in past years for displaying alphanumeric data. The VFD device generates light in much the same way as a cathode ray tube (e.g., early oscilloscope or TV display) does; that is, a material called phosphor emits light when it is bombarded by energetic electrons. The display uses a filament coated with material that generates free electrons when the filament is heated. The electrons are accelerated toward the anode by a relatively high voltage. When these high-speed electrons strike the phosphor on the anode, the phosphor emits light. A common VFD has a phosphor that emits a blue-green light that provides good readability in the wide range of ambient light conditions that are present in an automobile. However, other colors (e.g., red or yellow) are available by using other phosphors. Additional segments can be added to permit the display of alphabetic characters.

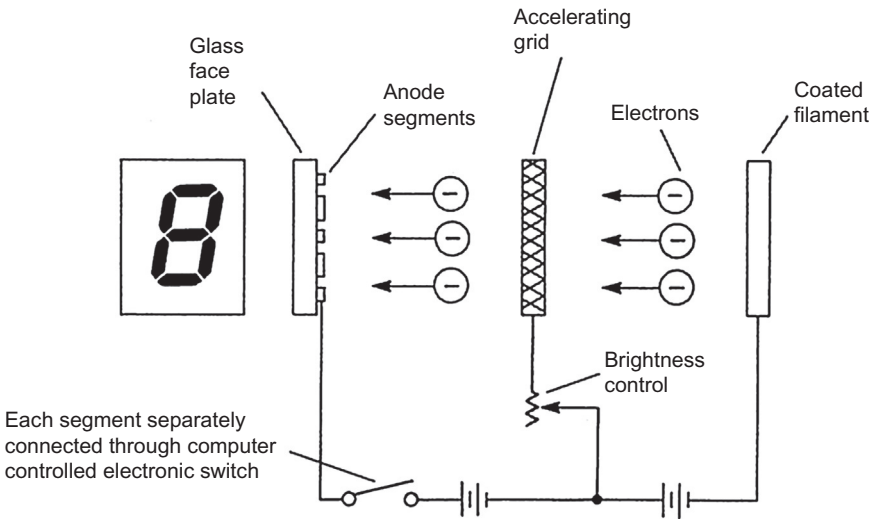


FIG. 8.15 Simplified vacuum-fluorescent display configuration.

The numeric characters are formed by shaping the anode segments in the form of a standard seven-segment character. The basic structure of a typical VFD is depicted in Fig. 8.15.

The structure consists of a pair of glass plates with attached electrodes separated by a third plate having an opening between these electrode plates and that provides a vacuum-tight seal around the edges. The plate, which faces out on the IP, contains thin metallic segments that constitute the anode electrodes for each display segment. The opposite plate contains a filament. The third plate supports the accelerating grid wires and provides the vacuum seal for the display device. The filament is a special type of resistance wire and is heated by passing an electric current through it. The coating on the heated filament produces free electrons that are accelerated by the electric field produced by a voltage on the accelerating grid. This grid consists of a fine wire mesh that allows the electrons to pass through. The electrons pass through because they are attracted to the anode, which has a higher voltage than the grid. The high voltage is applied only to the anode of the segments needed to form the character to be displayed. The instrumentation computer selects the set of segments that are to emit light for any given message. For those readers familiar with vacuum tube technology, the VFD is, in effect, a form of a CRT.

Since the ambient light in an automobile varies between sunlight and darkness, it is desirable to adjust the brightness of the display in accordance with the ambient light. The brightness is controlled by varying the voltage on the accelerating grid. The energy of the electrons striking the phosphor and the brightness of the light is an increasing function of the grid voltage. Fig. 8.16 shows the brightness characteristics for a typical VFD device.

A brightness of 200 fL (footlamberts) might be selected on a bright sunny day, whereas the brightness might be only 20 fL at night. The brightness can be set manually by the driver or automatically. In the latter case, a photoresistor is used to vary the grid voltage in accordance with the amount of

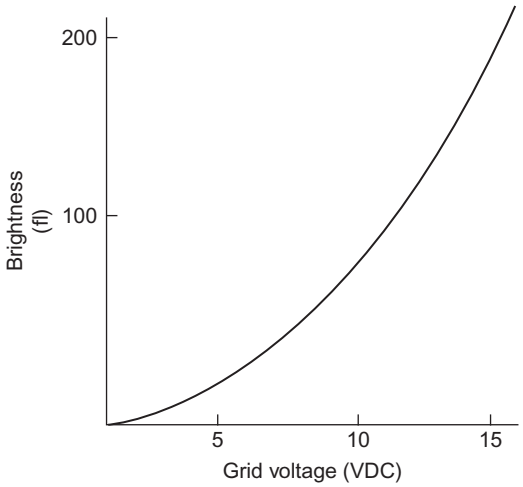


FIG. 8.16 Brightness control range for vacuum-fluorescent display.

ambient light. A photoresistor is a device whose resistance varies in proportion to the amount of light striking it (see the discussion of optical sensors in Chapter 5).

The VFD operates with relatively low power and operates over a wide temperature range. The most serious drawback for automotive application is its susceptibility to failure due to vibration and mechanical shock. However, this problem can be reduced by mounting the display on a shock-absorbing isolation mount.

ALPHA-NUMERIC DISPLAY

As mentioned in the introduction to this chapter, the ultimate display technology is the so-called FPD that is capable of displaying pictorial data (e.g., maps) and alphanumeric messages/data. In preparation for this subject, it is helpful to discuss simpler display types capable of displaying simple alphanumeric characters. All of the display devices, when used to present alphanumeric characters, require interface electronics that receive as input the output signals from the digital system and generate the electrical signals necessary to activate the display segments for the character to be displayed. Such interface is in the form of a so-called decoder circuit. For certain standard digital signal formats, the decoder may be packaged with the display circuitry. Each segment has its own electrical lead from which it is activated. The decoder maps the input m-bit binary signal into the segment leads that are used to create the particular character. The details of this decoder operation are explained later in this chapter.

In preparation for a discussion of advanced FPD technology, it is perhaps instructive to begin with a discussion of the details of the simplest digital numeric display. Such a display has the capability of displaying an N-digit decimal number in which each digit is displayed via a seven-segment electro-optical device. The seven segments for each digit are arranged as depicted in Fig. 8.17.

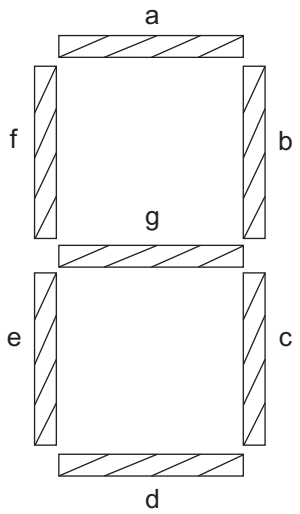


FIG. 8.17 Seven-segment digital display.

The individual segments are designated alphabetically from *a* through *g*. Each decimal number is displayed by activating a specific set of the seven segments. For example, numerical 0 is displayed by activating segments a-f and 8 by activating all seven segments.

The output of the instrumentation computer will be in a binary or binary-coded format. The selection of the subset of segments to be activated can be accomplished by a decoder circuit. Fig. 8.18 depicts a simplified block diagram of a seven-segment display.

For illustrative purposes, a simple system consists of a 4-bit output for each decimal digital display. This data can be transmitted serially or in parallel to the decoder circuit. The interconnection between the instrumentation computer and the decoder circuit can be via a fixed set of leads or it can be via an IVN as explained in the chapter on vehicle communication.

In the present illustrative example, the decoder generates the set of electrical signals required to activate the set of display segments required for the digit being displayed. The set of segments to

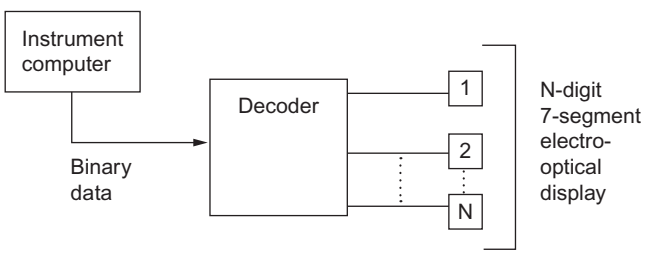


FIG. 8.18 Instrumentation system with display.

Truth table

Input				Outputs							
D	C	B	A	a	b	c	d	e	f	g	Display
0	0	0	0	1	1	1	1	1	1	0	0
0	0	0	1	0	1	1	0	0	0	0	1
0	0	1	0	1	1	0	1	1	0	1	2
0	0	1	1	1	1	1	1	0	0	1	3
0	1	0	0	0	1	1	0	0	1	1	4
0	1	0	1	1	0	1	1	0	1	1	5
0	1	1	0	1	0	1	1	1	1	1	6
0	1	1	1	1	1	1	0	0	0	0	7
1	0	0	0	1	1	1	1	1	1	1	8
1	0	0	1	1	1	1	1	0	1	1	9

FIG. 8.19 Decoder truth table.

be activated for each digit corresponding to a 4-bit binary number sent to the decoder denoted DCBA (where A = LSB) is given in the truth table presented in Fig. 8.19.

A simplified example of a decoder circuit is presented in Fig. 8.20 in which individual logic blocks are presented. These include all of the major logic blocks, the operation of which is given in Chapter 2 in which illustrative circuit examples are presented for logic blocks: AND/NAND, OR/NOR, X-OR, and inverter circuits. It is possible to trace the operation of each block from the 4-bit input DCBA (where A is the LSB) to the subset of segments that are activated by using Boolean algebra that is explained in Chapter 2. For example, the input 0001, corresponding to decimal 1, results in only segments b and c being activated. It is left as an exercise for the interested reader to show the segments that are activated for each input from decimal digit 0 through 9. The result of this analysis can be confirmed from the truth table above.

It should be noted that Fig. 8.20 is simplified in that it only depicts the relationship between the input DCBA and the corresponding segments. In actual operation, this system has inputs that enable the input to be loaded and to clear the display when it is time for refresh or for a new decimal digit to be displayed. Furthermore, there are similar multiple-segment displays with corresponding decoders that have the capability to display alphabetic symbols. A typical automotive display can present multiple digits or sequenced alphabetic symbols capable of depicting an array of messages.

The display devices that have been discussed to this point have one rather serious limitation. The characters that can be displayed are limited to those symbols that can be approximated by the segments that can be illuminated. Furthermore, illuminated warning messages such as “Check Engine” or “Oil Pressure” are *fixed* messages that are either displayed or not, depending on the engine conditions. The primary disadvantage of such ad hoc display devices is the limited flexibility of the displayed messages.

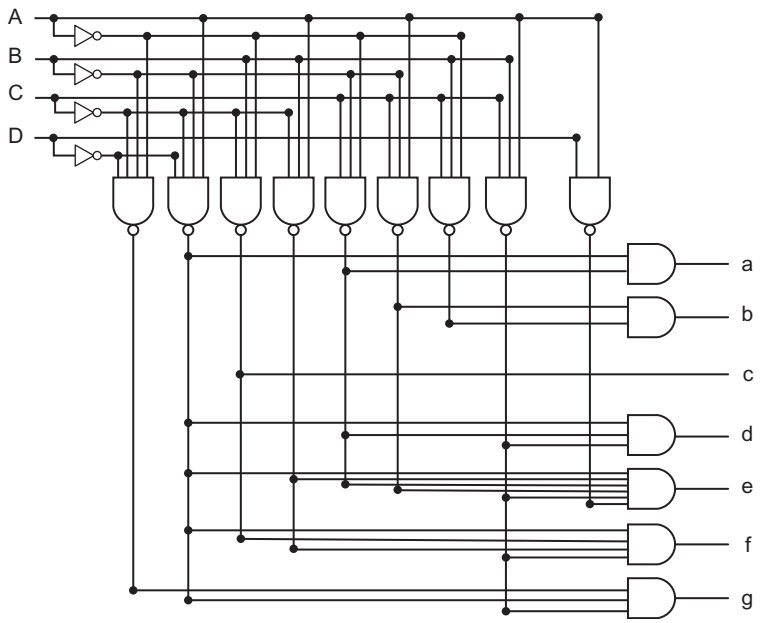


FIG. 8.20 Exemplary BCD to seven-segment display decoder.

FLAT PANEL DISPLAY INSTRUMENT CLUSTERS

Arguably, the display device with the greatest flexibility for presenting all types of data (including pictorial representations) is the so-called FPD (or cluster). This type of display is being used increasingly for display purposes in the aerospace industry, where it is used to display aircraft attitude information (sometimes pictorially), aircraft engine or airframe parameters, navigational data, and warning messages. It is known in the aerospace industry as the "glass cockpit." Clearly, the FPD has great potential for automotive instrumentation display and is coming into common use in vehicles. The FPD can be implemented with various electro-optical technologies as described above. We assume for convenience that TFT-LCD technology is employed in the following discussion.

A solid-state instrument cluster-type display consists of an array of TFT-LCDs arranged in a matrix format as depicted in Fig. 8.21. The display technology includes multiple-color display elements such that an RGB FPD is now available. However, for simplifying the explanation of an FPD, the explanation involves only a single element at a time. The individual elements in such a display are termed pixels. In this example, structure, only one TFT-LCD is active at any time. The intensity of the active pixel is controlled by circuitry connected within the microstructure that controls backlighting. The active pixel is selected via horizontal and vertical control circuitry. In the example FPD, each pixel has its own address. The presentation of alphanumeric or pictorial data requires activation of the associated pixels. This can be done by separately addressing and activating each pixel. However, this is a highly

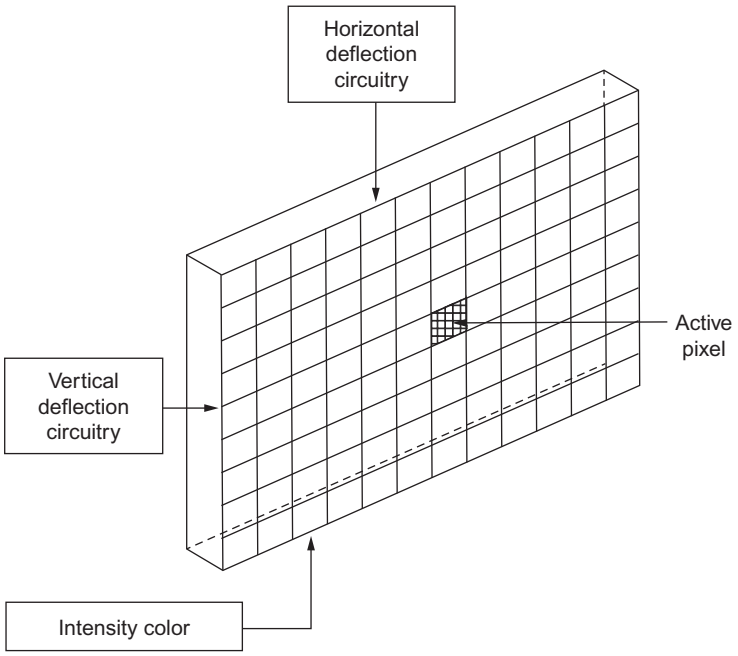


FIG. 8.21 Solid-state array-type display.

inefficient use of the instrumentation computer. An efficient use of this computer involves a scanning method of presenting pixel data in a stand-alone display controller as explained below in the form of a raster-type scan.

One scheme for achieving a solid-state raster-scan display device is to construct an array of elements that can be physically TFT-LCD as depicted in Fig. 8.21. These elements are interconnected with two grids of wires, one running vertically and one running horizontally. Each vertical wire is connected to all of the elements in a given column. Each horizontal wire similarly interconnects all of the elements in a given row.

The presentation of alphanumeric or graphic data requires activating the individual pixel elements that make up the visual pattern to be displayed. For simplicity, we consider only "black/white"-type display, although color display is a simple extension of the present concept. The location of any given pixel in the display is given by its coordinates in an x-y matrix. The mth column of the horizontal position of the display is given by x-coordinate x_m:

$$x_m = m\Delta X$$

where ΔX = the distance in the lateral direction between consecutive columns.

The vertical position of the nth row y_n is given by

$$y_n = n\Delta Y$$

where ΔY = the distance in the vertical (i.e., y) direction between consecutive rows.

One way of presenting the visual information in a display is to incorporate a raster (i.e., the name of the pattern of analog TV scanning)-type scan in which the position of any pixel is a repetitive function of time. In a raster type of scan, the pixels in a row are presented sequentially (e.g., from left to right), and the rows are presented sequentially (e.g., from top to bottom). For such a display, the pixel located at $x_m y_n$ is selected at time $t_{m,n}$ where

$$t_{m,n} = nT_{py} + mT_{px} \quad n = 1,2,\ldots,N,\ m = 1,2,\ldots,M \tag{8.17}$$

and where $T_{py} = MT_{px}$

and $T_c = NT_{py}$

where T_c = period required to scan the entire display

$$= \frac{1}{f_c}$$

f_c = picture cycle frequency (or refresh rate)

$$T_{px} = \frac{1}{f_x} \tag{8.18}$$

f_x = frequency at which columns are scanned

$$T_{py} = \frac{1}{f_y} \tag{8.19}$$

f_y = frequency at which rows are scanned.

In the example raster scan type of display, the scanning is done one row at a time from left to right beginning at the top row during each complete scanning cycle. For the nth row at time nT_{py}, m sequentially changes from 1 to M. At the completion of the scan for row n (i.e., at time $nT_{py} + MTpx$), the next row (i.e., $n+1$) is active, and horizontal scan begins again. The process continues until all N rows have been scanned. At this time (i.e., $t = NT_{py} + MT_{px}$), a cycle is complete, and the entire visual display has been presented. There are specific relationships between these frequencies: $f_y = Nf_c$ and $f_x = Mf_y$. The cycle frequency must be sufficiently fast that the image appears to the driver as a continuous display (e.g., f_c is in the range of 30–60 Hz typically).

One hypothetical configuration for implementing this raster-type scan is shown schematically in Fig. 8.22.

This configuration uses a separate counter and one of M select decoder for activating the desired column, and another counter and one of N select decoder select for activating the desired row. For the purposes of this discussion, it is assumed that whenever the mth column electrical lead and the nth row lead are simultaneously at high voltage, the pixel at $x_m y_n$ is active. By active, we mean that it is illuminated via backlighting in the corresponding pixel LCD activated via the TFT, which is also powered by the row/column signal. The control of whether a pixel is active or not in this hypothetical configuration is controlled by the digital display via logical signals E and E_c, which either enables the signal

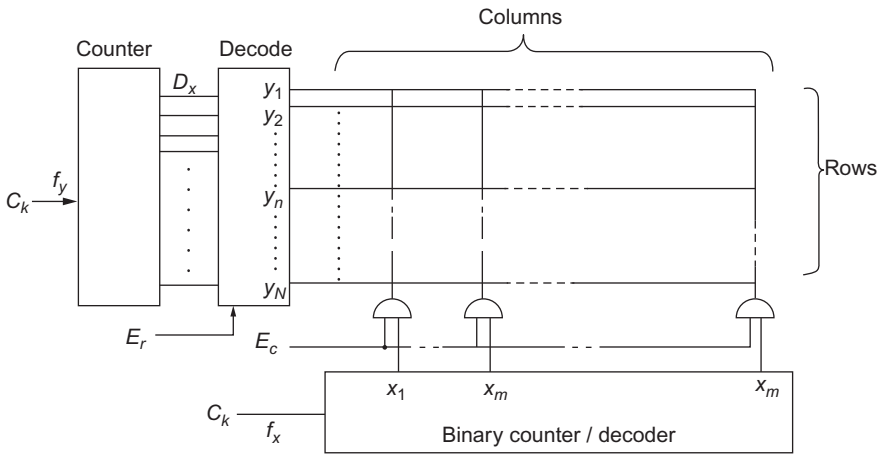


FIG. 8.22 Schematic illustration of representative pixel drive circuits.

for a given column or disables it via separate AND gate for each column output. Whenever a given pixel (e.g., $x_{m,n}$) is to be activated, the logical signals E_r and E_c are set high by the controller at time t_{mn}.

The column counter receives clock (Ck) pulses at frequency f_x. For each pulse received, the column counter is incremented by 1 continuing modulo M. Similarly, the row counter receives clock pulses at frequency f_y and is incremented by 1 for each received pulse (modulo N). A counter can readily be made to count modulo M or N for any pair of integers using appropriate logic circuits connected to the parallel-out counter leads (see Chapter 2). The decoder circuits are one of M and one of N select logic circuits that receive the parallel counter output signals. These circuits place a high voltage on the column number corresponding to the counter contents.

It is assumed that the column select period T_{px} is sufficiently long to activate any given pixel. It is further assumed that the cycle refresh period T_c is sufficiently short that the display can be perceived by the viewer as a complete picture. This perception is influenced by human visual persistency and the illumination period. It has long been known from analog TV that a picture refresh frequency of 30 Hz is sufficient to satisfy the visual persistency requirement.

In the above described example raster-type scan operation of a FPD, the counters and "clocks" at frequencies f_x and f_y are internal to the digital display controller. In such a configuration, the row and column counters can provide the address for the "pixel-active" binary value on signals E_r and E_c. Whenever this pixel is to be active, the digital system logical output, to each of the column select AND gates, is set high enabling the corresponding pixel. Whenever it is to be inactive, this signal is set to logical low, thereby inhibiting the column select signal from establishing the voltage output column lead, thereby rendering that pixel inactive.

A block diagram of the complete hypothetical FPD is shown in Fig. 8.23 in which the display control is depicted as a portion of the entire system.

In this system, the digital instrumentation system, which is microprocessor based and is under program control, receives input signals from all necessary sensors. The inputs may be analog or digital. For each analog signal, conversion to digital format is performed by an A/D converter. The digital control

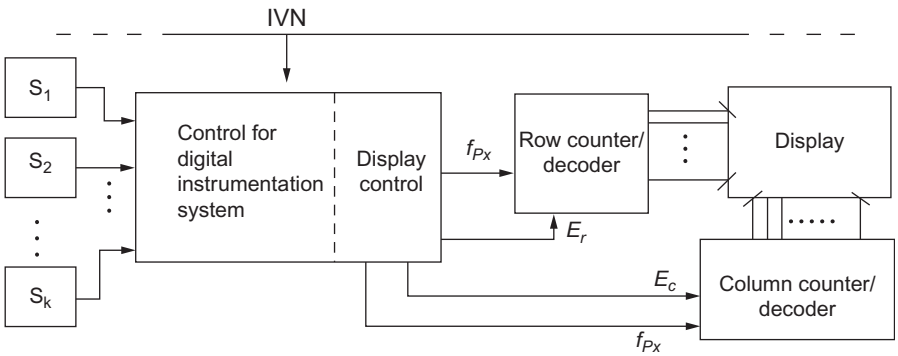


FIG. 8.23 Block diagram for flat-panel display controller.

system performs all signal processing operations, computing in this process an output appropriate for displaying each variable being measured and storing this value in RAM. In addition to the sensor inputs, which provide data for all important vehicle variables, there typically is at least one input from a vehicle network IVN (see Chapter 9) that connects the display system to all other vehicle electronic systems. For some sensors, the data is also supplied to the control via the IVN.

The logical value for each pixel, that is, whether active (i.e., on) or inactive (i.e., off) must be determined to achieve the desired display pattern. It is beyond the scope of this book to explain the software for creating any and all patterns to be displayed for a complex graphic, simulated analog (e.g., for vehicle speed) or pictorial display. However, whenever the display is to be alphanumeric at a specific location, the patterns for each pixel are known in advance and are readily stored in ROM. The display conversion process requires only selection of the pixel logical pattern for the desired alphanumeric character. Similarly, the data for pictorial displays such as digital maps are also stored in system memory, as explained later in this chapter, or it can be obtained via communication with an external infrastructure (see Chapter 9).

An alternative to a raster-type FPD is a so-called random access display. This type of display is advantageous for displaying patterns that either change relatively slowly or change at random (rather than periodically).

In this type of display rather than control of each pixel for every display cycle, the digital instrumentation system uses an intermediate RAM that we call a video RAM. Here, the term random access refers to the access of video RAM by the controller. The digital system transmits only changes to the display pattern when such change is required. The video RAM contains the logical value for each pixel in the display. This logical value E_{nm} corresponding to the pixel at $x_m y_n$ is determined by the digital system under program control.

A block diagram of a hypothetical random-access-type display is shown in Fig. 8.24.

In the display depicted in Fig. 8.24, the logical variables E_{mn} for each pixel are stored at memory locations corresponding to pixel locations $x_m y_n$. The counter decoder for x and y C/D_x and C/D_y, respectively, contain M- and N-bit binary numbers that are used as the address for each logical variable. In the system shown in Fig. 8.24, the x and y C/D sequentially scan the addresses for each pixel in a raster-type pattern. This pattern is repeated at cycle frequency f_c that continuously refreshes the display.

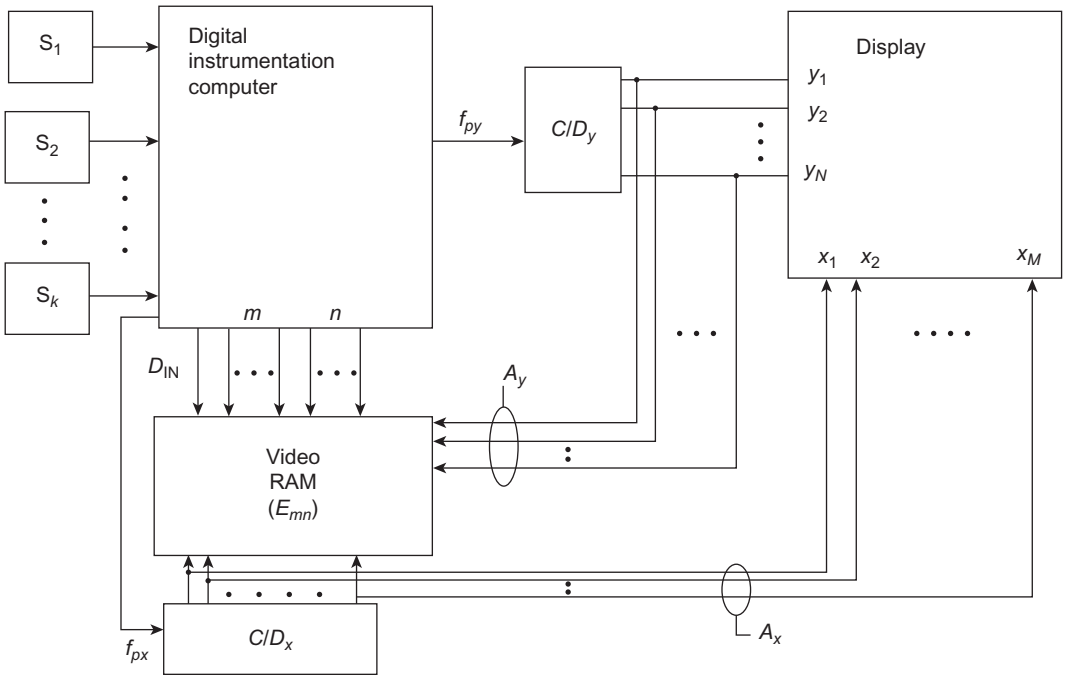


FIG. 8.24 Random access display block diagram.

The digital system can "write" data (see Chapter 3) into a memory location in video RAM (i.e., $x_m y_n$) at times when the display is not reading data during its scan pattern.

The two hypothetical FPD configurations discussed above are representative of a broad range of potential display technologies for automotive use. However, regardless of display technology used and even in cases for which the display is not changing, refresh is required within the time interval of human visual persistence in order to have a recognizable/readable display.

We next consider the structure and operation of an FPD controller that functions in conjunction with the instrumentation computer to cause the scanning display to be generated. A simplified block diagram for a system incorporating an FPD-type display with the associated controller is depicted in Fig. 8.25.

The source signals to be displayed (e.g., from the sensors) and instrumentation computer, which are microprocessor (MPU) based, shown at the left of this illustration have the same function as the corresponding components of the system in Fig. 8.2. The output of the instrumentation computer controls the FPD, working through flat-panel display controller (FPC).

In the example architecture of Fig. 8.25, it is assumed that the instrumentation computer communicates with the FPC via data and address buses (DB and AB) and controls its operation via a serial link along a line or set of lines labeled receiver/transmitter (R/T). However, many other choices of data link are possible. The data that are sent over the DB are stored in a special memory called display RAM. This memory stores digital data that are to be displayed in alphanumeric or pictorial patterns on the

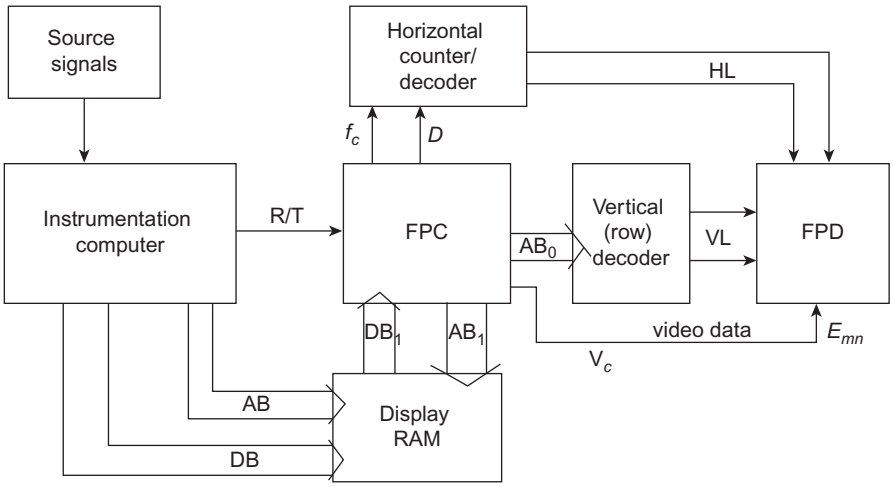


FIG. 8.25 Block diagram of automotive instrumentation with FPD.

FPD. The controller obtains data from the video RAM and converts them to the relevant video signal (V_c). At the same time, the controller activates the horizontal and vertical lines for each pixel in synchronism with the video signal.

The flat-panel controller in the example system (Fig. 8.25) itself incorporates an MPU for controlling the FPD. The data to be displayed are stored in the display RAM via the system buses under the control of the instrumentation computer. The operation of the MPU is controlled by programs stored in a display ROM (DROM). This ROM might also store data that are required to generate particular characters or pictorial symbols. The various components of the display controller are internally connected by means of data and address buses similar to those used in the instrumentation computer.

The operation of the display controller is under the control of the instrumentation computer. This computer transfers data that are to be displayed to the video RAM, via address bus (AB) and DB, and signals the display controller via control C. The FPC outputs a clock signal at frequency f_c to the horizontal counter/decoder and initiates counting via control D in Fig. 8.25. It is assumed for this example that the instrumentation computer transfers data one row at a time. The FPC loads the address of the active row into the vertical decoder circuit that activates the line for that vertical row. The FPC outputs the display video signal synchronously with the horizontal decoder such that the active pixel has the correct excitation.

The details of the transfer of data to the video generator and the corresponding generation of video signals vary from system to system. In the hypothetical system seen in Fig. 8.25, the display is assumed to be an array of TFT-LCD elements arranged in 240 rows vertically by 480 columns horizontally. Fig. 8.26 depicts a small section of the display in which the characters F and P are displayed. The dots are generated by switching on the active element at the desired location by reading the pixel logical variable E_{mn} at address given by the binary address $[x_m, y_n]$ and activating the elements of $E_{mn} = 1$ or having it remain inactive if $E_{mn} = 0$.

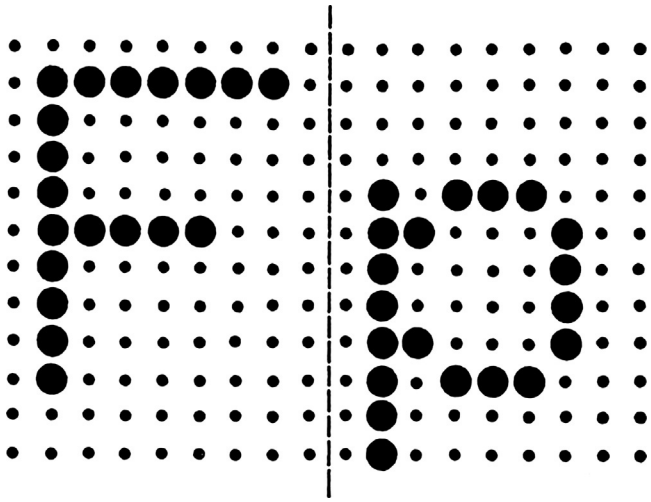


FIG. 8.26 Display of characters F and P.

The enormous flexibility of the flat-panel-type display offers the potential for a very sophisticated automotive instrumentation system. In addition to displaying the variables and parameters that have traditionally been available to the driver, the FPD-type display can present engine data for diagnostic purposes (see Chapter 11), vehicle comfort control system parameters, and entertainment system variables. It should be noted that IP configurations vary widely between the various automobile manufacturers and vehicle models. We have presented only an illustrative sample of IP example configurations here.

PICTORIAL DISPLAY CAPABILITY OF FPD

The multicolor high-resolution capability of an FPD permits display of unique symbols and alphanumeric characters. For example, one image that can be displayed on the HD FPD is the pictorial equivalent of an analog galvanometer-type display (e.g., vehicle speed and engine RPM). In such a display, the symbols presented replicate visually the image of the analog display including a pointer and a numeric scale. The appearance to the driver is visually the same as that of a galvanometer-type display. Essentially, any electromechanical display can be simulated pictorially on an FPD having sufficient resolution. The response time of a good quality FPD can reproduce the dynamics of an analog electromechanical display. In addition, the pictorial capability permits the display of maps that are created from data that are stored in memory or that can be transferred from a smartphone that has a wireless link to a vehicle computer as explained in Chapter 9. Such maps are termed digital maps and are used in conjunction with global position satellite (GPS) (see Chapter 9) for navigation applications. The next topic covered is the theory and presentation of digital maps. Then, following that topic is a discussion of touch-screen capability for certain configurations of FPD.

DIGITAL MAPS

One of the important applications of the FPD in vehicles is the display of digital maps that have an important function in vehicle navigations. Digital maps present two-dimensional displays of road maps similar to the paper equivalent. The scale for such maps is selectable either by the user or automatically by the electronic navigation system. The chapter on "Vehicle Communications" (Chapter 9) explains the mechanism by which the satellite-based GPS determines analytically the vehicle location relative to the position of four or more in-view satellites. The term "in view" refers to satellites whose position is such that a straight line from the vehicle to the satellite is sufficiently above the horizon that the signal received by the vehicle receiver has an adequate signal/noise for the receiver to process the transmitted data.

Except for off-road driving, the vast majority of vehicles are located on roads, highways, streets, or specific parking areas. In most cases, the digital map depicts a region surrounding the vehicle position and normally displays a symbol representing the vehicle position on the map at the appropriate scale.

A digital map is an electronically generated pictorial representation of a region of the earth's surface. However, the earth's surface is well known to not be planar. Rather, the surface of the earth can be modeled as a biaxial ellipsoid (i.e., closely approximated by a sphere) with local perturbations due to its topography. The coordinates of a point on this surface can be represented by so-called geodetic coordinates consisting of latitude φ, longitude λ, and height (h). The latter variable h is a point on the straight line from earth's center to the vehicle location relative to a point on the height baseline corresponding to the given point. One common baseline is the mean sea level. The elevation of any vehicle (land or water based or an aircraft) can be represented as its distance along a line through earth's center at φ, λ relative to the baseline.

A digital map, when displayed on a FPD, is inherently planar as is a paper map. The optimal depiction of a point within a region on the earth's surface by a point on a planar surface is achieved by the projection of the actual point on a plane that is tangent to the baseline surface at a point at or very near the center of the region being depicted. The map location of the point is given by the two-dimensional vector position of the point in any x-y Cartesian coordinate system. Normally for digital maps, the x-axis is east directed, and y-axis is north directed. The process of creating maps via projection on a planar surface is termed "cartography" and has been used for centuries in the creation of paper maps.

The location of a vehicle on a digital map is accomplished via GPS, the theory of which is explained both quantitatively and qualitatively in the chapter on *Vehicle Communications* (Chapter 9). There, it is explained how vehicle location is achieved by calculations based upon measured ranges (termed "pseudoranges") to a minimum of four satellites. The result of these calculations yields vehicle position in either geodetic or earth-centered, earth-fixed (ECEF) coordinates. The map vector position of the vehicle can be accomplished through coordinate transformation. This transformation involves matrix multiplication of the calculated position (e.g., in ECEF) with matrices of direction cosines of the various angles between the axes of the GPS coordinate system and the map axes. The matrices for such a transformation are presented in Appendix D. Vehicle navigation systems have the capability of displaying more than just the instantaneous position on a displayed map. A digital map consists of a two-dimensional vector position on the FPD of all symbols to be displayed for a given map segment at a given scale. These symbols include sufficient points to depict roads and other symbols that could

be found on an equivalent paper map. On most present-day systems, the user can enter a destination. The navigation system has sufficient data stored (e.g., in solid-state memory) to create a map that can depict present position and the selected destination. In doing so, the system must select the region to be depicted and an initial map scale. The latter can be changed by the user as desired. In addition, the capability exists to depict (e.g., via a colored set of lines and arcs) an optimal route to the destination based upon distances and historical traffic information. Moreover, many systems have the capability of receiving and displaying traffic congestion information. This latter function requires some form of communication infrastructure (see Chapter 9 on *Vehicle, Communication*).

TOUCH SCREEN

A FPD having any of the electro-optical technologies can be fabricated with touch-screen capability. A touch-screen-type display provides a mechanism for an input to the instrumentation system that is driving the display. The touch-screen input mechanism is employed in devices such as smartphones but is becoming increasingly employed in automotive FPDs/instrument clusters. In such displays, the electronic system that generates the signals that control the display causes symbols that are readily recognizable by the user. By touching the portion of the flat-screen containing a specific symbol with a finger or special-purpose stylus, the user is providing an input to the system for a specific action. For example, touch-screen input can serve to turn on and/or regulate the temperature of the HVAC system. In any event, the action to be taken by the vehicle instrumentation system depends upon the location on the FPD that is touched by the user. That is to say, the input to the system is determined by the location on the screen where the touching occurs, so the system must have the capability of measuring this touch location. In addition, the system also responds to time-dependent touch location when the touch location is moved by the user.

There are several technologies available for sensing the touch location including resistive, capacitive acoustic wave, and infrared. There are numerous relative advantages and disadvantages of each relative to the others. The particular choice of the sensing technology in automotive application is manufacturer-specific (as well as model). Each of these technologies involves a relatively thin structure on the outside of the FPD. Since the user must be capable of seeing the pattern depicted on the display, this structure (regardless of the technology used) must be highly transparent.

It is beyond the scope of the present book to cover all of these technologies, but the interested reader can find numerous references to each on the Internet. The touch-sensing methodology is illustrated by the following discussion of the capacitive technology.

As explained earlier in this book, capacitance is the property of electric charge storage (of opposite polarity) in a pair of conductors separated by an electric insulator (nonconductor) material. Such a structure is called a capacitor. The theory of capacitor operation and the concept of quantifying and modeling this operation involve the parameter called "capacitance." The electrostatic field theory for capacitor modeling is presented in Chapter 5. The voltage between the electrodes is proportional to the stored charge. For a linear capacitor, the capacitance C is defined

$$C = \frac{\mathrm{V}}{Q}$$

where V = voltage difference and Q = magnitude of the stored charge.

The units of capacitance are farads. For time-varying voltage V(t), the instantaneous current i flowing through a linear capacitor is given by

$$i = C\frac{d\mathrm{V}}{dt}$$

One representative method for sensing touch location is to create a matrix of capacitors by having a set of rows of conductors on one side of an insulating plate and a set of columns on the opposite face. These conductors form the electrodes of a multielement capacitance structure. When sensing touch position, there are two methods: self-capacitance and mutual capacitance. Regardless of which method is used, the presence of a finger or stylus in the vicinity of the crossing points of rows and columns alters capacitance in a measurable way. This variation in capacitance near the touch point yields methods of locating the touch point via capacitance measurement. For the purpose of explaining touch-screen technology, an illustrative mutual capacitance method is considered.

For mutual capacitance methodology, it is a common practice to form the row and column matrix by depositing an essentially transparent array using indium tin oxide ($InSbO_2$) on a glass or other transparent insulator, with the rows on one side and the columns on the other side of the insulator material. By depositing the conductor, it is straightforward to deposit a sequence of pads on each row and column in the vicinity of their crossing points. Such a pattern is illustrated in Fig. 8.27.

In this figure, the solid lines represent columns of conductor on the top surface and rows on the bottom surface. The intersection of a row and column is termed a "node" in the structure. In terms of a Cartesian coordinate system, the nth column is located at x_n and the nth row at y_n. Thus, any node corresponds to location x_m, y_n. The mutual capacitance between any row and any given column is due, in part, to the relative size and spacing of the conductors. However, it is also influenced by the presence of a finger or stylus. The touch location x_t, y_t is found by sequentially measuring the mutual capacitance

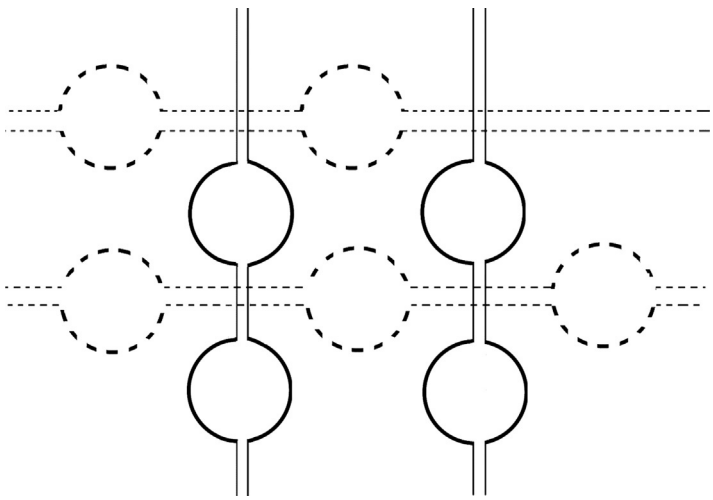


FIG. 8.27 Illustration of exemplary mutual capacitance configuration.

$C(m,n)$ at node m,n and subtracting it from the known no-touch capacitance $C_o(m,n)$ yielding differential mutual capacitance $\delta C_k(m,n)$ at time t_k:

$$\delta C_k(m,n) = C_o(m,n) - C_k(m,n) \quad k = 0,1,2,\ldots,\ m = 1,2,\ldots,M,\ n = 1,2,\ldots,N$$

The touch location is approximated closely by x_m, y_n for which the magnitude of $\delta C_k(m,n)$ is greatest. Curve-fitting algorithms exist for improved accuracy in estimating x_t, y_t.

The process of determining touch location involves the measurement of all n,m capacitances. There are several methods of measuring capacitance $C(m,n)$. An exemplary method involves connecting the capacitance to be measured to an oscillator and measuring the period or alternatively by the frequency of oscillation. One such oscillator is an astable multivibrator as described in Chapter 5 in the section on "Oscillator Methods of Measuring Capacitance." For such an oscillator, the period T is linearly proportional to $C(m,n)$ to which it is sequentially connected:

$$T(m,n) = aC(m,n)$$

where a is a known constant for the oscillator circuit as given in the above-named section of Chapter 5.

A simplified circuit/block diagram is depicted in Fig. 8.28. In this figure, the multivibrator output voltage $V_o(t)$ is binary valued with a high output (logic 1) for the period denoted T_{Hk} for the kth oscillator cycle with $T_{Hk}[C(m,n)]$ given in Chapter 5 and a low output corresponding to logic 0 for the remainder of the oscillator period T. Both high and low periods of an oscillator cycle are linearly proportional to the capacitance as explained in Chapter 5.

The clock is a high-frequency oscillator of frequency f_c. At the beginning of each oscillator period, the binary counter is edge triggered to reset to 0 output. During the interval $T_{Hk}(m,n)$, the AND gate enables the binary counter to count clock pulses. At the end of the logic high for the kth cycle, the binary counter has the binary equivalent of N_k where

$$N_k = \{\lfloor f_c T_{Hk}(m,n) \rfloor\} = \text{integer portion of } f_c T_{Hk}(m,n)$$
$$= a f_c C_k(m,n)$$

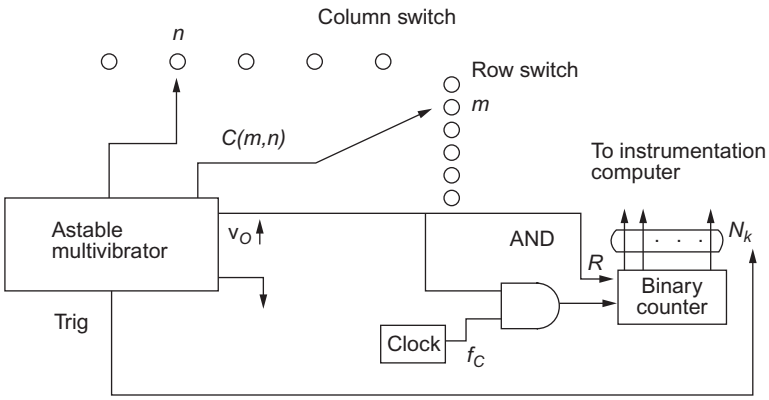


FIG. 8.28 Exemplary $C(m,n)$ measurement system.

where the formula for a is given in Chapter 5. That is, the binary counter contains a numerical value that is proportional to $C_k(m,n)$. During the transition from the high output to the low output of $v_o(t)$, the astable multivibrator generates a pulse that is called a trigger pulse. This trigger pulse can serve as an interrupt input to the instrumentation computer signaling that the data, which are proportional to $C_k(m,n)$, are available to be read on a parallel data path.

The touch location sensing requires measurement to $C(m,n)$ on a periodic basis at all nodes. In the block diagram of Fig. 8.28, the selection of any row is accomplished via electronic switching as depicted by the multi-input row switch and the column by the multi-input column switch. The switching is controlled by the instrumentation computer and can be accomplished physically by multiplexing-type hardware as explained in Chapter 2.

The algorithms for calculating $\delta C_k(m,n)$ and for finding m and n for maximum value are straightforward. Once the m and n for maximum $\delta C_k(m,n)$ are found, the touch position $(x_t, y_t) \simeq x_k(m), y_k(n)$ has been determined. The same computer that measures touch position also generates the display such that the sensed touch position corresponds to the user input associated with the displayed symbol at the touch location.

The mutual capacitance method of touch sensing enables multiple finger touch points to be determined. Such an input to the computer that generates the display permits the user to change the scale of the display as is commonly done with smartphones.

The automotive FPD with touch-screen capability can be a system that yields hands-free cell phone use for verbal and voice-generated text messaging through the cell phone infrastructure. Communication between the in-vehicle user and the other cell phone in use is through that infrastructure. However, the in-vehicle user can communicate via built-in microphone/speaker or through a smartphone that connects wirelessly to the vehicle via bluetooth communication system (or technical equivalent). For such a communication link to be established, the user must pair his/her smartphone with the built-in bluetooth system. With such an arrangement, the phone number of the vehicular system is that of the paired smartphone. The linking via a bluetooth communication is explained in the chapter on *Vehicular Communication* (Chapter 9).

Interaction with this system is both verbal and via a touch-screen input. For example, if a call is incoming to the vehicle, the instrumentation system can cause a symbol to be generated on the screen that is clearly identifiable by the user as an incoming call. The user touches this symbol on the screen to answer the incoming call. The user (e.g., the driver) can conduct the phone call hands-free and with minimal distraction from driving.

Similarly, when a phone call is initiated in the vehicle, the system is activated either by touching a symbol being displayed on the flat-panel screen or by a verbal command. Once activated, the system states through synthesized voice that it is ready to dial. The user/driver verbally gives the phone number for the intended call or states the name of the person to be called, and the system memory holds the corresponding number or name in its memory system.

The technology for implementing such calls is based on advanced software that is capable of generating synthesized vocal signals and speech recognition. The details of this software are beyond the scope of this book. However, sources are available for explaining this software in the public domain literature.

The use of an FPD with TS capability for the hands-free use of a cell phone by a driver while driving the vehicle can greatly improve safety relative to manual cell phone use. The earliest cell phones, when

used by the driver of a vehicle, created a serious potential safety hazard such as being distracted from viewing the road to look at the phone and dialing the phone, which required the driver to take his/her hands off of the steering wheel and eyes off the road.

Contemporary smartphones can be used hands-free without distracting the driver's attention from driving. Once the cell phone is connected via bluetooth to the vehicle audio system, it is possible to use the vehicle audio system and its associated computer-based system to control cell phone operation and to use the vehicle loudspeaker for the audio output of a cell phone call.

The FPD with TS capability can be used for a switched driver input to turn on any vehicle system in the same way as described for the cell phone. That is, a symbol representing a vehicle system that is to be operated by a vehicle front-seat occupant is displayed on the FPD. By touching this symbol, the system (e.g., HVAC) can be switched on or off by the user including the driver (with minimal distraction). In addition, it is possible for the FPD to display a symbol that can regulate the system being activated via touching (e.g., temperature of various vehicle zones). The touching of the TS capable FPD requires insignificant driver distraction and contributes to vehicle safety.

MEASUREMENT EXAMPLES

Having described the various components and implementation of automotive instrumentation, we now present some specific measurement examples. These examples present representative circuit diagrams or block diagrams that show specific components involved in each of the examples presented in this chapter. The instrumentation system being discussed is a digital system that involves multiplexing/demultiplexing for each of the examples. We begin with an example for the measurement of fuel quantify.

FUEL QUANTITY MEASUREMENT

During a measurement of fuel quantity, the MUX switch functionally connects the computer input to the fuel quantity sensor, as shown in Fig. 8.29. This sensor output is converted to digital format and then sent to the computer for signal processing. (*Note:* In some automotive systems the analog sensor output is sent to the instrumentation subsystem, where the A/D conversion takes place.)

Several fuel quantity sensor configurations are available. Fig. 8.30 illustrates the type of sensor to be described, which is a potentiometer connected via mechanical linkage to a float.

In Chapter 5, a potentiometer was introduced as a sensor for measuring throttle angular position. It also has application in certain fuel measuring instrumentation. Normally, the sensor is mounted so that the float remains laterally near the center of the tank for all fuel levels. A constant current passes through the sensor potentiometer, since it is connected directly across the regulated voltage source. The potentiometer is used as a voltage divider so that the voltage at the wiper arm is related to the float position, which is determined by fuel level.

The sensor output voltage is not directly proportional to fuel quantity in gallons because of the complex shape of the fuel tank. The computer memory contains the functional relationship between sensor voltage and fuel quantity for the particular fuel tank used on the vehicle.

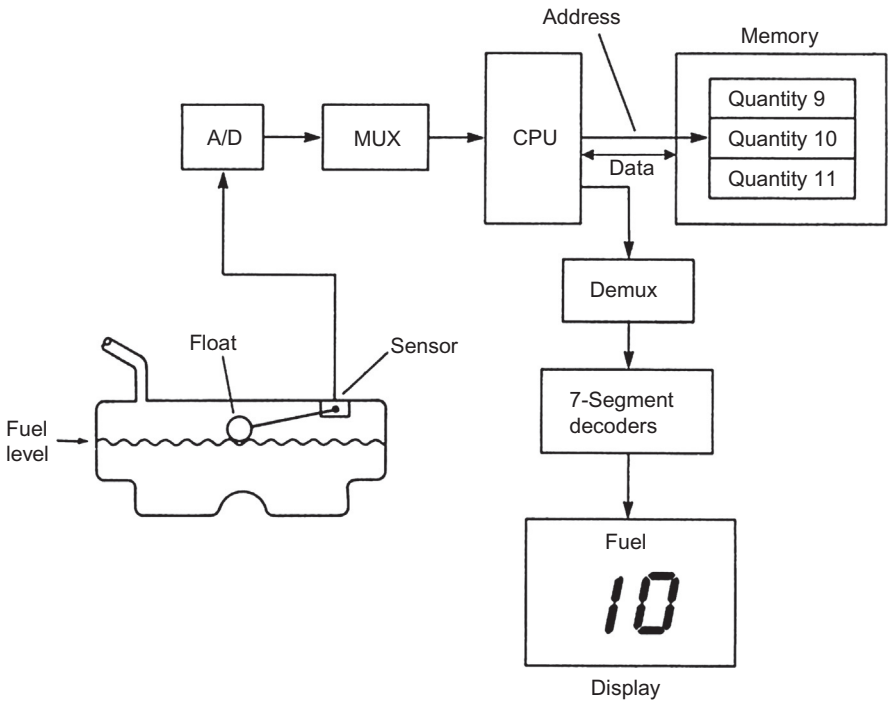


FIG. 8.29 Fuel quantity measurement system.

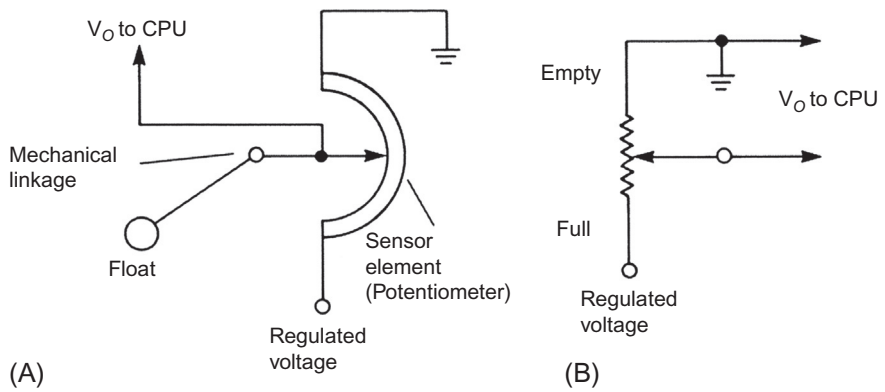


FIG. 8.30 Fuel quantity sensor configuration. (A) Configuration and (B) schematic.

The computer reads the binary number from the A/D converter (see Fig. 8.29) that corresponds to sensor voltage and uses it to address a particular memory location. Another binary number corresponding to the actual fuel quantity in gallons for that sensor voltage is stored in that memory location. The computer then uses the number from memory to generate the appropriate display signal—either analog or digital, depending on display type—and sends that signal via DEMUX to the display.

Computer-based signal processing can also compensate for fuel slosh. As the car moves over the road, the fuel sloshes about, and the float moves up and down around the average position that corresponds to the correct level for a stationary vehicle. The computer compensates for slosh by computing a running average of the fuel sensor voltage ($v_0(t)$ of Fig. 8.30). It does this by storing several samples over a few seconds and computing the arithmetic average of the sensor output or by low-pass filtering the sensor voltage. The oldest samples are continually discarded as new samples are obtained. The averaged output becomes the signal that drives the display. It should be noted that this is actually a form of digital filtering.

Let $v_n = v_0(t_n)$ be the fuel sensor voltages at the nth sample time (t_n). The actual fuel quantity is denoted F. The sensor voltage v_0 is a known function of F for any fuel/angle sensor combination such that the sensor terminal voltage is given by

$$v_0 = f_F(F) \tag{8.20}$$

The instrumentation computer (under program control) computes the sampled measurement F_n from v_n:

$$F_n = f_F^{-1}(v_n) \tag{8.21}$$

The short-term time average of fuel quantity F_{av} is given by

$$F_{av}(n) = \frac{1}{N}\sum_{m=1}^{N} F_{n-m} = \frac{1}{N}\sum_{m=1}^{N} f_F^{-1}(v_{n-m}) \tag{8.22}$$

The sloshing effect of fuel on fuel gauge indicated value also can be reduced by filtering the fuel sensor output. A block diagram of a fuel measurement instrumentation configuration with a LPF is shown in Fig. 8.31.

In this figure, the sensor output voltage is given by Eq. (8.20). The DSP is implemented in the instrumentation computer and includes the nonlinear correction block (NLC), which calculates the quantity of fuel F from the sampled sensor voltage as given in Eq. (8.21). The DSP also includes a LPF that has z-transfer function $H_{sp}(z)$. This latter calculation is done in a separate subroutine as a recursive algorithm. It should be noted that the short-term time average of fuel quantity also is effectively a form of LPF.

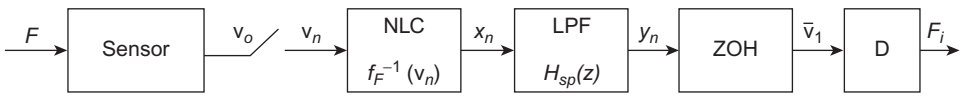


FIG. 8.31 Filtering fuel sensor signal.

It is assumed for the sake of illustration that the display device is an analog meter of the galvanometer configuration explained earlier in this chapter. It is further assumed that the scale is marked such that for a full tank θ is at full scale and for an empty tank $\theta = 0$ with fuel quantity F expressed as a fraction of full tank is given by deflection θ. The continuous-time transfer function for this type of display ($H_0(s)$) was shown earlier (Eq. 8.12) to be given by

$$\begin{aligned} H_D(s) &= \frac{\theta(s)}{v_0(s)} \\ &= \frac{K_c N B_r}{(Js^2 + sD + K)[(R_c + R_s) + sL_c]} \end{aligned} \tag{8.23}$$

A typical galvanometer display is designed such that the torque component proportional to the moment of inertia (J) is insignificant compared with the damping and spring torques. Thus, the transfer function is given approximate by

$$H_0(s) \cong \frac{K_c N B_r}{D L_C (s + s_0)(s + s_L)} \tag{8.24}$$

where $s_0 = \dfrac{K}{D}$

$$s_L = \frac{R_s + R_c}{L_c}$$

Using numerical values for a representative automotive analog fuel gauge, these frequency parameters are the approximate ranges given below:

$$s_0 \cong 0.5 - 2.0$$
$$s_L \cong 10^5 - 10^6$$

The large disparity in these pole locations makes the pole at s_0 the dominant pole and that at s_L a so-called insignificant pole. The result of this disparity is that the transfer function is approximately given by

$$\begin{aligned} H_D(s) &\cong \frac{K_c N B_r}{D(R_s + R_c)(s + s_0)} \\ &= \frac{K_D}{s + s_0} \end{aligned} \tag{8.25}$$

where K_D is a constant for the display that is given by

$$K_D = \frac{K_c N B_r}{D(R_s + R_c)} \tag{8.26}$$

Using the methods of Appendix B, it can be shown that the z-transfer function for the combination ZOH and display $H_D(z)$ is given by

$$H_D(z) = (1 - z^{-1}) Z\left(\frac{H_D(s)}{s}\right) \tag{8.27}$$

Representative values for K and S_0 are given by

$$K_D = 0.5$$
$$s_0 = 0.5$$

For a sample period of $T = .001$ s, the z-transfer function is given by

$$H_D(z) = \frac{(1 - z_1)}{(z - z_1)} \tag{8.28}$$

where $z_1 = e^{-s_0 T} = 0.9995$

The digital filter is chosen as a second-order Butterworth filter having a digital corner frequency $\Omega_c = 0.001$. It can be shown using the methods of Appendix B that the z-transfer function for this filter $H_{sp}(z)$ is given by

$$H_{sp}(z) = 10^{-5} \frac{[0.2462z^2 + 0.4924z + 0.2462]}{z^2 - 1.9956z + 0.9956} \tag{8.29}$$

The dynamic performance of this digital fuel measurement system can readily be demonstrated via simulation. The Simulink simulation model block diagram is shown in Fig. 8.32. The fuel tank is assumed to be one-half full ($F = 0.5$), and the fuel slosh is simulated in MATLAB/Simulink via a filtered white-noise source.

In this block diagram, fuel slosh is created using a random number generator filtered to yield a band-limited stationary random process. This random process is combined with the constant 0.5 representing the one-half full tank. The block labeled discrete transfer function is the second-order Butterworth digital filter having transfer function $H_{sp}(z)$ of Eq. (8.29). The final continuous-time transfer function represents the display dynamic response having gain $K_D = 0.5$ and bandwidth $s_o = 0.5$.

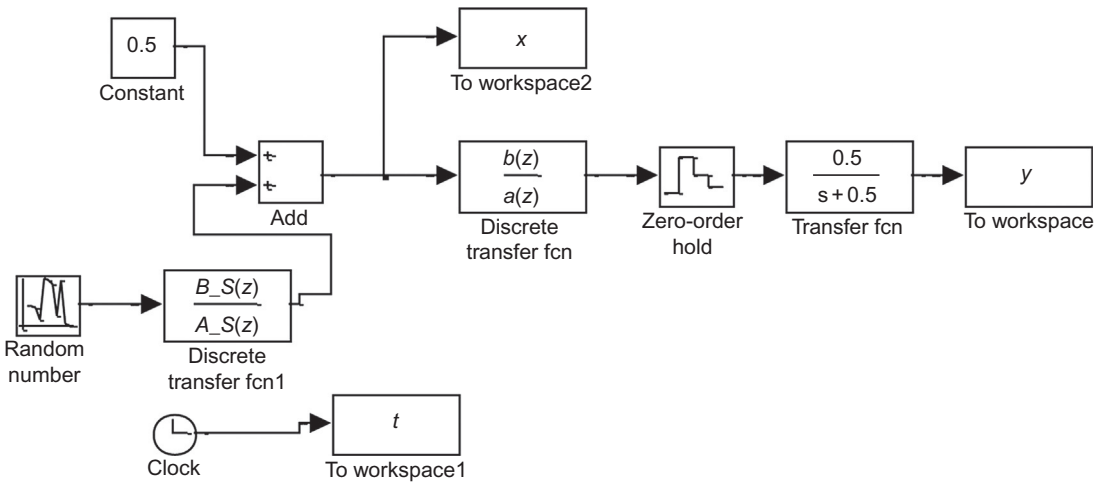


FIG. 8.32 Simulink model for fuel quantity instrument subsystem.

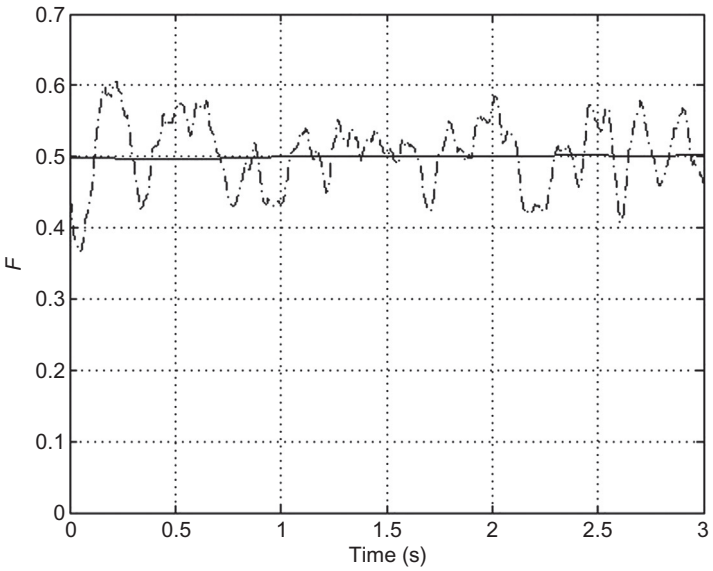


FIG. 8.33 Filtered fuel quantity *F* (solid line) and unfiltered fuel quantity (dashed line).

A sample of the system response is shown in Fig. 8.33 in which the dashed curve represents the unfiltered fuel measurement and the solid line represents the displayed value. The filtered display deviates only slightly (i.e., less than 1%) from the true value of $F = 0.5$ (i.e., $-1/2$ tank of fuel), but the random fluctuations due to fuel slosh are completely suppressed.

COOLANT TEMPERATURE MEASUREMENT

Another important automotive parameter that is measured by the instrumentation is the coolant temperature. The measurement of this quantity is different from that of fuel quantity because usually it is not important for the driver to know the actual temperature at all times. For safe operation of the engine, the driver only needs to know if the coolant temperature is more than a critical maximum value. A block diagram of the measuring system is shown in Fig. 8.34.

The coolant temperature sensor used in most cars is a solid-state sensor called a *thermistor*. Recall that this type of sensor was discussed in Chapter 5, where it was shown that the resistance of this sensor decreases with increasing temperature. Fig. 8.35 shows the circuit connection and a sketch of a typical sensor output voltage (v_o) vs. temperature (T) curve, which was calculated based on the model for a thermistor connected as in Fig. 8.35.

The sensor output voltage is sampled during the appropriate time slot and is converted to a binary number equivalent by the A/D converter. The computer compares this binary number to the one stored in memory that corresponds to the high-temperature limit. If the coolant temperature exceeds the limit,

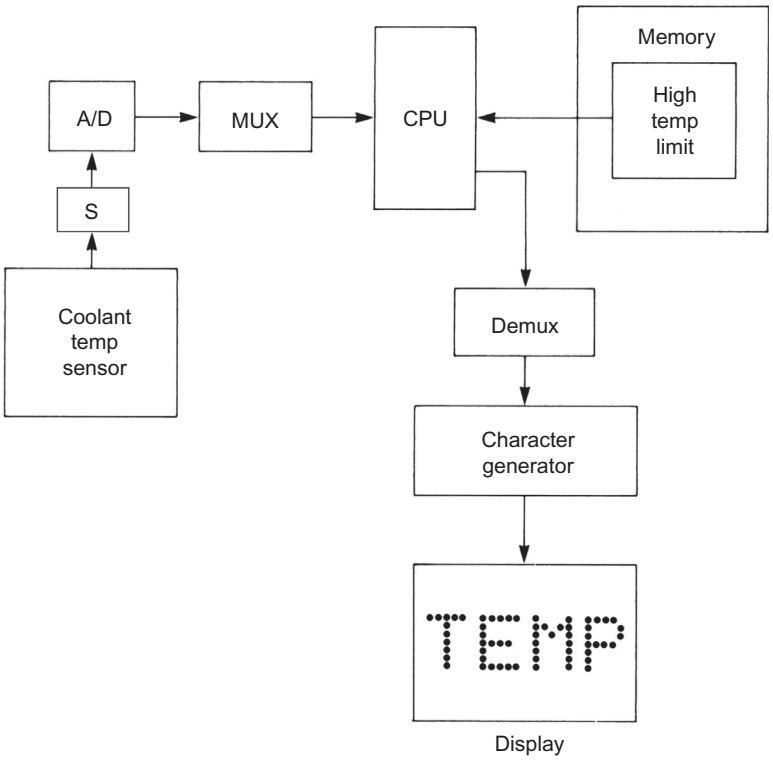


FIG. 8.34 Coolant temperature measurement.

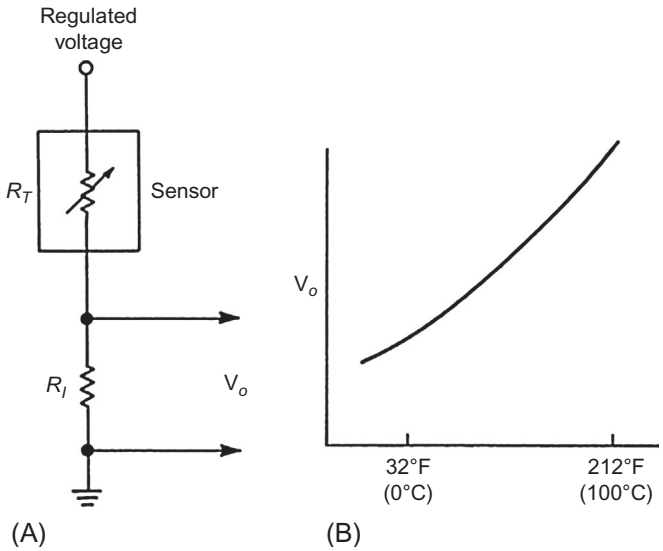


FIG. 8.35 Coolant temperature sensor circuit. (A) Schematic and (B) output V versus *T*.

an output signal is generated that activates the warning indicator. If the limit is not exceeded, the output signal is not generated, and the warning message is not activated. A proportional display of actual temperature can be used if the memory contains a cross-reference table between sensor output voltage and the corresponding temperature, similar to that described for the fuel quantity table.

OIL PRESSURE MEASUREMENT

Engine oil pressure measurement is similar to coolant temperature measurement in that it frequently uses a warning message display rather than an indicated numerical value although certain high-performance vehicles contain a display that either simulates an analog oil pressure gauge or uses a galvanometer-type display. Whenever the oil pressure is outside allowable limits, a warning message is displayed to the driver. In the case of oil pressure, it is important for the driver to know whenever the oil pressure falls below a lower limit. It is also possible for the oil pressure to go above an allowable upper limit; however, some manufacturers do not include a separate high-oil-pressure warning in the instrumentation.

The simplest oil pressure warning system involves a spring-loaded switch connected to a diaphragm. The switch assembly is mounted in one of the oil passageways such that the diaphragm is exposed directly to the oil pressure. The force developed on the diaphragm by the oil pressure is sufficient to overcome the spring and to hold the switch open as long as the oil pressure exceeds the lower limit. Whenever the oil pressure falls below this limit, the spring force is sufficient to close the switch. Switch closure is used to switch on the low-oil-pressure warning message lamp.

One of the deficiencies of this simple switch-based oil pressure warning system is that it has a single fixed low-oil-pressure limit. In fact, the threshold oil pressure for safe operation varies with engine load. Whereas a relatively low oil pressure can protect bearing surfaces at low loads (e.g., at idle), a proportionately higher-oil-pressure threshold is required with increasing load (i.e., increasing horsepower and RPM).

An oil pressure instrument that operates with a load- or speed-dependent threshold requires an oil pressure sensor rather than a switch. Such an oil pressure warning system is illustrated in Fig. 8.36. This system uses a variable-resistance oil pressure sensor (e.g. piezoresistive) such as seen in Fig. 8.37. Sensors of this type were discussed in Chapter 5. A voltage is developed across a fixed resistance connected in series with the sensor that is a known function of oil pressure. It should be noted that this assumed pressure sensor is hypothetical and used only for illustrative purposes.

During the appropriate measurement time slot, the oil pressure sensor voltage is sampled through the MUX switch and converted to binary numbers in the A/D converter. The computer reads this binary number and compares it with the binary number in memory for the allowed oil pressure limits. The oil pressure limit is determined from load or crankshaft speed measurements that are already available in the engine control system. These measurement data can be sent to the instrument subsystem via a MUX system as described with respect to Fig. 8.5 and over an IVN. These measurements serve as the address for a ROM lookup table to find the oil pressure limit. If the oil pressure is below the allowed lower limit or above the allowed upper limit, an output signal is generated that activates the oil pressure warning light through the DEMUX (see Fig. 8.7).

It is also possible to use a proportional display of actual oil pressure. A digital display can be driven directly from the computer. An analog display, such as a galvanometer, requires a D/A converter.

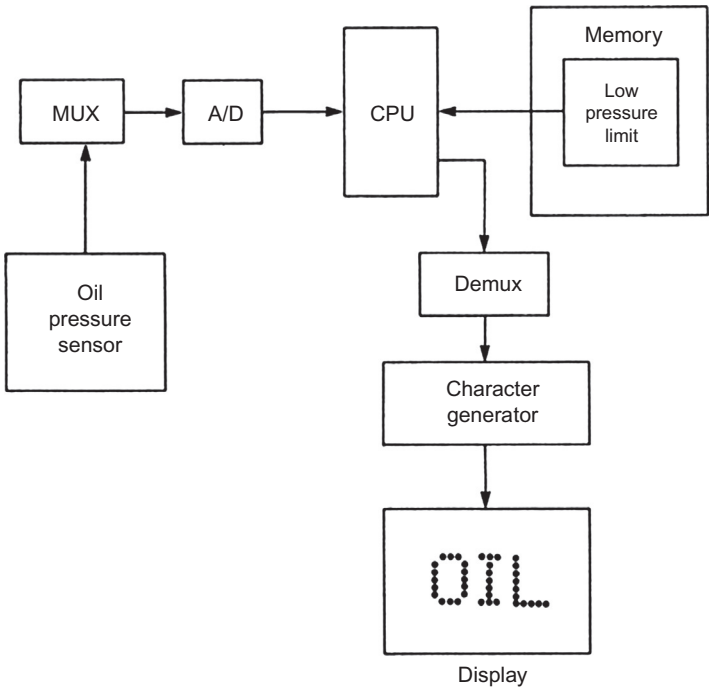


FIG. 8.36 Oil pressure measurement instrumentation.

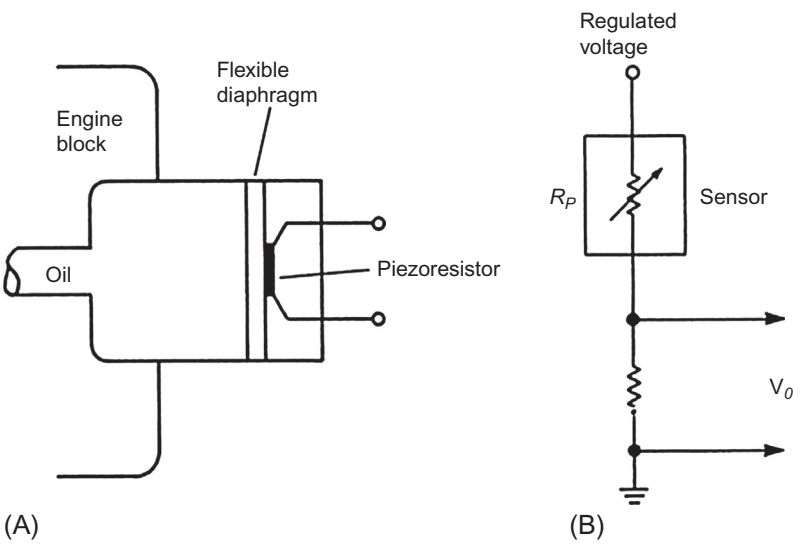


FIG. 8.37 Oil pressure sensor. (A) Configuration and (B) schematic.

VEHICLE SPEED MEASUREMENT

An example of a digital speed sensor has already been described in Chapter 7 for a cruise control system. The speed sensor is assumed to be of a structure such as is depicted in Fig. 5.10 or 5.13. In either of these sensors a single pulse is generated with the passage of each lug on the disk. A sensor of this type is assumed to be used for car speed measurements. The output of the speed sensor is a sequence of pulses at frequency f_p that is proportional to vehicle speed S:

$$f_p = k_s S \tag{8.30}$$

The sensor constant k_s is proportional to the number of lugs on the disk and the gear ratio between the shaft on which the disk is mounted to the drive axle.

A block diagram of the digital system (including the instrumentation computer) that determines vehicle speed from the speed sensor is depicted in Fig. 8.38. Since the sensor output pulse frequency is proportional to vehicle speed, a digital speed measurement can be obtained by counting pulses for a given specific time interval (τ). The pulse counting is accomplished via a binary counter (see Chapter 3). The time interval during which sensor output pulses are counted is determined by a control signal G from the instrumentation control system (ICS).

The electronic gate of Fig. 8.38 is functionally an electronically controlled switch (e.g., implemented by an FET; see Chapter 2) whose state (i.e., open or closed) is controlled by the binary-valued signal represented by logical variable G.

The ICS periodically outputs this logical control signal such that $G = 1$ corresponds to closed gate for which sensor pulses are sent to the counter and $G = 0$ corresponds to the gate open and counting is inhibited as given below:

$$\begin{aligned} G &= 1 \quad t_k \le t \le t_k + \tau \\ &= 0 \quad t_k + \tau < t < t_{k+1} \\ t_{k+1} - t_k &= T_s = \text{sample period} \end{aligned}$$

During the period in which $G = 1$, each sensor pulse causes the counter to increment by 1. Thus, at time $t_k + \tau$, the counter contains count P where

$$P = \{\lfloor f_p \tau \rfloor\}$$

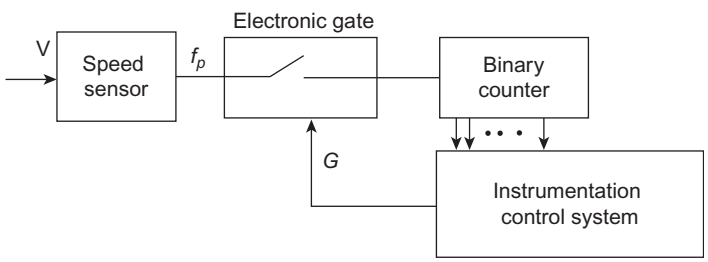


FIG. 8.38 Vehicle speed instrument subsystem.

where the brackets indicate the largest integer in the product $f_p\tau$. At some point during the post counting interval (i.e., $t_k + \tau < t < t_{k+1}$), the digital system generates signals necessary to transfer the counter contents to a memory location.

Under program control, the vehicle speed is computed from the counter contents as given below:

$$S = \frac{P}{k_s \tau}$$

where k_s is the speed sensor constant given above in Eq. (8.30).

The computer reads the number P in the binary counter and then resets the counter to zero to prepare it for the next count. After performing computations and filtering, the computer generates a signal for the display to indicate the vehicle speed. Although it is possible to display vehicle speed numerically, it is normally desirable to present speed using either a galvanometer-type analog display or a display that simulates an analog scale/pointer (e.g., FPD). A digital display can be directly driven by the computer. Either mph or kph may be selected. If an analog display is used, a D/A converter must drive the display. Both mph and kph usually are calibrated on an analog scale. A FPD is now commonly used for displaying such measurements. This display has sufficient flexibility and detailed resolution that graphic data or electronic maps can be shown to the driver as explained earlier.

The data required for such displays can, for example, be transmitted via an IVN link (e.g., CAN, as explained in Chapter 9) between the various onboard electronics systems. In the next chapter, we discuss high-speed intermodule digital communication systems.

TRIP INFORMATION FUNCTION OF THE SYSTEM

One of the functions of the electronic instrumentation computer is the trip information subsystem. This system has a number of interesting functions and can display many useful pieces of information, including the following:

1. Present fuel economy
2. Average fuel economy
3. Average speed
4. Present vehicle location (relative to total trip distance)
5. Total elapsed trip time
6. Fuel remaining
7. Miles to empty fuel tank
8. Estimated time of arrival

The trip information computer analyzes fuel flow, vehicle speed, and fuel tank quantities and then calculates information such as miles to empty, average fuel economy, and estimated arrival time. In the present chapter, English units are used because in the United States, these are the preferred units. The trip information subsystem in present-day vehicles is an extension of what was once a stand-alone computer-based system that could perform the functions listed above and that had a means (in some vehicles) for input of information by the driver (e.g., via a special keyboard). The following discussion of an exemplary trip information function is based on the assumption that the vehicle has a GPS-based navigation system.

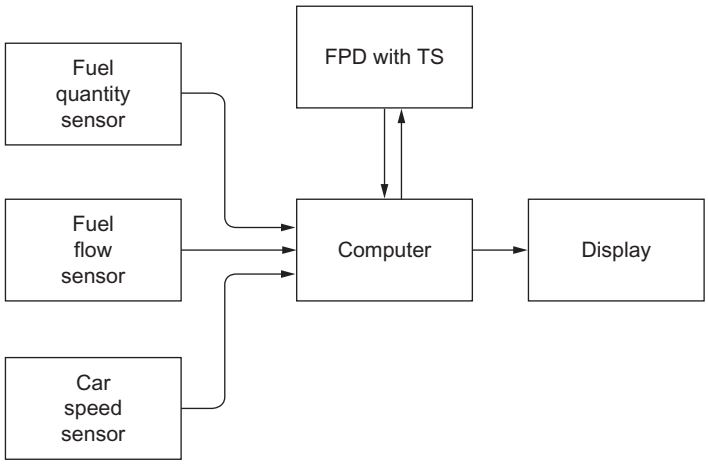


FIG. 8.39 Trip information system block diagram.

A block diagram of this system is shown in Fig. 8.39. Not shown in the block diagram are IVN, MUX, DEMUX, and A/D converter components, which are normally part of a computer-based instrument.

This system is assumed to be implemented as a set of special functions of the main automotive instrumentation system.

The vehicle inputs to this system come from the three sensors that measure the following variables:

1. Quantity of fuel remaining in the tank
2. Instantaneous fuel flow rate
3. Vehicle speed

Other inputs that are obtained by the computer from the navigation system include the following:

1. Starting location
2. Present position
3. Travel route along the electronic map

The driver enters inputs to the system through the TS capable FPD. At the beginning of a trip, the driver initializes the system and enters the destination and fuel cost. At any time during the trip, the driver can use the FPD to ask for information to be displayed.

The system computes a particular trip parameter from the input data. For example, instantaneous fuel economy in miles per gallon (MPG) can be found by computing

$$\text{MPG} = S/\dot{F}$$

where S is the speed in mph and $\dot{F}$ is the fuel consumption rate in gallons per hour.

Of course, this computation varies markedly as operating conditions vary. At a steady cruising speed along a level highway with a constant wind, fuel economy is essentially constant. If the driver

then depresses the accelerator (e.g., to pass traffic), the fuel consumption rate temporarily increases faster than speed, and MPG is reduced for that time. Various averages can be computed such that instant fuel economy, short-term average fuel economy, or long-term average fuel economy can be displayed.

Another important trip parameter that this system can display is the miles to empty fuel tank, D. This can be found by calculating

$$D = \text{MPG} \cdot Q \tag{8.31}$$

where Q is the quantity of fuel remaining in gallons. Since D depends on MPG, it also changes as operating conditions change (e.g., during heavy acceleration). In such cases, the calculation of miles to empty based on the above simple equation is grossly incorrect. The estimate of D for transient driving conditions (e.g., urban driving) can be improved relative to Eq. (8.31) by using short-term time-average values of MPG. However, this calculation gives a relatively correct estimate of the miles to empty for steady cruise along a highway in which operating conditions are mostly constant.

Still another pair of parameters that can be calculated and displayed by this system is distance to destination, D_d, and estimated time of arrival, ETA. The calculation of D_d is accomplished by summing the distances along each segment of the route being followed using the data from the digital map. The digital map data have been explained earlier in this chapter to have vector positions at a sufficient number of points along the route, which we denote as $\bar{P}(n)$ to be able to plot the route. For straight roads only, the beginning and end vector positions are necessary. For curved portions of the route, a number of vector positions are required. The distance between any two consecutive vector points, for example, $\bar{P}(m)$ and $\bar{P}(m+1)$ is given by $D_{m,n}$, which is the L_2 norm of the vector difference between these points:

$$D_{m,n} = \|\bar{P}(m+1) - \bar{P}(m)\|$$

Assume that the vehicle present position $\bar{P}_v$ is between these two vectors. The total distance to the destination $\mathrm{D_d}$ from that position is given by

$$D_d = \|\bar{P}(m+1) - P_v\| + \sum_{n=m+1}^{N} \|\bar{P}(n+1) - \bar{P}(n)\| \tag{8.33}$$

where $\bar{P}(N) =$ destination vector position.

In one example algorithm, the ETA can be found by computing the sum of time intervals along all remaining trip segments using published speed limits to estimate vehicle speed:

$$\text{ETA} = T_1 + \frac{\|\bar{P}(m+1) - \bar{P}_v\|}{S_a} + \sum_{n=m+1}^{N} \frac{\|\bar{P}(n+1) - \bar{P}(n)\|}{S_{Ln}} \tag{8.34}$$

where $S_a =$ actual present speed, $S_{Ln} =$ speed limit along segment from n to $n+1$, and $T_1 =$ present time.

The actual speed along the trip is influenced by many factors including driver-selected speed, traffic congestion, and road construction. Of course, drivers do not always necessarily maintain speed limits. It is possible to have adaptive algorithms that adjust ETA to trends in vehicle speed, particularly on segments of the routes that are on highways and along which drivers might travel somewhat above the local speed limit. Other adaptive algorithms are possible for navigation systems that obtain traffic and road construction data from a communication infrastructure as described in Chapter 9. Such adaptive algorithms can replace S_{Ln} in ETA calculations with a better estimate of actual speed.

The average fuel cost per mile (at any point on any given trip) C_{av} can be found by calculating

$$C = (D_P/\text{MPG}) \cdot \text{fuel cost per gallon}$$

where D_p = distance traveled from the start to the present position.

For a traditional vehicle that lacked GPS and associated navigation capability, the computer could roughly estimate the distance traveled to the present location D_P by subtracting the start mileage, D_1 (obtained from the odometer reading when the trip computer was initialized by the driver), from the odometer mileage, and use this D_P to calculate C_{av}. However, the position information D_P and trip starting location are found in vehicles that are equipped with GPS/digital map navigation systems more accurately than with the traditional trip computer. The other variables discussed above can be computed using the formulas given and where applicable, with navigation data.

In summary, vehicular electronic instrumentation displays important variables and parameters that are important for safe vehicle operation and for advising the driver of the state of each system whenever certain variables are out of limits. In addition, the measurement instrumentation obtains data required both for successful operation of the various electronic systems and for diagnosing problems with the various vehicular systems.

CHAPTER

VEHICLE COMMUNICATIONS

9

CHAPTER OUTLINE

IVN 462
CAN 464
CAN Bus Transceiver 467
CAN Electronic Circuits 469
Arbitration on CAN 472
Local Interconnect Network 472
FlexRay IVN 474
FlexRay Transceiver Circuit 477
MOST IVN 478
Vehicle to Infrastructure Communication 481
Vehicle-to-Cellular Infrastructure 482
Quadrature Phase Shifter and Phase Modulation (QPSR) 487
Short-Range Wireless Communications 488
Satellite Vehicle Communication 490
GPS Navigation 493
The GPS System Structure 500
Safety Aspects of Vehicle-to-Infrastructure Communication 503

Communication with vehicles (while moving) began with the introduction of AM radio receivers in the 1930s. In this same general era, two-way radio communication via radio was used by law enforcement agencies. The evolution of civilian two-way radio communications advanced relatively slowly until the introduction of cellular phones. Initially, they were often referred to as bag phones since they were relatively bulky and packages in bags. The early cellular phones (now just called cell phones) had telephone-type handsets. It is widely known that the advances in cell phone technology have been very rapid. It is also well known that cell phone communication requires an infrastructure of multiple cell phone transceiver stations (called cell towers). Other wireless communications between moving vehicles and multiple fixed or moving (i.e., satellite) transceivers are developing and have been developed. In this chapter, we refer to such communication as vehicle to infrastructure (often abbreviated as (V2I)).

Similar communication between sets of moving vehicles will be termed vehicle to vehicle and abbreviated (V2V). There are many applications to both V2I and V2V including vehicle safety and vehicle monitoring (e.g., truck fleet monitoring), which are discussed below in this chapter.

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-12-810434-7.00009-0

Copyright © 2017 Elsevier Inc. All rights reserved.

There is another very important application of communication technology, however, within any given vehicle (which we term in-vehicle communication (IVC)). This communication has multiple applications including exchange of data or status between vehicle systems or subsystems, diagnosis on board the vehicle of problems that have developed with the various electronically controlled systems, and optimization of overall vehicle performance.

To this point in the book, the various automotive electronic systems have been discussed as stand-alone systems. It has been presumed that each system or subsystem was configured with its own electronic control subsystem. In certain high-end vehicles, there are approximately 100 microcontrollers/microprocessors performing various tasks. The internal vehicle communication system forms a network within the vehicle leading to the possibility of shared computational capabilities and in extreme case (not yet commercially available) a single computational unit that could, in principle, perform all necessary computations. Moreover, it could optimize the interaction of certain subsets of individual vehicle electronic systems and improve overall vehicle performance. Such a network is termed in-vehicle network (IVN).

The notion of centralized versus distributed computing was an issue in aerospace vehicles (particularly fighter jets) in the days in which digital computation was first employed for control or instrumentation. However, in the case of distributed computation, any hardware failure or software glitch often would affect only one system, whereas, in the case of centralized computing, the entire electronic systems were affected by failures. Of course, in aerospace application, protection of the vehicle against a serious failure of a component was mitigated by system redundancy (normally in triplicate). This issue is also present in the design choice of centralized versus distributed computing capability for land vehicles.

IVN

Nonetheless, the benefits of networking land vehicle electronic systems far outweigh any potential benefits of isolated individual systems. The topology of a hypothetical system is depicted in Fig. 9.1. This configuration does not represent that of any particular vehicle model, but depicts the type of interconnection possible with an IVN. Most of the individual systems that are depicted have been discussed elsewhere in this book except for the comfort and entertainment systems and the system manager. The latter (when present in an IVN) provides control over the digital data link that provides the in-vehicle communication pathway, which is explained later in this section.

This system requires a set of momentary contact switches for inputs or a keyboard (KB) or a similar input device (e.g., touch pad) for operator control. The driver can, for example, select to display the entertainment system operation. This display mode permits the driver to select radio, tape, or CD and to tune the radio to the desired station and set the volume. In vehicle diagnostic mode, the flat-panel display can be configured to display the parameters required by the service technician for performing a diagnosis of any onboard electronic system (see Chapter 11).

In Fig. 9.1, several electronic subsystems are connected by the digital data link. Tying systems together this way has great potential performance benefits for the vehicle. Each automotive subsystem has its own primary variables, which are obtained through measurements via sensors. A primary variable in one subsystem might be a secondary variable in another system. It might not be cost-effective to provide a sensor for a secondary variable to achieve the best possible performance in a stand-alone subsystem. However, if measurement data can be shared via the digital data link, then the secondary

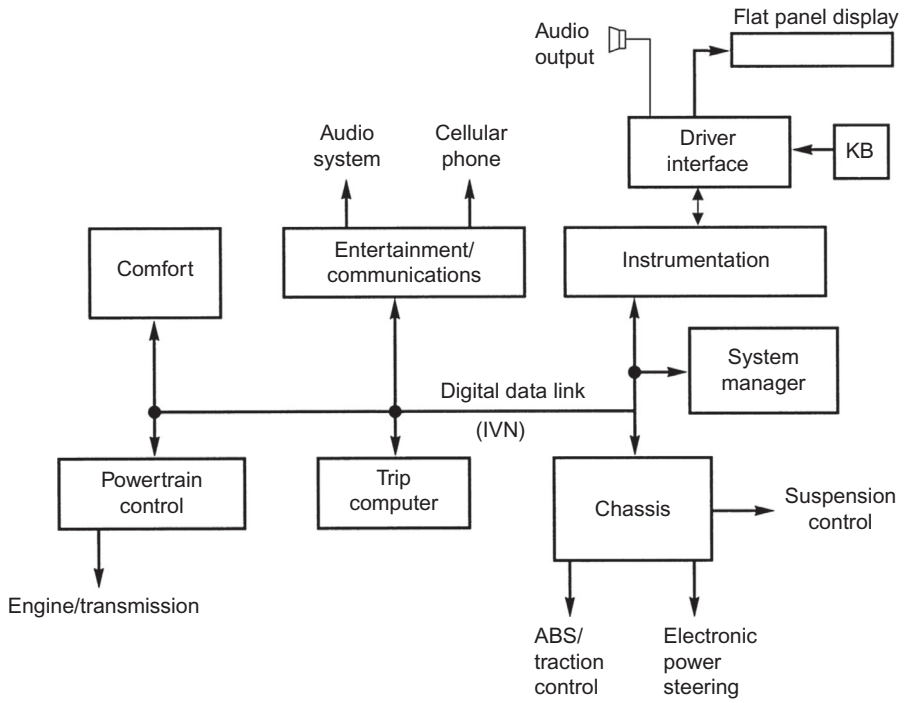


FIG. 9.1 Representative in-vehicle communication topology.

measurement is potentially available for use in optimizing performance. Furthermore, redundant sensors for measuring primary variables can be eliminated by an integrated electronics system for the vehicle. For example, wheel speed measurements are primary variables for ABS systems and are also useful in electronic transmission control, vehicle speed display, and others and are useful for such applications as trip computer or fuel range.

Any networking for IVC involves digital communication systems having both a specific hardware requirement and the software for controlling this communication. The physical link connecting vehicular electronic systems can have many forms. This link is termed a bus or, often, a medium. In addition to the physical bus, there is also a specific set of requirements for the format of any message that is sent along the bus. This format and all specifications for it are termed the protocol for the system.

There are several IVN configurations and protocols for application in land vehicles. Each has a very precise set of specifications for the hardware and message format. In this chapter, a number of IVN systems are discussed and compared with respect to performance and cost.

There are several components that are common to all types of IVNs. The bus for each is either a wire or a pair of wires or optical cable that passes through the vehicle close enough to all relevant electronic systems/subsystems to connect to them. In addition, there must be an interface between the bus and each device capable of receiving or transmitting messages to the device. In certain cases, there is a separate digital bus controller that determines how each device is enabled for sending a message.

There are two major strategies for determining individual device access to the bus for sending messages: (1) time-division multiplexing in which each device has a specific time interval in a complete cycle or (2) event-driven access in which a device transmits a message when a specific event occurs. In this latter case, it will frequently occur that two or more devices transmit simultaneously such that their messages overlap. When this occurs, the bus controller (if one exists) determines priority and sends commands for the devices to repeat sending the event-driven messages.

In the absence of a bus controller, some form of network arbitration is required for determining the priority of the use of the IVN whenever there is conflict between subsystems for its use. This arbitration feature can be handled by the system manager subsystem (see Fig. 9.1) if one exists in the system or is automatic in other cases.

One of the characteristics of an IVN is that messages are sent in a serial mode. Each message is digital and consists of multiple fields. One field is an identifier, another field contains data, and the remaining fields are IVN-specific and are discussed below for each IVN presented in this chapter.

CAN

We begin with an IVN that is called controller area network (CAN). This IVN was developed for vehicle use in the 1980s and has broad application in automotive systems including power train, suspension, and braking systems among other vehicle model-specific systems.

Essentially, CAN IVN provides a sophisticated communication system between various subsystems. Among the issues of importance for such a communication system are the protocol and message format. It is highly advantageous to have a standard protocol for all automobiles. The Society of Automotive Engineers (SAE) has developed a standard specification for CAN. The CAN IVN operates asynchronously at a data rate of up to 1 mbps for a distance of 40 m.

The basic message structure is derived assuming that the majority of data on the link are regularly sent. This means that the content of each message is known (only the actual data vary). The standards and specifications for the CAN network are given in a document published by SAE, which is given the designation (in the latest version at the time of this writing) *J-2284-3*.

In the CAN concept, each electronic subsystem that is connected to the CAN (called ECU in J-2284-3) incorporates communication hardware and software, permitting it to function as a communication module referred to as a gateway. CAN is based on the so-called broadcast communication mechanism in which communication is achieved by the sending gateway (which we call a transceiver) transmitting messages over the network (e.g., wire interconnect). Each message has a specific format (protocol) that includes a message identifier. The identifier defines the content of the message, its priority, and is unique within the network. In addition to the data and identifier, each message includes error-checking bits (e.g., cyclic redundancy check CRC) and beginning and end-of-file bits. In the most recent version of J-2284-3, the message identifier is 29 bits.

The CAN communication system has great flexibility, permitting new subsystems to be added to an existing system without modification, provided the new additions are all receivers. Each system connected to CAN may be upgraded with new hardware and software at any time with equipment that was not available at the time the car left the manufacturing plant or even when it left the dealer. Essentially, the CAN concept with its open architecture frees the development of new telematics applications from the somewhat lengthy development cycle of a typical automobile model with the help of AUTOSAR. Furthermore, it offers the potential for the aftermarket addition of new subsystems.

The SAE *J-2284-3* standard is a recommended practice document (one of many published by SAE) that defines the CAN protocol in terms of its physical layer and portions of the data link layer, message format, etc. It primarily focuses on a minimum standard level of performance from the CAN IVN implementation by any automotive manufacturer. All of the ECU's associated media are to be designed to meet component level requirements. By meeting component level requirements, the system level performance requirements are assured.

Physically, the CAN consists of a twisted pair of wires CAN_H and CAN_L whose voltages are specified by a pair of states: (1) dominant state and (2) recessive state. The CAN_H bus wire is fixed to a mean voltage level during the recessive state and is driven positive during the dominant bit state. The CAN_L bus wire is fixed to a mean voltage level during the recessive state and driven in the negative voltage direction during dominant bit state.

The recessive state is represented by an inactive state differential voltage (V_{diff}) between CAN_H and CAN_L that is approximately 0. The recessive state represents a logic 1-bit value. The dominant state is represented by a differential voltage between CAN_H and CAN_L greater than a minimum threshold value. The dominant state overwrites the recessive state and represents a logic 0-bit value. These voltages are depicted in Fig. 9.2.

Fig. 9.2A shows the individual voltages on the two lines. The differential voltage is depicted in Fig. 9.2C. The rejection of EMI is shown in Fig. 9.2B in which the differential voltage V_{diff} is unaffected by EMI, which changes both CAN_H and CAN_L by the same amount.

The SAE J-2284-3 standard gives a number of definitions of terms by which the CAN IVN can be understood. The term "media" refers to the physical structure/configuration that conveys the electrical transmission between ECUs on the network and as stated above is an unshielded twisted pair of wires. The term "physical layer" refers to the transmission of a bit stream over the physical media and deals with electrical, mechanical, functional, and procedural characteristics to access the physical media. The term "protocol" refers to a set of conventions for the exchange of information between ECUs on the CAN. It includes the specification of frame administration, frame transfer, and the physical layer. In this context, the frame is the formal arrangement of the sequence of bits over a specified time interval that constitutes the message.

The message format includes a message identifier (formerly 11 bits but later 29 bits). The actual encoding of the identifier is manufacturer-specific. The identifier defines the content of the message and its priority. The message also includes a field for the information being sent in the form of 8 data bytes. A set of error-checking bits is also included that might be of the form of "check sum" of the bits.

The CAN is capable of supporting data transfer between ECUs from a minimum of two to a maximum of 24. The topology of the CAN is depicted in Fig. 9.3, which illustrates a CAN with N ECUs.

The configuration of the CAN shown in Fig. 9.3 includes a connection to an off-board diagnostics tool ($ECU_{N\text{-}2}$) via a data link connector (DLC). Each ECU is connected to the CAN via a stub whose length (L_1) must satisfy

$$0 < L_1 \leq 1\,\text{m}$$

The stub length to the DLC L_2 has the same requirement as L_1. The off-board stub length (L_3) must satisfy

$$0 < L_3 \leq 5\,\text{m}$$

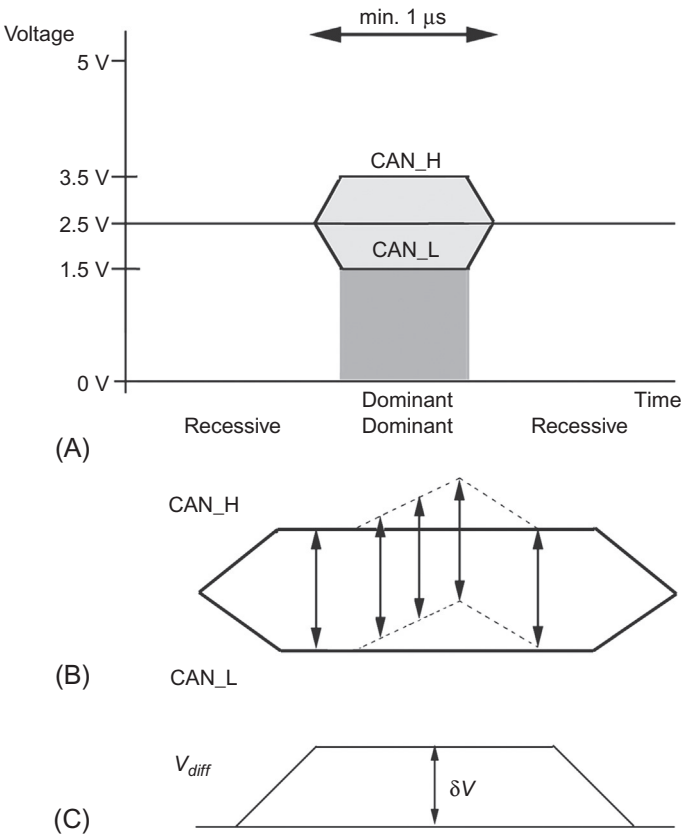


FIG. 9.2 CAN voltage levels. (A) CAN voltages, (B) individual CAN voltages with external interference, and (C) CAN differential voltage.

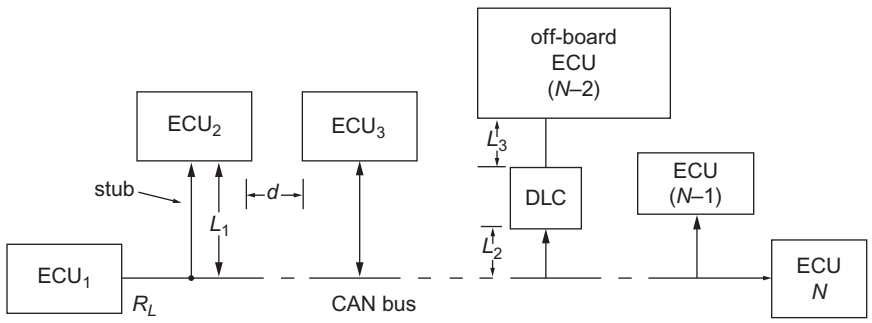


FIG. 9.3 CAN bus architecture.

The distance between and two ECUs including cable stubs (d) must satisfy

$$0.1 \leq d \leq 33\text{m}$$

The CAN must be terminated at either end with a resistance R_L that has tolerance range

$$118 \leq R_L \leq 132\Omega$$

The nominal value for R_L is 120 Ω. This resistance is connected between CAN_H and CAN_L wires. In addition, each ECU must present no more than 100 pF capacitance to ground and no more than 50 pF differential.

The physical media parameters for an unshielded twisted pair are also given in SAE J-2284-3. The characteristic impedance of the twisted pair z_o must satisfy

$$108 \leq z_0 \leq 132\Omega$$

The resistance/unit length R_l must be less than 0.070 Ω/m. The propagation delay for the media must be less than 5.5 ns/m. The basic CAN bit time requirements are a critical specification. In SAE J-2284-3, the bit time (t_{bit}) must satisfy $1990 \leq t_{\text{bit}} \leq 2010$ ns. A further constraint is that the nominal bit time must be a programmable multiple of the system clock period. For precise timing details, the reader is referred to SAE J-2284-3.

The SAE J-2294-3 also has specific requirements concerning electromagnetic compatibility. The electromagnetic radiation from the CAN and susceptibility to interference from other CAN electronic/electrical systems is specified in the SAE J-2284-3 standard. It is typical of SAE standard documents (including J-2284-3) that they evolve over time to accommodate technology advances and changes resulting, for example, from government-mandated regulatory changes. Regardless of such evolution, the basic concepts for the CAN network will remain the same.

The interface electronic block diagram is depicted in Fig. 9.4. In this figure, the CAN transceiver and controller are commercially available chips. The microcontroller refers to the ECU_3 depicted in Fig. 9.3 and controls the vehicular electronic system connected to the CAN bus as shown above. Also shown in Fig. 9.4 are the 120 Ω bus termination resistors (denoted R_{T}).

CAN BUS TRANSCEIVER

The CAN bus transceiver is a commercially available integrated circuit that handles the exchange of data between modules that are connected to the bus. It has a pair of terminals with one connected to the CAN_H line and the other to the CAN_L line. It is capable of both receiving and transmitting data along the CAN bus. For an understanding of its operation, it is helpful to refer to Fig. 9.5, which depicts the CAN_H and CAN_L voltage waveforms associated with a pair of modules S_1 and S_2.

In this figure, the time axis shows that successive bit times are alternately recessive or dominant. For convenience, each bit time is depicted as 1 μs duration. The graphs labeled T_{xn} or R_{xn} are the logic levels for transceiver in depicting what that transceiver is sending onto the bus (i.e., T_{xn}) and receiving from the bus (i.e., R_{xn}). These two line voltages are shown together in the graph whose ordinate is labeled V_{CAN}. The voltage on CAN_H is denoted as V_{CH} and is given by the solid line, and on CAN_L, the voltage is denoted as V_{CL} and is given by the dashed line. During the recessive time bits, the two voltages are approximately equal:

recessive $V_{CH} \simeq V_{CL} \simeq V_{CC}/2$

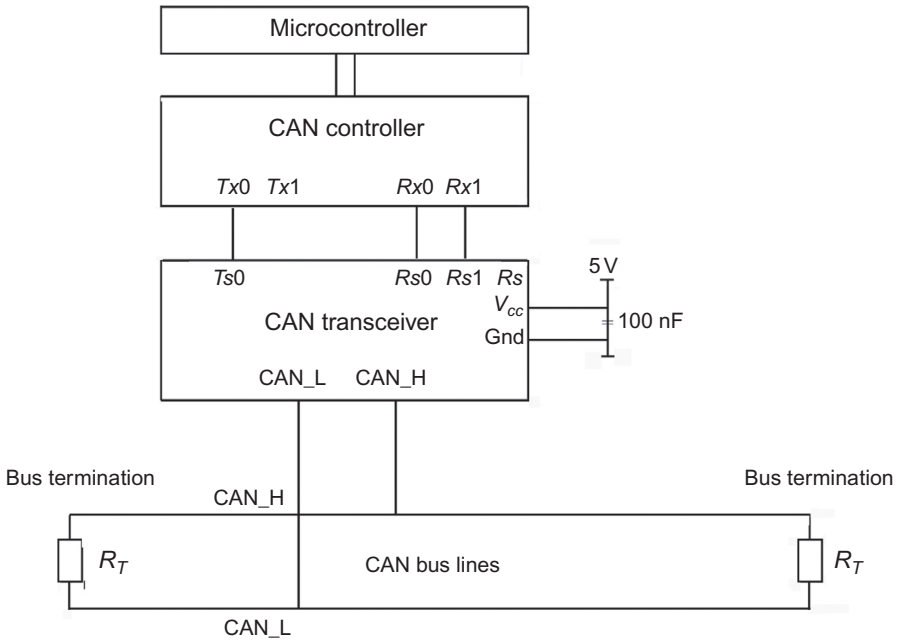


FIG. 9.4 CAN interface block diagram.

During the dominant time bit when a device is transmitting, the two voltages are given by

dominant

$$V_{CH} = \frac{V_{CC}}{2} + \frac{\delta V}{2}$$

$$C_{CL} = \frac{V_{CC}}{2} - \frac{\delta V}{2}$$

The differential voltage (V_{diff}) during the active dominant time bit is given by

$$V_{diff} = V_{CH} - V_{CL}$$
$$= \delta V$$

An international standard calls for the minimum value of δV to be 1.5 V. During recessive bits, both T_{xn} and R_{xn} are logic high. For the two modules whose voltages are depicted in Fig. 9.5, the receiver logical states (R_{x1} and R_{x2}) during the first dominant time bit are both low. Each transceiver generates the corresponding electrical signal in response to the V_{CH} and V_{CL} during dominant time bits.

Control of the transceiver is done by a microprocessor-based subsystem (or IC). There are commercially available integrated circuit CAN bus control devices. The IC manufacturer normally also has available development system that permits the user to program it to fit the IVN requirements. However, it is also possible to implement this control as a part of the vehicle system controller.

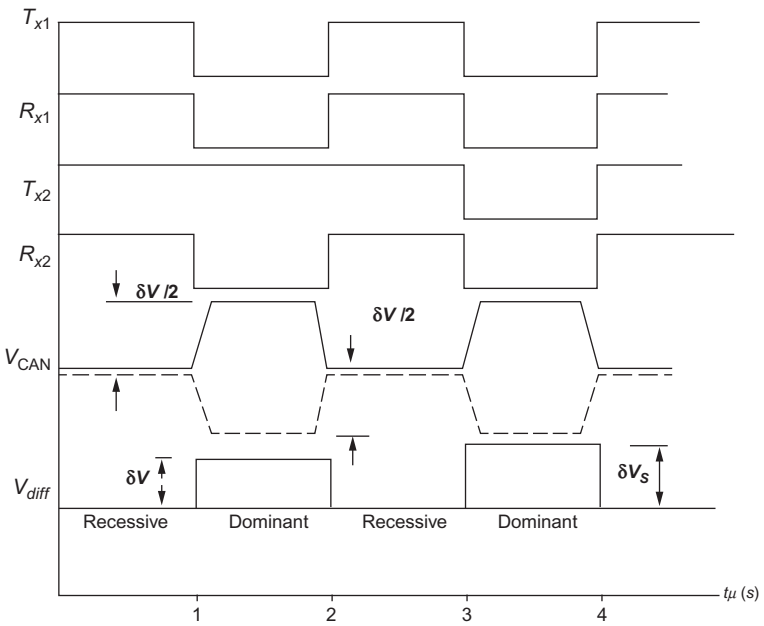


FIG. 9.5 CAN voltage waveforms.

CAN ELECTRONIC CIRCUITS

Fig. 9.6 depicts a representative circuit for a CAN transceiver. The upper portion of Fig. 9.6 containing two FETS is representative of the transmit circuitry, and the lower portion of Fig. 9.6 with an operational amplifier (op-amp) is representative of the receiver circuitry. In an ideal case during the recessive state, the voltages on both CAN wires would be $V_{cc}/2$ with $V_{diff}=0$. In the dominant state, $V_{diff}=\delta V$. In this case, the dominant state would correspond to $V_{diff}>0$. However, in practice with multiple nodes on a CAN network, there can be times when a small difference V_{diff} exists in a recessive state due to small fluctuations and small but nonzero interference/noise.

In practical CAN applications, it is helpful to require V_{diff} to exceed a threshold V_{th} to correctly identify dominant state and to minimize the probability of errors in the two states. In the representative receiver circuit of Fig. 9.6, this goal is achieved with the use of a zener (Z) diode in the output of an op-amp comparator circuit connected to CAN_H and CAN_L. It can be shown with reference to the discussion of operational amplifier circuits from Chapter 2 that voltage v_o is given by

$$v_o = V_{CH}\left(\frac{R_f}{R_s}+1\right)\left(\frac{R_s}{R_s+R_1}\right)-V_{CL}\frac{R_f}{R_s}$$

with R_1 chosen to be

$$R_1 = R_s^2/R_f$$

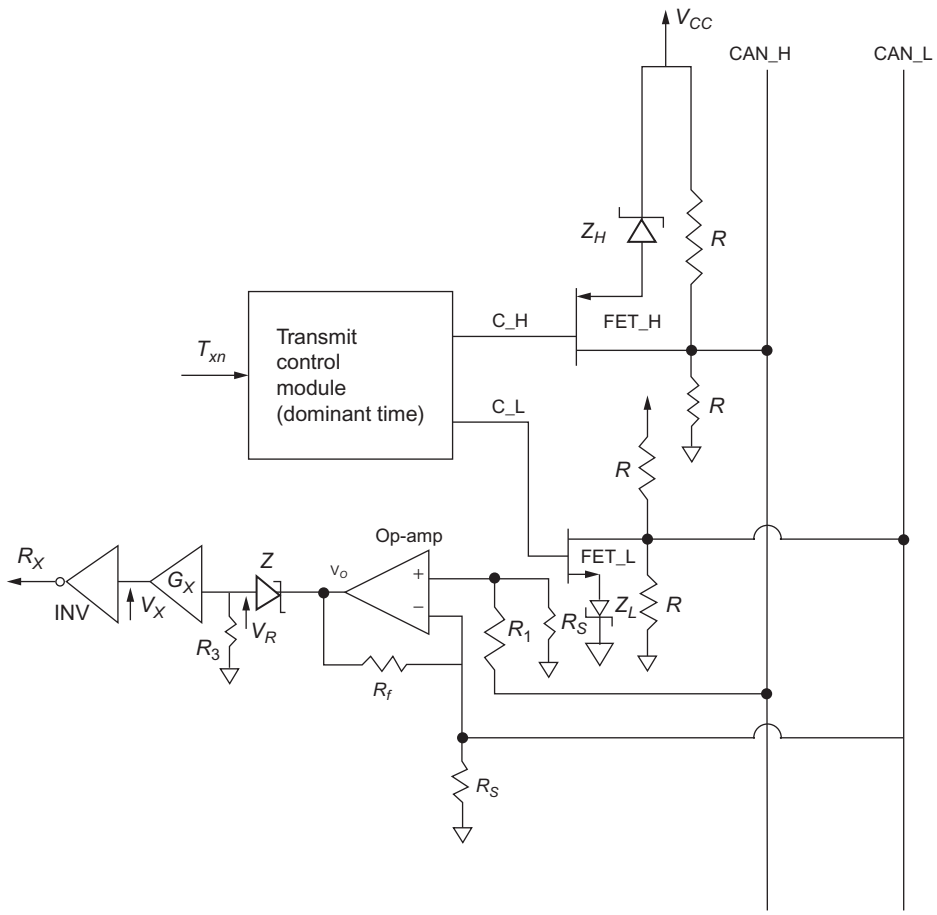


FIG. 9.6 Illustrative CAN transceiver circuitry.

v_o is given by

$$v_o = G V_{diff}$$

where

$$G = R_f / R_s$$

The voltage v_R across the resistance R_3 connected between the zener anode and ground is given by

$$\begin{aligned} v_R &= 0 && v_o < V_Z \\ &= G V_{diff} - V_Z && v_o \geq V_Z \end{aligned}$$

where $V_Z =$ zener voltage.

The threshold voltage for correctly detecting dominant state and not incorrectly detecting recessive state is determined by the parameters G and V_Z. For example, if the threshold voltage was chosen to be

1/2 δV, V_{diff} would be at $V_{th} = 1/2\ \delta V$, which would occur during the dominant state, and V_R would be 0 at $V_{diff} < V_{th}$ and would transition as V_{diff} across the threshold such that

$$\begin{aligned} G V_{th} &= V_Z \\ &= G\delta V/2 \end{aligned}$$

The parameters R_f and R_S should be selected such that

$$G = \frac{R_f}{R_S} = \frac{2V_z}{\delta V}$$

At the time during the transition from recessive to dominant occurs, the final value for V_R in the dominant state is denoted as V_{RD} which is given below:

$$V_{RD} = G\delta V - V_Z$$

The gain G_x of the amplifier that is connected to resistance R_3 chosen such that V_x corresponds to logic 1 where

$$V_x = G_x V_{RD} \Rightarrow \text{logic } 1$$

However, since the dominant state corresponds to logic 0 in CAN protocol, the output V_x must be inverted logically as depicted in Fig. 9.6 by the inverter (Inv).

The transmit portion of the circuit consists of a control section with two output control voltages C_H and C_L each connected to an FET. It should be noted that the circuit symbols for the FETs are not standard. It is left as an exercise for the interested reader to redraw FET-H and FET_L using the symbology and the theory or enhancement mode FETs to determine the channel polarity, the source and drain terminals for these two transistors based on the following description of their operation. During the recessive state, the FET transistors are in cutoff (i.e., virtually open circuit). The CAN_H and CAN_L lines are both held at voltage $V_{CC}/2$ by the pair of voltage dividers (i.e., series-connected resistances R connected between V_{CC} and ground). During dominant state, the control voltages drive the FET transistors into saturation such that

$$\left.\begin{aligned} V_{CH} &= V_{CC} - V_{ZH} \\ V_{CL} &= V_{ZL} \end{aligned}\right\} \text{Dominant state}$$

where

$$\begin{aligned} V_{ZH} &= \text{zener voltage of } Z_H \\ V_{ZL} &= \text{zener voltage of } Z_L \end{aligned}$$

The zener voltages satisfy the requirements for the voltages on CAN_H and CAN_L during the dominant state such that $V_{diff} = \delta V$ meets the minimum δV requirement.

One of the significant issues for a CAN IVN, especially under event-driven configurations, is the requirement to deal with simultaneous transmission from two or more vehicle systems. The process of handling such an occurrence is known as arbitration. The electrical result of simultaneous transmission by a pair of CAN bus transceivers can be seen with reference to Fig. 9.5. During the bit time from T_3 to T_4, both T_{x1} and T_{x2} are active. The differential voltage during this interval is denoted as δV_S in Fig. 9.5. With reference to the sending circuitry of Fig. 9.6, it can be seen that the output transitions of both transceivers are active, thus acting in parallel resulting $\delta V_S > \delta V$. Although δV_S is only slightly larger than δV, this result can be used during arbitration.

ARBITRATION ON CAN

Arbitration on the CAN IVN does not require a master node or bus regulator. Rather, arbitration is accomplished automatically during the identifier field of a frame. Each node on the CAN, when attempting to transmit data, receives the bits on the bus during the ID portion of a cycle. The transmission of data on the bus is only enabled if the bits on the bus during ID exactly match the device's ID bits. Whenever two or more nodes are attempting to transmit, the bus effectively performs the AND operation on the bits.

During transmission of ID bits, a dominant bit (logic 0) overwrites a recessive bit (logic 1). Whenever two nodes attempt to transmit, each node monitors the bus state. At any bit time in the sequence of ID bits transmitted, a node that detects a mismatch between its transmitted bus and the bus bit, it ceases transmission. Since logic 0 is dominant, the device with the lowest binary identifier effectively has the highest priority.

LOCAL INTERCONNECT NETWORK (LIN)

The CAN IVN has been widely used in automobiles for connecting systems such as power train, suspension, and steering. However, there are vehicular applications requiring far less capability (e.g., bandwidth) and are less costly than CAN. One of these is the local interconnect network (LIN) discussed next.

The LIN IVN is the least costly of the networking systems available for vehicle application and operates at data rates that are much smaller than for the CAN IVN and, in fact, provides a complement to an existing CAN IVN. However, its applications require far less speed and include such systems as door lock, window operation, engine cooling fan, HVAC motor control, light switches, interior lighting, seat position motor control, wiper control, interface to radio, navigation, and phone. It is a synchronous network with 20 kbps maximum rate and operates over a single wire with a serial communication protocol based upon the UART protocol. It operates at a nominal voltage of 12 V, but the power supply voltage (e.g., vehicle electrical bus) can be between 7 and 18 V.

The LIN IVN consists of a master node with multiple slave nodes. The master controls the LIN bus access synchronously so there are no simultaneous slave accesses (i.e., collisions) and no arbitration is required. Furthermore, all latency times are fixed. The LIN protocol has recessive and dominant states similar to those in the CAN protocol.

As in the case of the CAN IVN discussed above, each system/subsystem connected to LIN requires a transceiver. There are many commercially available transceiver IC's that function together with or include a microprocessor as a part of a node. Fig. 9.7 is a composite block diagram/circuit of a master node and a slave node as indicated by the dashed connection labeled slave. In the slave node, the master pull-up is not present. In the master node, the dashed connection to the vehicle system is not necessarily present and depends on the vehicle electronic system's configuration.

The components inside the dashed-line box constitute the transceiver portion of this composite block diagram/circuit. Included in the receiver portion of the transceiver components is a comparator (comp) that has been explained in Chapter 2. Resistors labeled R_c are a voltage divider that places a specific reference voltage of $V_{sup}/2$ on pin 1 of the comp. The LIN bus is connected to pin 2, and the output switches state as the LIN bus voltage crosses the reference voltage yielding an output voltage

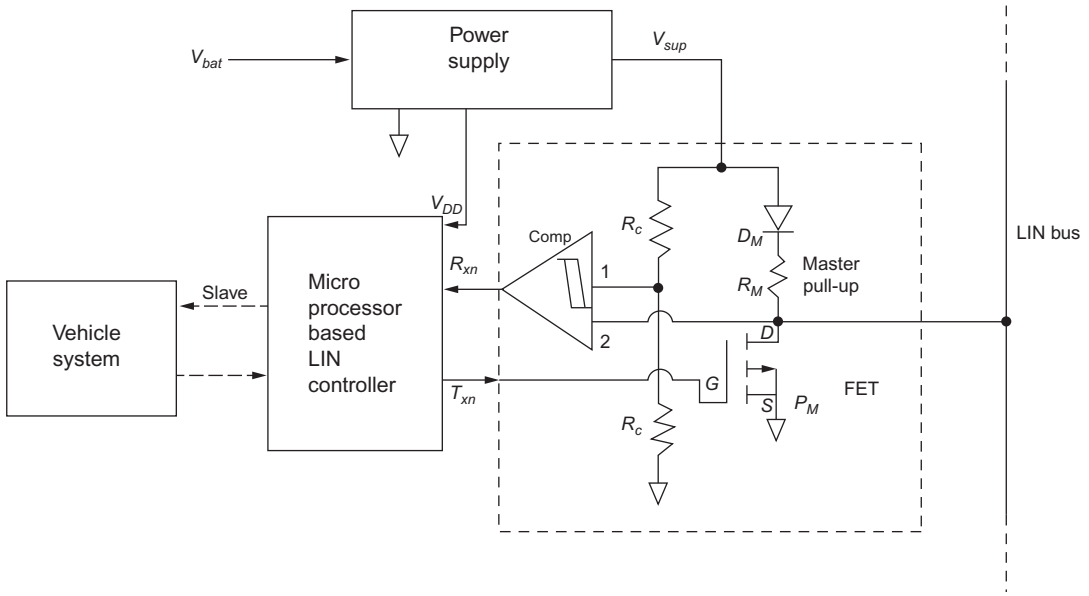


FIG. 9.7 Master and slave nodes block diagram.

corresponding to logic levels for either dominant (low) or recessive (high) states. This output is the received signal from the LIN bus that is connected to the LIN bus controller. The comp output is R_{xn} and is sent to the microprocessor-based LIN controller.

In the master node, the LIN bus controller provides necessary signals to the transmitted portion of the circuit for its operation. It generates an output T_{xn} that is supplied to the gate of the FET shown in Fig. 9.7. The FET drain D is connected to the LIN bus along with a pull-up circuit consisting of diode D_M and resistor R_M. The pull-up circuit holds the LIN bus high (recessive) until the T_{xn} signal drives the transistor to saturation, thereby lowering the LIN bus (to dominant state).

A slave node configuration is also depicted in Fig. 9.7. However, for the slave node, the master pull-up circuit is not present. Furthermore, the slave node LIN bus controller is connected via a communication bus (dashed lines in Fig. 9.7) to the vehicle system being served by that node.

The vehicle LIN IVN topology is depicted in Fig. 9.8. Data are transferred across the bus in messages having a specific format but with lengths that are selectable. Message transfers from

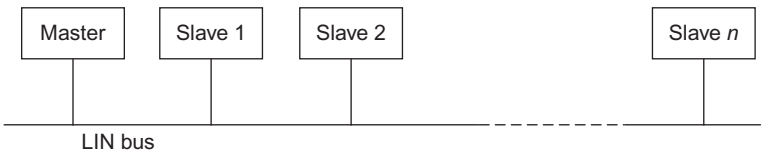


FIG. 9.8 LIN IVN topology.

master-to-slave, slave-to-master, or slave-to-slave. Message exchange begins with the master sending a so-called header consisting of (1) a synchronization break, (2) a synchronization byte, and (3) identification byte. The slaves respond with a data frame of between 2 and 8 data bytes followed by a check sum byte.

There are two bus states in the LIN protocol that are called active and sleep models. All nodes are in the active state, while data are on the bus. After a given time interval, the nodes enter the sleep mode but are triggered back to the active node by a frame known as WAKEUP. The master can issue WAKEUP from a programmed schedule. Any of the slaves can also transmit WAKEUP when the vehicle system software requires the activity, provided that the particular node has a transceiver with transmit capability (i.e., FET, R_m, and D_m) as part of the LIN system design.

There are a great many details of the LIN protocol that are beyond the scope of this book. The interested reader can obtain the LIN specifications Rev 2.2a from http://www.linsubbus.de.

FLEXRAY IVN

Another IVN is called FlexRay. This IVN has higher data rates than CAN but is more expensive to implement. It was developed early in the first decade of the 21st century by a consortium of automobile manufacturers. Its intended applications were for relatively high-performance vehicle systems needing high-speed communication, high reliability, and fault tolerance for safety (e.g., adaptive cruise control and drive-by-wire).It is a fault tolerant IVN with a data rate of 10 mbps. Not only a FlexRay bus driver can be configured for two separate data buses each consisting of a pair of twisted wires (similar to CAN), but also it can be configured for an optical bus. The twisted wires offer significant reduction in EMI susceptibility (similar to the CAN bus as explained in the section on CAN).

The FlexRay network topology can be of either a so-called multidrop topology or a star configuration. The multidrop topology is similar to CAN or LIN configurations in that the bus is connected to individual nodes with stubs or branches connecting the node to the bus.

The star topology consists of a group of nodes connected to a central active node via stubs/branches. The star configuration has reliability benefits relative to the multidrop topology since the failure of one segment allows the remaining system to continue functioning. These two topologies, in which each block representing a vehicle system has a label S, are depicted in Fig. 9.9. In the star topology, this system S_{ij} is the jth vehicle system connected to active node AN_i.

The two-wire FlexRay bus must be terminated at each end with a resistance that matches the characteristic impedance of the two-wire transmission line. Normally, each end is terminated in a resistance of 47.5 Ω.

The actual physical connection to the FlexRay bus is via a bus driver IC that provides differential transceiver capability with one pair of terminals labeled BP and BM, respectively. It also is connected to a protocol controller that is then connected to the vehicle system microcontroller.

The bus driver IC contains a number of electronic components including a transmitter, a receiver, a control module, wake-up detection, bus error detection, and modules that determine the mode of operation and send or receive data from the system for which it provides the connection to the bus. This IC has two modes of operation: normal mode and standby mode. During normal mode, data can be exchanged at high rates. During standby mode, communication is stopped, and the device has very-low-power consumption.

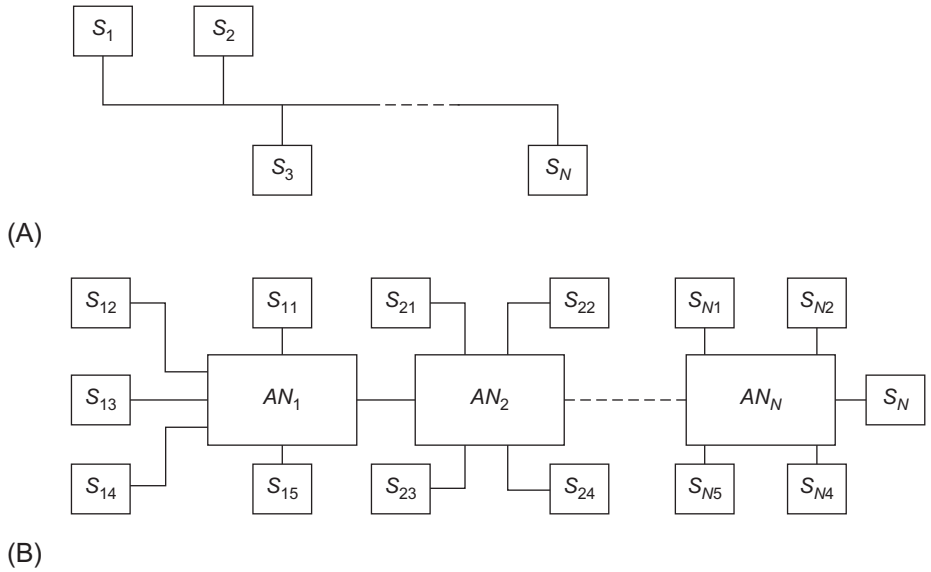


FIG. 9.9 FlexRay topology. (A) Multidrop topology and (B) star topology.

During normal mode operation, the bus voltages V_{BP} and V_{BM}, which are the voltages of wires *BP* and *BM*, respectively, can have three states: idle, bit 0, and bit 1. During an idle bit time, these voltages are

$$V_{BP} = \frac{V_{CC}}{2} = V_{BM}$$

The voltage difference V_{diff} is

$$V_{diff} = V_{BP} - V_{BM} = 0$$

During a bit 0, these voltages are

$$V_{BP} = \frac{V_{CC}}{2} - \frac{\delta V}{2}$$

$$V_{BM} = \frac{V_{CC}}{2} + \frac{\delta V}{2}$$

$$V_{diff} = -\delta V$$

and during a bit 1, they are

$$V_{BP} = \frac{V_{CC}}{2} + \frac{\delta V}{2}$$

$$V_{BM} = \frac{V_{CC}}{2} - \frac{\delta V}{2}$$

$$V_{diff} = \delta V$$

The magnitude of δV is specified in the FlexRay protocol. Fig. 9.10 depicts the three bus states in normal mode. During the standby mode, both *BP* and *BM* are essentially zero.

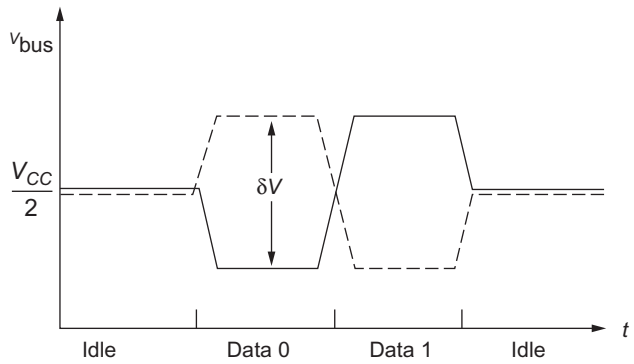


FIG. 9.10 FlexRay bus voltage states.

The communication cycle for a FlexRay IVN is the pattern for the time-domain multiplexing areas (TDMA) data format. The duration of the cycle is set during the network design process and is mostly in the range of 1–5 ms. Within each cycle, there are four main parts: (1) static segment, (2) dynamic segment, (3) symbol window, and (4) the network idle time. Each segment consists of a number of discrete-time slots. The static segment has slots assigned to specific vehicle systems that transmit data or receive data during the assigned slot. Deterministic data of this sort with a fixed latency are important for the operation of the associated vehicle system. The dynamic segment is used for exchange of event-based data and is somewhat similar in operation to the CAN IVN. Such a segment is useful in vehicles that have a number of relatively low-speed-noncritical systems and provide more time slot availability for the static segment. This dynamic segment is of a fixed predetermined length, thereby limiting the maximum number of event-driven data exchanges per cycle. The time slots in the dynamic segment are much smaller in duration than in the static segment and are termed minislots. The highest priority systems have minislots near the beginning of the dynamic segment. During the assigned minislot, the system ECU has a short interval to send data. The dynamic segment involves a mechanism similar to CAN arbitration.

The symbol window of each FlexRay cycle is used partly for system maintenance and partly to identify certain special cycles (e.g., start-up cycle). The network idle time is predetermined during the network design phase. The ECU of any system can use this time to make minor adjustments in timing as needed.

Bus error detection is accomplished via a module within the bus driver IC. There is also a so-called bus guardian that can protect the associated system from interference.

Each slot of a static or dynamic segment is divided into subsegments constituting a FlexRay frame and includes header, payload, and trailer. The header is 40 bits in length and contains 5 bits for status, 11 bits for frame ID, 7 bits for payload length, 11 bits for header CRC, and 6 bits for cycle count. The frame ID specifies the slot in which the frame is to be transmitted (static) and prioritizes event-triggered frames (dynamic). The payload length specifies the number of words (16 bit) that are transferred. The header cyclic redundancy check (CRC) is used to detect transfer errors. The cycle count advances by 1 each time a cycle starts. The payload is the actual data being transferred during the frame with a maximum of 254 bytes. The trailer has three consecutive 8-bit CRCs that are used to detect errors.

There are many more details in the FlexRay protocol that are beyond the scope of this book. These details are available from the FlexRay specifications that can be obtained online.

FLEXRAY TRANSCEIVER CIRCUIT

For the two-wire bus version of FlexRay, this transceiver must provide voltages to the two bus wires BP and BM. These voltages are tristate (i.e., having three possible levels corresponding to idle, bit 1, and bit 0) as described above. A representative circuit for the transmit portion of a FlexRay transceiver is depicted in Fig. 9.11. In this circuit, there are two, nearly identical, output driver circuits: one for BP and the other for BM wires. The voltages on the outputs of these subcircuits become the bus voltage BP and BM. Each subcircuit incorporates a p-channel MOSFET labeled *P* and an n-channel MOSFET labeled *N* in Fig. 9.11. The subscript for each label corresponds to the bus ID (i.e., *P* or *M*). During the idle state, all four FETs are in high-impedance state, and the two bus voltages are maintained at $V_{cc}/2$ by the pair of voltage dividers (i.e., resistances *R*).

The gate voltages of the FETs are determined in a transmit control block. During bit 1 state, the p-channel FET P_p is driven to a low resistance, which causes the voltage on BP line to rise to the value specified in FlexRay protocol, while the n-channel FET N_p remains at high resistance. During bit 1,

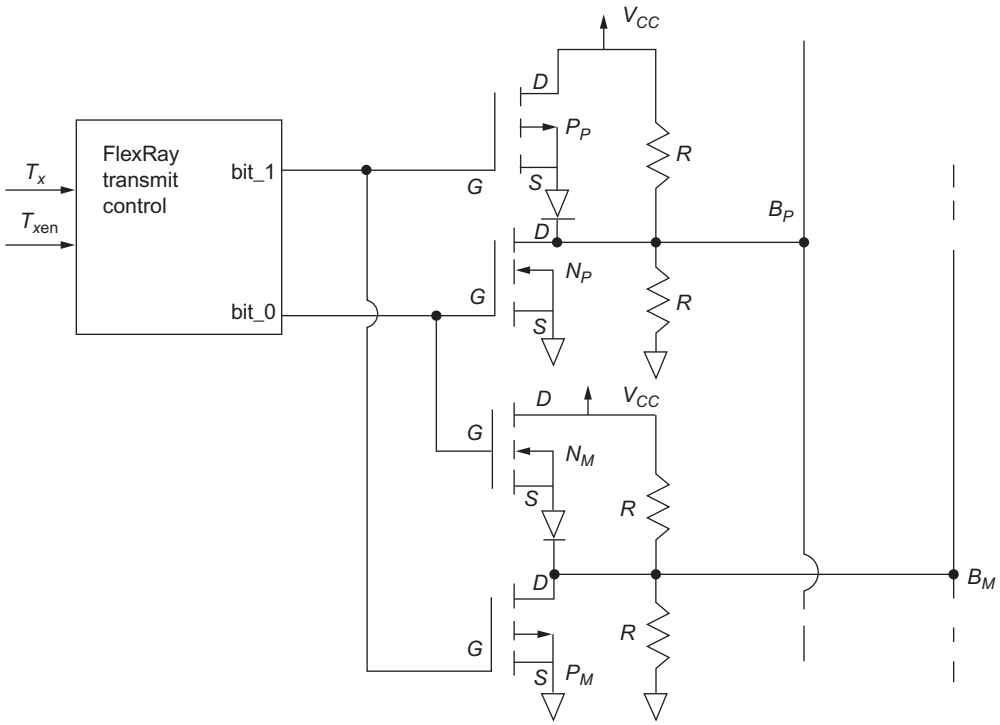


FIG. 9.11 Representative FlexRay transmit portion of a transceiver.

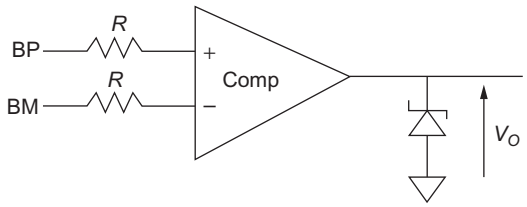


FIG. 9.12 Exemplary FlexRay transceiver receiver circuit.

the p-channel FET P_M is also driven to low resistance causing the voltage on BM to lower to the corresponding protocol requirement. During bit 0, the n-channel FET N_P is driven to low resistance causing voltage BP to drop to the required level, and n-channel FET N_M is driven low causing the voltage on BM to rise to the required value.

A representative FlexRay receiver circuit in a FlexRay transceiver is depicted in Fig. 9.12. The two wires of the FlexRay bus are connected to the inputs of a comparator circuit (comp). If required to prevent loading of the bus, a pair of series resistances R are connected between the bus wires and the comparator inputs. Examples of comparator circuits implemented with operational amplifiers are presented in Chapter 2. If the BP lead is connected to a noninverting op-amp input, the input impedance is sufficiently large that no series resistance is required. It also is possible to replace the resistance R connected to BM with a unity-gain high-input-impedance operational amplifier. In this exemplary circuit, the comparator output is connected to a zener diode cathode with the zener anode grounded.

If V_{diff} were exactly 0 V during the idle state, any polarity sensing comparator would suffice for detecting bit 1 or bit 0. However, in practice, V_{diff} can have small nonzero voltages due (e.g., to interference/noise or loading). In this case, a comparator circuit with hysteresis would suppress false detection of bit 1 or bit 0. During a bit 1 state, $V_{diff} = \delta V > 0$, and the output of the comparator would be the zener voltage (i.e., $v_o = V_z$ for bit 1). The zener diode could be selected such that V_z corresponds to logic 1. During bit 1, $V_{diff} = -\delta V$ and the comparator output forward biases the zener diode such that $v_o \simeq 0$, which corresponds to logic 0.

MOST IVN

Another IVN that is used on some vehicle models is called the media-oriented system transport (MOST). Unlike the previously discussed, IVNs, MOST is used for the interconnection of vehicle information/entertainment systems. The MOST system consists of a master and slaves (up to 64) in a logical ring topology. The preferred medium is a plastic optical fiber, although it can function with a pair of twisted wires. There are numerous variants of MOST with different data rates identified as MOST (N), where N is the data rate in megabits per second with $N = 25$, 50, 100, and 150. MOST also can operate both synchronously and asynchronously.

The MOST protocol is similar in certain respects to the previously discussed IVNs. The physical layer requires interface electronics to connect to the bus including a transceiver. It also has a network interface controller (NIC) for handling the exchange of data between the system being served and the

intended receiver system and providing the necessary format for transmission of data and responding to messages it is to receive. The transfer and addressing of messages are done by the NIC.

MOST differs, however, from the previously discussed IVN protocols in that it has a greater data rate than the others. It also differs in the class of vehicle systems that it serves. For example, the vehicle systems on the MOST IVN include digital audio broadcast (DAB) receivers, a tuner for satellite digital audio radio service (SDARS), mobile telephone, phonebook for the mobile telephone, navigation systems, and graphic display devices.

The MOST transceiver incorporates an LED for transmitting light pulses along the optical fiber. The operation of an LED is explained in Chapter 8. The wavelength of the light is in the red portion of the visible spectrum at 650 nm. The LED is mounted in a structure built around the optical fiber such that wherever the LED is activated with an electrical pulse a light pulse propagates along the optical fiber MOST bus.

An example circuit illustrating the electrical drive for the LED is depicted in Fig. 9.13. One of the NIC outputs is a sequence of bits $B(k)$ at voltage levels corresponding to 1, 0. In Fig. 9.13, a transistor (T_1) (bipolar in this example, but it could be an FET) driver controls the current level of the LED according to its specifications such that the intensity of the transmitted light pulse corresponds to MOST protocol specifications. The LED is in the emitter-to-ground segment of the circuit. It should be noted that the NIC has many other connections as described later in this section.

The MOST transceiver incorporates a PIN photodiode to receive optical data and convert them to electrical signals. In Chapter 2, the physical configuration of a diode was explained to be a junction between p-type- and n-type-doped semiconductor materials. The PIN diode has a thin layer of intrinsic (I) semiconductor between a p-type and n-type sections as depicted in Fig. 9.14. The I layer creates an expanded depletion region in the diode junction.

The PIN diode is fabricated with a transparent cover (C) such that input light can illuminate the I layer. The PIN diode is reverse biased as depicted in Fig. 9.14. Each photon entering the junction region creates hole-electron pairs. The number of such pairs is proportional to the intensity (I_ℓ) of the light illuminating the junction region. These charge carriers increase the reverse-bias saturation current $i_S(I_\ell)$ in an approximately linear relationship to (I_ℓ). Fig. 9.14B depicts a representative MOST receiver circuit in which an internal supply voltage V_{sup} reverse biases the photodiode. The reverse

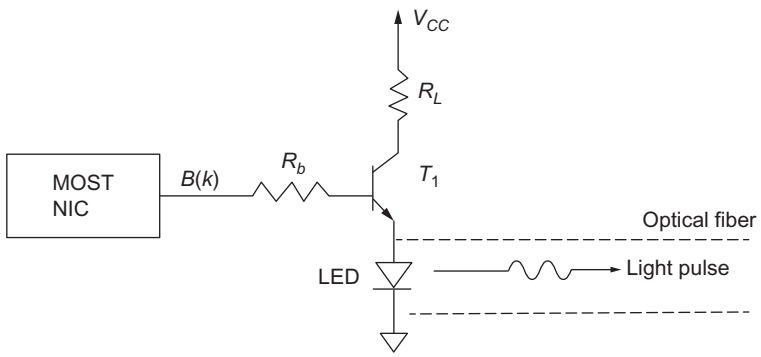


FIG. 9.13 Illustrative circuit for MOST LED.

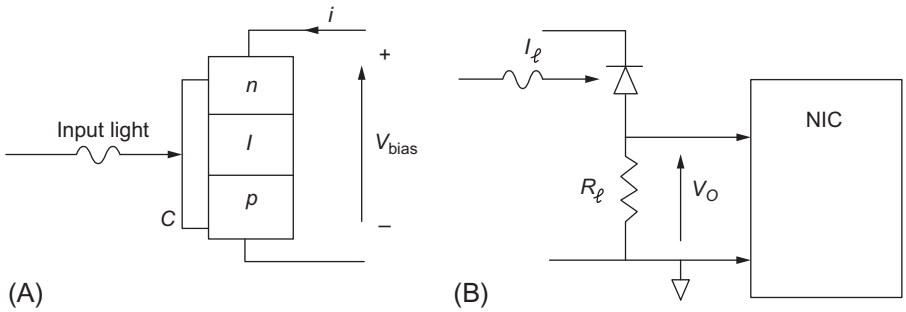


FIG. 9.14 (A) PIN diode structure and (B) MOST receive circuit.

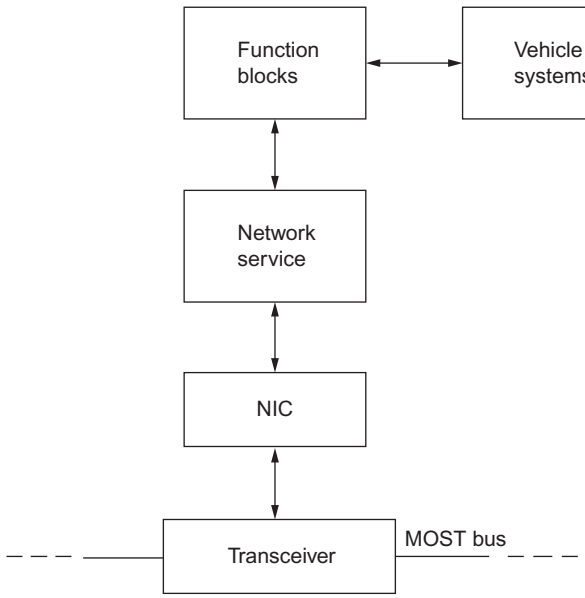


FIG. 9.15 Functional block diagram of MOST node.

saturation current passes through a load resistance (R_ℓ) creating an output voltage $v_o = R_\ell i_S(I_\ell)$. This voltage can be directly connected to the NIC receiver input provided that the amplitude during a light pulse bit is sufficient to represent a logical level. If v_o is insufficient, an amplifier can be placed between R_ℓ and the NIC. It would also be possible to insert a logical inverter if any particular configuration requires a light pulse to create a logic low.

A functional block diagram of a MOST IVN node is depicted in Fig. 9.15. Each node has interconnection with the MOST bus via the device that is termed a transceiver in this section of the book to relate this type of interface to those of other IVN protocols discussed above. The transceiver is

connected electrically to and controlled by the NIC, which, in turn, passes data to and from the MOST bus to the vehicle system(s) being served by the node. The MOST network requires a number of masters for different functions that can be packaged together in one electronic system. One of the masters is called a timing master. This master controls network timing and synchronization between the devices on the network. There is also a so-called network master that does the network setup and allocates addresses to the devices on the network. Another master, called the connection master, establishes synchronous communication channels between the various systems. There also is a master called the power master that continuously monitors all power supply operations. It also deals with power on for the system and system shutdown.

Communication between vehicle systems that are connected to the MOST bus is done through the function blocks (called F blocks in MOST protocol). The system sending the message does not require it to have the address of the receiving system F block. The address and transfer of a message is done by the sender NIC. If more than one vehicle system has the same F block, each will be given a separate address by the master.

Data are sent along the MOST bus in digital frames. Each frame can be made up of three communication channels. One channel is the synchronous channel that can stream data. Another channel is the asynchronous channel. This channel handles packed data with relatively large data block size with relatively large bandwidth. There also is a channel called the control channel. This channel is for event-driven transmission. It operates with relatively low bandwidth 10 kbps. It normally involves relatively short length.

The MOST protocol there are a number of frame bit sequence configurations. As an illustration, we consider a MOST 50 frame. It consists of 128 bytes on 1024 bits. The first 11 bytes are the header that includes a descriptor of the boundary between synchronous and asynchronous data and 4 bytes of control channel and an administrative section. The next data area is 117 bytes long and consists first of synchronous data and the next of asynchronous data. The boundary between these two channels is variable, which is the reason the boundary descriptor is required. The system administration portion of the header has a number of functions. One function involves recognition in delays associated with conversion between optical and electrical data. The delay in any node has one value for an active node and another for a passive one. The system must incorporate these delays when controlling exchange of data. In addition, unused channels can be detected. The system administration in the master can reallocate channels.

The optical medium has the advantage of EMI immunity. Also, the fiber is lighter in weight than a corresponding wire bus. However, the maximum working temperature is 85°C, which precludes the use of optical fiber in the engine compartment. Nevertheless, the infotainment applications of the MOST protocol do not require it to be used in high-temperature regions of the vehicle.

There are many details of the MOST protocol that are beyond the scope of this book. The interested reader can learn any level of detail of interest by consulting the MOST specification.

VEHICLE TO INFRASTRUCTURE COMMUNICATION

In addition to the communication between vehicle systems via the various IVNs discussed above, there are several significant communication channels to and from the vehicle. Such communication between a moving vehicle and the outside world requires an infrastructure capable of transmitting and receiving

signals via an electromagnetic radiation-type (radio) link. Communication to and from the vehicle is termed vehicle-to-infrastructure communication. There are several infrastructure media available in vehicles. The oldest such infrastructure is the commercial broadcast stations (first AM and then AM-FM) and has always been a one-way medium from broadcast transmitters to vehicle receivers. The configuration of radio frequency V2I has evolved over the years since the earliest AM radio days.

As of the time of the writing of this book, there are several vehicle-to-infrastructure (V2I) communication channels/media. These, of course, include cellular telephone and satellite media. As is almost universally known, cell phones provide voice communication and text messaging as well as Internet connection from moving vehicles essentially the same as from fixed locations. These features are standard with a majority of the handheld cell phones. However, there also is the capability to install a built-in cell access point in the vehicle.

The vehicle-to-satellite infrastructure has three major categories: audio or video entertainment, concierge service, and GPS navigation. The audio/video communication is a one-way streaming of data from the satellite to the vehicle. These signals may be sent directly to an ad hoc audio or video unit or may be handled via an IVN such as the MOST IVN described above. The concierge service is well represented by GM's OnStar system. Originally, this service included a connection from the vehicle to the cellular infrastructure and then to a concierge. The concierge could provide information to the vehicle verbally and also could complete a telephone connection. This latter application provided "hands-free" telephone dialing, which has safety implications in that the driver, when using the service, does not need to take "eyes of the road" for dialing a number.

VEHICLE-TO-CELLULAR INFRASTRUCTURE

A detailed discussion of cell phone technology is far too large for this book. Rather, there are numerous publications dealing exclusively with this technology. However, a limited overview of the technology is informative for the way in which vehicle-to-infrastructure technology is evolving. One of the important issues in cell phone technology is the available bandwidth in relationship to the very large pool of users. Any given cell phone channel will have multiple simultaneous users, both fixed and vehicle born. The cell phone infrastructure must provide a means of accommodating these multiple users. On the other hand, the bandwidth associated with an individual user for voice communication occupies only a few kilohertz of bandwidth. The sharing of the communication channel by multiple users can be termed, in the broadest sense, multiplexing.

There are many forms of multiplexing communication channels. One of these multiplexing schemes involves periodic sampling of each individual communication signals (e.g., voice). The sampled analog data are converted to digital data (via A/D conversion; see Chapter 3) and are transmitted over the channel during an assigned time slot. Each user is assigned a time slot within a communication cycle. At the receiving end of the channel, the digital data are converted back to analog for vocal communication via D/A conversion (see Chapter 3) and sent to the receiver. Of course, for text messages, the A/D and D/A conversions are not necessary.

For multiple users, each user is assigned a time slot, and that user is given access to the communication channel during the assigned time slot during each cycle. This type of multiplexing is known as

time-division multiplexing access (TDMA). There is a fundamental limit on the minimum duration of the cycle time for any given signal bandwidth. This cycle time also is the sampling interval (T_s) for each user that, for verbal communication, must satisfy the so-called Nyquist sampling rate, which is given by

$$T_s \leq \frac{1}{f_m}$$

where f_m is the maximum frequency in the signal being sampled and sent over the communication channel.

Cellular systems have a number of frequency intervals assigned to each commercial carrier. Within each band, a given carrier can use multiple carrier frequencies each of which is modulated with the information being transferred. With a TDMA multiplexing, each group of users within a given cycle would be modulated on a separate carrier. At the receiving end of the communication channel, the signals are recovered by demodulation, and each individual signal is reconstructed from its time slot.

Another significant multiplexing technique is known as code division multiple access (CDMA). In a CDMA system, all user signals are transmitted over the same frequency band but are separated by assigning each a unique code (e.g., PN or Walsh codes). At the receiving end, the signals are separated by a device known as a correlator. Each correlator only produces a significant output when the input matches the unique code for which it is assigned.

What is actually happening in CDMA is that the narrow band signal (less than 10 kHz), such as is the case for cell phones, is spread over a very much larger bandwidth by a very wide band code that is associated with a particular user. Such a technique is known as the spread-spectrum technique. The codes used in CDMA spread spectrum are orthogonal codes in a specific sense. The term orthogonal is analogous to the property of orthogonal vectors. The product of a vector $\bar{x}$ with an orthogonal vector $\bar{y}$ is 0 (i.e., $\bar{x} \cdot \bar{y} = 0$). This same property can be applied to specific binary codes. One such code that is used in some cell phone CDMA systems is known as the Walsh code.

The Walsh codes are generated by a matrix algorithm defined by the rank of the matrix n and denoted as W_n where

$$W_{2n} = \begin{bmatrix} W_n & W_n \\ W_n & \bar{W}_n \end{bmatrix} \quad n = 1, 2, \ldots, N$$

with $W_1 = 0$ and where the overbar indicates the logical complement of all elements of the matrix. For example, W_2 and W_4 are given by

$$W_2 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

$$W_4 = \begin{bmatrix} W_2 & W_2 \\ W_2 & \bar{W}_2 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

In the practical implementation of CDMA, $N = 64$. A Walsh code is specified by the notation W_{mN} where m is the row number of Walsh matrix W_N. For example, Walsh code $W_{34} = [0\,0\,1\,1]$.

The spectrum spreading is accomplished by performing the logical exclusive *OR* (*XOR*) with a code chosen for a particular user. The logical *XOR* (denoted $\oplus$ in Boolean algebra) is defined for any pair of bits by the following truth table:

A	B	A⊕B
0	0	0
0	1	1
1	0	1
1	1	0

Algebraically, the XOR function relates simply to the following combination of AND (·) and OR(+):

$$A \oplus B = \bar{A} \cdot B + A \cdot \bar{B}$$

This can be implemented in logical circuitry as depicted in Fig. 9.16.

The CDMA coding with spread spectrum can be illustrated with a simple example for a single user transmitting data sequence T_x 1 0 1 with an assigned code W_{24}. The spread-spectrum code is achieved by *XOR*ing T_x with W_{24} with the spread-spectrum transmission sequence denoted as T_{xss} as shown in the following table for three cycles:

	Cycle 1	Cycle 2	Cycle 3
T_{xm}	1	0	1
W_{24}	1010	1010	1010
T_{xss}	0101	1010	0101

where m is the bit number in the sequence and the cycle number. At the receiving end, the spread spectrum received sequence R_{xss} is ideal for perfect transmission without noise given by $R_{xss} = T_{xss}$. Recovery of the data for the given user is accomplished by correlating R_{xss} with W_{24} (i.e., the code for the given user). Correlating these two is accomplished by XORing and averaging the resulting bits

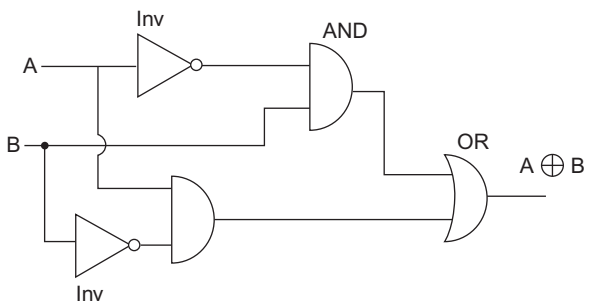


FIG. 9.16 Circuit equivalent of *XOR*.

for each cycle. The despreading for the example user is accomplished by *XOR*ing R_{xss} with assigned code W_{24} and computing the average of the *XOR*ed bits for each cycle as shown below:

	Cycle 1	Cycle 2	Cycle 3
	0101	1010	0101
R_{xss}	1010	1010	1010
W_{24}	1111	0000	1111
$Y_{nm} = R_{xss} \oplus W_{24}$			
$R_x(m)$	1	0	1

where

$$R_x(m) = \frac{1}{4}\sum_{n=1}^{4} Y_{nm}$$

In the above table, Y_{nm} is the nth bit in the mth cycle. The received bit for cycle m ($R_x(m)$) is the average over the four *XOR*ed bits Y_{nm}, $n = 1, 2, 3,$ or 4:

$$m = 1,\ 2,\ 3$$

It can be seen by comparing $R_x(m)$ with $T_x(m)$ that the transmitted signal has been correctly recovered. In the present idealized representation of recovering data from the spread-spectrum CDMA, the average of bits for each cycle for the correlation process is distinctly different for the user of an assigned code to any other user who was assigned a different Walsh code. The result of this correlation for any Walsh code must be either 0 or 1 for the correct identification of the uses assigned to the particular Walsh code. This identification correlation result is shown in the above table. Any other value (in the ideal noise-free and error-free case) will be a fraction between 0 and 1 indicating the wrong user code.

We consider next a second user assigned Walsh code $W_{34} = [0011]$. The following table illustrates the result of correlating T_{xss} for the first user with the Walsh code of the second user:

	Cycle 1	Cycle 2	Cycle 3
	0101	1010	0101
R_{xss}	0011	0011	0011
W_{34}	0110	1001	0110
$Y_{nm} = R_{xss} \oplus W_{34}$			
$R_x(m)$	0.5	0.5	0.5

Note that for each cycle above, the correlation of the received data with W_{34} is 0.5, which indicates that R_{xss} is not from user #2 and, therefore, is not passed by the system to that user. Since the correlation of R_{xss} with W_{24} is either 1 or 0, these data are passed by the system to the receiver for the user that was assigned code W_{24}.

The above example of CDMA theory is a simplified illustration. In actual cell phone CDMA, the Walsh codes are from W_{64}. That is, there are 64 orthogonal codes (or chips) that greatly increase spectrum spreading relative to the W_4 matrix example. This large code sequence is highly effective in correctly identifying each user in the presence of noise that occurs in practice with actual cell phone transfer of data. In addition, the practical CDMA uses an analog representation of T_{xss} such that the spread spectra from multiple users can be added together algebraically before being transmitted

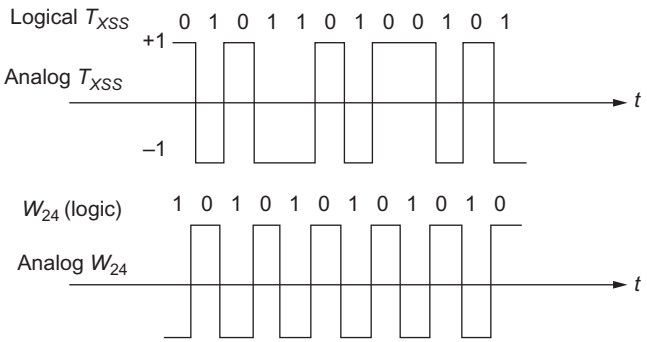


FIG. 9.17 Analog versions of T_{xss} and W_{24}.

on a carrier signal. In the analog conversion, a logic 1 is represented by a signal corresponding to -1 and a logic 0 by a +1. The corresponding waveforms for the T_{xss} of the above example are illustrated in Fig. 9.17.

The correlation at the receiving end is accomplished by averaging the product of the analog R_{xss} with W_{24}. The signal that is transmitted is the sum of the analog versions of the T_{xss} for each of the active users. This composite signal modulates the carrier by a scheme that permits signed values of the combined sum of all T_{xss} analog signals. At the receiving end, the composite analog spread-spectrum signal is obtained by demodulation of the transmitted signal. The demodulated combined signal is sent to each of the correlations for all assigned codes. In this way, the data for each user are sent to the intended receiver when the correlation of the data is consistent with the assigned code for the user pair.

The cell phone carrier frequencies are in various portions of the electromagnetic spectrum including a band near 2 GHz. The modulation method varies somewhat among the main cell phone providers. One type of modulation method involves phase shifting the transmitted carrier signal with the composite analog T_{xss} data. In such a scheme, the instantaneous amplitude $A(t)$ of the modulated carrier can be modeled as follows:

$$A(t) = A\sin(\omega_c + \phi(t))$$

where $A =$ constant, $\omega_c = 2\pi f_c$, $f_c =$ carrier frequency, and $\phi(t) =$ instantaneous phase:

$$= \Phi_m \sum_{k=1}^{K} T_{xss}(k) \quad k = 1, 2, \ldots, K \leq 64$$

where $\phi_m =$ constant for the modulating circuit and $T_{xss}(k) =$ analog T_{xss} for kth user.

Often, CDMA transmitting system incorporates a quadrature carrier in which $\phi(t)$ includes a fixed $\pi/2$ phase shift. In such cases, the in-phase and quadrature-modulated signals are sent over the same channel. The phase shift $\phi(t)$ due to the data T_{xss} consists of integer multiples of a fixed amount of phase shift due to the nature of the T_{xss} signal. Such a modulation scheme is known as phase-shift keying (PSK) or quadrature-phase-shift keying (QPSK) in the case of quadrature carriers.

QUADRATURE PHASE SHIFTER AND PHASE MODULATION (QPSR)

In QPSK modulation schemes, it is necessary to have two carrier signals of equal amplitude at frequency f_c that have a phase difference of 90 degrees ($\pi/2$ rad). A circuit that can generate two signals of equal amplitude that are 90 degrees out of phase is depicted in Fig. 9.18.

The input to the circuit is a sinusoid $V_s(t)$ at carrier frequency f_c that in complex notation is given by

$$V_s(t) = V_s e^{j\omega_c t}$$

where $\omega_c = 2\pi f_c$.

The two output voltages $V_1(t)$ and $V_2(t)$ complex amplitude are given by

$$V_1 = \frac{V_s / j\omega_c C}{R + \dfrac{1}{j\omega_c C}} = \frac{V_s}{1 + j\omega_c CR}$$

$$V_2 = \frac{V_s R}{R + \dfrac{1}{j\omega_c C}} = \frac{jV_s\omega_c CR}{1 + j\omega_c CR}$$

To achieve the desired phase difference, the R and C are chosen such that

$$\omega_c RC = 1$$

In this case, voltages V_1 and V_2 (in complex notation) are given by

$$V_1(t) = \frac{V_s}{\sqrt{2}} e^{j\left(\omega_c t - \frac{\pi}{4}\right)}$$

$$V_2(t) = \frac{V_s}{\sqrt{2}} e^{j\left(\omega_c t + \frac{\pi}{4}\right)}$$

The phase difference is $\dfrac{\pi}{2}$ (90 degrees), and the amplitudes are equal as desired.

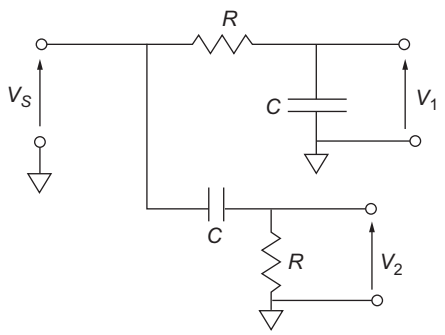


FIG. 9.18 Quadrature-phase signal generation.

The above analysis of the circuits shown is based upon ideal circuit components R and C. In practice, achieving close approximations of actual circuit components to ideal depends upon the carrier frequency and the method of fabrication (e.g., IC). However, the technology exists today to achieve the desired result.

The combination of resistance and capacitance can be used to cause the phase variation of a signal if either circuit element is variable. The major portion of modern electronic systems is implemented in ICs. One of the readily fabricated IC circuit components is a diode. A reverse-biased diode has junction capacitance due to the fixed charge in the depletion region. The size of the depletion region varies with the magnitude of the reverse-bias voltage. The technology exists in IC fabrication to optimize diode junction capacitance variation with applied voltage.

In a PSK modulation system, the carrier frequency is several orders of magnitude greater than the modulating signal. This frequency separation permits the application of the phase modulating signal to be applied to a diode through a signal path that is effectively isolated from the circuit supplying the carrier frequency signal to the variable capacitance diode. For example, the portion of the circuit shown in Fig. 9.18 that supplies voltage V_1 could include a variable capacitance diode across capacitance C. A modulating signal applied to the diode would cause the phase of the voltage corresponding to V_1 to vary. If the modulating signal V_m was applied to the diode whose capacitance c_d then that capacitance would be given by

$$C_d(V_m) = C_d(O) + \delta C_d(V_m)$$

The corresponding phase of v_1 would be given by

$$\phi_1 = \tan^{-1}[\omega_c R(C + C_d(V_m))]$$

In most applications, the variation in phase with modulation voltage $V_m(\delta\phi_1(V_m))$ is essentially linear in V_m, although, for phase-shift keying, a linear relationship is not necessary since the modulating signal is binary-valued and only needs to produce binary-valued phases of the carrier.

The structure of a cell phone network includes several components including the mobile station (user), the base transceiver station, the base station control system, and the mobile switching center. The network incorporates numerous base stations distributed over wide geographic areas and multiple mobile switching centers. Each such station covers an area within the line of sight of the mobile user, which area is determined, in part, by the height of the associated antenna. For the user, the source/destination of signal/data transfer is the base transceiver. The data being transferred between the mobile station and the destination are directed from the base station to the switching center where the connection between the mobile use and its intended receiver is made via the switching centers. Once the signal to or from a mobile use is passed through the base and switching centers from either and of the user pair, the messages are in the network and are handled with correct routing.

There are many other details associated with CDMA base cellular communication that are beyond the scope of this book, which has many other topics to discuss. These details are available to the interested reader from various Internet sites.

SHORT-RANGE WIRELESS COMMUNICATIONS

Wireless communication between any pair of users (humans or electronic systems) uses radio (or light) as the medium. It requires at least one transceiver at either end of the link. Previously discussed V2I communication has involved intertransceiver distances of relatively long range

(e.g., tens or hundreds of kilometers or thousands of kilometers for satellites). There are numerous applications for short-range wireless communication that can be considered as belonging to one of the three classes: class 1 with a range up to 100 m, class 2 with a range up to 10 m, and class 3 with a range of about 1 m. The range of any wireless ink is influenced in part by the obstacle line-of-sight distance and the transmitter power. Any such wireless communication within a vehicle would belong to class 2 or 3.

We consider as an example of short-range wireless communication the system known as Bluetooth, which is a widely used and relatively reliable short-range communication system. This type of short-range communication system has existing and multiple potential future applications, both within and to and from vehicles. For example, a class 3 Bluetooth device provides a wireless link between a cell phone and a headset. Such a device can be used within a vehicle enabling hands-free cell phone communication. For this application, the vehicle is equipped with a built-in Bluetooth system. The cell phone Bluetooth device "pairs" with the corresponding vehicular system. This means that the cell phone is connected via a wireless link to the vehicle. Some of the technical aspects of pairing are explained later in this section. In many vehicles, the built-in Bluetooth is connected to the system that controls a flat-panel display sometimes with "touch-screen" capability. In addition, the controller that is coupled to the Bluetooth system in many cases is programmed for voice recognition such that a driver can dial the destination phone verbally without having to be visually distracted from driving. Such a system with the Bluetooth link from cell phone to vehicle can have important safety implications.

An example of wireless communication to and from the vehicle is the potential to connect vehicle electronic systems to service-bay computers. A wired connection capability already exists (e.g., as part of CAN) but is limited in length by CAN protocol specifications. A class 2 Bluetooth system could provide a wireless length of as much as 10 m.

In addition, a Bluetooth wireless link is possible between cell phones and the vehicle from outside. A potential application might be to unlock a car when keys and key fobs are accidentally locked inside a vehicle (e.g., a cell phone application in which the customer contacts a car dealer or a representative at the OEM who could transmit a code for unlocking the vehicle).

There also are a great many in-vehicle potential applications such as wireless connection of in-vehicle systems. For example, a sliding door control could be implemented with a short-range wireless link rather than flexible cables. Many potential applications are reported in existing automotive literature. In addition, there are potential applications for short-range wireless lengths to and from existing IVNs (e.g., CAN).

The Bluetooth short-range telecommunication system operates with 79 individual carriers at a frequency band within the microwave portion of the spectrum that involves radio waves with a wavelength of just under 5 in. Commonly, antennae for such application are approximately one-fourth of the wavelength or less in length. This makes packaging Bluetooth devices convenient.

Each Bluetooth device has a controller that controls the built-in transceiver. A pair of Bluetooth devices, when sufficiently close together (i.e., within the range of the corresponding classes) mutually detect transmissions from the other devices and are programmed to "pair up" or lock onto the same carrier frequency. At this point (i.e., once paired), the two devices synchronously change carrier frequency randomly, but both devices change to the same frequency. The communication between the devices then is a spread-spectrum technique that is known as frequency hopping. Frequency-hopping spread spectrum differs from the CDMA spread spectrum described earlier in this chapter, but it

achieves much of the same result as that of the CDMA by increasing channel capacity (via spread spectrum) and reducing sensitivity to interference/noise.[1]

SATELLITE VEHICLE COMMUNICATION

Direct communication between satellites and vehicles is commonplace in present-day vehicles. Satellite transmission of audio (or video) on multiple channels is an example of infrastructure to vehicle communication for entertainment purposes. In addition, the global positioning system (GPS) is available for navigation purposes in vehicles that are equipped with GPS receivers (both receivers installed in vehicles and in cell phones can provide navigation in vehicles). There is a detailed discussion of the theory and operation of vehicular GPS later in this chapter. However, we begin the discussion of satellite infrastructure to vehicles for entertainment.

The satellite radio uses a 2.3 GHz carrier (microwave S-band) for North America. It uses a technique known as digital audio broadcasting (DAB). One of the issues in transmission at these high carrier frequencies is multipath interference in which multiple reflections create a signal at the receiver that is the sum of numerous versions of the transmitted signal with relatively large time differences that can, in certain circumstances, yield a low or null signal amplitude. The multipath problem in such communication is exacerbated in a moving vehicle.

The multipath problem along with other problems including interference and noise has been eliminated or, at least, significantly reduced by the use of DAB in which a spread spectrum for the signal being sent is accomplished by a unique form of frequency multiplexing known as orthogonal frequency-division multiplexing (OFDM). In an OFDM system, there are N subcarriers having frequencies f_n where the separation of frequencies Δf is given by

$$\begin{aligned} \Delta f &= f_{n+1} - f_n = \frac{1}{T} \\ f_n &= \frac{n}{T} \quad n = 1,2,\ldots,N \end{aligned} \tag{9.1}$$

where T is the duration of data units. In DAB the data are sampled in the audio spectrum. Each subcarrier can be modeled as follows (in complex form):

$$V_n(t) = V_n e^{i\left(\frac{2\pi nt}{T}\right)} \tag{9.2}$$

where V_n is the amplitude of the nth subcarrier. In practice, these amplitudes are nearly identical. Orthogonality of the subcarriers is represented by the following relationship:

$$\begin{aligned} &\frac{1}{T}\int_O^T V_n(t)V_m(t)dt = V_n^2 \qquad m = n \\ &= \frac{V_n V_m}{2\pi(n-m)}\left[e^{j2\pi(n-m)} - 1\right] = 0 \quad m \neq n \end{aligned} \tag{9.3}$$

[1]Synchronous, random frequency changing between transceiver pairs was originally invented by film actress Hedy Lamarr and composer George Antheil at the beginning of World War II.

The audio input signal v(t) is sampled at period T/N yielding an input sequence (v_n) where

$$\begin{aligned} v_n &= v_o(t_n) \\ t_n &= nT/N \quad n = 1,2,\ldots,N \end{aligned} \tag{9.4}$$

The maximum value for T/N must satisfy the Nyquist criterion explained earlier.

In addition to suppression of multipath interference, the orthogonality of the subcarriers mitigates against channel-to-channel cross talk. The input data sequence is encoded (analogous to the CDMA scheme described in the discussion of cell phone technology). The code allows for error correction yielding data. The coded data are modulated on the N subcarriers at what is commonly called baseband since later in the process, modulation on the S-band carrier occurs. The data are transmitted in blocks such that a time sequence of data is converted to a set on N parallel modulated channels.

The input sequence of data (v(n)) is converted to a parallel stream of coded symbols C_m, $m = 0,1,2,\ldots,N-1$ for later modulation on a separate orthogonal subcarrier. The resulting parallel data structure is analogous to a discrete Fourier transform (DFT) in which each complex symbol C_m represents the amplitude of the modulation on the orthogonal subcarrier.

It is perhaps instructive to review the formal definitions and structure of DFT and its inversion called the inverse discrete Fourier transform (IDFT). The discrete Fourier transform (DFT) is the conversion from a time-domain-based sequence $x_n(n = 0,1,\ldots,N-1)$ to a sequence of complex discrete frequency-domain values $X(k\Omega)$ and is defined

$$X(k) = \sum_{n=0}^{N-1} x_n e^{-jk\Omega n} \quad 0 \le k \le N-1 \tag{9.5}$$

where $\Omega = \dfrac{2\pi}{N}$ = fundamental digital frequency and N = number of samples in the input sequence.

Thus, if a continuous-time waveform (e.g., an audio signal) is sampled at time $t_n = nT_s$ over an interval $0 \le t \le NT_s$, the result is the sequence x_n:

$$x_n = x(nT_s) \quad n = 0,1,\ldots,N-1$$

The DFT of this sequence as defined above yields digital frequency representation of $\{x(nT_s)\}$. The corresponding inverse discrete Fourier transform (IDFT) is a transformation from the discrete frequency-domain sequence $X(k)$ to a time-domain-based sequence $x(n)$:

$$x(n) = \sum_{k=0}^{N-1} X(k) e^{jk\Omega n}$$

where $\Omega = \dfrac{2\pi}{N}$.

In the ODFM process, a time-domain sequence ($c(n)$) is first converted to a parallel set of complex values C_m via a serial to parallel and a coding process. The sequence C_m (complex valued) is returned to a discrete-time sequence $c(n)$ by means of an inverse discrete Fourier transform using a highly computationally efficient inverse Fourier transform known as inverse fast Fourier transform (IFFT) where $c_n(n)$ is given by

$$c(n) = \sum_{m=0}^{N-1} C_m e^{j\frac{m2\pi n}{N}}$$

The above equation for the nth time sample $c(n)$ can be interpreted as the sum of N frequency-domain representation of N frequencies ($e^{jn\lambda_m}$) amplitude-modulated by complex value C_m. The digital frequency of the subcarrier for C_m is λ_m where

$$\lambda_m = \frac{2\pi m}{N}$$

Thus, the actual modulation of the data sequence on a set of subcarrier frequencies takes place in the IFFT operation.

Before the sequence is transmitted, there is another step required to prevent interference between adjacent blocks due to multipath interference. This step involves modifying the sequence by inserting a so-called cycle prefix in which each block contains a subsequence of samples from the previously transmitted block over a time interval t_n–T_g. The sequence of data symbols sent is converted to a continuous-time signal $s(t)$ via a D/A converter. The baseband signal $s(t)$ for the nth block can be represented in complex notation by the following:

$$s(t) = \sum_{m=1}^{N-1} C_m e^{j\frac{2\pi mt}{T}} \quad (n-1)T - T_g \le t \le nT$$

That is, assuming a perfect D/A, the analog signal is the sum of a set of amplitude-modulated orthogonal subcarriers $e^{j2\pi f_m t}$ at frequencies f_m where

$$f_m = \frac{m}{T}$$

The data signal is amplitude-modulated on the 2.3 GHz carrier frequency (denoted f_c). In amplitude modulation of a carrier, the modulated signal $S_m(t)$ is given by

$$S_m(t) = S_o(1 + \varepsilon s(t))\cos(\omega_c t)$$

where ε is the modulation index and S_o is the carrier amplitude. To avoid distortion of the modulating signal $s(t)$, this index should be chosen such that

$$\varepsilon s(t) < 1$$

In the case of amplitude-modulated satellite radio, the transmitted signal $S_T(t)$ is given by

$$S_T = S_O\left[1 + \varepsilon\,|C_m|\cos\left(\frac{2\pi t}{T} + \phi_m\right)\right]\cos(2\pi f_c t)$$

where $|C_m|$ = absolute value of C_m and ϕ_m = phase angle of C_m.

The resulting spectrum of S_T is given by the sequence of frequencies ($f_s(m)$):

$$\{f_s(m)\} = \left\{f_c \pm \frac{m}{T}\right\} \quad m = 0, 1, \ldots, N-1$$

Other modulation schemes also can be used such as phase-shift modulation with the same result of upconverting the signal being transmitted $s(t)$ from baseband to the set of frequencies ($f_s(m)$).

At the receiving end, the process of recovering the original analog signal is the reverse of the process at the transmitter. The first step is to demodulate the carrier and convert the information to baseband. This is accomplished by multiplying $S_T(t)$ with a local oscillator at the carrier frequency and low-pass filtering the product that recovers $s(t)$.

The baseband analog signal is sampled at times t_n resulting in the sequence ($c(n)$) being recovered. The output of the sampled signal is given by

$$s(t_n) = s(n)$$
$$= \sum_{m=0}^{N-1} C_m e^{j\frac{2\pi mn}{N}}$$

The coded data C_m can be recovered by taking the discrete-time Fourier transform DFT using the highly efficient algorithm fast Fourier transform (FFT). The FFT performs an operation as given below:

$$C_m = \frac{1}{N} \sum_{n=0}^{N-1} s(n) e^{jm\Omega n} \quad m = 0, 1, \ldots, N-1$$

This process yields the parallel data C_m because the digital frequencies are orthogonal. The original analog signal is obtained by decoding the parallel data and generating an output sequence v_n, which is a replica of the sampled input audio at the transmitter. Decoding can be accomplished by correlating the coded data C_m with the code sequence that created C_m from the input sequence analogous to signal recovery in CDMA and yielding the serial sequence $c(n)$. The analog audio $v(t)$ is then obtained by sending the recovered sequence $c(n)$ to a D/A converter.

In DAB, there are multiple channels that are sent using the process described above. Each channel is available at the receiver end of the satellite-to-vehicle communication system. One scheme for handling multiple channels is to use time-domain multiplexing during the input sampling process and assigning a specific time slot to each channel during each cycle.

The advanced signal processing and the OFDM with coding have resulted in a reliable satellite-to-vehicle communication system. Another important application of satellite-to-vehicle communication is GPS navigation, which has become commonplace in vehicles and is discussed in technical detail in the next section of this chapter.

GPS NAVIGATION

The GPS navigation system, global positioning system (GPS), has provided the capability of some relatively sophisticated vehicle navigation systems. Initially intended for aircraft position measurements and navigation, it has been successfully adapted for use with land vehicles. As explained below, a GPS-equipped vehicle has the capability for relatively precise and accurate measurements of the vehicle position. This position information combined with electronic versions of maps yields the capability to navigate optimally between any two locations without requiring any paper road maps.

The GPS system consists of 24 satellites arranged in groups of four in each of six orbital planes inclined at 55 degrees spaced 60 degrees apart in longitude and at a nominal altitude of 11,000 nm above the local surface (i.e., orbital semimajor axis $\approx$ 26,600 km). At any given time for any given receiver location, a subset (I) of satellites are available for use by the receiver.

Each satellite carries a precise (atomic) clock and repetitively transmits its position and time (i.e., ephemeris data). The user equipment consists of a receiver along with its own precise clock.

By measuring the time difference δt from the transmission of the signal to its reception, the receiver obtains a measurement of the transit time from satellite to receiver, which yields an estimate of the range R from the satellite to receiver:

$$R = c\delta t$$

where c is the speed of propagation of the satellite-transmitted signal. If the receiver and satellite clocks were perfectly synchronized then, in principle, the measurement of δt would yield the range from the receiver to (known) satellite position. A set of three measurements to three satellites could ideally yield the solution for the user position from these measurements. However, it is, in practice, impossible to exactly synchronize these two clocks. Moreover, the receiver clock is less precise than the satellite atomic clock. The actual measured time difference between satellite i ($i = 1, 2, \ldots I$) clock and receiver clock time yields an estimate of R (denoted R_i) called pseudorange. Because of the receiver clock uncertainty, at least four measurements are required to estimate vehicle position. The pseudorange model for satellite i is given by

$$\begin{aligned} R_i &= c\delta t_m \\ &= c\delta t + B + n_e \\ &= c\delta t + n \end{aligned}$$

where δt_m = measured time difference, and B is a bias resulting from the receiver clock error Δt_c,

$$B = c\Delta t_c$$

that is, Δt_c is the error between true GPS time as carried by the satellite and the receiver clock time and c is the propagation speed of the GPS signal. In addition, there are other error sources n_e (e.g., due to fluctuations in c as the transmitter signal passes through the atmosphere) that are discussed later in this chapter. The combined errors are denoted as n in the pseudorange model.

For an understanding of GPS navigation principles, it is helpful to first understand a simplified ideal system for determining the position of a point in a 3D coordinate system. The unknown location of a point in a coordinate system can be determined if the distance (straight line) to three noncolinear points of known locations is found. Let the distances to the three points be denoted as R_1, R_2, and R_3. For the GPS system, a spherical coordinate system with origin at the center of the earth has the position of each satellite given by radial distance R_s, latitude θ_s, and longitude φ_s. However, for a more simplified explanation of GPS operation, a Cartesian coordinate system is perhaps more readily understandable. It should be noted that the transformation of coordinate systems involves relatively simple matrix operations. A more detailed discussion of the relevant coordinate system is presented later in this section.

We begin the discussion of the determination of vehicle position with the process for an ideal, error-free set of distance measurements. In this ideal case, only three distance measurements are required to solve for vehicle location. The location of the unknown point x, y, z (in a Cartesian coordinate system) is the intersection of three spheres of radii R_1, R_2, and R_3 whose centers are at the known locations denoted as $x_1 y_1 z_1$, $x_2 y_2 z_2$, $x_3 y_3 z_3$, respectively. The coordinates of the unknown point can be found by a process known as trilateration in which the three quadratic equations of the surfaces of the three intersecting spheres are solved for x, y, z.

These three equations for the spheres centered at $x_i y_i z_i$, respectively, for $i = 1, 2, 3$ are given by

$$i=1: \quad R_1 = \left((x_1-x)^2+(y_1-y)^2+(z_1-z)^2\right)^{\frac{1}{2}}$$
$$i=2: \quad R_2 = \left((x_2-x)^2+(y_2-y)^2+(z_2-z)^2\right)^{\frac{1}{2}}$$
$$i=3: \quad R_2 = \left((x_3-x)^2+(y_3-y)^2+(z_3-z)^2\right)^{\frac{1}{2}}$$

There are multiple methods of solving these three equations, but since they are quadratic, a method must be found to select the correct solution. One trilateration solution in somewhat simplified satellite/vehicle configuration is presented in Appendix E.

A far superior method of estimating vehicle position from pseudorange measurements involves the use of a so-called Kalman filter. A Kalman filter provides an optimal linear estimate of a state vector in the presence of noise in that it yields the estimate with the smallest possible RMS error between the true and estimated state vector.

Although there are numerous Kalman filter implementation schemes, for the purpose of the present discussion, we assume one of the simplest forms that includes a signal model, a measurement model, and a filter model. In this section of the chapter, a vector is denoted by a symbol with an overbar. The signal model is based upon a state vector $\bar{X}(k)$ at a discrete time $t_k = k\delta t \quad (k = 0,1,2,\ldots)$ in which the state vector includes the position coordinates of the receiver along with other components as explained below. The signal model yields the progression of the state vector from time t_k to t_{k+1}:

$$\bar{X}(k+1) = F\bar{X}(k) + \bar{w}(k)$$

where $F =$ state-transition matrix.

In this model, $\bar{w}(k)$ is a vector of random processes that represent random variations in the state vector. For example, in a vehicle, $\bar{w}(k)$ includes random fluctuations in vehicle speed due to hills, driver action, traffic, etc. The state vector for the present Kalman filter configuration and the state-transition matrix are given later in this section after the structure of the measurement model is presented.

The measurement model in which the measurement is denoted as $\bar{z}(k)$ is given by the following linear model:

$$\bar{z}(k) = H(k)\bar{X}(k) + \bar{n}(k)$$

where H is the measurement matrix and $\bar{n}(k)$ is a vector of random errors. In the case of GPS, there are many sources of errors explained below. These error sources are independent, stationary (normally approximately white Gaussian) random processes. These random error properties are important in the formulation of the Kalman filter applied to GPS.

The Kalman filter can have a number of structures depending upon various factors including the variables being measured and the nature of the errors involved. Essentially, the input to a Kalman filter is a set of measurements of the desired variable that involve random errors or noise. In the application of Kalman filters to GPS-based navigation as in the case of a trilateration estimate of position, the measurements are the distance from a set of satellites (that are above the horizon). Due to the imperfections of these measurements in practice, these distances are termed pseudorange. For navigational purposes, the goal of the GPS is to obtain a discrete-time estimate of the state vector $(\bar{X}(k))$ of the vehicle at time $t_k = k\delta t$, which is denoted as $\hat{X}(k)$ that includes its position and velocity of motion in a navigationally

significant coordinate system (e.g., geodetic). For the purposes of explaining GPS operation, it is convenient to choose a Cartesian coordinate system having z-axis through the center of the earth and its origin at the center of the earth. Such a coordinate system is known as Earth-centered, Earth-fixed (ECEF) Cartesian coordinate system. For land-based vehicles (rather than high-speed aircraft), the x,y plane that is tangent to the earth at a point in the vicinity of the vehicle at the start of navigation is useful for locating a vehicle over sufficient distances to provide vehicle navigation. This coordinate system is chosen for the discussion of GPS for land vehicles, but it is readily adaptable to other coordinate systems depending upon the distance traveled by the vehicle (e.g., across a continent). The following explanation, which is relatively simple, should provide sufficient background for the interested reader to investigate GPS (with Kalman filter) for other coordinate systems (e.g., geodetic).

In the present (somewhat simplified) signal model, the state vector includes the position of the vehicle (x,y,z) at discrete times:

$$t_k = k\delta t \quad k = 0, 1, 2, \ldots$$

and δt is the discrete-time sample interval. This interval is taken in this section of the chapter to correspond to the GPS sampling interval. In developing the signal model for the Kalman filter, the following vehicle model is assumed:

$$\begin{aligned} x(k+1) &= x(k) + \delta t \dot{x}(k) \\ y(k+1) &= y(k) + \delta t \dot{y}(k) \\ z(k+1) &= z(k) + \delta t \dot{z}(k) \end{aligned}$$

The vehicle position in the Cartesian coordinate system $\bar{P}(k)$ is given by the following vector:

$$\bar{P}(k) = [x(k), y(k), z(k)]^T$$

The vehicle velocity vector $\bar{v}(k)$ is given by

$$\bar{v}(k) = [\dot{x}(k), \dot{y}(k), \dot{z}(k)]^T$$

where

$$\begin{aligned} \dot{x}(k) &= \left.\frac{dx}{dt}\right|_{k\delta t} \\ \dot{y}(k) &= \left.\frac{dy}{dt}\right|_{k\delta t} \\ \dot{z}(k) &= \left.\frac{dz}{dt}\right|_{k\delta t} \end{aligned}$$

Note that, in this chapter, the superscript T refers to the transpose of the vector/matrix.

The vehicle speed at time $t_k(s(k))$ is the norm of $\bar{v}(k)$:

$$s(k) = \|\bar{v}(k)\|$$

As explained earlier in this chapter, the GPS measurements are the pseudorange $R_n(k)$ to N satellites where

$$R_n(k) = \sqrt{(x_n(k) - x(k))^2 + (y_n(k) - y(k))^2 + (z_n(k) - z(k))^2} + n_n(k) \quad n = 1, 2 \ldots I$$

where the square-root portion represents the true range and $n_n(k)$ the random error in the kth measurement of the pseudorange to satellite n at time $t_k = k\delta t$ and where the satellite n position in the Cartesian coordinates at t_n is denoted as $(x_n(k), y_n(k), z_n(k))$.

Unfortunately, the measurement is nonlinear, whereas the example Kalman filter requires a linear relationship between measurement and the state vector. Fortunately, for the relatively large satellite-to-vehicle distances and the sample rate, the change in pseudorange between successive samples ($\delta R(k)$) is sufficiently small that a linear approximation to $\delta R(k)$ can be formulated in the Kalman measurement model as given below:

$$\begin{aligned}\bar{z}(k) &= \delta R_n(k) = R_n(k) - R_n(k-1)\\ &= c_{n1}\delta x(k) + c_{n2}\delta y(k) + c_{n3}\delta z(k) - (c_{n1\delta}\delta x_n(k)\\ &\quad + c_{n2\delta}\delta y_n(k) + c_{n3\delta}\delta z_n(k))\end{aligned}$$

where

$$\begin{aligned}\delta x(k) &= x(k) - x(k-1) = \delta t\dot{x}(k)\\ \delta y(k) &= y(k) - y(k-1) = \delta t\dot{y}(k)\\ \delta z(k) &= z(k) - z(k-1) = \delta t\dot{z}(k)\end{aligned}$$

$$\left.\begin{aligned}\delta x_n(k) &= x_n(k) - x_n(k-1) = \delta t v_{sx}(k)\\ \delta y_n(k) &= y_n(k) - y_n(k-1) = \delta t v_{sy}(k)\\ \delta z_n(k) &= z_n(k) - z_n(k-1) = \delta t v_{sz}(k)\end{aligned}\right\} n = 1, 2\ldots N$$

and where

$$\begin{aligned}c_{n1}(k) &= \left.\frac{\partial R_n}{\partial x}\right|_{R_n(k-1)} = \frac{x(k-1) - x_n(k-1)}{R_n(k-1)} \simeq -\frac{x_n(k-1)}{R_n(k-1)}\\ c_{n2}(k) &= \left.\frac{\partial R_n}{\partial y}\right|_{R_n(k-1)} = \frac{y(k-1) - y_n(k-1)}{R_n(k-1)} \simeq -\frac{y_n(k-1)}{R_n(k-1)}\\ c_{n3}(k) &= \left.\frac{\partial R_n}{\partial z}\right|_{R_n(k-1)} = \frac{z(k-1) - z_n(k-1)}{R_n(k-1)} \simeq -\frac{z_n(k-1)}{R_n(k-1)}\end{aligned}$$

The above approximations for the direction cosines $c_{ni}(k)$ are very close provided the vehicle is close to the origin of the x, y, z coordinate system because the satellite positions are large compared with the vehicle position.

The velocity vector for satellite n is $\bar{v}_s$ is given in terms of its Cartesian coordinates:

$$\bar{v}_s(k) = \left[v_{sx}(k), v_{sy}(k), v_{sz}(k)\right]^T$$

Each satellite orbit is known with sufficient accuracy that the components of $\bar{v}_s$ are known at the vehicle receiver. The linearized measurement equation for N satellites can be written in the following matrix form:

$$\bar{z}(k) = H(k)X(k) + n(k)$$

where

$$\begin{aligned}\bar{z}(k) &= [z_1(k), z_2(k)\ldots z_n(k)]^T\\ z_n(k) &= R_n(k) - R_n(k-1) \quad n = 1,2,\ldots,N\end{aligned}$$

and where $H(k)$ is given by

$$H(k)=\delta t\begin{bmatrix}0&0&0&c_{11}(k)&c_{12}(k)&c_{13}(k)&-c_{11}(k)&-c_{12}(k)&-c_{13}(k)\\0&0&0&c_{21}(k)&c_{22}(k)&c_{23}(k)&-c_{21}(k)&-c_{22}(k)&-c_{23}(k)\\\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&\vdots\\0&0&0&c_{N1}(k)&c_{N2}(k)&c_{N3}(k)&-c_{N1}(k)&-c_{N2}(k)&-c_{N3}(k)\end{bmatrix}$$

The state vector for the Kalman filter is augmented to account for the satellite motion and is given by

$$\bar{X}(k)=\left[x(k),y(k),z(k),\dot{x}(k),\dot{y}(k),\dot{z}(k),v_{sx},v_{sy},v_{sz}\right]^T$$

The state-transition matrix F is given by

$$F=\begin{bmatrix}1&0&0&\delta t&0&0&-\delta t&0&0\\0&1&0&0&\delta t&0&0&-\delta t&0\\0&0&1&0&0&\delta t&0&0&-\delta t\\0&0&0&1&0&0&0&0&0\\0&0&0&0&1&0&0&0&0\\0&0&0&0&0&1&0&0&0\\0&0&0&0&0&0&1&0&0\\0&0&0&0&0&0&0&1&0\\0&0&0&0&0&0&0&0&1\end{bmatrix}$$

The Kalman filter provides the optimum linear estimate of $X(k)$ that is denoted as $\hat{X}(k)$ and is obtained recursively by the following model:

$$\hat{X}(k+1)=(F-K(k)H(k))\hat{X}(k)+K(k)\bar{z}(k)$$

where $K(k)$ = Kalman filter gain.

The derivation of $K(k)$ is given in Appendix E. The recursive estimates $\hat{X}(k)$ begin with an initial estimate $\hat{X}(0)$ that could, in principle, be obtained by trilateration, but other means are available depending upon the system configuration and the application.

The state vector $\hat{X}(k)$ for the Kalman filter GPS gives position and velocity of both the vehicle/receiver and the satellites. In vehicle navigation, the satellite data need not be provided (displayed) to the user. In this case, the model for the navigation output is normally only the vehicle position, although there are applications that provide vehicle speed. The output of the GPS/Kalman filter for vehicle position is the estimate of the position vector $\bar{P}(k)$ with the estimate denoted as $\hat{P}(k)$ and is given by

$$\hat{P}(k)=C\hat{X}(k)=[\hat{x}(k),\hat{y}(k),\hat{z}(k)]^T$$

where, for position only data, the matrix C is given by

$$C=\begin{bmatrix}1&0&0&0&0&0&0&0&0\\0&1&0&0&0&0&0&0&0\\0&0&1&0&0&0&0&0&0\end{bmatrix}$$

In most vehicles, the GPS position is displayed on a map that is presented on a flat-panel electronic display as described in Chapter 8 on instrumentation. In displaying position on the map, the vehicle

position must be converted from the ECEF Cartesian coordinates for the Kalman filter (if these coordinates were used) to map coordinates (e.g., latitude and longitude). This conversion is accomplished via a matrix transformation.

Although the elevation z relative to the origin is available in the $\hat{P}(k)$ vector, for land vehicle navigation, there is no need to display elevation. Moreover, the flat-panel display is two-dimensional. The transformation of $\hat{P}(k)$ in ECEF Cartesian coordinates to map coordinates $\hat{P}(\text{map})$ involves a simple model:

$$\hat{P}(\text{map}) = M\hat{P}(\text{ECEF}) + T$$

where M is a transformation matrix and T is a vector that translates the converted vector to the appropriate origin of the map coordinates.

In Appendix E, which derives the equations for the GPS estimate of vehicle position, there is a simulation of the estimate of the x,y positions of a vehicle over a coordinate plane tangent to the earth. For this simulation, a vehicle is traveling along a road with a curve initially at freeway speed and then gradually slowing. For illustrative purposes, this simulation includes four satellites within the line of sight of the vehicle. Fig. 9.19 is a plot of the true vehicle position depicted by a solid curve, the GPS-estimated position is a dashed curve, and a series of position estimates based upon trilateration are presented with the symbol **+**. The vehicle true vector position is denoted as $\bar{P}(k)$ where $k = 1,2,\ldots,240$. The GPS position estimate error (in ft) is denoted as $e(k)$ where $e(k)$ is given by the following norm of difference between true and estimated position vectors:

$$e(k) = \left\| \bar{P}(k) - \hat{P}(k) \right\|.$$

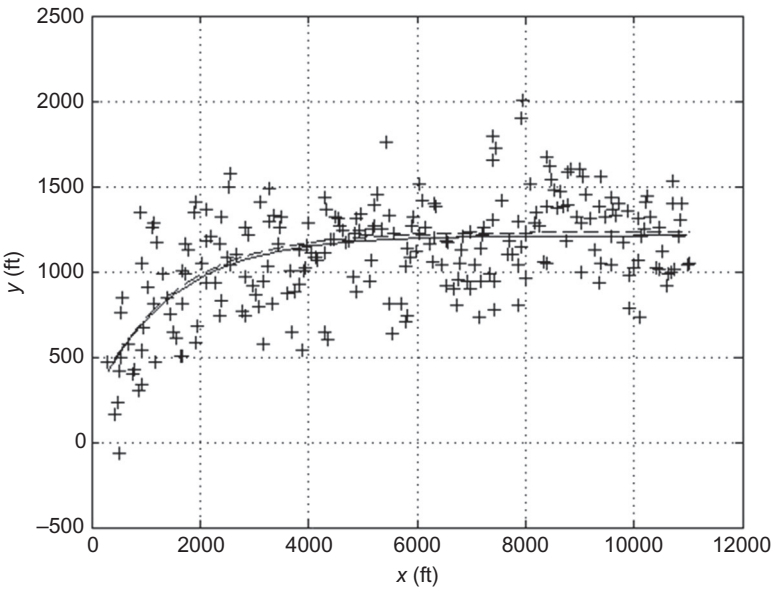


FIG. 9.19 Vehicle position: true solid line, GPS estimate dashed-line trilateration +.

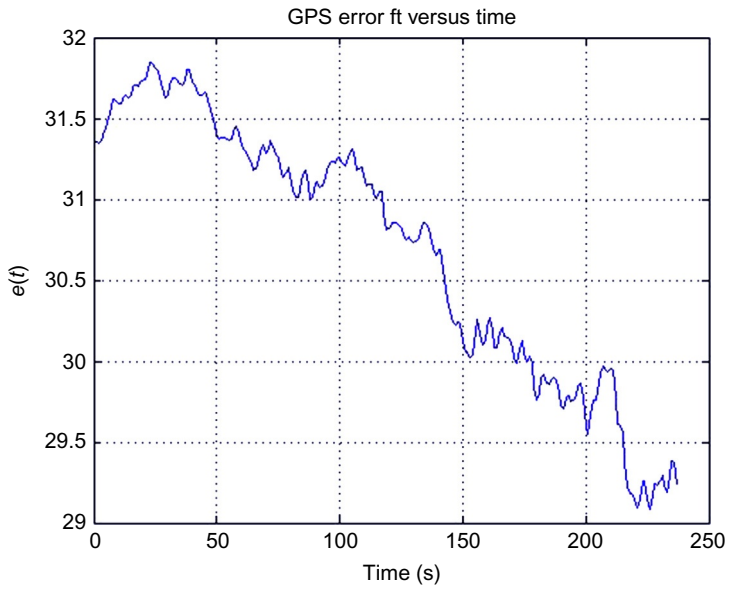


FIG. 9.20 GPS error feet versus time.

In all cases, the GPS-estimated error is very much smaller than the trilateration estimate errors. Fig. 9.20 is a plot of error in the Kalman filter estimate $e(k)$ for the simulation. For the example of random error in the simulation, the RMS error in trilateration is 178 ft. For the Kalman filter, the error begins at about 32 ft and asymptotically approaches 29 ft with an RMS of about 1 ft. The numerical values obtained for these errors are not necessarily represented by an actual GPS receiver or trilateration because the hypothetical error was taken simply to illustrate the Kalman filter. On the other hand, the relationship between the errors is representative of the relative error magnitudes for the two methods of obtaining vehicle position from satellite pseudorange measurements.

THE GPS SYSTEM STRUCTURE

The structure of the GPS navigation system consists of three major segments: the space segment (the satellites), the control segment, and the user receiver systems. The satellite must be capable of transmitting its position and the correct GPS time continuously. A major function of the control segment is to periodically upload to each satellite data from which this position can be computed. Periodic updates to these ephemeris data are required owing to orbital perturbations and changes due to lunar-solar perturbations; drag due to particulate matter at orbital distances; asphericity of the Earth's gravitational potential; and magnetic, static electric forces in orbit.

The control segment configuration is depicted in Fig. 9.21. The monitor stations receive the GPS signals (same as a mobile user). These signals can be used to evaluate ephemeris errors and satellite clock errors. These stations are located at Colorado Springs, Kwajalein, Diego Garcia, Ascension Island, and Hawaii. These stations measure pseudorange values form the satellites as they come into

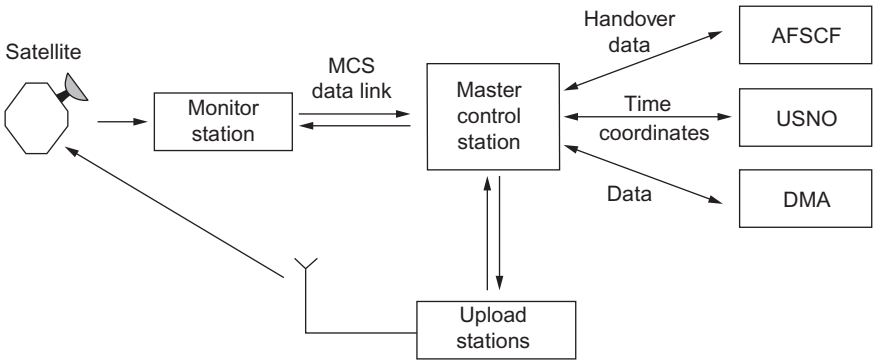


FIG. 9.21 GPS control configuration.

view. These measurements are used to determine ephemeris and clock errors. In addition, these stations monitor local meteorologic data that are useful for correcting for tropospheric delays. The data and corrections obtained by these monitor stations are sent to the master control.

The master uploads navigation messages to the satellites via the stations at Ascension, Diego Garcia, and Kwajalein. The satellites are continuously controlled via the master control to avoid cumulative errors that would occur in the absence of this control function.

There are numerous error sources in GPS navigation solutions, including satellite ephemeris errors, propagation errors and uncertainties, and clock errors. These errors are exacerbated by poor geometry, which increases the uncertainty in position. Such uncertainty is represented quantitatively by a parameter known as geometric dilution of position (GDOP).

The ephemeris errors result from imperfect prediction of satellite position. Propagation errors and uncertainties result from ionospheric and tropospheric refraction index variation. The ionospheric refraction is determined largely by free-electron density and carrier frequency. The index of refraction, n, for propagation through the ionosphere is defined as

$$n = \frac{c_o}{v_\phi}$$

where v_ϕ is the phase velocity and c_o is the vacuum speed of light.

At any carrier frequency, f, the index of refraction is given by

$$n = \sqrt{1 + \left(\frac{f_c}{f}\right)^2}$$

where f_c is the plasma frequency:

$$f_c = \frac{1}{2\pi}\sqrt{\frac{N_e e^2}{m\varepsilon_0}} \cong 9\sqrt{N_e}$$

where N_e is the electron density (number/m^3), e is the electron charge, m is the mass of electron, and ε_0 is the permittivity of free space.

On the other hand, tropospheric index of refraction is independent of carrier frequency but is influenced by the partial pressure of water vapor in an approximate relationship as represented by index of refraction, n:

$$n \cong 1 + \frac{K_1}{T}\left(p + \frac{K_2 p_w}{T}\right)$$

where p is the atmospheric pressure, T is the absolute temperature, p_w is the partial pressure of water, and $K_{1,2}$ are constants.

The path length change due to refraction ΔL is given by

$$\Delta L = \int_0^R (n-1)\,ds,$$

where R = distance to the satellite, $\cong$ pseudorange, and s = coordinate along the propagation path.

This expression can be rewritten approximately in terms of receiver altitude h_0 and elevation angle ϕ_0 to the satellite relative to the horizon:

$$\Delta L = \int_{h_0}^{H} \frac{n-1}{\sin\phi_0} dh$$

where H is the satellite elevation above the Earth. If the atmosphere is assumed to be exponential, then

$$n - 1 \cong (n_0 - 1)e^{-bh}$$

where (for a standard day)

$$n_0 \cong 1.00032$$
$$b \cong 0.000145/\text{m}$$

Typical values are

$$\Delta L \cong 2.2\,\text{m} \quad \text{for} \quad \phi_0 = 90\,\text{degrees}$$
$$\cong 25\,\text{m} \quad \text{for} \quad \phi_0 = 5\,\text{degrees}$$

The influence of satellite geometry is given via the GDOP. The GDOP can be computed from the matrix G formed from the direction cosines to the satellite that are in H. The nth row of G is denoted as G_n:

$$G = [G_1, G_2, \ldots G_N]^T$$

where

$$G_n = [c_{n1}, c_{n2}, c_{n3}] \quad n = 1, 2 \ldots N$$

$$\text{GDOP} = \left[\text{trace}\left\{\left[G^T G\right]^{-1}\right\}\right]^{\frac{1}{2}}$$

With a given value for GDOP, the RMS error is given by

$$\sigma = \text{GDOP}\,\sigma_o$$

where σ_o is the minimum position error that results for optimal satellite geometry and is due to the error sources listed above. The review given here in this chapter of GPS theory is only an overview. The actual theory is much more involved than can possibly be presented in the present book. However, there are numerous publications that explain GPS theory in a much more advanced way for the interested reader.

SAFETY ASPECTS OF VEHICLE-TO-INFRASTRUCTURE COMMUNICATION

One of the major issues in telematics is how to present the information and services that are potentially available to the driver without distraction from the driving tasks. Of course, the various services can be made available to passengers without necessarily distracting the driver. For example, video monitors in rear seats can provide entertainment, game playing on any standard computer Internet terminal via onboard DVD, or wireless connection, be it cell phone or satellite links.

On the other hand, the use of any subsystem that provides information such as is described above is potentially distracting to the driver. The simple act of dialing a standard cell phone requires the use of at least one hand and at least a momentary look at the cell phone. Some state legislatures are passing laws prohibiting the driver's use of a standard cell phone while driving.

The driver's distraction through cell phone use is somewhat alleviated by voice-activated cell phone dialing, which is available in some vehicles and in which the cell phone user verbally gives the phone number, speaking each digit separately. Included within the cell phone dialing system is a very sophisticated algorithm for recognizing speech. Speech recognition software identifies spoken words or numbers based on patterns in the waveform at the output of a microphone into which the user speaks. There are two major categories of speech recognition software: speaker-dependent and speaker-independent.

Speaker-dependent software recognizes the speech of a specific individual who must work with the system. The user is prompted to say a specific digit a number of times until the software can reliably identify the waveform patterns associated with that particular speaker. By this process, the system is "trained" to the individual user. It may not be capable of recognizing other users to whose speech it has not been trained.

Speaker-independent voice recognition software can recognize spoken digits regardless of the user. It is generally more sophisticated than speaker-dependent speech recognition. Unfortunately, it is also prone to recognition errors in excess of the speaker-dependent systems.

The cell phone connection can also be used to provide online navigation or other services by contacting a service with operators trained to provide this type of service. Alternatively, the cell phone can be used to provide an Internet connection to an online navigation service that transmits data to the car for display on an electronic map. In addition, it is well known to users of smartphones that, with built-in GPS and electronic maps, they can have the capability of providing navigation directly to the user, both visually and (synthesized) verbally.

A concierge service such as "OnStar" provides the capability of completely hands-free telephone connection and is an alternative to voice recognition for verbal cell phone dialing. The driver (or other occupants) can signal OnStar via a single push button. An operator receives the phone number to be dialed verbally and can complete a phone connection. The driver can complete a phone call without ever having to divert his/her attention from driving.

In the event of an accident, a vehicle that also has a GPS navigation system can alert the operator of such a service system of the accident and relay car coordinates such that emergency vehicles can be directed to the accident scene without requiring intervention or verbal communication with any occupant. The sensing of a crash can be accomplished via the sensor used for airbag deployment or via a dedicated independent crash sensor.

At the time of the writing of this book, there is a movement within government agencies, private industry, and academia to develop initial concepts and standards for a communication infrastructure for vehicles that will be required for safe operation of autonomous vehicles. The US DOT has advanced funding for a relatively large *Connected Vehicle Implementation Pilot* program. Such a system also can provide vehicle-to-vehicle communication of data related to safety. Of course, verbal vehicle-to-vehicle communication is well established via cell phones. However, autonomous or semi-autonomous vehicles in relatively close proximity can intercommunicate data such as position vector velocity and routing. The exact future of this type of vehicle-to-infrastructure and vehicle-to-vehicle communication is still in a developmental phase and cannot be discussed in any greater detail at the time of this writing.

One of the key elements in the development of vehicular communications including IVN, V2I, and/or V2V is the software for controlling all systems. One of the important open software standards is AUTOSAR. As vehicular communication technology expands (as it inevitably will do), there will be a great increase in the software. With AUTOSAR-based software, there is significant code reusability and portability that helps reduce software development costs. AUTOSAR is explained in detail in Chapter 3. There are a number of commercially available tools for AUTOSAR that support embedded software in the various IVNs. These can readily be found on an Internet search for the interested reader.

CHAPTER

ELECTRONIC SAFETY-RELATED SYSTEMS

10

CHAPTER OUTLINE

Airbag Safety Device 505
Blind Spot Detection 512
Automatic Collision Avoidance System 515
Lane Departure Monitor 521
Tire Pressure Monitoring System 522
Enhanced Vehicle Stability 524

Safety aspects of any vehicle can be classified into two major areas. The first to be discussed in this chapter involves occupant protection in the event of an accident, for example, collision with another vehicle or a large object. The second area involves techniques to avoid such accidents including blind spot object detection, collision avoidance, and enhanced vehicle stability (EVS).

The first topic discussed in this chapter is airbags, which are an important supplement to seat belts for occupant protection. This section of the chapter reviews the basic theory of operation of airbag systems and explains some of the significant improvements in these systems with respect to maximizing occupant protection.

AIRBAG SAFETY DEVICE

An airbag is one of the major electronics-based occupant protection devices. An airbag is a gasbag that is stored at specific locations within the vehicle in an empty collapsed configuration. In its stored state, it is covered by normal automotive interior panels that are readily ruptured upon airbag inflation. The theory of operation is perhaps best begun with an explanation of the earliest airbags that were deployed in some production automobiles in the 1970s. In such early vehicle deployment, the focus was on protection of the driver from impact with the steering wheel in the event of a frontal (or nearly frontal) collision.

An airbag system consists of the airbag itself, an inflation device, a digital control system, and one or more sensors. During an essentially frontal collision, the vehicle experiences a very large deceleration as the vehicle is crushed by the impact forces associated with a collision. The earliest sensors employed by airbag systems were electromechanical switches, such as are depicted in Fig. 10.1, that were normally open but were closed whenever deceleration reached a predetermined level that was

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-12-810434-7.00010-7

Copyright © 2017 Elsevier Inc. All rights reserved.

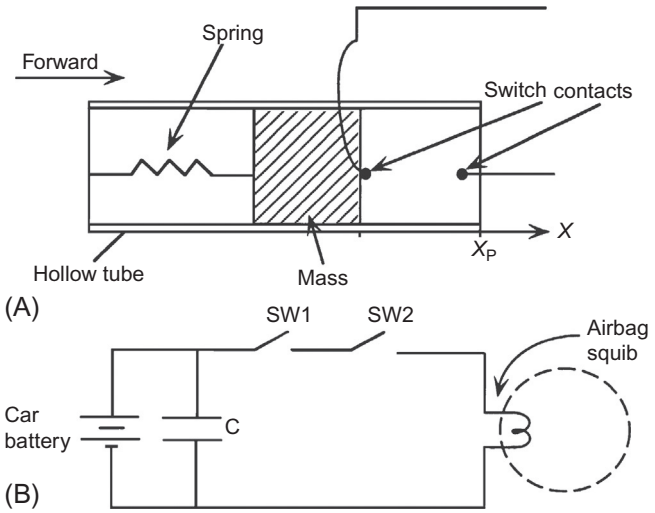


FIG. 10.1 Airbag deployment system. (A) Traditional airbag switch configuration. (B) Circuit for airbag activation.

associated with collision dynamics. At this level of deceleration, an electric signal was sent to the igniter of the airbag inflator. In the earliest configurations, the airbag was fully inflated in 30–40 ms.

The concept of an airbag system is that an inflated airbag can act as a cushion that can isolate or partially isolate occupants from an impact with various body parts with the internal vehicle structure. Initially, during the early 1970s, it was believed by many in the industry that airbags might have become a substitute or replacement for seat belts since data had suggested that seat belt use generally was relatively low. At the present time, airbags are considered a supplementary safety system to the primary system of seat belts by the automobile industry and are required by regulatory authorities.

Airbag deployment in vehicles has evolved from the initial steering wheel locations intended for driver protection. Vehicles were later provided with airbags for front seat passengers and then along vehicle sides including the side doors and the side curtain airbag to provide protection in the event of a side impact. Other airbag locations and configurations tend to be somewhat manufacturer-specific, the details of which are beyond the scope of this book. Rather, this book is intended to explain the details of the electronic components and the theory of operation.

The practical implementation of the airbag has proved to be technically challenging. At car speeds that can cause injury to the occupants, the time interval for a crash into a rigid barrier from the moment the front bumper contacts the barrier until the final part of the car ceases forward motion is of the order of a second. Table 10.1 lists required airbag deployment times for a variety of test crash conditions.

For an understanding of the conceptual framework of an airbag electronic system, it is, perhaps, helpful to briefly review a somewhat simple example of an early airbag concept. The earliest concepts involved protecting the driver and front seat passenger from essentially frontal-only collisions.

One of the early configurations that was intended to protect occupants from longitudinal axis deceleration employed a pair of acceleration switches SW1 and SW2 as depicted in Fig. 10.1. Each of these switches is in the form of a mass suspended in a tube with the tube axis aligned parallel to the

Table 10.1 Airbag deployment times

Test library event	Required deployment time (ms)
9 mph frontal barrier	ND
9 mph frontal barrier	ND
15 mph frontal barrier	50.0
30 mph frontal barrier	24.0
35 mph frontal barrier	18.0
12 mph left angle barrier	ND
30 mph right angle barrier	36.0
30 mph left angle barrier	36.0
10 mph center high pole	ND
14 mph center high pole	ND
18 mph center high pole	ND
30 mph center high pole	43.0
25 mph offset low pole	56.0
25 mph car to car	50.0
30 mph car to car	50.0
5 mph curb impact	ND
20 mph curb drop-off	ND
35 mph Belgian blocks	ND

Note, ND = nondeployment.

longitudinal car axis. Fig. 10.1A is a drawing of the accelerator switch configuration. Fig. 10.1B is a circuit diagram for an early airbag system.

The two switches, which are normally open, must both be closed to complete the circuit for firing the airbag. When this circuit is complete, a current flows through the igniter that activates the charge. A gas is produced (essentially explosively) that inflates the airbag.

In early airbag systems, the switches SW1 and SW2 are placed in two separate locations in the car. Typically, one is located near the front of the car and one in or near the front of the passenger compartment (some automakers located a switch under the driver's seat on the floor pan).

Referring to the sketch in Fig. 10.1, the operation of the acceleration-sensitive switch can be understood. Under normal driving conditions, the spring holds the movable mass against a stop, and the switch contacts remain open. During a crash, the force of acceleration (actually deceleration of the car) acting on the mass is sufficient to overcome the spring force and move the mass. For sufficiently high car deceleration, the mass moves forward to close the switch contacts. In a real collision at sufficient speed, both switch masses will move to close the switch contacts, thereby completing the circuit and igniting the chemical compound to inflate the airbag.

An approximate dynamic model for the mechanical crash sensor is given below:

$$M\ddot{x} + D\dot{x} + F_c + Kx = 0 \tag{10.1}$$

where M = mass of the movable element, D = viscous friction coefficient, F_c = coulomb friction force (stiction), and K = spring constant.

The acceleration of the mass $(\ddot{x})$ is related to vehicle acceleration (a) or deceleration $(-a)$ by the following:

$$\ddot{x} = -a$$

The motion of the movable mass is the solution to the following:

$$D\dot{x} + F_c + Kx = Ma \tag{10.2}$$

Whenever the mass displacement exceeds the spacing to the switch contact (x_p) (i.e., $x = x_p$), the contacts close, and the action described above proceeds.

Fig. 10.1 also shows a capacitor connected in parallel with the battery. This capacitor is typically located in the passenger compartment. It has sufficient capacity that, in the event the car battery is destroyed early in the crash, it can supply enough current to ignite the squib.

The evolution of airbag sensing technology advanced to electronic sensor. In such systems, the role of the acceleration-sensitive switch is played by an analog accelerometer along with electronic signal processing, threshold detection, and electronic driver circuit to fire the squib. Fig. 10.2 depicts a block diagram of such a system.

The accelerometers A1 and A2 are placed at locations similar to where the switches SW1 and SW2 described above are located. Each accelerometer outputs a signal that is proportional to acceleration (deceleration) along its sensitive axis. As an illustrator of the characteristic waveform from an accelerometer, Fig. 10.3 presents measurements of a 3200 lb (curb weight) vehicle that was crashed into a rigid barrier at 30 mph.

Under normal driving conditions, the acceleration at the accelerometer locations is <1 g. However, during a collision at a sufficiently high speed, the signal increases rapidly. Signal processing can be employed to enhance the collision signature in relation to the normal driving signal. Such signal processing must be carefully designed to minimize time delay of the output relative to the collision deceleration signal. A comparison of the deceleration profile of Fig. 10.3 for this crash with the deployment requirements of Table 10.1 illustrates the complexity of the signal processing necessary to properly deploy the airbag.

As explained above, the original electromechanical deceleration sensors have been replaced with solid-state accelerometers. In addition, airbag systems are equipped with other sensors including

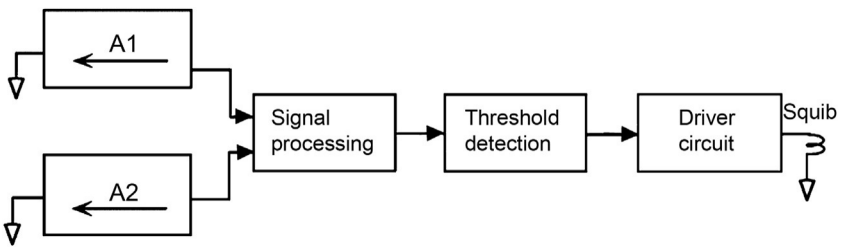


FIG. 10.2 Accelerometer-based airbag system.

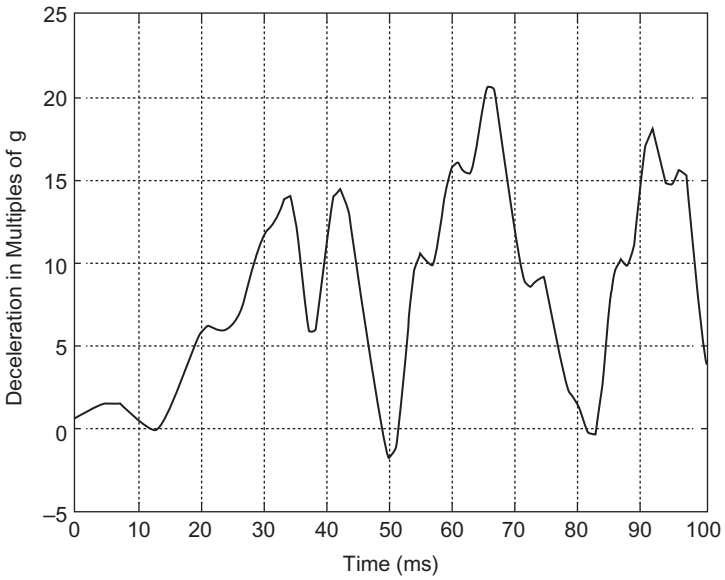


FIG. 10.3 Acceleration data for 30 mph crash.

sensors for side door pressure, wheel angular speed, brake pressure, seat occupancy, and gyroscopic sensors for angular position/rate measurement and impact sensors. The electronic control for an airbag system receives and processes signals from all of the associated sensors. The algorithms for this signal processing are relatively complex since the central system must distinguish signals due to a collision from signals that are produced during normal driving. In the latter case, the signals from the sensors depend upon the vehicle speed, the condition of the road (or off-road terrain), and sometimes vehicle maneuvers resulting from driver input. For example, a vehicle traveling over a road with significant potholes or a vehicle striking a curb during an improperly executed turn can involve rapid vehicle dynamics that result in signals from the sensors that can emulate a collision.

Regulatory requirements for airbag deployment are usually specified in terms of a specific collision. For example, the frontal collision into a rigid barrier at a specific speed (e.g., ∼14 mph) requires airbag deployment. Such a collision will cause a very specific set of signals from the various sensors for a particular vehicle model and/or configuration. If the threshold for airbag deployment was set too low and if the driver's airbag was deployed when the vehicle hit a severe pothole (or curb), the occupant safety would be reduced because the airbag would obstruct the driver's visibility and could impact his/her ability to control the vehicle.

As an illustration of the complexity of the signal processing required by the airbag control system, Fig. 10.3 depicts the actual output signal waveform of an accelerometer during a frontal barrier collision for an actual vehicle. In addition to signal waveform complexity, the detection of a collision must occur within a few milliseconds of the initial contact of the vehicle with the colliding object. The time for full inflation of a representative airbag from the trigger (igniter) signal is 30–40 ms. The vector

magnitude and orientation of the net deceleration of the portion of the vehicle where sensors are located can be determined by combining signals from accelerometers.

As a simple illustrative example, we consider a pair of accelerometers located under the driver seat with one having sensitive axis along the vehicle's x-axis; this is its longitudinal axis (i.e., fore and aft). Another accelerometer is oriented with its sensitive axis along the transverse horizontal axis (denoted y-axis). For illustrative purposes, it is assumed that these sensors are linear with output voltages v_x and v_y, respectively as explained in Chapter 5, and given by

$$v_x = K_s a_x$$
$$v_y = K_s a_y$$

A third accelerometer can be placed at the same location and oriented along the vehicle's third (i.e., z) axis yielding voltage v_z where

$$v_k = K_s a_z$$

These accelerations are given in terms of a convenient inertia reference, which, in most cases, is earth-fixed with x and y parallel to the vehicle axes at the onset of the collision and parallel to the road surface. During the course of a collision, a rotation of the vehicle axes can occur as a function of the vehicle object orientation at impact. However, during the initial phase of the collision within which the airbag is to be deployed, this rotation of axes is sufficiently small that it doesn't have to be taken into account.

The magnitude of the collision is represented by the magnitude a of the total acceleration vector $\bar{a}$ and is given by

$$\bar{a} = \left[a_x a_y a_2\right]^T$$
$$a = \|\bar{a}\|$$

The angle of $\bar{a}$ in the transverse plane (x,y) is denoted θ_t and is measured relative to vehicular x-axis and given by

$$\theta_t = \tan\left(a_y / a_x\right)$$

For an essentially frontal collision, $\theta_t = \pi$. When the control unit detects from a that a collision is in progress, the driver and passenger airbags will deploy.

The signal processing for crash sensing involves algorithms that perform transformations on samples of the output of the various sensors. The waveform amplitude from any given sensor (e.g., accelerometer) is an increasing function of the severity of the crash. For example, the crash intensity into a barrier of a given vehicle model increases with impact speed. The waveform of any of the accelerometers also depends upon not only the impact speed but also the lateral impact point and is different for different vehicle models. The waveform of the output of the sensors such as the accelerometers contains the information that must be extracted to detect a crash and to distinguish a crash from a noncrash-related impact (e.g., pothole) within a sufficiently short time interval to deploy the airbag when needed. As an illustration of the signal processing used for crash detection, consider the output of an accelerometer that has its sensitive axis along the vehicle's x-axis. The sampled output of this sensor is denoted x_k and is given by

$$x_k = K_s(a_x(t_k) + n_v(t)) \tag{10.3}$$

where $t_n = k\delta t \quad k = 0,1,2,\ldots$

n_v = noise generated due, in part, to vehicle vibration.

One of the simplest, but not highly reliable, algorithms for crash sensing is to compare the sensor data with a threshold value. For example, a frontal barrier crash involves a_x measurements. A potential algorithm is given below

$$x_k < x_{Th} \rightarrow \text{no airbag activation}$$
$$x_k \geq x_{Th} \rightarrow \text{airbag inflation procedure}$$

where x_{Th} = threshold level for airbag inflation.

Another signal processing that can improve performance of the crash detection algorithm is filtering. The sequence of sampled data $\{x_k\}$ is the input to a filter that generates output sequence y_n where

$$y(n) = \sum_{k=n-k}^{n} h_{n-k} x_k \tag{10.4}$$

where $h_{n\text{-}k}$ = impulse response of the filter:

$$y(n) = y_a(n) + y_{nv}(n)$$

The filter output consists of a signal component $y_a(n)$ that optimally replicates the sampled version of the vehicle acceleration (scaled in amplitude) and a noise component $y_{nv}(n)$ that is ideally of negligible amplitude. Depending upon the signal waveform, its spectral content, and the noise power spectral density, there are certain optimal filters that can maximize the signal/noise of the measured acceleration.

It is possible in crash detection algorithms involved in airbag systems to simultaneously process the various sensors. In such a system, frontal collisions for a given vehicle model might have a relatively narrow range of waveforms associated with a frontal crash at a given vehicle speed at the time of the crash. In this case, a matched filter (MF) with impulse response coefficients selected from stored values in memory for a given interval of speed can optimize crash signal detection. The theory of a MF is explained in the section of this chapter devoted to vehicular radar for crash avoidance. A MF can be an optimal signal processing for detecting the waveform associated with a frontal crash. MFs for certain other crashes are theoretically possible.

For a pure side impact, $\theta_t \pm \pi/2$. However, for a purely side impact in which another vehicle collides with the one being considered, a door pressure sensor could be expected to give the earliest indication of the collision. In this case, the side airbags will be inflated. The crash data from National Highway Transportation Safety Administration (NHTSA) show that the majority of collisions are neither fully frontal nor fully side impacts. The actual inflation of the airbag can be conducted in one or two stages to obtain an optimum bag position relative to the occupant. Any given crash scenario for a given vehicle configuration can have an optimal airbag deployment as controlled by the airbag control system for the particular collision as determined by vehicle sensors and signal processing algorithms.

There are some vehicles that incorporate pneumatic cylinders that, when pressurized, increase seat belt tension. Such cylinders receive pressure from canisters that contain airbag inflation materials. By increasing seat belt tension, the occupant is restrained from impact-induced motion. Such restraint can reduce any impact-related occupant/airbag impacts.

One of the algorithms used for airbag activation/inflation is based upon vehicle speed. This algorithm continuously monitors change in vehicle speed δV over an interval T. In terms of continuous time variables, $\delta V_T(t)$ is found by integrating acceleration along the vehicle's x-axis. The model for $\delta V_T(t)$ at any time is given by

$$\begin{aligned}\delta V_T &= \int_{t-T}^{t} a_x(t')dt' \\ &= \frac{1}{K_s}\int_{t-T}^{t} \mathrm{v}_x(t')dt'\end{aligned} \tag{10.5}$$

The crash detection algorithm involves comparing $\delta V_T(t)$ with a threshold value δV_{Th} yielding a binary-valued variable A where

$$\begin{aligned}A &= 0 \quad \text{if } \delta V_T > \delta \mathrm{v}_{th} \\ A &= 1 \quad \text{if } \delta V_T \leq \delta \mathrm{v}_{th}\end{aligned} \tag{10.6}$$

where v_{th} = threshold value:

$$\delta \mathrm{v}_{th} < 0$$

A representative algorithm for deployment of the airbag is represented in the present illustration by a binary-valued variable denoted D. A third binary-valued variable is the vehicle speed $V(t)$ where

$$\begin{aligned}B &= 0 \quad V < 14\,\text{mph} \\ B &= 1 \quad V \geq 14\,\text{mph}\end{aligned} \tag{10.7}$$

The airbag inflation activation occurs at the instant D changes from 0 to 1 where (in Boolean algebra notation)

$$D = A \cdot B \tag{10.8}$$

Thus, the decision to deploy requires $\delta V_T(t)$ to be algebraically less than the threshold value, and the vehicle speed must equal or exceed 14 mph in this representative example crash detection algorithm. The particular values for parameters T and δV_T are vehicle-specific.

Many vehicles include roll sensing via an angular rate/position sensor equivalent to gyroscopic sensors. A solid-state angular rate sensor is explained in detail in Chapter 5. For an accident involving a vehicle that rolls about its longitudinal axis due to lateral acceleration (e.g., in a very tight turn/cornering) acting through a relatively high cg, the sensor and control unit can detect when the vehicle is in a rollover maneuver. If the vehicle is equipped with side curtain airbags, these will be deployed during the rollover accident. Such airbags, in addition to the all-important seat belts, can offer significant occupant protection.

BLIND SPOT DETECTION

Another vehicular electronic safety-related system is known as "blind spot detection" (BSD). The term refers to a deficiency in the traditional means of attempting to give drivers a full 360° viewing capability around the vehicle. The driver's primary viewing attention when traveling forward is the region in front of and somewhat to either side of the vehicle. Although vehicles are designed and built with multiple windows, there are nontransparent portions of the vehicle necessary to provide the physical structure. For example, the occupant compartment includes the vehicle roof with supporting structures (e.g., A and B door posts). Basically, except for windows, the vehicle structure restricts the driver's field of view. The traditional devices intended to improve the field of view were the rearview and side-view mirrors. Although such mirrors have provided valuable assistance to the driver for the

regions covered by the mirrors, there have always been regions around the vehicle in which parts of the vehicle obstructed the driver's view. These regions have been termed "blind spots."

In this chapter, several electronic systems that effectively eliminate blind spots are discussed. These systems either can improve the driver's field of view or are capable of issuing warnings to the driver about other vehicles or objects that are a potential collision threat.

Safety in driving has many components including collision avoidance between two or more moving vehicles or between any given moving vehicle and a fixed or essentially immovable object. A necessary (but not always sufficient) requirement to avoid collisions is the need for the driver to detect a potential collision and, where possible, to take an evasive action by either steering or braking or both. Traditionally, the driver could only detect potential collisions visually. In most vehicles, the driver has good visibility forward and to either side. Visibility to the rear is aided by rearview and side-view mirrors. However, depending upon the vehicle structure, there can be obstacles that block visibility in certain directions and locations surrounding the vehicle. This section of the chapter discusses the means of dealing with these blind spots.

In contemporary vehicles, there are numerous electronic means to improve the driver's ability to detect objects in otherwise blind spots. One such spot is the region immediately behind the vehicle. When the vehicle is in reverse (e.g., backing out of a parking space), there is potential for running into an unseen obstacle. It is particularly hazardous if there are any small children walking behind a vehicle in reverse. One of the early electronic BSD systems is a rearview video system. Such a system includes a video camera (normally solid state) having a relatively large field of view. The video signals are sent to the digital system that controls the flat-panel display most likely via an in-vehicle network (IVN) (see Chapter 8). The flat-panel display, which is explained in Chapter 8 on *instrumentation*, can present the two-dimensional image as "seen" by the video camera (see Chapter 5). In most vehicles that are equipped with a rearviewing video system, the system is automatically activated whenever the vehicle is in reverse, and the vehicle flat-panel display is dedicated to presenting the rearview image.

In addition to visual object detection by the video system, some vehicles are equipped with ultrasonic object detection aids. Such systems incorporate multiple (two or more) ultrasonic transducers. The ultrasound carrier frequency for each transducer is unique. The corresponding control system generates signals that cause the transducers to emit short bursts of ultra-high-frequency sound. The beam width of the transmitted sonic pulse is sufficient to cover the region behind the vehicle for which a potential collision can occur. Any object within this region reflects the sonic pulse. The transducers receive the reflected pulses at the transmitted frequency and send the received signal to the control system. If the reflecting object has sufficiently small lateral dimensions, the reflected wave will have a pulse duration essentially the same as the transmitted pulse. The round-trip time from the transducer and from the object $t_n(k)$ for transducer n is given by

$$t_n = r_n / C_S$$

where r_n = distance to the object from transducer n and C_S = speed of sound.

For a single object, the location of the object in a vehicle-based Cartesian coordinate system can be found by trilateration, the theory of which is explained in Chapter 9 on *vehicle communication* in the section that explains GPS navigation.

If the object is within a region of potential collision, an alert message is given to the driver. For example, a relatively short narrow cylindrical post is often not readily visible to the driver via the

rearview video system, particularly in a low light or low color contrast or rain environment. If the system senses that the driver has not detected the object, the controller can highlight it on the display.

When a vehicle is traveling forward, there are BSD systems capable of alerting the driver to a potential collision For example, if a driver is changing lanes on a multilane road/highway and another vehicle is approaching in the destination lane in a blind spot, there is a potential for collision. There are technologies that can detect objects such as this example in a blind spot and issue a warning to the driver.

One such technology involves the use of low-power short-wavelength radar. A number of antennas mounted around the vehicle (primarily facing to the rear and side) can transmit pulses of microwave power from an internal transmitter. If there is a vehicle within the predetermined blind spot, the radar signal is reflected from this vehicle and received by the radar system. The radar system can measure the distance to another vehicle and its relative speed along the line of centers. The distance r to the other vehicle is given by

$$r = c\delta t \tag{10.9}$$

where δt = round-trip time from transmit to receive and c = speed of propagation of the transmitted radar pulse.

The relative speed S_r is obtained from the frequency shift of the received radar pulse:

$$S_r = c\frac{\delta f}{f_t} \tag{10.10}$$

where

$$\delta f = f_t - f_r$$

f_r = received frequency and f_t = transmitted frequency.

Assuming that the overtaking vehicle does not reduce speed and the given car continues a lane change, the time to collision δt_c is approximately given by

$$dt_c \simeq \frac{r}{S_r}$$

The criteria for alerting the driver can include numerous factors. For example, in the above case of radar detecting an overtaking vehicle, one criterion might be the time required for lane change to be very small compared with δt_C. In this exemplary scenario, a lane deviation/departure system can be employed. Such a lane change detection system is explained later in this chapter.

Alternative BSD systems employ optical sensing of vehicles in blind spots. At least one vehicle model incorporates a pair of cameras, each one mounted in a side-view mirror housing. The theory of operation of such a camera is explained in Chapter 5 and is based on the operation of an array of photo detectors, which are explained in Chapter 2. Each of these cameras has a field of view along the side of the vehicle rearward. This system is helpful for lane changing on a multilane road/highway. When the turn signal is switched on to indicate a lane change, the image from the camera on the side to which the vehicle is about to move is displayed on the flat-panel display screen. This display will reveal another vehicle in one of the blind spot locations in the lane to which the vehicle is moving. Another sensor system employed for BSD uses LIDAR, which is explained in Chapter 5.

An extension of the above system involves signal processing of the optical image. The algorithms and software for optical object detection have existed for decades and have been used to locate objects

on a conveyer belt for robot pickup in industry. If all vehicles had exactly the same shape, such algorithms could be used to detect vehicles in blind spots. However, the two-dimensional image contained in the signal from any camera of a vehicle in a blind spot depends upon the size and shape of the vehicle, as well as its distance from the camera. On the other hand, algorithms for detecting changes in the patterns of video camera signals can identify an object in a blind spot that is approaching the camera. Furthermore, the changes in the image can be processed to obtain an estimate of the speed of the approaching vehicle relative to the camera.

The details of object detection algorithms are beyond the scope of this book. However, a simple example can illustrate the types of signal processing that can be employed in BSD. This example involves detecting and locating an edge of an object. For a camera with color capability, the camera generates three signals for red R, green G, and blue B.

For illustrative purposes, we consider a simple rectangular green object on a white background. The electric green signal is a function of the Cartesian coordinates of the camera field of view and is denoted $V_G(H, V)$ where H represents the horizontal position and V the vertical position. The edges of this object are associated with discontinuities in the derivatives $G_H(H, V)$ and $G_V(H, V)$:

$$G_H(H, V) = \frac{dV_G}{dH}$$

$$G_V(H, V) = \frac{dV_G}{dV}$$

The edges of the object can be represented by contours $V_E(H_E)$ where the derivatives are discontinuous.

For BSD, some vehicle models have more than two cameras and can detect other vehicles in all of the blind spot locations for the vehicle. Advanced signal processing algorithms of the signals from these cameras go far beyond the simple edge detection algorithm discussed above. However, these algorithms are largely proprietary and are not available for discussion here. On the other hand, a publicly documented method of image detection known as *convolutional neural network* (CNN) has been used for object identification and, in principle, could be used for BSD. The CNN electronically performs operations on video data that are not unlike animal visual cortex in operation on the distribution of individual pixels.

AUTOMATIC COLLISION AVOIDANCE SYSTEM

Any time a vehicle is moving, there is the theoretical potential for a collision with another vehicle or with a fixed object. The traditional method of avoiding a collision has been proper driving by the driver. However, collisions occur most often, particularly with other vehicles, in relatively high traffic density areas.

Another electronic system that improves driving safety is a so-called collision avoidance system (CAS). A CAS incorporates millimeter wavelength radar and/or laser-based lidar for detecting a potential collision with another vehicle. Such a radar/lidar sensor functions the same as traditional radar in detective objects along a given path or direction. It does so by transmitting a relatively low beam width pulse and receiving reflected signals. Radar was initially developed during World War II for detecting and locating aircraft or ships at sea that could be carrying destructive weapons. For radar to be effective, it must distinguish between pulses that are reflected by the object/vehicle that is to be detected from

background radiation or electromagnetic noise. A reflected signal will have essentially the same pulse waveform as the transmitted pulse. The carrier frequency of a received signal, which is reflected by the object being detected, will differ from the transmitted carrier frequency by the Doppler shift due to motion of the reflected object. The received frequency shift due to motion δf is given by

$$\delta f = \frac{\dot{r}}{c} f_c \tag{10.11}$$

where

$$\dot{r} = \frac{dr}{dt}$$

r = distance from the radar antenna to the object and c = speed of propagation of the radar signal.

The lateral velocity of the moving object (i.e., velocity in a plane orthogonal to the line from the antenna to the object) v_l is given by

$$v_l = r\dot{\theta} \tag{10.12}$$

where

$$\dot{\theta} = \frac{d\theta}{dt}$$

θ = angular position of the object relative to a reference direction from the transmitting antenna.

A block diagram of a traditional radar system is depicted in Fig. 10.4.

In Fig. 10.4, the component labeled *Circ* is a circulator that isolates the highly sensitive receiver from the relatively powerful transmitted pulse. For a circulator, an input signal at port *a* is coupled with very low insertion loss to port *b*. Port *c* is virtually isolated from signals entering port *a*. However, an input to port *b* is coupled with low insertion loss to port *c*. With this component, a receiver capable of receiving low signal levels is prevented from saturation due to the transmitted pulse. An alternative to the traditional circulator is a switch (called a transmit/receive (TR) switch) that places a short circuit across the receiver input during transmission and connects the receiver to the antenna (Ant) when the

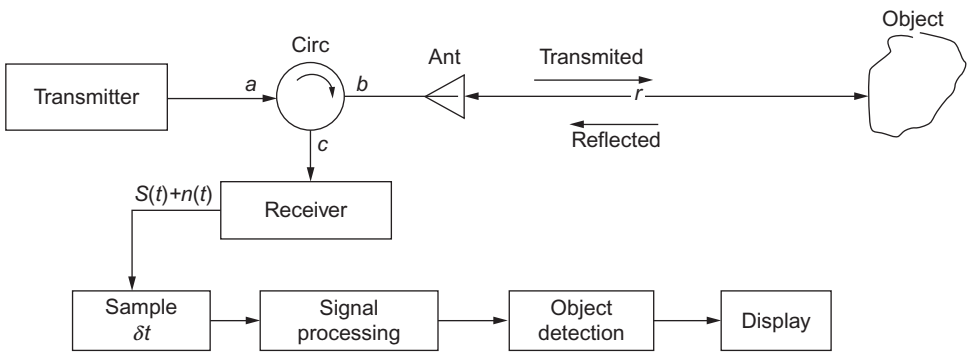


FIG. 10.4 Traditional radar block diagram.

transmitter pulse is ended. The time interval between successive transmitted pulses must be at least equal to the round-trip time for electromagnetic propagation from the antenna to the object at the maximum range for which object detection is required. For automotive frontal collision avoidance, the distance will normally be such that avoidance functions taken by the vehicle have sufficient time to be employed. In certain cases, this distance will be mandated by government agencies. However, the individual vehicle manufacturer may choose to use a maximum detection distance that exceeds government regulations.

Although the power level of transmitted radar signals is relatively low, the radar cross section of another vehicle with which a collision is possible is relatively large, and the reflected signal may be large relative to ambient noise. On the other hand, there are circumstances for which the reflected signal to ambient noise is insufficient for an early detection of potential collision object to have an acceptable probability of detection. Examples of situations with relatively low probability of early detection include inclement weather and relatively small-object radar cross section (e.g., small fixed posts near a road or objects with low reflectivity for millimeter wave radar).

In such cases, signal processing can be employed to maximize signal/noise, thereby enhancing probability of early detection of an object having the potential for collision. The traditional signal processing employed in radar for optimal object detection in the presence of noise is a so-called matched filter (MF). In collision avoidance radar, an MF would provide early detection of objects that might otherwise not have a sufficient probability of detection for acceptable vehicle safety.

The traditional MF was a continuous time analog device that would have as input the received radar reflected signal. Such a filter is designed with a transfer function (or impulse response) derived from the waveform of the signal that is to be detected. A contemporary MF is implemented in discrete time as a digital filter. As explained in Appendix B, a digital filter can be modeled in terms of the convolution of the input x_k with the sequence of impulse response coefficients h_k to produce an output y_n:

$$y_n = \sum_{k=-\infty}^{\infty} h_{n-k} x_k$$

The input consists of a signal component s_k and a noise component n_k:

$$x_k = s_k + n_k$$

where n_k is a stochastic random process that is uncorrelated with s_k.

In the case of a radar system, the transmitted signal is an amplitude-modulated carrier in which the modulation has a pulse waveform. The radar receiver demodulates the signal coming from the antenna. In this case, the signal sequence $\{s_k\}$ is a discrete time version of the envelope of the modulated signal that is sampled at a rate that is sufficient to detect the radar pulse waveform. The envelope of the amplitude-modulated transmitted signal is modeled by periodic waveform $S_T(t)$:

$$S_T(t+nT) = S_T(t) \quad n = 0, 1, 2 \tag{10.13}$$

where T = period of the modulating signal.

The received signal is a time-delayed reduced amplitude version of the transmitted signal. An ideal model for $S(t)$ is a rectangular pulse having waveform given by

$$\begin{aligned} S_T(t) &= S_o \quad 0 \le t \le \tau \\ &= 0 \quad \tau \le t \le T \end{aligned} \tag{10.14}$$

The received signal is modeled by

$$S(t) = \gamma S_T(t - T_R) \tag{10.15}$$

where γ = relative amplitude of the received signal, $T_R = 2R_o/C$ = round-trip time, R_o = range from antenna to object, and C = speed of propagation of the radar signal.

Fig. 10.5 depicts the transmitted and received signals for a radar system such as could be employed in a vehicle CAS. In this figure, the demodulated signal and noise ($S(t)+n(t)$) are sampled at time t_k yielding a discrete time input x_n to the signal processing:

$$\begin{aligned} x_k &= x(t_k) \\ &= S_k + n_k \end{aligned} \tag{10.16}$$

where

$$t_n = k\delta t \quad k = 0, 1, 2, \ldots, K$$

$S_k = S(t_n)$ = sampled signal, $n_k = n(t_k)$ = sampled noise.

The sample that is closest to the time T_R is denoted (k_o), which is the closest integer to the ratio $T_R/\delta t$:

$$k_o = \left\{ \left\lfloor \frac{T_R}{\delta t} \right\rfloor \right\}$$

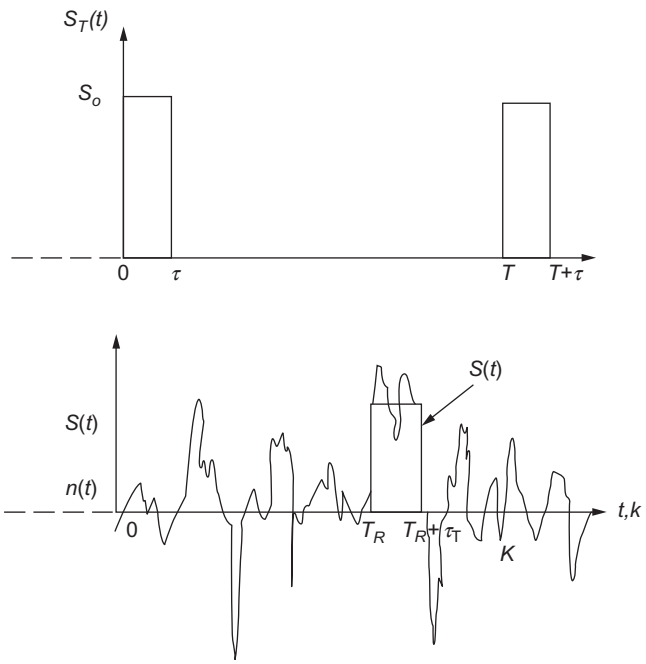


FIG. 10.5 Transmitted and received radar signals.

The signal processing filter generates output y_k, which consists of a signal component y_S and a noise-related component y_n given by

$$y_S(n) = \sum_{K=o}^{K} h_{n-k} S_k \quad n = 1,2,\ldots,K \tag{10.17}$$

$$y_n(n) = \sum_{k=o}^{K} h_{n-k} n_k \tag{10.18}$$

The output sequence $\{y_S(n)\}$ can be represented as a vector of length K denoted

$$\bar{y}_s = [y_S(1), y_S(2), \cdots, y_S(K)]^T$$

The output component $y_n(n)$ is a sequence of samples of a random process having standard deviation σ_n. The signal-to-noise ratio (SNR) for a radar cycle of duration T can be defined:

$$SNR = \frac{\bar{Y}_S^T \cdot \bar{y}_S}{\sigma_n^2} \tag{10.19}$$

The MF that maximizes SNR has impulse response coefficients $h_{mF}(k)$:

$$h_{mF}(k) = \alpha S_T(K-k)$$

where α is chosen such that the maximum values of $y_S(n)$ do not exceed the maximum digital word length of the digital system. In effect, the MF impulse response coefficients are a reversed time version of $S_T(k)$ (i.e., analogous to a mirror image).

It should be emphasized at this point that a MF or other signal processing is fundamentally required for a collision avoidance radar. Moreover, it has the capability to extend the early detection range of a potential collision object/vehicle. Early detection enables the system to initiate preparatory steps in case automatic collision avoidance should become necessary due to a lack of timely action by the driver.

The potential for a collision with a vehicle equipped with a CAS requires some form of tracking of the potential collision object/vehicle by the control system using continuous monitoring of radar signals. The details of collision object tracking algorithms are manufacturer-/model-specific and are largely proprietary. However, it is possible to present a hypothetical exemplary CAS with representative algorithms.

In order to be useful for collision avoidance in a vehicle, automotive radar must have the capability to scan a region in front of the vehicle. Such radar is also useful for adaptive cruise control. Traditional radar systems (e.g., those developed during WWII) achieve scanning by mechanically rotating the antenna. However, technology has been developed that achieves scanning electronically without requiring any mechanical means. One means of electronic scanning is the so-called phased array method. In this technology, a set of antennas that are properly sized for the carrier wavelength are arranged in an array. For an automotive radar, the antennas would be placed such that the radiation beam pattern is forward of the vehicle. These antennas are all driven from the same carrier signal source but at different relative phases. The combined radiation pattern is a relatively narrow beam whose major direction is determined by the relative phases of the antenna. By electronically varying the relative phases of the multiple antennas, the combined radiation beam can be electronically scanned to cover the region in front of the vehicle for locating objects or other vehicles in area that could potentially lead to collisions.

When the potential for a collision has been determined by processing the scanned radar return signals, the collision avoidance algorithms are activated. The initial phase of collision avoidance is for the system to issue warning messages to the driver verbally, visually, or both. If the driver takes no evasive action and the collision threat continues or worsens, the next phase of the collision avoidance involves automatic action by the system through either steering or braking (or both). Automatic steering input to the vehicle requires an actuator that is coupled mechanically to the vehicle steering system. Such an activator can be a hydraulic or pneumatic cylinder that moves under the influence of pressure in the cylinder. That pressure can be regulated by solenoid-type or motor-driven valves that can, in turn, be operated by the CAS. Automatic steering is explained in Chapter 12, which discusses autonomous vehicles.

Similarly, automatic brake application is readily possible on essentially any anti-lock brakes (ABS)-equipped vehicle via electrically controlled valves. We will illustrate a hypothetical braking-type collision avoidance algorithm with a very unusual but technically possible scenario. This example involves a vehicle that is equipped with a CAS that is traveling along a single-lane tunnel. Another vehicle entered the tunnel ahead of the example vehicle and is stopped due to a mechanical failure. Evasive steering is assumed to not be an option for collision avoidance due to the single lane assumed for the example. However, automatic braking is an option for collision avoidance. It is further assumed that the tunnel configuration is such that the collision avoidance radar has detected the stalled vehicle sufficiently far away that the potential collision can be avoided by braking.

The simplified algorithm for collision avoidance in this example can be explained in terms of the radar measurements and vehicle dynamic motion. The distance from the example vehicle antenna to the stalled vehicle is denoted $r(t)$. The rate of closure between the two vehicles is given by the vehicle speed S when

$$s(t) = \dot{r} = \frac{dr}{dt} \tag{10.20}$$

Assuming brakes are applied at time t_o with a constant deceleration a, the time required to stop the example vehicle δt_s is given by

$$\delta t_s = \frac{S(t_o)}{-a} \tag{10.21}$$

where $a = -\ddot{r}$.

The distance traveled by the example vehicle from t_o to the stopping point is denoted d and is given by

$$d = \frac{1}{2} a \delta t_S^2 \tag{10.22}$$

The minimum stopping distance $d_{\min}$ is determined by the maximum deceleration with maximum braking force $F_{b\max}$. In this simplified illustrative example, the braking force is modeled linearly in terms of the effective braking tire/road friction coefficient μ:

$$F_b = \mu W$$

where W is the vehicle weight.

In this simplified linear model, factors such as the transfer of the normal force on the tires to the front tires are ignored. Nevertheless, the maximum deceleration $a_{\max}$ is given by

$$a_{\max} = \frac{g F_{b\max}}{W} \tag{10.23}$$

where $a_{\max} = g\mu_{\max}$

and where $F_{b\max}$ = maximum braking force and $\mu_{\max}$ = maximum friction coefficient.

The minimum stopping distance $d_{\min}$ corresponds to $a_{\max}$ and is given by

$$d_{\min} = \frac{S^2(t_o)}{2g\mu_{\max}} \tag{10.24}$$

An extreme version of the hypothetical collision avoidance algorithm in this example would be that if the driver has taken no action whenever

$$d_{\min} = r(t)$$

the maximum braking is applied at the time to for which $r(t_o) = d_{\min}$. In this idealized example, the vehicle will come to a stop at $r = 0$.

Of course, this simplified illustrative example of collision avoidance through braking is not realistic. Any practical algorithm would be designed to avoid such an accident through braking that is less than the theoretical maximum possible. However, a practical CAS can have an emergency brake activation system that could apply maximal braking in the event of a sudden emergency situation (e.g., when another vehicle pulls in front of the CAS-equipped vehicle at distances that are at or less than minimum possible stepping distance). In this case, the CAS can often mitigate the ensuing collision impact and reduce injuries to occupants from those sustained at the speed involved with no braking.

The algorithms employed in an actual vehicle CAS are more complicated than the above simplified example. These take into account a variety of potential collision sceneries. Furthermore, for a properly configured CAS, steering inputs are possible to avoid a collision in some circumstances. At the time of this writing, CAS via steering is not widely accepted by car manufacturers. However, nearly all US manufacturers plan to have CAS by automatic braking for essentially all new cars sold in the United States by 2022.

LANE DEPARTURE MONITOR

Another electronic safety system is a system that works on roads that have clearly marked lanes and is called a *lane departure monitoring* (LDM). An LDM has an optical sensor having a field of view forward of the vehicle, the center of which is aligned with the vehicle's longitudinal axis. The sensor is a form of video camera that sends the video signal to a computer that is capable of identifying the painted stripes associated with the lane in which the vehicle is traveling. Alternatively, there are lidar systems (as explained in Chapter 5) that can detect lane markings. The system determines the vehicle position within the lane. Ideally, the lane markings should be symmetrical with respect to the vehicle axis. Unless the driver has activated a turn signal indicating an intentional lane change, the system monitoring algorithms generate a warning to the driver. Such a system can have great safety implications in situations in which the driver has become distracted or is falling asleep.

The lane position detection algorithms are potentially simpler than vehicle in BSD algorithms. Except on curves, the lane markings are straight lines with predictable patterns in the video signal v(x,y) where x,y are the coordinates associated with the region ahead of the vehicle.

TIRE PRESSURE MONITORING SYSTEM

Beginning in 2007, automatic tire pressure warning systems have been required on all new vehicles sold in the United States. Tire pressure has been known to be an issue in land vehicle performance and safety since the introduction of pneumatic tires. For example, tire rolling resistance varies inversely with tire pressure and is a component of the power required to move a vehicle. However, a more serious issue than power required is vehicle safety. For example, a sudden loss of pressure in any tire due to a puncture from an external object can result in a complete loss of steering control and often leads to a very serious accident.

It also is possible for a tire to gradually lose pressure due to a faulty valve or the partial puncture of a tire by an object such as a wood or sheet metal screw. For such a puncture, tire pressure loss can take hours or even days before the pressure loss is visible or can be easily detected while driving. One of the safety aspects of gradual tire pressure reduction (particularly in front tires) is the effect that it can have on steering. For example, differential pressure between a pair of front tires causes a steering moment due, in part, to differential cornering stiffness that can create an undesirable yaw unless manual compensation is provided by the driver. This type of undesirable steering input requires an increase in driver attention to maintain the vehicle in the proper lane of a road or to maneuver around a curve correctly.

The gradual loss of tire pressure can lead to hazardous driving conditions resulting, for example, from insufficient driver steering input. In addition, the loss of pressure due to partial puncture can lead to a sudden loss of pressure at speeds that are sufficient to cause the partial puncture to become a total puncture by dislodging the obstacle causing a partial puncture. It is an important aspect of driving safety that the driver should be aware of the tire pressure reduction/loss. The development of technology for alerting the driver of a vehicle automatically led to the requirement of the automatic tire pressure warning system (TPWS) on new vehicles sold in the United States. A TPWS is configured to warn the vehicle driver of unacceptably low pressure or high pressure. The system is required to operate for 7 years.

The output of the TPWS is a warning message displayed to the driver on the vehicle instrument paned and/or via an audio warning message. The more advanced TPWS systems have a mechanism for identifying the tire (or tires) that have low pressure. A TPWS is an electronic system that in addition to a display, includes a pressure sensor (in each tire) and sometimes a temperature sensor, an A/D converter, an electronic control system, a transmitter, an antenna, and, in some systems, a receiver for receiving control signals from the vehicle and a battery that provides the electric power required for operation. There are numerous pressure sensor configurations that are suitable for using a TPWS as described in the chapter on *sensors and actuators*. However, there is no practical means of sending the electric signal from a rotating wheel to the vehicle body via wires. Rather, a wireless connection via low-power audio link is implemented as explained below.

One of the factors involved in the design of such an electronic system is the environment. The TPWS must be mounted within the tire and is physically very small (e.g., $\sim$19 cm^2). Two of the system components that are affected by the small TPWS size requirements are the battery and the antenna(s). A common choice for TPWS batteries are the miniature Li ion (often called *coin*) batteries. Although the current drain for TPWS is relatively low, power management is important for meeting the lifetime requirement, the mechanisms for which are discussed later in this section of the chapter.

The other important component having a design that is strongly affected by size restrictions is the antenna. The majority of tire pressure monitoring system (TPMS) transmits the pressure data at low power and at 315 MHz in the United States. The efficiency of any antenna is determined by its configuration and size in relationship to the wavelength of the *r-f* carrier. In the case of TPWS, the wavelength is ~1 m. There are a few different implementations of a TPWS antenna, each having a low radiation resistance. For example, relatively common antenna consists of a rectangular loop of wire that extends ~1 cm above and 2 cm across mounted on the circuit board on which the electronic components are mounted. The antenna impedance $Z_{ant} = R_a + j\,X_a$ where typical values are in the ranges given below:

$$0.07 \lesssim R_o \lesssim 0.1\,\Omega$$
$$60 \lesssim X_a \lesssim 65\,\Omega$$

The transmitter driving the antenna has a source impedance $Z_S = R_S + j\,X_S$. The transfer of electric power from any source to any load having impedance Z_ℓ is maximum when

$$Z_\ell = Z_S* \tag{10.25}$$

where the asterisk refers to complex conjugate. For the purpose of minimizing the battery power drain, it is important to maximize the efficiency of the transmission of pressure data from each tire to the vehicle receiver(s). This efficiency requires a matching network between the transmitter and the antenna that cancels the reactance and transforms the antenna resistance to match that of the source as closely as possible. The high ratio of X_a/R_a means that the matching network will be relatively frequency-sensitive, which requires a relatively stable carrier frequency.

In order for the transmitted tire pressure data to be made available to the vehicle instrumentation, there must be a receiver on the vehicle. In many TPMS configurations, there is a separate antenna mounted in each wheel well. This provides the means of identifying the tire associated with the pressure measurement. The individual tire also can be identified by a code transmitted by the module in each tire. Whenever tires are rotated or replaced with new tires, the system must receive the ID data for the tire on each wheel.

An alternative method of tire ID is to have a low-frequency transmitter (LFT) mounted near the wheel in each wheel well. Such LFTs operate on 125 kHz. In such cases, the LFT transmits a signal that interrogates the TPMS (i.e., triggers a transmission of pressure data). The vehicle instrumentation is programmed to read the sensor ID so that installation of new tires will have the correct TPM ID for each wheel.

A block diagram of a representative TPMS is depicted in Fig. 10.6. The TPM depicted in this figure does not include an LF system. The entire electronics are powered by a coin battery that also is not depicted. The modulator puts the data on the transmitter carrier frequency signal in one of several possible modulation methods. One of the methods is phase shift keying, which is explained in the chapter on *vehicle communication*. The receiver demodulates the received signal and sends the data to the instrumentation system/computer that processes the data with regard to the measured pressure by comparing it with the value for which a warning message is to be displayed. Department of Transportation regulations demand that the driver be warned whenever the pressure is 25% below manufacturer recommended values.

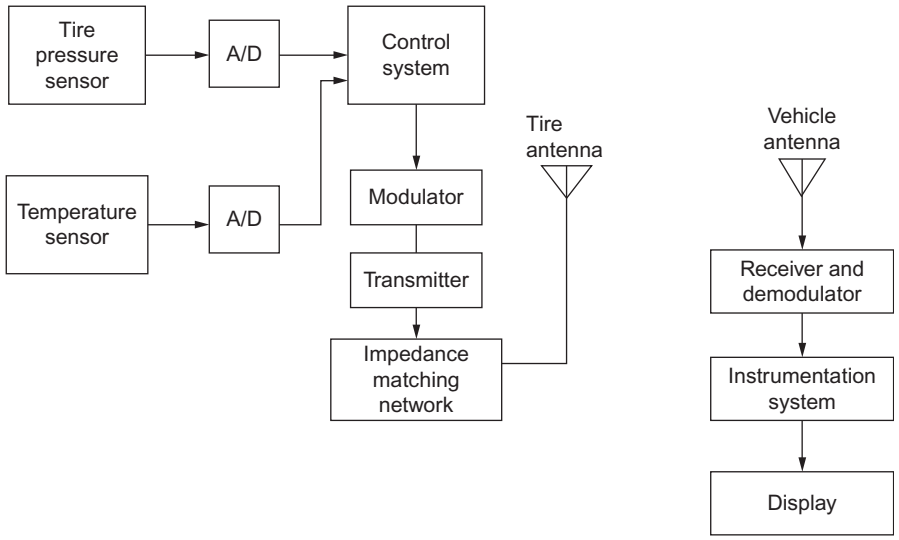


FIG. 10.6 TPMS-1. Tire pressure monitoring system block diagram.

ENHANCED VEHICLE STABILITY

For any vehicle equipped with an ABS system, the capability of individually applying brakes under electronic control also has the capability of improving cornering/maneuvering. Individual brake application produces a moment about the vehicle's vertical axis that directly influences yaw rate. In effect, such brake application provides a steering input in addition to any front wheel (or rear wheel) steering angle. An EVS is an electronic control system that is capable of preventing the loss of steering control from driver input or poor vehicle/road characteristics. For example, in an excessive oversteer condition, brakes can be applied to the wheels on the outside of the turn.

There are several components that are fundamentally required for any given EVS. These include an accumulator for storing pressure (either pneumatic or hydraulic) needed to operate the relevant brake mechanism. In addition, a valve is required for regulating the pressure from the accumulator to the brake mechanism. Such a valve can be binary-valued (e.g., with a solenoid actuator) or it can be proportional as operated by an electric motor. Sensors also are required to measure individual wheel angular speeds and steering input angle. In addition, vehicle motion sensors are required to measure yaw rate (r) and lateral acceleration (a_y). Representative examples of such sensors or automotive systems are explained in the chapter on *sensors and actuators*. Additionally, some EVS include sensors for measuring vehicle roll rate and/or roll angle and vehicle heading and vehicle ground velocity.

An EVS also includes a digital control system that receives as inputs the variables that are measured by the various sensors. The control system operates with algorithms that evaluate cornering to assess the potential for the loss of driver control. Whenever the loss of control has either occurred or is about to occur, the system generates a control signal that applies the appropriate brake to assist vehicle heading control or to regain controlled steering by the driver.

In some cases, when a vehicle is maneuvering at a speed that exceeds a critical speed for a given vehicle traveling under specific conditions (explained later in this section), the vehicle reaches a condition known as oversteer (also explained later in this section), which can result in the loss of directional control and which can lead to an accident. For certain EVS configurations, the EVS control system can reduce speed by reducing throttle angle.

Enhanced stability refers to the stability of vehicle dynamic motion during a steering input maneuver. A quantitative study of a representative EVS requires dynamic analytic models for a vehicle during such maneuvers. For an understanding of vehicle motion during steering input maneuvers, it is helpful to begin with the dynamic model for a vehicle traveling around a curve with a constant radius of curvature (R) at a constant speed (u_o). For an initial explanation of EVS operating theory, it is possible to use the linearized differential equations from the portion of Chapter 7 covering vehicle steering. The translational motion of a vehicle over a level surface while maneuvering over a horizontal surface with Cartesian coordinate x',y' as depicted in Fig. 7.32 is represented by Eq. (7.113) of Chapter 7. For the purposes of the present discussion, this equation is rewritten in the following form:

$$M\dot{v} + \left\{ Mu_o + \frac{2}{u_o}(aC_F - bC_R) \right\} r + \frac{2(C_F + C_R)\mathrm{v}}{u_o} = 2C_F\delta_F \tag{10.26}$$

The variables in the above equation are defined in the section of Chapter 7 associated with Eq. (7.113) in that chapter and depicted in Fig. 7.32. The equation of rotational motion about the vehicle cg is given in Eq. (7.114) of Chapter 7, which is rewritten in the following form:

$$I_{zz}\dot{r} + \frac{2(a^2C_F + b^2C_R)r}{u_o} + \frac{2(aC_F - bC_R)\mathrm{v}}{u_o} = 2aC_F\delta_F \tag{10.27}$$

For the purpose of investigating the lateral directional stability of a vehicle on a curved path, it is instructive to first obtain a single second-order equation in yaw rate r. This equation can be obtained by differentiating the rotational motion equation with respect to time and multiplying by Mu_o:

$$Mu_oI_{zz}\ddot{r} + 2M\left(a^2C_F + b^2C_R\right)\dot{r} + 2(aC_F - bC_R)M\dot{v} = 2Mu_oaC_F\dot{\delta}_F \tag{10.28}$$

The translational motion equation can be solved for $M\dot{v}$ yielding:

$$M\dot{v} = 2C_F\delta_F - \left\{ Mu_o + \frac{2}{u_o}(aC_F - bC_R) \right\} r - \frac{2(C_F + C_R)}{u_o}\mathrm{v} \tag{10.29}$$

The variable v can be eliminated by solving the rotational equation for v and substituting the translational motion of the equation for $M\dot{v}$ into the rotational equation of motion. After simplification, the second-order equation in yaw rate is given by

$$\begin{aligned} &I_{zz}Mu_o\ddot{r} + \left[2M\left(a^2C_F + b^2C_R\right) + 2I_{zz}(C_F + C_R)\right]\dot{r} \\ &\quad + \left[4C_FC_R\ell^2 - 2Mu_o^2(aC_F - bC_R)\right]r/u_o = 2Mu_oaC_F\dot{\delta}_F + 4C_FC_R\ell\delta_F \end{aligned} \tag{10.30}$$

where $\ell = a + b =$ wheelbase.

A simplified analysis of vehicle directional stability can be done for a vehicle traveling at a constant speed u_o along a curved level road. In this case, the radius of curvature R for this path can be expressed in terms of the yaw rate r as given below.

For this simplified study, it is assumed that the steering angle is fixed (i.e., $\dot{\delta}_F = 0$). The radius of curvature R is fixed by the road configuration and is taken as constant for an interval of travel. The driver sets the speed u_o and the steering angle δ_F. The yaw rate r is given by

$$r = u_o/R$$

For steady motion along the curve, $\ddot{r} = 0 = \dot{r}$, and the differential equation is reduced to the following algebraic equation:

$$\left[4C_F C_R \ell^2 - 2Mu_o^2(aC_F - bC_R)\right] r/u_o = 4C_F C_R \ell \delta_F$$

The required steering input for this motion along the curve of radius R is given by

$$\delta_F = \frac{1}{R}\left[\ell - \frac{Mu_o^2(aC_F - bC_R)}{2\ell C_F C_R}\right]$$

For an interpretation of the steering input, the vehicle speed, and the vehicle stability characteristics, it is helpful to define a parameter η_S that is called the steering coefficient (also known as the understeer coefficient), which is defined below:

$$\eta_S = -\frac{M(aC_F - bC_R)g}{2\ell C_F C_R}$$

Using this parameter, the above equation relating δ_F to u_o and R is given by

$$\delta_F = \frac{\ell}{R}\left(1 + \frac{\eta_S u_o^2}{g\ell}\right)$$

The parameter η_S characterizes the vehicle behavior while being steered along a curve. A vehicle is said to have neutral steer whenever $\eta_S = 0$. For this case, the required steering input δ_F is constant versus vehicle speed u_o. The parameter η_S depends, in part, on the vehicle configuration and the tire/road interface. Vehicles for which $\eta_S < 0$ are said to be oversteered. For an oversteered vehicle, the steering input changes sign when the speed exceeds a so-called critical speed u_{crit}. This critical speed is given by

$$u_{crit} = \sqrt{\frac{g\ell}{-\eta_S}} \quad \eta_S < 0 \tag{10.31}$$

Examination of Eq. (10.30) reveals that an oversteered vehicle becomes unstable whenever $u_o > u_{crit}$. Instability in vehicle yaw can be determined by the roots of the characteristic equation associated with Eq. (10.30), which can be written as

$$\left(C_1 s^2 + C_2 s + C_3\right) r(s) = (Mu_o aC_F s + 4u_F C_R)\delta_F(s) \tag{10.32}$$

where

$$\begin{aligned} C_1 &= Mu_o I_{zz} \\ C_2 &= \left[2M\left(a^2 C_F + b^2 C_R\right) + 2I_{zz}(C_F + C_R)\right] \\ C_3 &= \left[4C_F C_R \ell^2 - 2Mu_o^2(aC_F - bC_R)\right]/u_o \\ &= 4C_F C_R \ell^2 \left[1 + \eta_S \frac{u_o^2}{g\ell}\right] \end{aligned}$$

The roots to this characteristic equation can be real or complex depending upon the coefficients. The yaw-rate response to steering input is characterized by the transfer function $H_r(s)$:

$$H_r(s) = \frac{r(s)}{\delta_F(s)} \tag{10.33}$$

A stable response of yaw rate to steering input is assured provided the real parts of the roots to the characteristic equation are negative, that is, for complex roots s_1 and s_2 where

$$s_{1,2} = \frac{-C_2 \pm \sqrt{C_2^2 - 4C_1C_3}}{2C_1}$$

However, if $C_3 < 0$, both roots are real, and one is positive. Root $s_1 > 0$ if $C_3 < 0$ and is given by

$$s_1 = \frac{-C_2 + \sqrt{C_2^2 + 4C_1|C_3|}}{2C_1} > 0$$

It can be shown that $C_3 < 0$ if $u_o > u_{crit}$, which means that an oversteered car can have an unstable exponentially growing yaw rate whenever vehicle speed exceeds the critical speed:

$$r(t) = r(0)e^{s_1 t}$$

where $r(0) =$ yaw rate at which u_o becomes $> u_{crit}$.

An alternate interpretation of the linear steering model for a vehicle traveling at a constant speed (i.e., without braking or accelerating) along a level circular arc of radius R can be developed by including the vertical force components on the vehicle. Fig. 10.7 depicts the vertical forces including vehicle weight $W = Mg$.

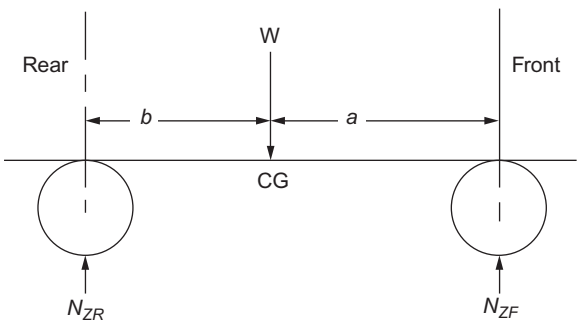


FIG. 10.7 Vertical forces on the vehicle.

The vertical forces front (N_{zF}) and rear (N_{zR}) are given by

$$\begin{aligned} N_{zF} &= \frac{bW}{\ell} \\ &= \frac{bMg}{\ell} \\ N_{zR} &= \frac{aW}{\ell} \\ &= \frac{aMg}{\ell} \end{aligned}$$

The cornering stiffness coefficients are proportional to these vertical forces that, by assumption, are laterally symmetrical.

The understeer parameter η_S can be expressed in terms of the vertical forces as derived below:

$$\begin{aligned} \eta_S &= -\frac{Mg}{\ell}\left(\frac{aC_F - bC_R}{C_F C_R}\right) \\ &= \left[\frac{N_{zF}}{2C_F} - \frac{N_{zR}}{2C_R}\right] \end{aligned} \tag{10.34}$$

Similarly, the steering model can be expressed in terms of the lateral forces that are applied to the tires due to the lateral acceleration a_y. For the linear approximate model in which the slip angles are small, the lateral forces F_{yF} and F_{yR} are given approximately by

$$\begin{aligned} F_{yF} &= \frac{b}{\ell} Ma_y \\ &= \frac{N_{zF}}{g} a_y \\ F_{yR} &= \frac{a}{\ell} Ma_y \\ &\frac{N_{zR}}{g} a_y \end{aligned}$$

The tire slip angles (for lateral symmetry) are given by

$$\begin{aligned} \alpha_F &= \frac{F_{yF}}{2C_F} \\ \alpha_R &= \frac{F_{yR}}{2C_R} \end{aligned} \tag{10.35}$$

The difference in tire slip angles (front to rear) is given by

$$\begin{aligned} \alpha_F - \alpha_R &= \frac{F_{yF}}{2C_F} - \frac{F_{yR}}{2C_R} \\ &= \left(\frac{N_{zF}}{2C_F} - \frac{N_{zR}}{2C_R}\right)\frac{a_y}{g} \\ &= \eta_s \frac{a_y}{g} \end{aligned} \tag{10.36}$$

The above equation can be interpreted to mean that for an understeered vehicle for which $\eta_S > 0$, the front tire slip angle is greater than the rear (i.e., $\alpha_f > \alpha_R$). For neutrally steered vehicle, the front and rear tire slip angles are identical. For an oversteered vehicle for which $\eta_S < 0$, the rear tire slip angle exceeds the front slip angle (i.e., $\alpha_2 > \alpha_1$).

The individual steering stability assessment and brake control algorithms are manufacturer-specific and often vehicle model-specific. The details of commercially available EVS algorithms and operation are (or may be) proprietary and sufficiently different from one another that only exemplary EVS controls are discussed in the present chapter. For the purpose of explaining such an exemplary EVS in a simplified discussion, it is assumed that the relevant vehicle model is linear and is of the form of Eqs. (7.117) through (7.121). However, in the case of EVS, the D matrix is given by

$$D = \begin{bmatrix} 2C_F & 0 \\ 2_aC_F & D_{22} \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{10.37}$$

where

$$D_{22} = \pm W_\ell/2$$

W_ℓ = *lateral distance between wheel planes of symmetry.*

The sign of D_{22} *is determined from the steering angle and has the opposite sign:*

$$\text{sgn}(D_{22}) = -\text{sgn}(\delta_F)$$

The input vector u is given by

$$u = [\delta_F, F_b]^T$$

where F_b = braking force applied to the relevant wheel by the road surface.

For the present exemplary EVS, a single output y is taken to be the yaw rate r. That is, in conventional state variable formation, the output is attained as given below:

$$r = Cx$$

where $C = [0, 1, 0, 0]$.

The linearity of the model permits a separation of the state vector and output into two components, one due to the driver input δ_F and the other due to EVS braking:

$$x = x_\delta + x_b \tag{10.38}$$

where x_δ is due to δ_F input and x_b is due to F_b input. A pair of state variable equations can be written for the linear model as follows:

$$\dot{x}_\delta = Ax_\delta + B_\delta\delta_F \tag{10.39}$$

$$\dot{x}_b = Ax_b + B_bF_b \tag{10.40}$$

where $B_\delta = G^{-1}D_\delta$
$B_b = G^{-1}D_b$

and where $D_\delta = \left[C_f, aC_f, 0, 0\right]^T$
$D_b = [0, D_{22}, 0, 0]^T$.

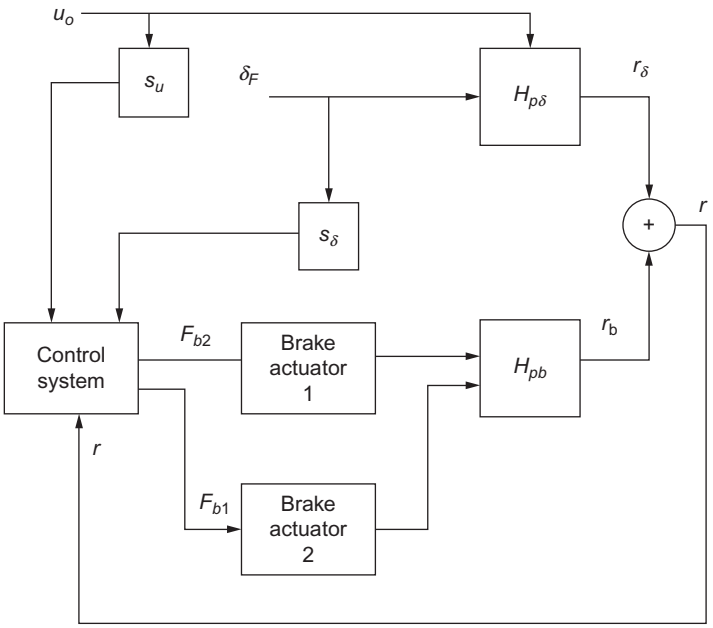


FIG. 10.8 Exemplary EVS block diagram.

An illustrative block diagram for the exemplary EVS is depicted in Fig. 10.8.

In this block diagram, the vehicle dynamic plant model is characterized by operational transfer functions $H_{P\delta}(s)$ and $H_{Pb}(s)$ (due to F_{bn}). From the theory presented in Appendix A, these linear operational transfer functions can be shown to be given by

$$\begin{aligned} H_{P\delta}(s) &= r_\delta(s)/\delta_F(s) \\ &= C(sI_4 - A)^{-1} B_\delta \end{aligned} \tag{10.41}$$

and

$$\begin{aligned} H_{Pb}(s) &= r_b(s)/F_{bn}(s) \quad n = 1 \text{ or } 2 \\ &= C(sI_4 - A)^{-1} B_b \end{aligned}$$

where I_4 is a four-dimensional identity matrix.

The combined yaw rate $\dot{r}$ has two components:

$$r = r_\delta + r_b$$

where r_δ = component of r due to δ_F and r_b = component of r due to brake force F_b.

An illustrative representation steering control algorithm is based upon the relationship developed above for yaw rate r, vehicle speed u_o, and steering angle:

$$\frac{r}{u_o} = \frac{\delta_F/\ell}{1 + \left(\eta_s u_o^2/g\ell\right)} \tag{10.42}$$

In principle, η_s is known from the vehicle configuration and the cornering stiffness values $C_F\, C_R$. However, these two latter coefficients depend, in part, on the tire/road interface friction, which can change significantly with weather conditions, for example, wet/icy roads. However, with appropriate sensors for measuring δ_F, u_o, and r, the value for η_s can be computed. Solving the above equation for η_s yields,

$$\eta_s(\delta_F) = \left(\frac{\delta_F u_o}{\ell r_\delta} - 1\right)\frac{g\ell}{u_o^2} \tag{10.43}$$

Whenever $\dfrac{r_\delta}{u_o} > \dfrac{\delta_F}{\ell}$, the understeer coefficient $\eta_s < 0$ corresponding to an oversteered vehicle. It is possible to estimate u_{crit} from η_s. Whenever $u_o > u_{crit}$, the oversteered vehicle becomes unstable and can skid out of control.

If a brake is applied such that r_b has the opposite sign of r_δ, the net yaw rate is reduced, and the vehicle stability is returned. Depending upon the type of actuator used, the brakes can be applied proportionally and continuously for a proportional valve or the brakes can be pulsed with a duty cycle, which yields the desired r_b for a solenoid operated valve. That is, by applying the appropriate brake (in this case brake(s) on the outside of the turn), then

$$\text{sgn}(r_b) = -\,\text{sgn}(r_\delta)$$

and the combined r can be reduced to the point of stable cornering. The control algorithm can be designed to apply the stabilizing brake for a condition that is reached before steering becomes unstable. This can occur if the vehicle speed u_o is increasing and approaching u_{crit}. It can also occur if road conditions are changing with u_{crit} decreasing due to reduced road/tire friction. For example, when driving along a curve and entering a rain shower, the water on the road increases, which can cause C_F and C_R to be reduced by a factor of $\epsilon (\epsilon < 1)$. In this case, u_{crit} is decreased by a factor of ϵ compared with its dry value. The EVS can estimate the trend of u_o toward u_{crit} and apply stabilizing brakes.

It must be emphasized that the above example of EVS is not intended to explain any existing, practical commercial system. Rather, it is intended to illustrate concepts involved. In any practical vehicle, the cornering dynamics are represented by nonlinear models. Moreover, an EVS is not designed or implemented to improve cornering performance such as might be desirable for certain racing vehicles. An EVS is intended to improve safety by preventing the loss of steering control.

CHAPTER

DIAGNOSTICS 11

CHAPTER OUTLINE

Electronic Control System Diagnostics 534
Service Bay Diagnostic Tool 536
Onboard Diagnostics 536
Model-Based Sensor Failure Detection 538
General Model-Based Diagnostics 539
Diagnostic Fault Codes 543
Onboard Diagnosis (OBD II) 555
Misfire Detection 555
Model-Based Misfire Detection System 555
Expert Systems in Automotive Diagnosis 567

From the earliest days of the commercial sale of the automobile, it has been obvious that maintenance is required to keep automobiles operating properly. Of course, automobile dealerships have provided this service for years, as have independent repair shops and service stations. Until the early 1970s, however, a great deal of the routine maintenance and repair was done by car owners themselves, using inexpensive tools and equipment. However, the Clean Air Act affected not only the emissions produced by automobiles but also the complexity of the engine control systems and, as a result, the complexity of automobile maintenance and repair. Car owners can no longer, as a matter of course, do their own maintenance and repairs on certain automotive subsystems (particularly, the engine). In fact, the traditional shop manual used for years by technicians for repairing cars is rapidly becoming obsolete and is being replaced by electronic technician aids.

As will be shown later in this chapter, the trend in automotive maintenance is for the automobile manufacturer to distribute all required documentation, including parts lists (with figures) and repair procedures in electronic format via a dedicated communication link (e.g., via satellite) or via CD supplied to the service technician. The repair information is then available to the technician at the repair site by use of a PC-like workstation.

Onboard digital systems can also store diagnostic information wherever a failure or partial failure occurs in a component or subsystem. The relevant information can then be stored in a memory (e.g., RAM) that retains the information even if the car ignition is switched off. Then, when the car is delivered to a repair station (e.g., at the dealer), the technician can retrieve the diagnostic information electronically.

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-12-810434-7.00011-9

Copyright © 2017 Elsevier Inc. All rights reserved.

The change from traditional fluidic/pneumatic engine controls to microprocessor-based electronic engine controls was a direct result of the need to control automobile emissions and has been chronicled throughout this book. However, little has been said thus far about the diagnostic problems involved in electronically controlled engines. This type of diagnostics requires a fundamentally different approach than that for traditionally controlled engines because it requires more sophisticated equipment than was required for diagnostics in preemission control automobiles. In fact, the best diagnostic methods use special-purpose computers that are themselves microprocessor-based.

ELECTRONIC CONTROL SYSTEM DIAGNOSTICS

Each microprocessor-based electronic subsystem has the capability of performing some limited self-diagnosis. A subsystem can, for example, detect a loss of signal from a sensor or detect an open circuit in an actuator circuit and other failures. As long as the subsystem computer is still functioning, it can store fault codes for detected failures. Such diagnosis within a given subsystem is known as *onboard diagnosis*.

Some limited self-diagnostics have been available in power train control from the earliest days of microprocessor-based control systems. However, the Environmental Protection Agency (EPA) has developed regulations mandating a relatively high level of diagnosis for components and subsystems that can adversely effect exhaust emissions when failed or in degraded performance. These regulations are known as "onboard diagnostics II" (OBD II). They require that the vehicle has within its electronic control/instrumentation systems the capability of essentially continuously monitoring the performance of the vehicle emission control systems. The details of this regulation and specific implementation schemes are discussed later in this chapter.

Whenever a fault in a component or system is detected, a code, specific to the failure/degraded performance known as a "fault code," is stored in memory. Various techniques for detecting such failures are discussed later in this chapter. If the fault has the potential to degrade the emission control system beyond allowable limits, OBD II requires that the driver be alerted via a "check engine" message on the instrument panel.

However, a higher level of diagnosis than the onboard diagnosis is typically done with an external computer-based system that is available in a service shop. Data stored in memory in an onboard subsystem are useful for completing diagnosis of any problem with the associated subsystem. Such diagnosis is known as *off-board diagnosis* and is usually conducted with a special-purpose computer.

In order for fault code data to be available to the off-board diagnosis computer, a communication link is required between the off-board equipment and the particular subsystem on board the vehicle. Such a communication system is typically in the form of a serial digital data link. However, the details of such a communication link are discussed in Chapter 9 in which communication via an in-vehicle network (IVN) is explained. A serial data link transmits digital data in a binary time sequence along a communication path or network (e.g., pair of wires). Before discussing the details of onboard and off-board diagnosis, it is perhaps worthwhile to review briefly automotive digital communications.

It was shown in the Chapter 9 that the various electronic subsystems (ECUs) in a contemporary vehicle are connected together via an IVN (e.g., the CAN network). For example, in Fig. 9.3, one of the connections to the exemplary CAN bus is a data link (denoted DLC) that is a portal from the vehicle to the off-board diagnosis system. A connection is made to this diagnostic system when the vehicle is in an authorized repair facility (e.g., car dealer) for maintenance/repair.

Fig. 11.1 depicts a representative connection of an off-board connection of a so-called diagnostic scan tool to an automotive DLC. The diagnostic scan tool depicted in Fig. 11.1 is portable and can be carried in the vehicle when it is being test-driven by a maintenance technician as discussed later in this chapter.

The scanner has access to address and data buses of the subsystem containing the memory in which the relevant fault codes are stored. The scanner then sends addresses to the memory locations where the fault codes are stored and retrieves any fault code in each memory location associated with fault code storage. The scanner also includes a display device where it displays the fault code. Some diagnostic systems include storing the clock time of the occurrence of the fault. Such a system is useful for

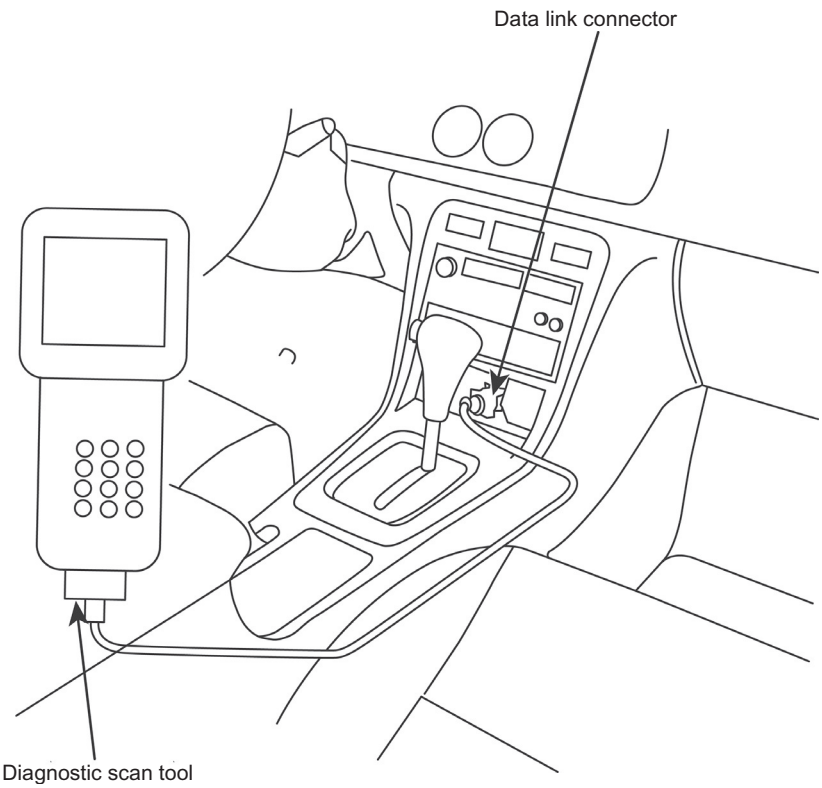


FIG. 11.1 Illustration of diagnostic scan tool connection to vehicle.

diagnosing intermittent faults (i.e., those that come and go randomly and are challenging for the technician to find). In addition to the portable scan diagnostic tool (PSDT), there is a service bay diagnostic tool (SBDT) that is often on a movable cart but is not small enough to be carried on board for test drivers.

SERVICE BAY DIAGNOSTIC TOOL

An alternative to the onboard diagnostics is available in the form of a service bay diagnostic system. This system uses a computer that has a greater diagnostic capability than the vehicle-based system because its computer is typically much larger and has only a single task to perform—that of diagnosing problems in vehicle electronic systems.

Service bay diagnostic systems are computer-based instruments that are capable of reading fault codes that are stored by the onboard diagnostic systems (e.g., via the DLC described in Chapter 9). In addition, they have electronic versions of the equivalent of shop manuals and recommended procedures for diagnosing specific problems from the stored fault codes and information and problem descriptions from the driver.

In certain circumstances, fault codes, by themselves, are insufficient to fully diagnose a given problem. In the cases, the off-board diagnostic system can present a sequence of steps that require action by the service technician which, when followed, can complete the diagnosis of a problem. Of course, it should be emphasized that fault codes are only applicable to those automotive systems/subsystems that have electrical or electronic components. Other subsystem/components require the knowledge and experience of the service technician to perform diagnosis/repair. For example, a failed or partially failed wheel bearing is not a failure that will have a stored fault code. The diagnosis and repair of problems in automobiles will always require competent, knowledgeable, trained technicians.

On the other hand, the electronic content in contemporary vehicles continues to increase with each new vehicle configuration/model. Thus, it is clear that electronic diagnostic methods will continue to proliferate.

In addition to storing and displaying shop manual data and procedures, a computer-based service bay diagnostic system has the theoretical capability to automate the diagnostic process itself. In achieving this objective, the technicians' terminal has the capability to incorporate what is commonly called an *expert system* that is explained in detail later in this chapter.

ONBOARD DIAGNOSTICS

Onboard diagnostics are dictated largely by the need for each automobile to meet the requirements of OBD II regulations. As stated above, any component/subsystem having the potential to adversely affect exhaust emissions must be evaluated for its performance. In addition, however, on a power train system level, the onboard diagnostics must be capable of detecting engine misfire. A misfire is any failure of any cylinder (during an engine cycle) to experience normal combustion. It can include a complete misfire in which ignition fails to cause combustion to occur. Partial combustion in which only a portion of the fuel/air mixture is combusted also can constitute a misfire by OBD II standards. A misfire can

degrade the performance of the catalytic converter since the exhaust gas constituents and concentrations are outside the limits in which it is intended to function.

Any engine can experience an occasional, spurious misfire (or partial misfire). However, when the severity and frequency of occurrence exceeds certain tolerance limits, the catalytic converter performance is degraded, and exhaust emissions can exceed the EPA-mandated limits. For such an occurrence, the warning message must be displayed, and the owner should seek repairs for the vehicle. The format for this warning message varies with vehicle model, but it is often an illuminated "check engine" display. For convenience in the present chapter, this warning message is termed "fault indication lamp" or FIL since it is actually illuminated due to a component/system fault. An exemplary method of detecting misfires is described in detail later in this chapter.

On a component level, there are many individual components that can adversely affect exhaust emissions during periods of degraded performance. For example, the heated exhaust gas oxygen concentration sensor (HEGO; see Chapter 5) that is used for closed-loop fuel control can experience a failure or partial failure. For a warmed engine running with fuel control in closed-loop mode, the voltage waveform of the HEGO sensor will have certain patterns when the sensor is operating normally. This voltage should be cycling between its normal high voltage level (about 1 V) and its low voltage level (about 0.1 V). Moreover, the mean value of the sensor voltage will lie within a relatively narrow band that is approximately midway between the high and low voltage levels.

Any deviation in the HEGO sensor voltage waveform is an indication of a potential HEGO sensor failure or degraded performance. However, there are other potential causes of waveform parameters (HEGO sensor) outside expected limits. For example, the fuel control system could have experienced a failure and could be unintentionally fueling the engine too rich or too lean. In addition, one or more fuel injectors could have failed resulting in excessively rich or lean mixture.

The OBD II requirement at this point is to illuminate the FIL warning and set the appropriate fault codes. These might include separate codes corresponding to the conditions (1) HEGO sensor voltage a steady high or (2) a steady low, (3) HEGO sensor voltage failure to cycle, (4) mean HEGO sensor voltage above limits, or (5) below limits.

When the vehicle is brought to the service facility, the service technician will normally connect the appropriate off-board system to the DLC or its functional equivalent (see Fig. 9.3) and transfer all fault codes from the onboard memory to the scan tool. With all fault codes present, the service technician can follow a set of procedures to diagnose the failures.

Another component that can fail affecting exhaust emissions is a fuel injector. Not all fuel injector failures are detectable with onboard diagnostics. However, the power train control system can monitor fuel injector current and terminal voltage. Measurement of these quantities can detect an open or short circuit in the fuel injector solenoid coil. Of course, should either condition be detected, OBD II requires the driver alert message and storage of the appropriate code and identification of the affected cylinder.

Still, another important component requiring monitoring is the catalytic converter. As stated earlier in this book, there is no cost-effective way of measuring the regulated exhaust gas concentrations on board the vehicle. On the other hand, it is possible to obtain some assessment of the catalytic converter conversion efficiency by placing a second HEGO sensor in its output side. The primary HEGO sensor for fuel control is located upstream of the catalytic converter. Recall from Chapter 6 that in closed-loop mode, the fuel control continuously cycles from rich to lean of stoichiometry and from lean to rich. During periods of relatively rich mixture, the exhaust gas oxygen concentration is low. This exhaust

gas enters the catalytic converter with the low O_2 concentration where the converter acts as an oxidizer. The reverse is true for a relatively lean mixture. A comparison of the primary and secondary HEGO sensor voltages can serve as an indication of relative converter efficiency.

MODEL-BASED SENSOR FAILURE DETECTION

The performance of certain sensors can be evaluated via model-based calculation from measurements of other sensors. For example, the MAF is an important sensor for setting fuel injector base pulse duration (see Chapter 6). A calibration change in the MAF sensor can lead to misfueling (relative to stoichiometry) particularly in open-loop mode of fuel control. It is important to maintain proper fuel/air mixture regardless of the controller mode of operation.

An independent check on MAF calibration is possible (theoretically) for any engine that also has an MAP sensor and an intake air temperature sensor. Assuming that these sensors are functioning correctly, the mass flow rate $\dot{M}_a$ into the intake system is given by

$$\dot{M}_a = \dot{V}_a \delta_i \tag{11.1}$$

where $\dot{V}_a$ = volume flow rate and δ_i = intake air density.

Using tables of volumetric efficiency (η_V) that is explained in Chapter 4 for the engine as a function of throttle angle and RPM, the volume flow rate $\dot{V}_a$ is given by

$$\dot{V}_a = \frac{D_e n \eta_V(\theta_t, R)}{2} \tag{11.2}$$

where D_e = engine displacement, $n = R/60$, R = RPM, and θ_t = throttle angle.

The intake air density is given by Eq. (11.3)

$$\delta_i = \frac{\delta_0 p_i T_0}{p_0 T_i} \tag{11.3}$$

where

δ_0 = sea level standard day air density
p_0 = sea level standard day air pressure
T_0 = sea level standard day air temperature
p_i = intake manifold air pressure
T_i = intake manifold air temperature.

In principle, the MAF sensor calibration can be evaluated by comparing the measured value of $\dot{M}_a$ from it with the calculated value from temperature and pressure measurements. Unfortunately, the cost of adding extra sensors makes it unattractive to an automobile manufacturer to implement such a method unless these sensors were already in place for other control applications. Nevertheless, this hypothetical example illustrates the potential for cross-checking the performance of various sensors.

It has been shown (e.g., see Chapter 6) that power train control uses numerous angular speed sensors. As shown in Chapter 5, many of these use a magnetic or optical sensor in conjunction with a disk having multiple lugs. Many failure modes are possible with such sensors. For example, a magnetic speed sensor incorporating a permanent magnet and a coil can experience shorting of a portion of the coil turns. This type of failure leads to a lower than normal output voltage at any given speed.

Although speed measurement from such a sensor is based upon the frequency of counted pulses, the failure can occur whenever the voltage amplitude falls below the level at which the signal processing can detect the presence of some (or all) of the pulses produced within one revolution. In addition, it is possible for one of the lugs to come off the disk (e.g., due to manufacturing defect). In either type of failure, the signal processing will compute an incorrect angular speed for the sensor.

One method for detecting missing pulses uses the controller clock to record the time of occurrence $t_k(k=1,2,\ldots,K)$ of each of the pulses in the incoming sequence during any given engine cycle where K is the number of lugs on the disk. The controller can also obtain the differential time between successive pulses

$$\delta t_k = t_k - t_{k-1} \quad k = 1,2,\ldots,K$$

If a sensor failure of the types described above has occurred, then there will be at least one differential time that is significantly different from the others in a full sequence during an engine cycle. The measurements of angular speed for this particular time differential are termed an outlier. Several standard algorithms exist for detecting and removing these erroneous measurements from the data collected during each cycle of operations and setting a fault code.

GENERAL MODEL-BASED DIAGNOSTICS

There is a computer-based diagnostic method of detecting and identifying failures in components of vehicular electronic systems. The detection of a failure or an incipient failure of a component requires a dynamic analytic model of the system or subsystem that incorporates the component. The failure or incipient failure is accomplished with a unique digital system that is called a "failure detection and identification" (FDI) system.

In order to implement an FDI, the system containing the component must be capable of being modeled by a linear multidimensional state variable equation of the following form:

$$x = A\dot{x} + Bu$$

where

$x = [x_1, x_2, \cdots, x_N]^T$ = state vector
A = state transition matrix
B = input matrix
u = input to the system.

The dimensionality of this equation is given by

$$x \in R^N$$
$$A \in R^{N\times N}$$
$$B \in R^{N\times m}$$
$$u \in R^m$$

The system model with a failure in the input due, for example, to an actuator failure is given by

$$\dot{x} = Ax + Bu + f_i \tag{11.4}$$

where f_i = vector model for the failure in the input term.

For example, an actuator is an electromechanical device in which the input variable u is determined by an analog voltage v_s that is the output of the electronic control for which the input model is given by

$$u = K_a v_s \tag{11.5}$$

where K_a = actuator calibration constant. A calibration change due to a failure or degradation in actuator performance is given by a change δK_a in K_a from its nominal design value

$$K_a = K_{ad} + \delta K_a \tag{11.6}$$

where K_{ad} = design calibration.

In this case, the failure event vector f_i is given by

$$f_i = B\delta K_a v_s \tag{11.7}$$

Although the above model is linear and many vehicular systems are modeled by nonlinear equations, the FDI method can be made applicable to nonlinear system by suitable linearized models within domains of a system operating point.

The FDI is formulated as a special state estimator (a Luenberger state estimator) for calculating the estimate $\hat{x}$ of x. The model for the FDI is given by

$$\dot{\hat{x}} = A\hat{x} + Bu + D(y - \hat{y}) \tag{11.8}$$

where $y = Cx \in R^k$

$$\hat{y} = C\hat{x} \in R^k$$

D = special feedback matrix and C = measurement matrix.

The vectors y and $\hat{y}$ represent measurements of the state variables and the estimated values, respectively. For measurement of a subset of k variables, the measurement matrix has dimensionality given by

$$C \in R^{N \times k} \tag{11.9}$$

The FDI output is an error vector e that is defined

$$\begin{aligned} e &= y - \hat{y} \in R^k \\ &= C(x - \hat{x}) \end{aligned} \tag{11.10}$$

The unique capability to identify the failed or failing component is based on the direction of the FDI error output vector e in the k dimensional output vector space. This direction is determined with the design of the feedback matrix D which, in turn, is derived from the error vector f_i. The algorithm for computing D is present in Appendix D. In addition to controlling the output vector direction based on e, the design of D involves selecting the parameters that establish the dynamic response of the FDI as characterized by the state error vector dynamics as given below:

$$\begin{aligned} \dot{\epsilon} &= \dot{x} - \dot{\hat{x}} \\ &= (A - DC) \in + f_i \\ &= G \in + f_i \end{aligned} \tag{11.11}$$

where $G = A - DC$

Eq. (11.11) is a first-order N-dimensional differential equation in which f_i is the forcing function. The FDI estimator must be stable which requires the following:

$$Re(s_j) < 0 \quad j = 1,2,\cdots,N$$

where s_j is the roots of the characteristic equation

$$\det(sI(N) - G) = 0$$

where I(N) is an N-dimensional identity matrix.

Whenever the actuator fails or partially fails and no other system change takes place, the FDI output error vector is $e(f_i)$. The identification of this failure is accomplished by computing the angle ϕ of the output vector relative to $e(f_i)$. This angle ϕ is given by

$$\phi = \cos^{-1}[\langle e(f_i), e\rangle / (\|e(f_i)\| \cdot \|e\|)]$$

where

$\langle e(f_i), e\rangle$ = inner product of $e(f_i)$ and e
$\|e\| = L_2$ norm of the vector.

In the case of a failure in the actuator for which the FDI has been designed, the angle ϕ in an ideal noise free case is identically 0. This angle is independent of the magnitude of e. However, the magnitude of $e(f_i)$ (i.e., $\|e\|$) is proportional to the magnitude of the failure (i.e., $\|(f_i)\|$). The procedure for detecting and identifying the failure of the component for which the FDI is designed is to calculate the angle ϕ of e relative to $e(f_i)$. Then, for any time the system's operating, ϕ is either 0 (in an ideal case) or within a very small interval about zero in the presence of measurement or process noise (in y), the procedure is to evaluate the magnitude of failure relative to the level of unacceptable (or unsafe) performance of the component.

The performance of an FDI for detecting failure in an actuator can be illustrated with the example of the steering actuator in a vehicle equipped with automatic steering as discussed in Chapter 7 and presented in detail in Chapter 12. In the present example, it is assumed that steering is by front wheels only, although the dynamic models for the plant and the FDI are based on the four-wheel steering model given in Chapter 7. The model parameters are those presented in association with the model and are taken from an actual vehicle. The state variable model is presented in Eqs. (7.117)–(7.120) in Chapter 7, with the state vector given by Eq. (7.116).

It is assumed that the actuator is analog and regulated by the output of an automatic steering control system by voltage v_s. The actuator model of Eq. (11.7) yields the failure event vector f_1 and is parallel to B, which is an N-dimensional column vector for a scalar (i.e., $m = 1$) input u:

$$f_i = B\delta K_a v_s \in R^N$$

The D matrix, for the example, FDI, is computed in Appendix D.

A simulation was conducted using the vehicle parameters given in Chapter 7 in association with the steering model. The error residuals in the example simulation were put at 5%, 10%, and 15% calibration error (i.e., $\Delta K/K_a$). Fig. 11.2 is a plot of the individual components of $e(f_i)$ for a 10% reduction in actuator calibration for a constant steering input such as occurs for a vehicle traveling at a constant speed along a curve with a constant radius of curvature.

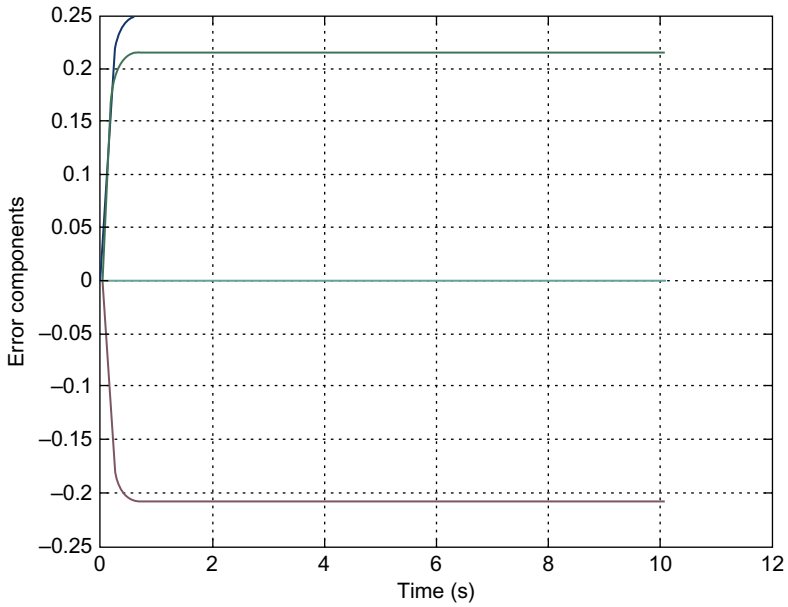


FIG. 11.2 Error residuals due to steering actuator calibration failure 10%.

The difference in angle for these three levels of degradation is $<10^{-6}$ radian. The magnitude of the failures for these three simulated failures is proportional to the calibration change fraction. A measure of the degradation in actuator performance under steady-state conditions (i.e., $u = \text{const}$) that is denoted d is given by

$$d = \frac{\|Ge(f_i)\|}{\|B\| \cdot \|u\|}$$

For this steady state, $d = \delta K_a / K_{ad}$ and directly yields a measure of the fractional change in calibration. This measure of degradation in performance is useful diagnostic information.

The FDI methodology explained in this chapter and illustrated in Appendix D is applicable to the detection and identification of an actuator in any system. The only requirement is that the system incorporating the component must be capable of being modeled in a state variable form with a multiple component state vector.

One of the main issues involved in the FDI is the generation of an output error residual e due to other changes in the system. For the automatic steering system, an example of such a change is variation in the cornering stiffness of the tires (e.g., due to wet or icy road). A simulation was run with the FDI that is designed for steering actuator failure but with a 10% reduction in cornering stiffness for all wheels. The error vector for this change is denoted $e(I)$. The angle ϕ between $e(I)$ and the error for f_i $e(f_i)$ is denoted ϕ_I and is given by

$$\phi_I = \cos^{-1}[\langle e(I), e(f_i)\rangle / (\|e(f_i)\| \cdot \|e(I)\|)]$$

This directional difference between the error vector for which the FDI was designed and that due to a related environmental change (e.g., due to ice) of 34 degrees is significant for the correct identification of actuator failure relative to the environmental change.

Various temperature (e.g., coolant temperature) sensors are used in power train control whose accuracy affects emission control. The very wide range of temperatures over which a vehicle operates essentially precludes a practical and reliable means of checking the calibration accuracy. On the other hand, open- and short-circuit conditions can be detected by monitoring sensor terminal voltages. However, depending upon the design, it may be possible to establish limits on the possible range of indicated temperatures. Any reading outside this range can be considered a detected fault.

In addition to those sensors that must be monitored for OBD II requirements, there are sensors that are important for safe reliable vehicle operation. For example, an oil pressure measurement is important to assure that proper lubrication is present for all engine rotating parts. Such sensors can be monitored with respect to open-/short-circuit conditions.

In addition, the proper operation of electric system is important. The charging of the battery via the alternator can be monitored by both the terminal voltage and the current flow. In particular, an alternator voltage level that is too low or too high for engine operating conditions is an indication of an alternator problem.

As will be demonstrated below, the diagnostic capability provided in any modern microprocessor-based electronic control system (although somewhat limited) can provide valuable assistance to the service technician. These diagnostic functions are performed by the microprocessor under the control of stored programs and are normally performed when the microprocessor is not fully committed to performing normal control calculations. While it is beyond the scope of this book to review the actual software involved in such diagnostic operations, the diagnostic procedures that are followed and explanations of onboard diagnostic functions can be reviewed, by example.

During the normal operation of the car, there are periods during which the performance of various electrical and electronic components is monitored via the vehicle instrumentation system (see Chapter 8). Whenever a fault is detected, the data are stored in memory using a specific fault code. At the same time, the controller generates or activates an FIL warning lamp (or similar display) on the instrument panel indicating that service is required provided the fault affects the emission control system or affects safety.

The onboard diagnostic functions have one potential major limitation—they do not necessarily detect intermittent failures reliably. For the traditional onboard diagnostic system to detect and isolate a failure, the failure had to be nonreversible and persistent. In an onboard diagnostic system, if the electronic control module stores trouble codes that are automatically cleared by the microprocessor after a set number of engine cycles have occurred without a fault reappearing, then intermittent failure detection is precluded. However, it is possible in certain vehicles for the system to be put into a fault-recording mode. Many times such a fault-recording mode can identify intermittent failures. Detection of intermittent failures is possible using FDI-based system provided the dynamic response is sufficient. The FDI dynamic response can be made sufficiently fast by proper choice of the eigenvalues (see Appendix D).

DIAGNOSTIC FAULT CODES

The Society of Automotive Engineers (SAE) has developed a set of recommended practices that provides a standard set of diagnostic fault codes for those component/system faults that are common to all vehicle models. By standardizing fault codes, a qualified independent service technician can diagnose

certain problems on any vehicle using a universal scan tool. Each individual car manufacturer defines its own fault codes for any component or system that is not encompassed by the standard set.

The SAE-defined code has the format P0xxx. A partial list is given in Table 11.1 as an example of a subset of fault codes. Manufacturer-specific codes can have a format P1xxx as illustrated in Table 11.1.

Table 11.1 Fault Code Sample

FIL Indication	Affected Component
P0106 (5)	Manifold absolute pressure circuit range/performance
P0107 (3)	Manifold absolute pressure circuit low input
P0108 (3)	Manifold absolute pressure circuit high input
P0112 (10)	Intake air temperature circuit low input
P0113 (10)	Intake air temperature circuit high input
P0116 (86)	Engine coolant temperature circuit range/performance
P0117 (6)	Engine coolant temperature circuit low input
P0118 (6)	Engine coolant temperature circuit high input
P0122 (7)	Throttle position circuit low input
P0123 (7)	Throttle position circuit high input
P0131 (1)	Primary heated oxygen sensor circuit low voltage (sensor 1)
P0132 (1)	Primary heated oxygen sensor circuit high voltage (sensor 1)
P0133 (61)	Primary heated oxygen slow response (sensor 1)
P0135 (41)	Primary heated oxygen sensor heater circuit malfunction (sensor 1)
P0137 (63)	Secondary heated oxygen sensor circuit low voltage (sensor 2)
P0138 (63)	*Secondary heated oxygen* sensor circuit high voltage (sensor 2)
P0139 (63)	Secondary heated oxygen sensor slow response (sensor 2)
P0141 (65)	Secondary heated oxygen sensor heater circuit malfunction (sensor 2)
P0171 (45)	System too lean
P0172 (45)	System too rich
P1300 or same as P03001	Misfire of multiple cylinders detected
P0301 (71)	Cylinder 1
P0302 (72)	Cylinder 2
P0303 (73)	Cylinder 3
P0304 (74)	Cylinder 4
P0305 (75)	Cylinder 5
P0306 (76)	Cylinder 6
P0335 (4)	Crankshaft position sensor circuit low input
P0336 (4)	Crankshaft position sensor range/performance
P0401 (80)	Exhaust gas recirculation insufficient flow detected
P0420 (67)	Catalyst system efficiency below threshold
P0441 (92)	Evaporative emission control system incorrect purge flow

Table 11.1 Fault Code Sample—cont'd

FIL Indication	Affected Component
P0500 (17)	Vehicle speed sensor circuit malfunction
P0505 (14)	Idle control system malfunction, automatic transaxle
P1107 (13)	Barometric pressure circuit low input
P1108 (13)	Barometric pressure circuit high input
P1297 (20)	Electric load detector circuit low input
P1298 (20)	Electric load detector circuit high input
P1361 (8)	Top dead center sensor intermittent interruption
P1362 (8)	Top dead center sensor no signal exhaust gas recirculation—EGR
P1381 (9)	Cylinder position sensor intermittent interruption
P1382 (9)	Cylinder position sensor no signal
P1459 (92)	Evaporative emission purge flow switch malfunction
P1491 (12)	EGR valve lift insufficient detected
P1498 (12)	EGR valve lift sensor high voltage
P1508 (14)	Idle air control valve circuit failure
P1607 (A)	Engine control module internal circuit failure A

These diagnostic trouble codes (DTCs) will be indicated by the blinking of the malfunction indicator lamp (MIL) with the SCS service connector connected.

The procedure for diagnosing one or more problems during vehicle repair/maintenance begins with the service technician connecting the off-board diagnostic scan tool to the DLC or via an equivalent communication connection. With the ignition switch on and a diagnostic tool connected, data associated with any or all faults are automatically transferred to the diagnostic tool. In addition to the individual fault codes that were stored, additional data indicating the engine and associated components/systems operating conditions may also be transferred (depending on the vehicle model and manufacturer).

In most cases, the most advanced diagnostic tools are computer-based having relatively large databases related to diagnostic and repair procedures. Commonly, these procedures are presented to the service technician in the form of a flowchart (not unlike a flowchart for a computer algorithm). The flowchart appears on the diagnostic tool visual monitor (e.g., computer display) in a graphic/pictorial form. Although the procedures to be followed in the flowchart depend on the particular fault, it is possible to illustrate the procedures with a representative example. The example taken here considers a failure in the primary HEGO. Refer to Chapter 6 for a review of the role played by this important sensor in closed-loop fuel control and Chapter 5 for the theory of its operation.

Once the vehicle has been taken (possibly driven) to an authorized repair facility, the diagnosis begins with the service technician connecting the diagnostic (scan) tool to the DLC. If the onboard diagnostic subsystem has detected an HEGO sensor low voltage fault, it will store the fault code P0131 in memory (see Table 11.1). When properly connected, either of the scan tools (i.e., PSDT or SBDT) will display this code to the service technician. For our example situation, it is presumed that the service bay scan tool

has the relevant diagnostic flowchart stored internal to its computer and will display (either automatically or at the command of the technician) a flowchart such as is depicted in Fig. 11.3.

The first step in the procedure of this flowchart involves verification that a fault has actually occurred and persists. This verification is accomplished during a test-drive with a fully warmed vehicle that is conducted by the service technician with the PSDT connected and configured to measure and display the primary HEGO sensor terminal voltage (V_{HEGO}). If this voltage does not satisfy the condition ($V_{\text{HEGO}} \leq 0.1$ V), the system is deemed to be functioning properly at the time of the test-drive, and the FIL is considered to be intermittent. For this test-drive outcome, a separate path (denoted B in Fig. 11.3) is to be followed. This path includes the recommendation that the service technician examine the wiring associated with the HEGO sensor to check for broken or loose wires or connectors. If no wiring problem is found and the vehicle has experienced other such FIL warnings, the instructions may be to install special recording equipment in the vehicle and either return it to service or repeat the test-drive.

On the other hand, if $V_{\text{HEGO}} \leq 0.1$ V, the flow path directs the service technician to measure fuel pressure. If this pressure is outside limits specified in the service manual, it must be repaired. After repairs are completed, step C involves returning to the flowchart at the point indicated.

If the fuel pressure is within limits, the service technician is directed to electrical tests of the HEGO. With the engine switched off, the HEGO is disconnected from the wiring harness. A diagnostic scan tool is connected via a set of leads with clip-on ends to the sensor terminals of the HEGO (note that for a HEGO, there is also a pair of connectors for the heating element; see Chapter 5). The engine is then started and allowed to idle. The HEGO voltage is measured by the scan tool. At this point in the flowchart, there is a break from point A to the continuous point A at the top right of the flowchart.

If the condition $V_{\text{HEGO}} \leq 0.1$ V is met, it is the HEGO sensor itself that has failed, and it is replaced. To confirm that the problem has been resolved, the technician returns to point D in the procedure. Assuming that the problem is resolved, the procedure will end at point B where it will become concluded that the problem is fixed. If this condition is not met, the sensor is functioning and the problem of low HEGO sensor voltage may be in the wiring harness.

The next step in the flowchart involves testing the wiring harness from the HEGO to the engine control unit (ECU). The HEGO sensor wiring harness is removed from the ECU, and a wire continuity test is performed using either the scan tool or any available multimeter. If there is either intermittent or no continuity in the sensor wires from the ECU to the sensor end and if there are short circuits either between the two sensor leads or from either to ground, the harness is faulty and must be repaired (if possible) or replaced. Again, the procedure will be repeated at D, and if the problem is resolved, the procedure will end in step B with a conclusion that the system is repaired.

If there is continuity, the problem must be in the ECU itself. The service technician is directed to replace the ECU with a known good one. If the problem with low HEGO sensor voltage disappears, a permanent replacement ECU is installed. At this point, there is a return to point D, and if the problem has been resolved, the exit at step B is taken.

If, after the vehicle is returned to service, the FIL is illuminated and the scan tool detects fault P0131 again, the problem is an intermittent fault. Among the possible options is the choice of installing a recording device that can, over a period of time, collect data to identify that an intermittent fault has occurred. It is also possible to replace the HEGO sensor and its wiring harness and continue road testing.

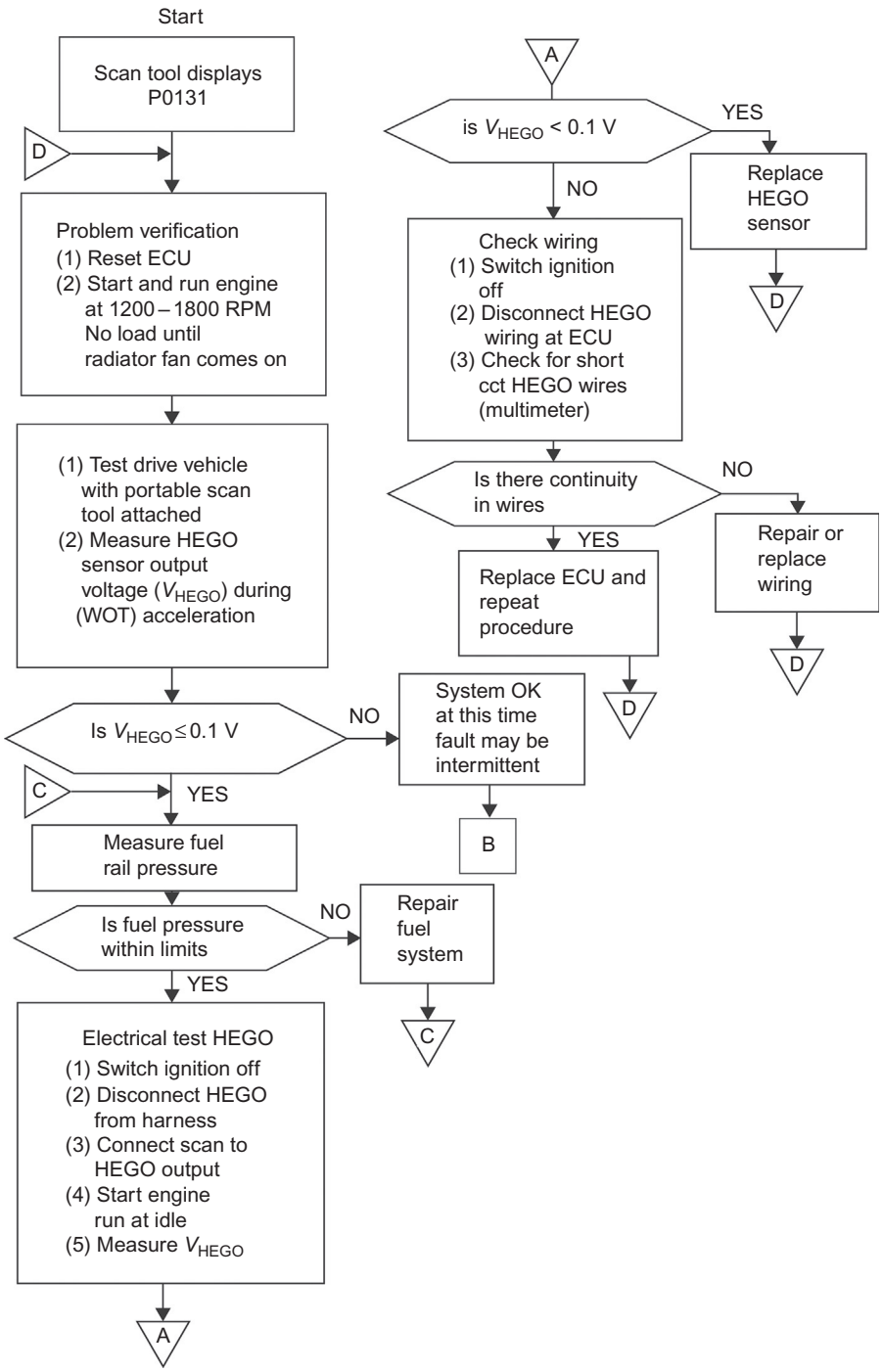


FIG. 11.3 Flowchart for diagnosing fault in primary HEGO.

Another example procedure will be illustrated here by following the steps necessary to respond to the specific fault code P0133, which indicates that the HEGO sensor has slow response. Recall from the discussion in Chapter 6 that the HEGO sensor switches between ~0.1 and 1 V as the mixture switches between the extreme conditions of lean and rich. Recall also that this voltage swing requires that the HEGO sensor must be at a temperature above 200°C. Fault code P0133 means that the HEGO sensor may not swing above or below its cold voltage of ~0.5 V, and that the electronic control system will not go into closed-loop operation (see Chapters 4 and 6) or that the transitions are too slow for closed-loop control to function. Possible causes for fault code P0133 include the following:

- HEGO sensor is not functioning correctly.
- The connections or leads are defective.
- The control unit is not processing the HEGO sensor signal.

Further investigation was required to attempt to isolate the specific problem.

To check the operation of the HEGO sensor, the average value of its output voltage is measured using the scan tool (or a multimeter). The desired voltage is displayed on the scan tool. Using this voltage, the service technician follows a procedure outlined in Figs. 11.4 and 11.5. If the voltage is <0.37 or >0.57 V, the service technician is asked to investigate the wiring harness for defects.

If the HEGO sensor voltage is between 0.37 and 0.57 V, tests are performed to determine whether the HEGO sensor or the control unit is faulty. The service technician must jumper the HEGO sensor leads together at the input to the control unit, simulating a sensor short circuit, and must read the sensor voltage value using the PSDT (or a suitable multimeter). If this voltage is <0.05 V, the control unit is functioning correctly, and the HEGO sensor must be investigated for defects. If the indicated sensor voltage is >0.05 V, the control unit is faulty and should be replaced.

A further test of the proper HEGO sensor dynamic (switching) operation as part of the engine control is illustrated in the flowchart of Fig. 11.5. In this diagnostic procedure, the goal is to ascertain whether the HEGO sensor operation results in closed-loop mode of engine control. As explained in Chapter 6, the engine must be sufficiently warmed before closed-loop operation is activated. The first step in the flowchart is to run the engine and monitor coolant temperature. Once this temperature exceeds a given threshold level, the HEGO sensor should be operating properly even if the heater has failed. The technician is directed to run the engine at fast idle and monitor HEGO sensor voltage. Under these conditions, the sensor should be switching. If the voltage is constant, the sensor has failed and must be replaced.

If the sensor voltage is variable, it must switch from <0.3 to more than 0.6 V. If it does not, it must be replaced. If it does meet this condition, the service technician is directed to determine if closed-loop mode is activated or not. The PSDT tool is configured to read a binary-valued parameter that is termed "closed-loop indicator" (CLI). If CLI $= 0$, the HEGO sensor switching is insufficient to cause closed-loop operation to occur, and the sensor is replaced. If CLI $= 1$, the sensor is OK, and this diagnostic procedure is complete. It should be noted that the ECU could also have failed, but diagnosis of this problem would follow a different flowchart.

In addition to measurement of HEGO sensor, the scan tool can be used by a service technician to measure other variables or parameters as suggested above. We consider, for example, the throttle position sensor that provides an important input to the electronic engine control as explained in Chapters 4–6. The onboard diagnostic can detect out-of-limit values for this sensor and display fault codes, for example, P0122 for voltage below a lower limit or P0123 for voltage above a high limit. However,

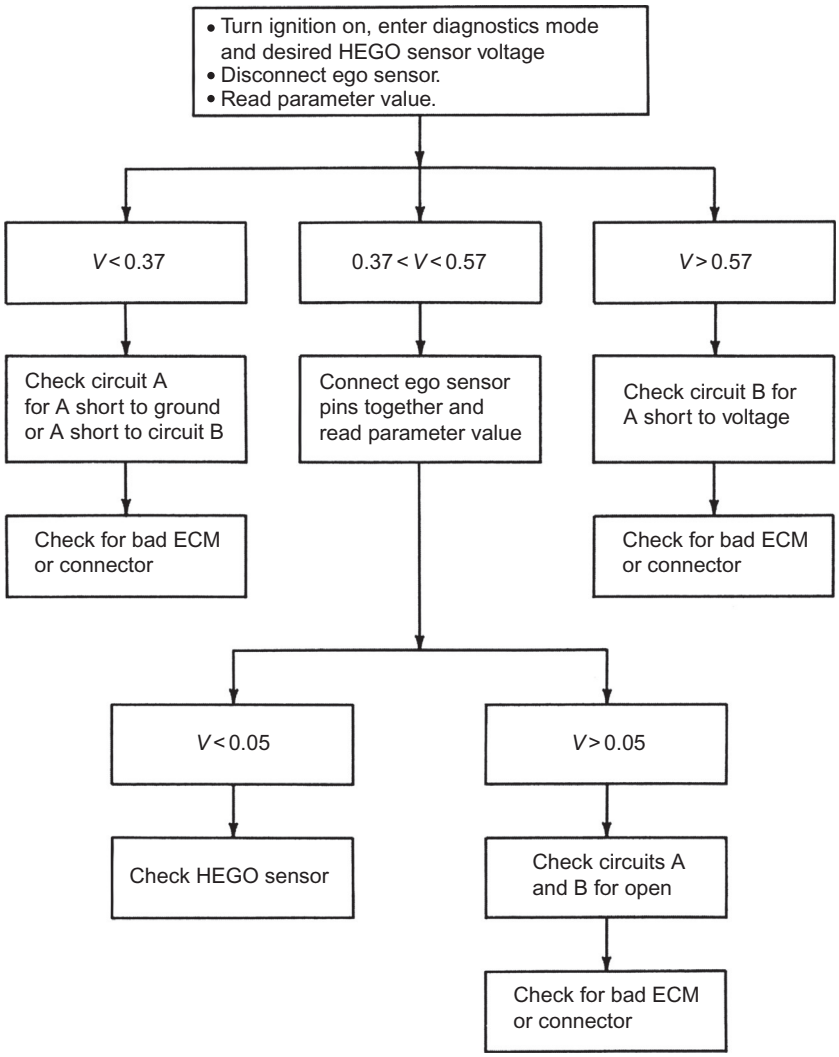


FIG. 11.4 Flowchart for diagnosis of HEGO sensor output voltage problems.

other throttle position sensor faults are possible that are not detected by the exemplary onboard diagnostic system. A change in calibration of this sensor will normally result in an incorrect computation of fuel injector base pulse duration (see Chapter 6). Such a calibration failure could result from a change in the supply voltage to the sensor. Even though no fault code is set for such a failure (in this hypothetical example), a service technician with sufficient experience and knowledge may suspect such a failure if the vehicle driver reports an apparent reduction in performance under certain driving conditions.

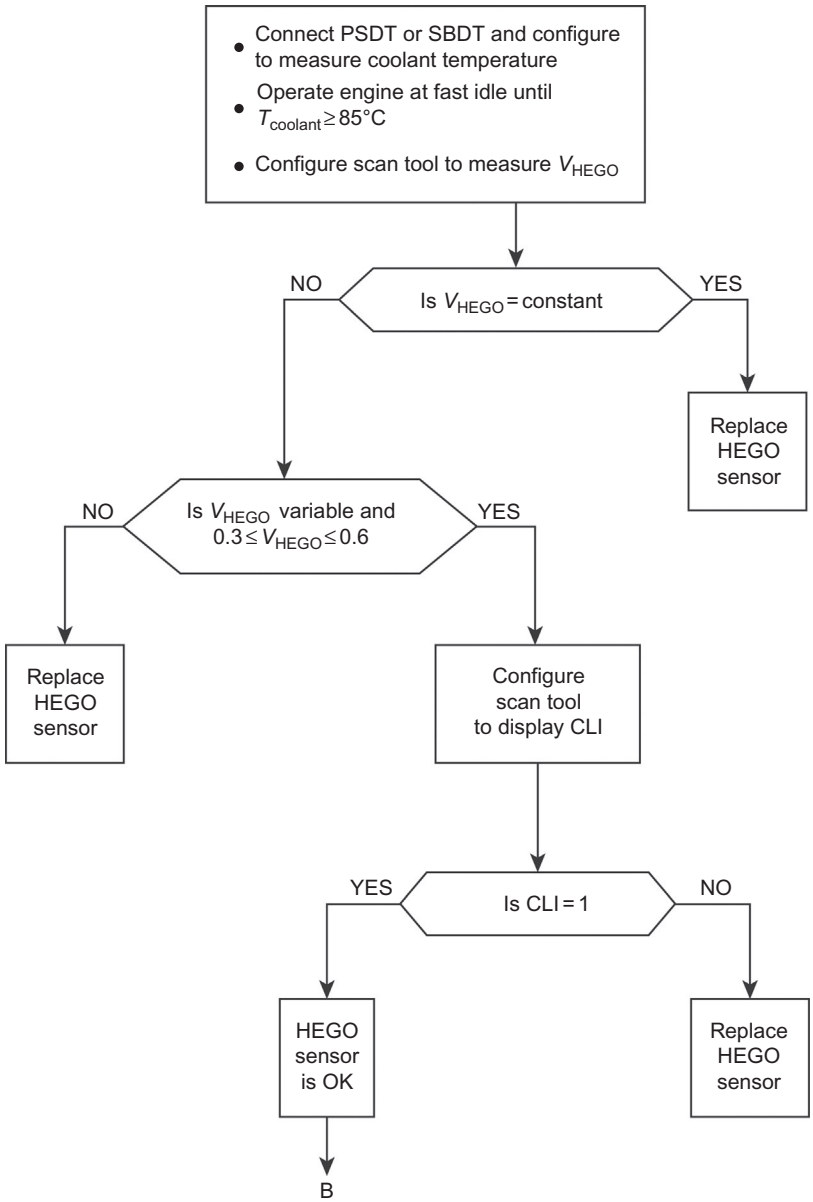


FIG. 11.5 Flowchart for HEGO sensor proper switching test.

The technician can configure the scan tool to measure the throttle position sensor voltage. Then, with the ignition switch in the on position but with the engine not running, the service technician can measure the voltage as the throttle is depressed. Although it is theoretically possible to independently measure throttle angular position θ_t and to obtain a plot of sensor voltage $V_t(\theta_t)$, normally, it is sufficient for diagnostic purposes to qualitatively examine the voltage as the throttle is changed. This sensor voltage should change smoothly and roughly linearly with θ_t. Similar measurements can be made on other sensors that might have developed partial failures (e.g., calibration shift) that are not sufficient to be detected by the onboard diagnostic system.

In addition to parameter and variable measurements, the diagnosis of problems with various switches is often desirable or even necessary. Various examples of the important function of certain switches have been explained in previous chapters. For example, in a traditional cruise control system, the brake pedal switch has the critical safety-related function of disconnecting the throttle actuator from the throttle linkage in a cruise control system (see Chapter 7) when the driver applies the brakes. The onboard diagnostic system cannot detect a failure in this switch unless there is an independent means of sensing that brakes are applied (e.g., via a brake pressure sensor). Owing to this potentially inherent limitation of the onboard diagnostic system, it is desirable to perform a sequence of switch tests during a routine vehicle servicing procedure. The evaluation of various switches can be implemented automatically via the diagnostic tool with the involvement of the service technician. Such an automatic switch test procedure was implemented in at least one production vehicle.

We illustrate this switch test procedure with the above exemplary system in which the scan tool is configured to display two-digit diagnostic codes. The two-digit codes and associated circuit are presented in Fig. 11.6. In this figure, the relevant diagnostic codes displayed on the scan tool are represented by digits AA.

For this example, the switch tests involve diagnostic codes 71–80 and provide checks on the switches indicated in Fig. 11.6.

To begin the switch tests, the service technician must depress and release the brake pedal. If there is no brake switch failure, then the code advances to 71. If the display does not advance, then the control unit is not processing the brake switch signal, and further diagnosis is required. For such a failure, the service technician locates the specific flowchart (such as seen in Fig. 11.7) for diagnosis of the particular switch failure and follows the procedure outlined. The detailed tests performed by the service technician are continuity checks that are performed with the PSDT or a multimeter. Fig. 11.8 depicts the cruise control brake circuit diagram.

Whenever any switch test fails, a diagnostic flowchart is called up by the service technician, and its steps are followed in the sequence displayed on the scan tool. Similar procedures are followed for each switch test in the sequence. This procedure sequence is as follows:

(1) With code 71 displayed, depress and release brake pedal for normal operation, the display advances.
(2) With code 72 displayed, depress the throttle from idle position to wide-open position. The control unit tests the throttle switch, and advances the display to code 73 for normal operation.
(3) With code 73 displayed, the transmission selector is moved to drive and then neutral. This operation tests the drive switch, and the display advances to code 74 for normal operation.
(4) With code 74 displayed, the transmission selector is moved to reverse and then to park. This tests the reverse switch operation, and the display advances to code 75 for normal operation.

Switch tests

Two-digit display code

Code AA	Circuit being tested
71	Cruise control brake switch
72	Throttle switch
73	Drive circuit
74	Reverse circuit
75	Cruise on/off
76	Set/coast
77	Resume/acceleration
78	Instant/average
79	Reset
80	Air conditioning clutch

FIG. 11.6 Switch test sequence.

(5) With 75 displayed, the cruise control is switched from off to on and back to off, testing the cruise control switch. For normal operation, the display advances to 76.

(6) With code 76 displayed and the cruise control switch on, depress and release the set/coast button. If the button (switch) is operating normally, the display advances to 77.

(7) With 77 displayed and with the cruise control instrument on, depress and release the resume/acceleration switch. If the switch is operating normally, the display advances to 78.

(8) With 78 displayed, depress and release the instant/average button on the trip information computer (TIC). If the button is working normally, the code advances to 79.

(9) With 79 displayed, depress and release the reset button on the TIC panel. If the reset button is working normally, the code will advance to 80.

(10) With 80 displayed, depress and release the rear defogger button on the climate control head. If the defogger switch is working normally, the code advances to 70, thereby completing the switch tests.

This exemplary diagnostic tool can also be used to display certain engine parameters with the engine running (either in the service bay or on a road test). The scan tool gives the measurement and the normal range for the parameter.

Fig. 11.9 shows the parameter values in sequence for a traditional exemplary vehicle. Parameter 01 is the angular deflection of the throttle in degrees from idle position.

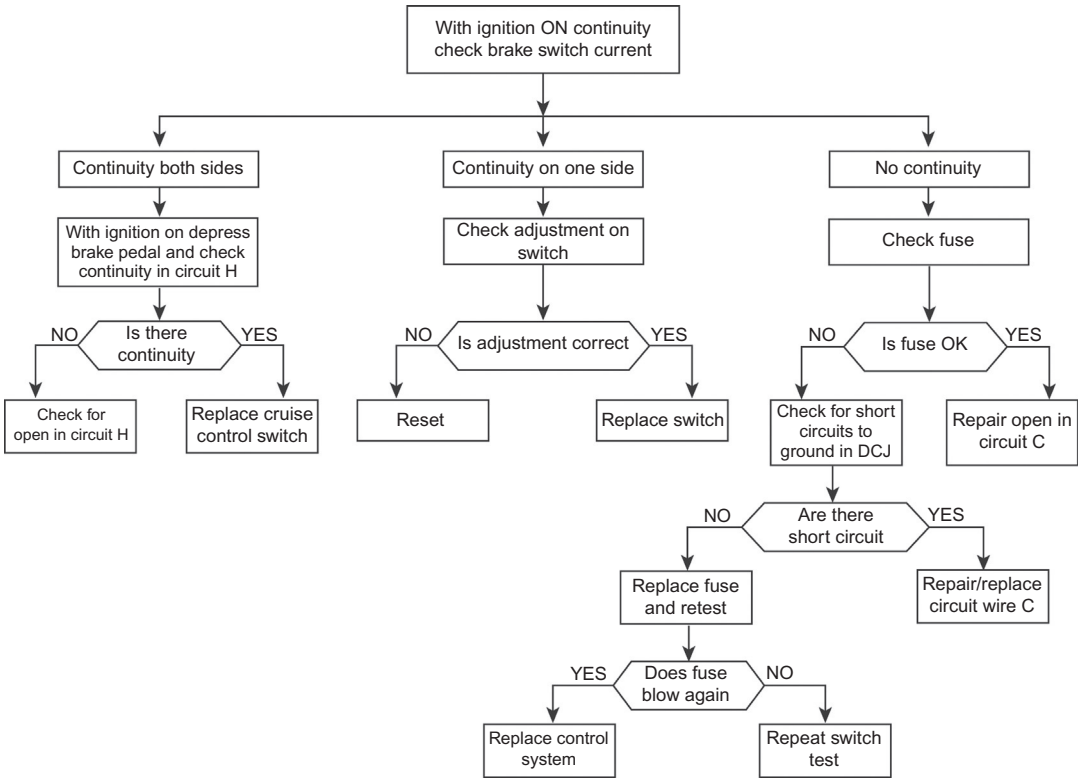


FIG. 11.7 Flowchart for cruise control brake circuit.

Parameter 02 is the manifold absolute pressure in kilopascals. The range for this parameter is 14–99, (for a normally aspirated engine) with 14 representing about the maximum manifold vacuum. Parameter 03 is the absolute atmospheric pressure in kilopascals. Normal atmospheric pressure is roughly 90–100 kPa at sea level. Parameter 04 is the coolant temperature, and Parameter 05 is the intake manifold temperature.

Parameter 06 is the duration of an exemplary traditional fuel injector pulse with in milliseconds. Refer to Chapters 4–6 for an explanation of the injector configuration, the pulse widths, and the influence of these pulse widths on the air/fuel mixture.

Parameter 07 is the average value for the HEGO sensor output voltage. Reference was made earlier in this chapter to the diagnostic use of this parameter. Recall that the HEGO sensor switches between about 0.1 and 1 V as the mixture oscillates between lean and rich. The displayed value is the time average for this voltage, which varies with the duty cycle of the mixture.

Parameter 08 is the spark advance in degrees before TDC for a representative engine. This value should agree with that obtained using a SBDT configured in the engine analyzer mode. Although it is not shown in Fig. 11.9, Parameter 09 is the number of ignition cycles that have occurred since a trouble

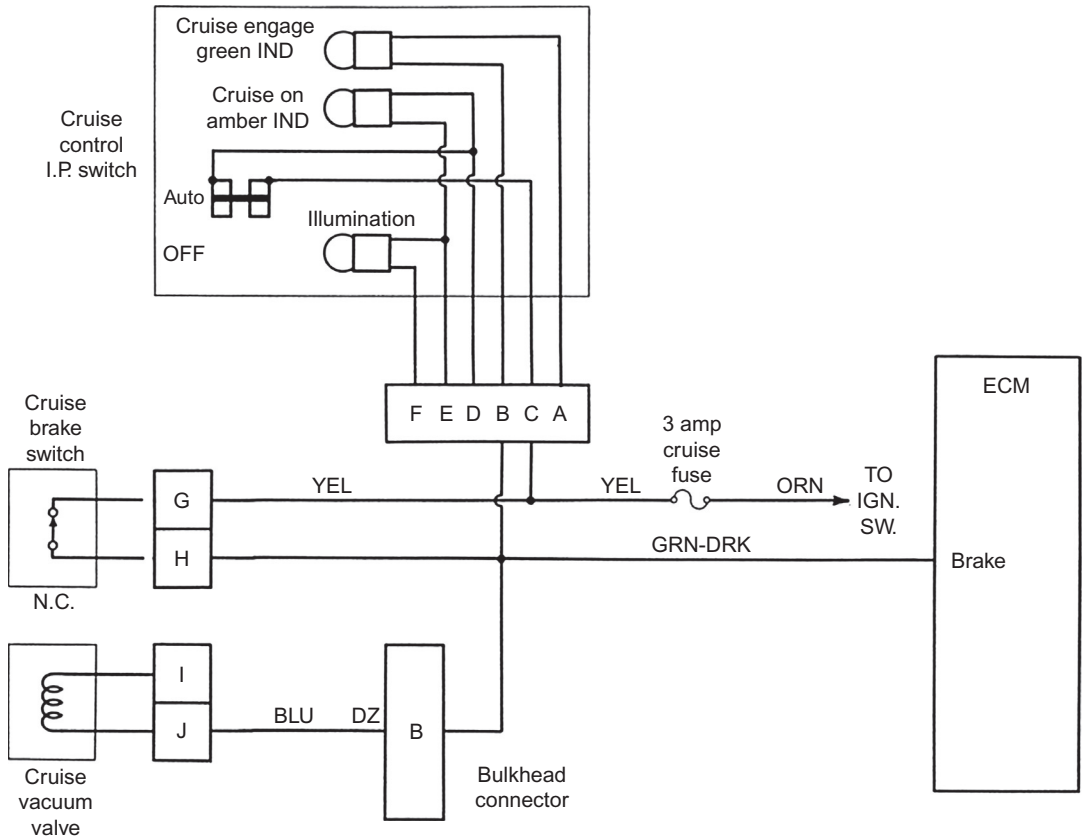


FIG. 11.8 Cruise control brake circuit.

Parameter number	Parameter	Normal range
01	Throttle position	0-31
02	Manifold pressure	14-99
03	Barometric pressure	80-99
04	Coolant temperature	0-99
05	Manifold air temperature	9-99
06	Injector pulse duration	0-9.9
07	HEGO sensor voltage	0.5-0.6
08	Spark advance (degrees)	0-25

FIG. 11.9 Chart of exemplary engine parameters with normal ranges.

code was set in memory. If an OEM specific number of such cycles have occurred without a fault, this counter is set to zero, and all trouble codes are cleared.

Parameter 10 (not shown in Fig. 11.9) is a logical (binary) variable that indicates whether the engine control system is operating in open or closed loop (i.e., the CLI). A value of 1 corresponds to closed loop, which means that data from the HEGO sensor are fed back to the controller to be used in setting injector pulse duration. Zero for this variable indicates open-loop operation, as explained in Chapters 5 and 6. Parameter 11 is the battery voltage.

ONBOARD DIAGNOSIS (OBD II)

As mentioned in the introduction to this chapter, onboard diagnosis has also been mandated by government regulation, particularly if a vehicle failure could damage emission control systems. The relatively severe requirement for onboard diagnosis is known as OBD II. This requirement is intended to ensure that the emission control system is functioning as intended.

Automotive emission control systems, which have been discussed in Chapters 4 and 6, consist of fuel and ignition control for the three-way catalytic converter and controls for EGR, secondary air injection, and evaporative emission. The OBD II regulations require real-time monitoring of the performance of the emission control system components. For example, the performance of the catalytic converter must be monitored using a temperature sensor for measuring converter temperature and a pair of HEGO sensors (one before and one after the converter).

MISFIRE DETECTION

Another requirement for OBD II is a misfire detection system. It is known that under misfiring conditions (failure of the mixture to ignite), exhaust emissions increase. In severe cases, the catalytic converter itself can be irreversibly damaged. Standardization of the hardware and communication protocols for OBD II had been provided by SAE standards. The OBD II requirement includes two standardized connectors, one of which must be built into the vehicle DLC. There are five SAE communication standards for communications protocol (e.g., J1850 pwm).

The only cost-effective means of meeting OBD II requirements involves electronic instrumentation. Owing to intellectual property issues, it is not feasible to present an actual misfire detection system used by any particular automotive manufacturer. Rather, we present a hypothetical misfire detection system that is mathematical model-based and has been tested under laboratory conditions and in actual road tests.

MODEL-BASED MISFIRE DETECTION SYSTEM

A model-based method of detecting engine misfires requires a dynamic model for the power train of sufficient detail and accuracy to be able to represent the relationship between the instantaneous torque fluctuations and the corresponding fluctuations in crankshaft instantaneous angular speed $\omega_e(t)$. It is shown later in this section that measurements of $\omega_e(t)$ can be used as the basis for misfire detection in accordance with the following model. The instantaneous net torque T_n applied at the flywheel consists of the algebraic sum

$$T_n[\theta_e(t)] = T_i[\theta_e(t)] + T_R[\theta_e(t)] + T_{Fp}[\theta_e(t)] - T_l[\theta_e(t)] \tag{11.12}$$

where

$\theta_e(t)$ = crankshaft instantaneous angular position
$T_i[\theta_e(t)]$ = indicated torque
$T_R[\theta_e(t)]$ = torque due to inertial forces of reciprocating components
$T_{Fp}[\theta_e(t)]$ = friction and pumping loss torque
$T_l[\theta_e(t)]$ = load torque from transmission.

The indicated torque is the torque that is applied to the crankshaft due to cylinder pressure during combustion acting on the piston area (A_p) through the instantaneous lever arm $\ell(\theta_e)$ of the connecting rod crankshaft throw structure (see Chapter 4, Fig. 4.10). The friction component of T_{Fp} is due to the sliding friction of all moving surfaces, and the pumping component of T_{Fp} is the torque required to pump the fuel air mixture into each cylinder and pump the exhaust gases out of the engine through the exhaust system.

The reciprocating torque is the torque applied to the crankshaft due to the inertial forces associated with the reciprocating motion of the piston/connecting rod/crankshaft throw. This torque amplitude increases quadratically with RPM but can be computed with great accuracy for any given engine configuration from the known geometry and component masses.

For the purposes of illustrating the present concept for misfire detection, a number of simplifying assumptions are made. There is negligible loss of model robustness by assuming that the crankshaft is infinitely stiff and experiences insignificant torsional motion in response to the torque fluctuations. It is also adequate for the present purposes to assume that the connecting rod is sufficiently long relative to the crankshaft throw (R_c) and that the piston pin offset is negligible such that the indicated torque due to the power stroke of the mth cylinder is given by

$$T_m(\theta_e) = A_p R_c (p_c - p_o) f_m(\theta_e) \tag{11.13}$$

where

$$f_m(\theta_e) = \sin(\theta_e - \theta_m)\left[1 + \frac{(R_c/L_c)\cos(\theta - \theta_m)}{\sqrt{1 - (R_c/L_c)^2 \sin^2(\theta_e - \theta_w)}}\right]$$

and where

L_c = connecting rod length
R_c = crankshaft throw
p_c = cylinder pressure
p_o = atmospheric pressure
where $\theta_m = \theta_e$ at TDC for cylinder m.

The origin for θ_e is taken as the crankshaft angle for the number 1 cylinder at TDC for compression/combustion strokes. The indicated torque is the sum of the indicated torque for all M cylinders of an M cylinder engine:

$$T_i(\theta_e) = \sum_{m=1}^{M} T_m(\theta_e)$$

The reciprocating torque associated with the mth cylinder are given by

$$T_{Rm}(\theta_e) \cong M_{eq} R_c^2 f_T(\theta_e)\left[f_T(\theta_e)\dot{\omega}_e + f_B(\theta)\omega_e^2\right] \tag{11.14}$$

where $\omega_e = \dfrac{d\theta_e}{dt}$

$$f_T(\theta_e) = \sin(\theta_e - \theta_m) + \frac{(R_c/L_c)\sin[2(\theta_2 - \theta_m)]}{2\sqrt{1-(R_c/L_c)^2\sin^2(\theta_2-\theta_m)}}$$

$$f_B(\theta_e) = \frac{R_c}{L_c}\left\{\frac{\cos[2(\theta_e-\theta_m)]}{\sqrt{1-(R_c/L_c)^2\sin^2(\theta_e-\theta_m)}}\right\} + \left(\frac{R_c}{L_c}\right)^3 \frac{\sin^2(\theta_e-\theta_m)}{4\sqrt{(1-R_c/L_c)^2\sin^2(\theta_e-\theta_m)^3}} + \cos(\theta_e-\theta_m) \tag{11.15}$$

where M_{eq} = sum of the mass of the piston, wrist pin, and one-third of the connecting rod.

The combined reciprocating torque T_R is given by

$$T_R(\theta_e) = \sum_{m=1}^{M} T_{Rm}(\theta_e) \tag{11.16}$$

The model for T_R was developed by the first author of Ref. 1 on p. 564.

For the purposes of modeling the engine for misfire detection, it is possible to approximate $T_{Fp}(\theta_e)$ with a linearized model as given below:

$$T_{Fp}(\theta_e) \cong R_e \omega_e$$

where R_e = linearized friction coefficient.

The net torque $T(\theta_e)$ applied to the crankshaft is the sum of the components:

$$T(\theta_e) = T_i(\theta_e) + T_R(\theta_e) + T_{Fp}(\theta_e) \tag{11.17}$$

The instantaneous torque produced by any reciprocating engine fluctuates as a function of θ_e as given by Eq. (11.17) and the models given for the components derived above. In addition to the variation in lever arm, there is a significant variation in cylinder pressure $p_c(\theta_e)$. The combination of p_c variation reciprocating forces and lever arm variation results in a variation in the indicated torque $T_i(\theta_e)$ for each cylinder during each engine cycle. For an ideal engine with perfect and identical combustion in each cylinder for the engine at a fixed load and RPM, the net torque would have a perfectly periodic waveform for each cylinder in each cycle.

However, in a practical engine, there is a level of fluctuation in the cylinder to cylinder indicated torque. On the other hand, the normal fluctuation in cylinder to cylinder T_i is very small compared with the case of a misfiring cylinder. The peak torque generated by a misfiring cylinder is significantly smaller than that generated by cylinders with normal combustion.

The present method of misfire detection in an engine is based upon a metric that represents the nonuniformity in torque generation from cylinder to cylinder in any given cycle. If every cylinder produced exactly the same torque during a given engine cycle, the fluctuations in $T(\theta_e)$ would have exactly the same extrema (i.e., relative maximum and relative minimum). However, this situation is never achieved in practice due to variations in fueling and combustion. Nevertheless, these extrema are nearly the same for a normal running engine.

On the other hand, for one or more misfires (or partial misfires), these extrema are significantly different. That is, the nonuniformity in $T(\theta_e)$ is relatively small for normal engines and increases

significantly for misfire conditions. The present method of misfire detection is based on a metric for torque nonuniformity in the form of a $2M$-dimensional vector for a given engine cycle for an M cylinder engine, which is denoted $\bar{n}$ and is given by

$$\bar{n} = \delta\bar{T} - \delta T_{av}\bar{u} \tag{11.18}$$

In this definition for $\bar{n}$, the $\delta\bar{T}$ and $\delta\bar{T}_{av}$ are derived from extrema of $T(\theta_e)$, which are denoted T_m and T'_m. In the following, we begin with a vector of extrema $\bar{T}$:

$$\bar{T} = \left[T_1, T'_1, \ldots, T_M, T'_M\right]^T \in R^{2M}$$

where

$T_m = T_m\left(\theta_e^m\right)$ = relative maximum of T_m
θ_e^m = crankshaft angle at which T_m occurs
and where $T'_m = T_m(\theta_{em})$ = relative minimum of T_m
$\theta_{em} = \theta_e$ at which T'_m occurs.

That is, the extremal values for $T(\theta_e)$ are characterized by

$$\begin{aligned} &\left.\frac{dT}{d\theta_e}\right|_{\theta_e^m} = 0 \\ &\left.\frac{d^2T}{d\theta_e^2}\right|_{\theta_e^m} < 0 \\ &\left.\frac{dT}{d\theta_e}\right|_{\theta_{em}} = 0 \\ &\left.\frac{d^2T}{d\theta_e^2}\right|_{\theta_{em}} > 0 \end{aligned} \qquad m = 1, 2, \cdots, M \tag{11.19}$$

Note that during certain engine operating conditions (e.g., low load and high RPM), it is possible to have more than one relative maximum or minimum for a given cylinder cycle (e.g., due to reciprocating forces). If this is the case, θ^m closest to TDC and θ_m closest to BDC must be chosen.

The average of these extrema is given by

$$T_{av} = \frac{1}{2M}\sum_{m=1}^{M}\left[T_m + T'_m\right] \text{(average of extrema per cycle)} \tag{11.20}$$

The vector $\delta\bar{T}$ in Eq. (11.18) is given by

$$\delta\bar{T} = \left[\delta T_1 \delta T'_1 \cdots \delta T_m \delta T'_m \cdots \delta T_M \delta T'_M\right]^T$$

where

$$\begin{aligned} &\delta T_m = T_m - T_{av} \\ &\delta T'_m = T'_m - T_{av} \qquad m = 1, 2, \cdots, M \\ &\delta T_{av} = \frac{1}{2M}\sum_{m=1}^{M}\left[\delta T_m - \delta T'_m\right] \quad \text{(average torque deviation magnitude)} \end{aligned}$$

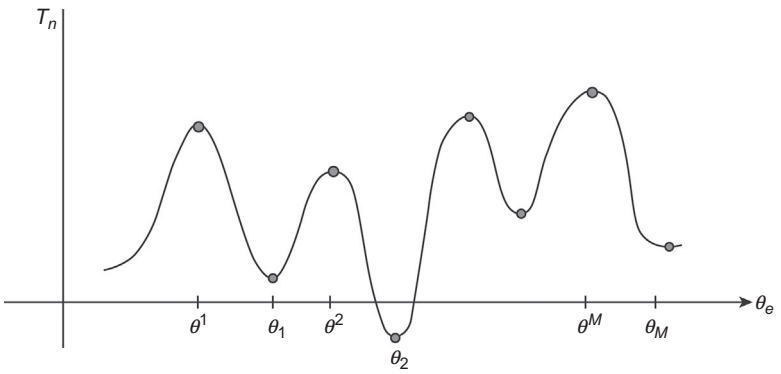


FIG. 11.10 Illustrative torque waveform and its extrema.

In Eq. (11.18), $\bar{u}$ is a $2M$-dimensional vector of 1's with alternating signs is given by

$$\bar{u} = [1, -1, 1, -1 \ldots 1, -1]^T \quad \in R^{2M}$$

Fig. 11.10 illustrates (qualitatively) the nonuniformity vector samples for a hypothetical torque waveform. Note that for perfectly uniform torque waveform $\delta T'_m = -\delta T_m$ (for all m) with the result that $\bar{n}$ is a $2M$-dimensional vector with all elements zero.

The presence of a misfire can readily be detected by a scalar n derived from the L_2 norm of the vector $\bar{n}$:

$$n = \|\bar{n}\| \quad L_2 \text{ norm} \tag{11.21}$$

The misfire torque nonuniformity vector $\bar{n}$ is demonstrated with some specific samples of data from a four-cylinder engine. The torque vectors for three circumstances are denoted:

$\bar{T}$	typical normal operation
$\bar{T}_2$	partial misfire cylinder number 2
T_3	partial misfire cylinder number 3

In addition, for comparison purposes, an ideal torque vector $\bar{T}_0$ that corresponds to uniform torque production is presented. As stated above, all of the components for $\bar{n}(T_0)$ are identically 0. The various vectors and average values as defined above are given in Table 11.2.

All of the calculations required for Eq. (11.18) are performed, and the results are given in Table 11.2 and the corresponding nonuniformity metric vector $\bar{n}$ for each sample. The L_2 norm for the $\bar{n}$ metric for each sample is given by

$\|\bar{n}_0\| = 0$	(ideal combustion)
$\|\bar{n}\| = 3.9$	(normal combustion)
$\|\bar{n}_2\| = 41.7$	(misfire in cylinder 2)
$\|\bar{n}_3\| = 41.2$	(misfire in cylinder 3)

Table 11.2 Illustrative Samples of $\bar{n}$

T_0 = [70, 7, 70, 7, 70, 7, 70, 7]
T_0_av = 38.5
d_T_0 = [31.50, −31.50, 31.50, −31.50, 31.50, −31.50, 31.50, −31.50]
$d_T\text{av}_0$ = 31.5
n_0 = [0, 0, 0, 0, 0, 0, 0, 0]
T = [67.2000, 7.0000, 65.1000, 7.7000, 68.6000, 8.4000, 63.7000, 8.0500]
T_av = 36.969
d_T = [30.231, −29.969, 28.131, −29.269, 31.631, −28.568, 26.731, −28.919]
$d_T$av = 29.181
n = [1.052, −0.788, −1.050, −0.088, 2.45, 0.613, −2.45, 0.262]
T_2 = [63.7, 12.6, 15.4, 10.5, 63.7, 11.2, 63.0, 11.2]
T_2_av = 31.4125
d_T_2 = [32.287, −18.812, −16.012, −20.912, 32.287, −20.212, 31.587, −20.212]
$d_T\text{av}_2$ = 20.038
n_2 = [12.25, 1.225, −36.05, −0.875, 12.250, −0.175, 11.550, −0.175]
T_3 = [66.5, 13.3, 65.1 14.7, 17.5, 15.4, 63.0, 18.2]
T_3_av = 34.212
d_T_3 = [32.287, −20.912, 30.887, −19.512, −16.712, −18.812, 28.787, −16.012]
$d_T\text{av}_3$ = 18.812
n_3 = [13.475, −2.10, 12.075, −0.70, −35.525, 0.00, 9.975, 2.80]

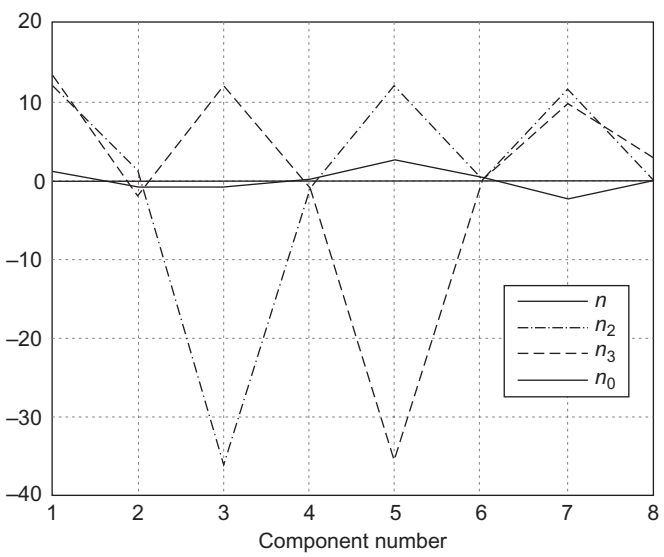


FIG. 11.11 Plot of the components for the sample nonuniformity vectors.

For the samples with misfire, the L_2 norms are roughly a factor of 10 greater than for normal engine operation. As a further illustration of the nonuniformity metric, Fig. 11.11 is a plot of line graphs connecting the individual components of each sample. The line segments connecting the points are only presented in the figures to visually simplify identification of the components and to show the contrast between the normal and misfiring n vectors. The n_o in this figure is the abscissa line of the plot.

In addition to detecting that a cylinder is misfiring, it is also desirable for diagnostic purposes to identify the cylinder that is misfiring. One method for identifying the individual cylinder is to compute the angle of the nonuniformity vector $\bar{n}$ with respect to $\bar{n}$ for a known nonmisfiring (normal combustion) vector denoted $\bar{n}_{\text{norm}}$. The angle $\phi(\bar{n}, \bar{n}_{\text{norm}})$ can be computed using the following equation:

$$\phi(\bar{n}, \bar{n}_{\text{norm}}) = \cos^{-1}[(<\bar{n}, \bar{n}_{\text{norm}}>)/(\|\bar{n}\| \cdot \|\bar{n}_{\text{norm}}\|)]$$

where $<\bar{n}, \bar{n}_{\text{norm}}>$ = inner product of $\bar{n}, \bar{n}_{\text{norm}}$.

As an illustration of this method, the angles for the examples given in Table 11.2 are as follows:

$$\phi(\bar{n}_2, \bar{n}) = 72 \text{ degrees}$$
$$\phi(\bar{n}_3 \bar{n}) = 132 \text{ degrees}$$

Although the data are not given in Table 11.2, a separate sample for misfire on cylinders 1 and 4 yielded the following:

$$\|\bar{n}_1\| = 41.8$$
$$\phi(\bar{n}_1, \bar{n}) = 107 \text{ degrees}$$
$$\|\bar{n}_4\| = 41.4$$
$$\phi(\bar{n}_4, \bar{n}) = 43.8 \text{ degrees}$$

Clearly, the norm $\|\bar{n}\|$ can detect a misfire either on a continuous basis or on an intermittent misfire condition. The angle $\phi(\bar{n}, \bar{n}_{\text{norm}})$ provides information for identifying the cylinder that is misfiring in the present representative example. In this exemplary angle model for misfiring cylinder identification, the vector $\bar{n}_{\text{norm}}$ is simply a reference vector. It is possible to determine a reference vector (denoted $\bar{n}_{\text{ref}}$) such that the angles calculated $\phi(\bar{n}_m, \bar{n}_{\text{ref}})$ have a maximum separation, thereby optimizing the identification of the misfiring cylinder(s).

The instrumentation for obtaining the torque vector is explained later in this section of the chapter. The computations involved in obtaining and processing this vector are readily accomplished in an on board digital computer that could either be a stand-alone device. These computations could also be accomplished within the power train controller.

One of the issues in detecting misfire with the L_2 norm of vector n is the randomness of combustion for even the best available engines. For an engine operating normally (without misfire), the nonuniformity index ($\|\bar{n}\|$) has a random process component. However, with a single-cylinder misfire, there is a very large increase in $\|\bar{n}\|$ compared with the mean and 3σ interval for normal combustion as demonstrated next.

The actual misfire detection is done on a statistical hypothesis testing basis. An experimental test of the misfire detection method was conducted in which there are three conditions expressed as hypothesis H_0, H_1, and H_2 where

$H_0 \rightarrow$ normal engine operation,
$H_1 \rightarrow$ misfire in a single cylinder within an engine cycle,

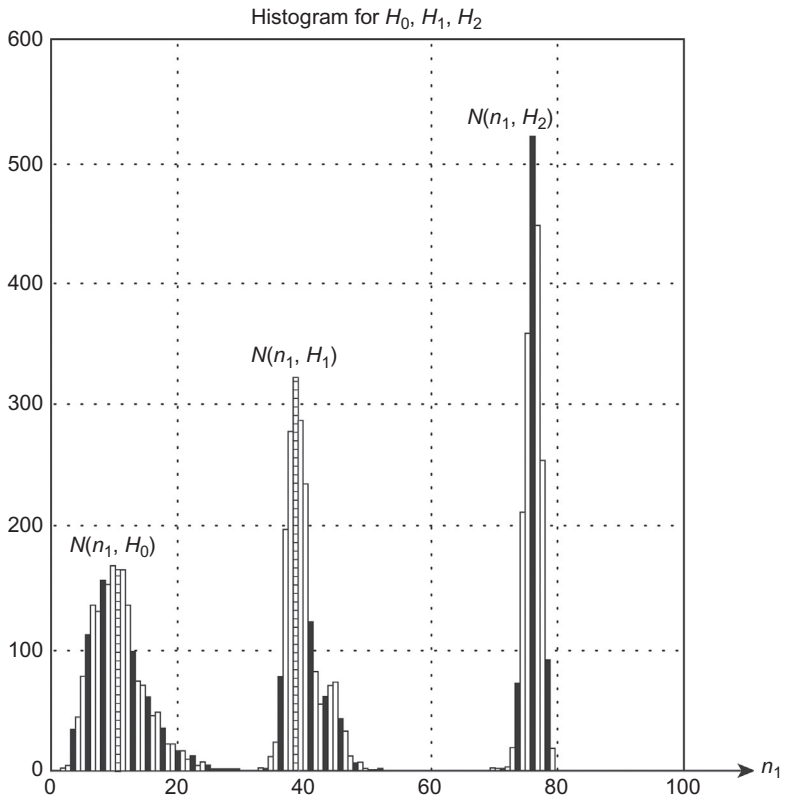


FIG. 11.12 Histograms of nonuniformity index.

$H_2 \rightarrow$ misfire in two cylinders within an engine cycle.

The tests were conducted on a four-cylinder engine having port fuel injection on each cylinder. The engine control system was programmed to interrupt fuel injection on one or two cylinders or on none. Instrumentation (explained later) was constructed that obtained the L_1 norm of n (n_1) for each of several thousand engine cycles. Fig. 11.12 is a plot of the histogram for these data in which the distribution centered near $n_1 \cong 10$ corresponded to H_0.

The distribution centered near $n_1 \cong 40$ corresponds to H_1 and that centered near $n_1 \cong 80$ corresponds to H_2. This histogram consists of the number of occurrences at the value n_1 for each hypothesis H_i, N (n_1, H_i) $(i = 0, 1, 2)$ of nonuniformity index n_1. The specific hypothesis under any test was determined by the number of cylinders that were caused to be misfired in the associated control instrumentation (i.e., 0, 1, 2 misfiring cylinders).

The detection of misfire can be based on a variety of criteria. For example, a simple statistical test can be a threshold comparison. Let N_{av} (H_0) be the mean value for n_1 under H_0 and N_{av} (H_1) be the mean value for n_1 under H_1. A threshold n_t is chosen such that

$$n_t = [N_{av}(H_0) + N_{av}(H_1)]/2 \tag{11.22}$$

The criterion for misfire is as follows:

$$n_1 > n_t \rightarrow \text{misfire}$$
$$n_1 < n_t \rightarrow \text{no misfire}$$

There are two types of error associated with the above misfire criterion:

$$n_1 < n_t \text{ for an actual misfire(missed detection)}$$
$$n_1 > n_t \text{ for no misfire(false alarm)}$$

It should be noted that a similar statistical study was conducted using other threshold values. Choosing the threshold as done above yields approximately equal costs to both missed detection and false alarms.

The above method of detecting misfires (based only on $\|\bar{n}\|$) does not, by itself, identify the cylinder(s) that is (are) misfiring. The nonuniformity index vector $\bar{n}_1$ can be used as a further onboard diagnosis tool to assist the repair technician in identifying the misfiring cylinder(s). For an otherwise properly running engine, a unique vector $(\bar{n})$ tends to be associated with the misfire in each cylinder. Assume initially that the above misfire detection indicates only a single cylinder is misfiring.

The unique "signature" nonuniformity index for a consistent misfire in cylinder m will have nonuniformity vector $\bar{n}(m)$. This "signature" can be obtained during engine control development by running the engine with cylinder m purposely disabled (i.e., via fuel or spark). Data for the nonuniformity vector $\bar{n}(m)$ are given by the statistical average of $\bar{n}$ over a sample of K engine cycles:

$$\bar{n}(m) = \frac{1}{K}\sum_{k=1}^{K} \bar{n}_k \qquad m = 1,2,\cdots,M \tag{11.23}$$

where $\bar{n}_k$ = nonuniformity vector for the kth engine cycle.

Each of these M vectors is directed to a point in a $2M$-dimensional space. One method of identifying the misfiring cylinder is done by finding the shortest "distance" from a nonuniformity vector $\bar{n}$ to these vectors. This vector distance (for the kth engine cycle) in $2M$-dimensional space $\bar{\delta}_k(m)$ is given by Eq. (11.16)

$$\bar{\delta}_k(m) = \bar{n}(m) - \bar{n}_k \tag{11.24}$$

where $\bar{n}$ is the measured nonuniformity vector for an engine cycle in which a single-cylinder misfire has been detected. The problem of isolating the misfiring cylinder is reduced to finding the cylinder number m_o, which yields the smallest L_2 norm for the vector distance

$$\min_m \left(\left\| \bar{\delta}_m \right\|_2 \right) = \left\| \bar{\delta}_{m_o} \right\|_2 \tag{11.25}$$

That is, cylinder m_o ($m_o = 1, 2, \ldots, M$) has the minimum $\left\| \bar{\delta}_{m_o} \right\|_2$ and is identified as the misfiring cylinder.

As demonstrated earlier with respect to some samples of $\bar{n}$, an alternate approach to identifying the misfiring cylinder is to compute the angle $\phi(k)$ between the reference nonuniformity vector $\bar{n}_{ref}$ and $\bar{n}_k$:

$$\phi(k) = \cos^{1}\left[< \bar{n}_{ref}, \bar{n}_k > / \left(\left\| \bar{n}_{ref} \right\| \cdot \|\bar{n}_k\| \right) \right]$$

The cylinder that is misfiring will have the minimum angle, difference $\phi(k) - \phi(n_m, n_{ref})$ for the kth cycle.

If cylinder m_o consistently misfires (as opposed to a random pattern) then by setting an appropriate flag in the diagnostic memory, the repair technician can know which cylinder should be analyzed for problems. This type of information greatly reduces the off-board diagnosis and maintenance effort. Often, vehicles experience intermittent failures. A relatively simple onboard analysis program can evaluate the frequency of and the consistency of an intermittently misfiring cylinder.

Although the above method has great potential for detecting and diagnosing misfire problems, it cannot be directly implemented since there is no cost-effective method of measuring torque; however, the torque fluctuations δT_n lead directly to crankshaft speed fluctuations that are measurable with a simple, inexpensive noncontacting sensor. We explain below the relationship between torque and crankshaft angular speed fluctuations. This relationship can be developed from a dynamic model for the power train as explained next.

For misfire detection purposes, an estimate of $T(\theta_e)$ can be obtained from a sliding mode observer (SMO) based upon a relatively straightforward system for measuring crankshaft angular speed (ω_e). This method and the analytic models for T_i and T_R and the theory of the SMO for torque estimation are given in an excellent paper by Rizzoni et al.[1] The model from which this SMO is built for an automatic transmission-equipped vehicle with unlocked torque converter is given below:

$$\begin{aligned} J\dot{\omega}_e &= T_i(\theta_e) - T_R(\theta_e) - R_e\omega_e - T_1(\theta_e) \\ &= T_n(\theta_e) \end{aligned} \tag{11.26}$$

where J = moment of inertia of engine rotating parts and T_l = load torque on the engine output.

For the purposes of illustration, we consider the special case in which the vehicle is traveling under steady-state conditions with an unlocked torque converter (see Chapter 6) for which T_l is a constant. This term can be neglected in the computation of torque fluctuations (as is done here).

Combining Eqs. (11.12)–(11.16) with Eq. (11.26) yields the following model for $\dot{\omega}_e$

$$\dot{\omega}_e = \frac{1}{J + M_{eq}R_c^2 f_T^2(\theta_e)}\left\{(p_c - p_o)A_pR_cf_T(\theta_e) - M_{eq}R_c^2 f_T(\theta_e)f_B(\theta_e)\omega_e^2 - R_e\omega_e\right\} \tag{11.27}$$

The equations for T_i and T_R have been given previously. Rewriting the above equation in state vector form with state vector x given by

$$x = [x_1 x_2]^T \qquad x_1 = \theta_e, x_2 = \omega_e$$

yields $\dot{x}_1 = x_2$

$$\dot{x}_2 = \frac{1}{J + M_{eq}R_c^2 f_T^2(x_1)}\left\{(p_c - p_o)A_pR_cf_T(x_1) - M_{eq}R_c^2 f_T(x_1)f_B(x_1)x_2^2 - R_ex_2\right\} \tag{11.28}$$

It is shown below that both x_1 and x_2 are measurable with inexpensive noncontacting sensors. Let the measurement of state vector x_1 be denoted y_1 and the measurement of x_2 be denoted y_2 The SMO for the estimate of x_2 (which is denoted $\hat{x}_2$) is given by

$$\dot{\hat{x}}_2 = \frac{1}{J + M_{eq}R_c^2 f_T^2(y_1)}\left\{-A_{\text{SMO}}\,\text{sgn}\left[f_T(y_1)(\hat{x}_2 - y_2)\right]A_pR_cf_T(y_1) - M_{eq}{}^2R_c^2 f_T(y_1)f_R(y_1)y_2^2 - R_ey_2\right\} \tag{11.29}$$

where A_{SMO} = SMO gain and sgn( ) = sign function of argument.

[1]Rizzoni G., Drakunov S., Wang Y.-Y. On line estimation of indicated torque in IC engines via sliding mode observer. In American Control Conference, 1995, pp. 2123–2127.

The SMO gain requirement is that it be larger than the maximum value that can occur for $(P_e - P_o)$:

$$A_{\text{SMO}} > \max(P_c - P_o) \tag{11.30}$$

The estimate of indicated torque is obtained as the output of the first-order filter given by

$$\begin{aligned} \tau\dot{v} + v &= -A_{\text{SMO}} \operatorname{sgn}[f_T(y_1)(\hat{x}_2 - y_2)] A_p R_c f_T(y_1) \\ \hat{T}_n &= v \end{aligned} \tag{11.31}$$

Using this SMO to estimate $\hat{T}_n$, it is possible to form the vector $\bar{T}$ from which misfire detection is possible as explained above since the extrema of $\hat{T}_n(\theta_e)$ closely approximates T_m and T'_m.

The measurement of crankshaft angular position and speed can readily be made using a noncontacting sensor such as that depicted in Fig. 11.13 and as explained in Chapter 5.

In Fig. 11.13, the ferromagnetic disk (with lugs) is attached to the crankshaft. However, for the accuracy in measurements of θ_e as required for SMO estimation of torque, there is a minimum number of lugs on the ferromagnetic disk. Experiments have shown that use of the starter ring gear that typically has 30–50 teeth is sufficient for these measurements.

For illustrative purposes, it is convenient to consider these measurements at a relatively slowly changing RPM. In this case, the crankshaft angular speed $\omega_e(t)$ is given by

$$\omega_e(t) = \Omega_e + \delta\omega_e(t) \tag{11.32}$$

where $\Omega_e = \dfrac{\pi \text{RPM}}{30}$ = short-term time average of ω_e and $\delta\omega_e(t)$ = variation in ω_e due to δT_n.

This angular speed is actually in the form of a frequency-modulated (FM) carrier frequency in which Ω_e acts as the carrier frequency and $\delta\omega_e(t)$ is the modulation. It should be noted that $\Omega_e \gg \max(\delta\omega_e)$ for essentially all normal engine operating conditions.

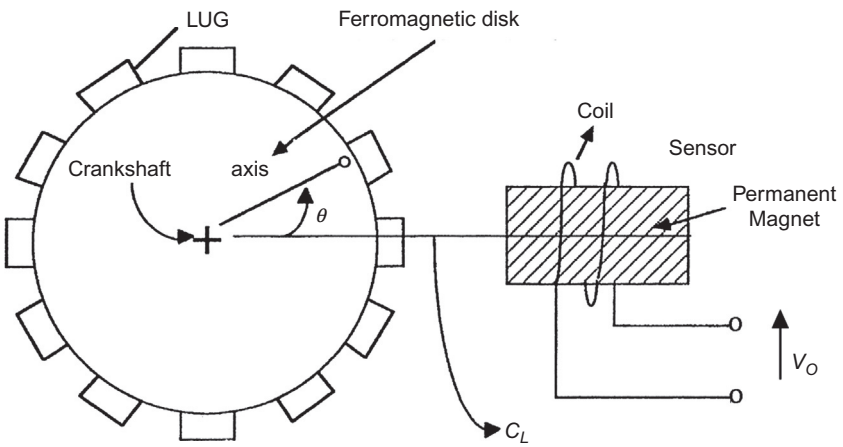


FIG. 11.13 Noncontacting crankshaft angular speed sensor.

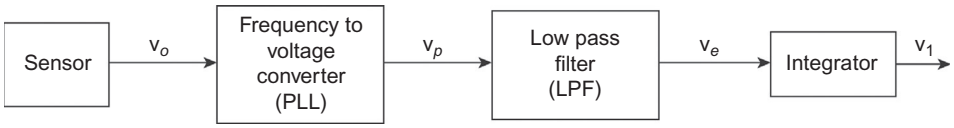


FIG. 11.14 Block diagram for ω_e measurement.

The crankshaft instantaneous angular position $\theta_e(t)$ is given by

$$\theta_e(t) = \theta_o + \int_o^t \omega_e(t')dt' \tag{11.33}$$

where $\theta_o = \theta_e(0) =$ phase reference.

The phase reference can be established relative to the engine cycle via a camshaft once/revolution noncontacting sensor (see Chapter 5).

The sensor output signal $v_0(t)$ is given by

$$v_0(t) = f[M_d\theta_e(t) + \psi] \tag{11.34}$$

where $M_d =$ number of lugs on disk and $\psi(t) =$ random process (error in the sensor output).

The function $f(\cdot)$ is the waveform associated with the sensor configuration. Fortunately, the electronic signal processing required to measure $\omega_e(t)$ can be obtained using either analog or digital electronic signal processing. Fig. 11.14 shows a block diagram for an analog signal processing.

The "frequency-to-voltage converter" is in effect an FM demodulator that can be implemented with a circuit known as a "phase-locked loop" (PLL). The PLL is an electronic closed-loop system. Chapter 2 presents a detailed explanation of the PLL and presents a model of frequency demodulation. It's output voltage $v_p(t)$ is given by

$$v_P = K_{\text{PLL}}(M_d\omega_e(t) + \dot{\psi})$$

The low-pass filter (LPF) passes the first term and suppresses that portion of the spectrum of $\dot{\psi}$ that lies outside the LPF pass bands, thereby yielding the measurement of ω_e needed for the SMO to compute $\hat{T}_n$. Appendix A explains LPF theory and presents design models for an analog LPF. For the present analysis, it is assumed that this portion of $\dot{\psi}$ is negligible. The crankshaft angular position can be obtained by integrating the LPF output voltage. Using the integrator circuit described in Chapter 2, the integrator output voltage v_I is given by

$$\begin{aligned} v_I &= \frac{1}{\tau_I}\int_0^t V_e(t')dt' \\ &= \frac{K_{\text{PLL}}M_d}{\tau_i}[\theta_e(t) + \theta_o] \end{aligned} \tag{11.35}$$

where $\tau_i =$ integrator time constant.

Of course, digital integration as explained previously (e.g., see Chapter 7) can also be used to obtain v_I. The phase origin for this measurement of $\theta_e(t)$ is established via the once/revolution camshaft sensor. The measurement of θ_e is required as part of the computation of the nonuniformity index $\bar{n}$.

In a contemporary implementation, the measurement of $\omega_e(t)$ is done in discrete time based upon successive samples of $v_o(t)$. As explained in Chapter 5, a sensor such as is depicted in Fig. 11.13 generates an output waveform that crosses zero whenever one of the lugs on the disk lies along the centerline (C_L) of the disk sensor axis. Let t_k be the time of the kth zero crossing of the sensor output voltage, and let δt_k be given by

$$\delta t_k = t_k - t_{k-1}$$

The kth sample of $\omega_e(t)$ is denoted $\omega_e(k)$ and is given by

$$\omega_e(k) = \frac{2\pi}{M_d \delta t_k} \tag{11.36}$$

If M_d is sufficiently large, the sequence $\{\omega_e(k)\}$ will be an unaliased sample of $\omega_e(t)$.

The instantaneous crankshaft angular position $\theta_e(k)$ within a 2 revolution cycle is given by Eq. (11.28)

$$\theta_e(k) = \theta_e(t_k) \qquad k = 1, 2, \cdots, 2M_d \tag{11.37}$$

This sampled crankshaft angular position is readily obtained by passing the sensor through a zero crossing detector (ZCD) (see Chapter 2) and counting the output pulses using a binary counter (see Chapter 2 for an explanation of a counter) as explained in Chapter 6. The counter should be reset by a signal from the once/revolution camshaft sensor. This signal is also sent to a ZCD and then to the binary counter reset input. This configuration will automatically count zero crossings of the crankshaft sensor of Fig. 11.13 modulo $2M_d$ for each engine cycle.

Using the instrumentation above for measuring ω_e and θ_e provides the necessary data for a calculation of $\hat{T}_n$ using the SMO and the nonuniformity index $\bar{n}$. The misfire detection proceeds using the estimate of $\bar{T}$ according to the procedure explained earlier.

The above hypothetical method of misfire detection has been shown to reliably detect misfires both in a laboratory environment and in actual road tests. For a test vehicle equipped with an automatic transmission total errors of $<1\%$ have been achieved for the exemplary misfire detection in actual road tests. Although intellectual property considerations preclude discussing the actual misfire detection methods used by any automotive manufacturer, many of the components of the hypothetical system are to be found in some of them.

EXPERT SYSTEMS IN AUTOMOTIVE DIAGNOSIS

An expert system is a computer program that employs human knowledge to solve problems normally requiring human expertise. The theory of expert systems is part of the general area of computer science known as artificial intelligence (AI). The major benefit of expert system technology is the consistent, uniform, and efficient application of the decision criteria or problem-solving strategies. We consider next a hypothetical expert system devoted to automotive diagnosis.

The diagnosis of electronic engine control systems by an expert system proceeds by following a set of rules that embody steps similar to the diagnostic charts in the shop manual. The diagnostic system can receive fault codes from the onboard diagnostic. The system processes these codes logically under program control in accordance with the set of internally stored rules. However, as explained above, not

all faults are detected by the onboard diagnostic system. Testing of various systems and components by the service technician as directed by the expert system aids the diagnosis of problems. The hypothetical expert system-based diagnostic procedure also is designed to receive inputs from the service technician based on such tests. The end result of the computer-aided diagnosis is an assessment of the problem and recommended repair procedures. The use of an expert system for diagnosis has the potential to improve the efficiency of the diagnostic process and can thereby reduce maintenance time and costs.

The development of an expert system requires a computer specialist who is known in AI parlance as a *knowledge engineer*. The knowledge engineer must acquire the requisite knowledge and expertise for the expert system by interviewing the recognized experts in the field. In the case of automotive electronic engine control systems, the experts include the design engineers, the test engineers and technicians, involved in the development of the control system. In addition, expertise is developed by the service technicians who routinely repair the system in the field. The expertise of this latter group can be incorporated as evolutionary improvements in the expert system. The various stages of knowledge acquisition (obtained from the experts) are outlined in Fig. 11.15.

It can be seen from this illustration that several iterations are required to complete the knowledge acquisition. Thus, the process of interviewing experts is a continuing process.

Not to be overlooked in the development of an expert system is the personal relationship between the experts and the knowledge engineer. The experts must be fully willing to cooperate and to explain their expertise to the knowledge engineer if a successful expert system is to be developed. The personalities of the knowledge engineer and experts can become a factor in the development of an expert system.

Fig. 11.16 represents the environment in which an expert system evolves. Of course, a digital computer of sufficient capacity is required for the development work. A summary of expert system development tools that have been used in the past and that are potentially applicable for a mainframe computer is presented in Table 11.3.

It is common practice to think of an expert system as having two major portions. The portion of the expert system in which the logical operations are performed is known as the *inference engine*. The various relationships and basic knowledge are known as the *knowledge base*.

The general diagnostic field to which an expert system is applicable is one in which the procedures used by the recognized experts can be expressed in a set of rules or logical relationships. The automotive diagnosis area is clearly such a field. The diagnostic charts that outline repair procedures (as outlined earlier in this chapter) represent good examples of such rules.

To clarify some of the ideas embodied in an expert system, consider the following example of the diagnosis of an automotive repair problem. This particular problem involves failure of the car engine to start. It is presumed in this example that the range of defects is very limited. Although this example is not necessarily commonly encountered, it does illustrate some of the principles involved in an expert system.

The fundamental concept underlying this example is the idea of condition-action pairs that are in the form of IF-THEN rules. These rules embody knowledge that is presumed to have come from human experts (e.g., experienced service technicians or automotive engineers).

A typical expert system formulates expertise in IF-THEN rules. The expert system of this example consists of three components:

(1) A rule base of IF-THEN rules
(2) A database of facts
(3) A controlling mechanism

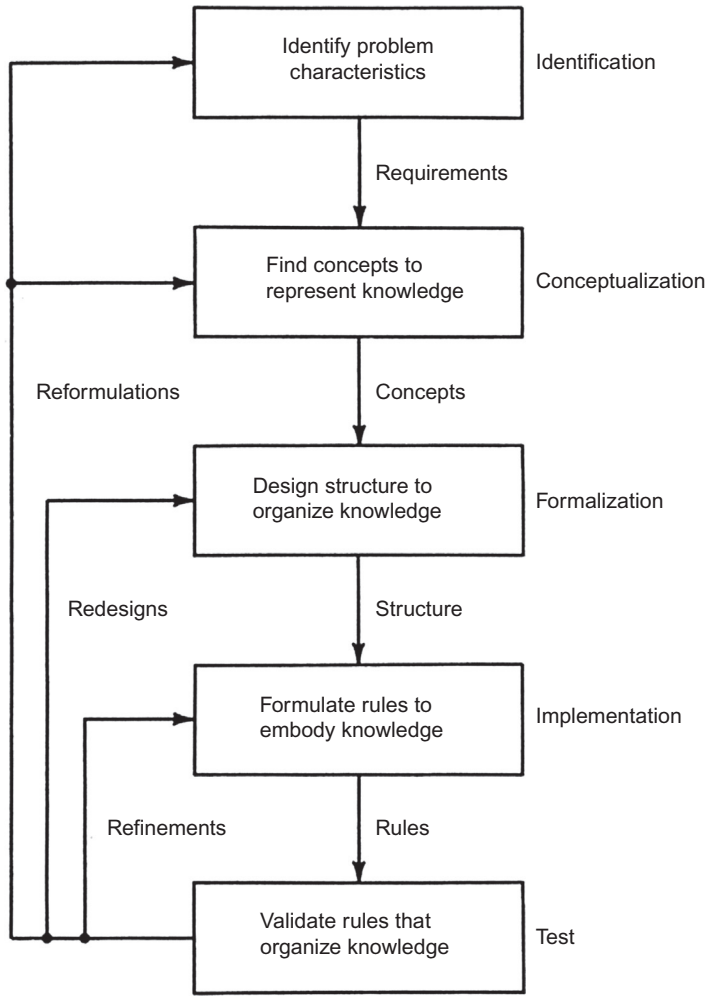


FIG. 11.15 Expert system development procedure.

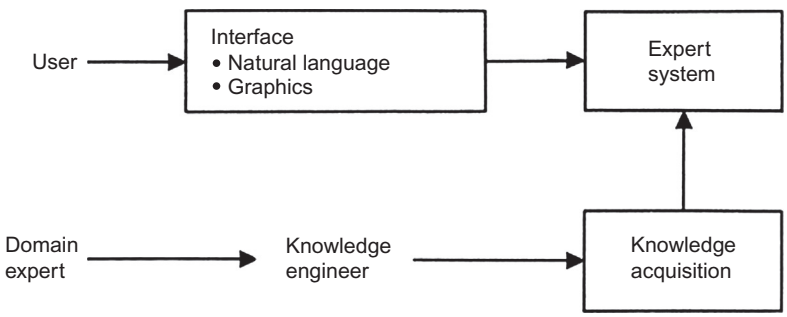


FIG. 11.16 Environment of an expert system.

Table 11.3 Expert System Developing Tools for Mainframes

Name	Company	Machine
Ops5	Carnegie Mellon University	VAX
S.1	Teknowledge	VAX Xerox 1198
Loops	Xerox 1108	
Kee	Intelligenetics	Xerox 1198
Art	Inference	Symbolics

Each rule of the rule base is of the form of "if condition A is true, then action B should be taken or performed." The IF portion contains conditions that must be satisfied if the rule is to be applicable. The THEN portion states the action to be performed whenever the rule is activated (fired).

The database contains all the facts and information that are known to be true about the problem being diagnosed. The rules from the rule base are compared with the knowledge base to ascertain which are the applicable rules. When a rule is fired, its actions normally modify the facts within the database.

The controlling mechanism of this expert system determines which actions are to be taken and when they are to be performed. The operation follows four basic steps:

(1) Compare the rules with the database to determine which rules have the IF portion satisfied and can be executed. This group is known as the *conflict set* in AI parlance.
(2) If the conflict set contains more than one rule, resolve the conflict by selecting the highest priority rule. If there are no rules in the conflict set, stop the procedure.
(3) Execute the selected rule by performing the actions specified in the THEN portion, and then, modify the database as required.
(4) Return to step 1 and repeat the process until there are no rules in the conflict set.

In the present simplified example, it is presumed that the rule base for diagnosing a problem starting a car is as given in Fig. 11.17.

Rules R2 through R7 draw conclusions about the suspected problem, and rule R1 identifies problem areas that should be investigated. It is implicitly assumed that the actions specified in the THEN portion include "add this fact to the database." In addition, some of the specified actions have an associated fractional number. These values represent the confidence of the expert who is responsible for the rule that the given action is true for the specified condition.

Further suppose that the facts known to be true are as shown in Fig. 11.18.

The controlling mechanism follows step 1 and discovers that only R1 is in the conflict set. This rule is executed, deriving these additional facts in performing steps 2 and 3:

- Suspect there is no spark.
- Suspect too much fuel is reaching the engine.

At step 4, the system returns to step 1 and learns that the conflict set includes R1, R4, and R6. Since R1 has been executed, it is dropped from the conflict set. In this simplified example, assume that the conflict is resolved by selecting the lowest numbered rule (i.e., R4 in this case). Rule R4 yields the

R1: IF starter turns engine but it fails to start
THEN suspect no fuel reaches engine OR
suspect there is no spark OR
suspect too much fuel is reaching engine

R2: IF suspect no fuel reaches engine AND
gas gauge works AND
gas gauge is on empty
THEN gas tank is empty (0.95)

R3: IF suspect no fuel reaches engine AND
gas gauge is not on empty AND
temperature is less than 32 °F
THEN fuel line is frozen (0.75)

R4: IF suspect no fuel reaches engine AND
can smell gas
THEN break in fuel line (0.65)

R5: IF suspect no fuel reaches engine AND
gas gauge is not on empty AND
do not smell gas
THEN water in gas tank (0.5) OR
gas gauge broken (0.6)

R6: IF suspect too much fuel is reaching engine AND
can smell gas
THEN mixture is too rich (0.7)

R7: IF suspect there is no spark AND
gas gauge not on empty AND
(weather is damp OR weather is rainy)
THEN spark plug wires are wet (0.6)

FIG. 11.17 Simple illustrative automobile diagnostic rule base.

Gas gauge works
Starter turns engine but it fails to start
Gas gauge is not on empty
Can smell gas

FIG. 11.18 Starting database of known facts.

additional facts after completing steps 2 and 3 that there is a break in the fuel line (0.65). The value 0.65 refers to the confidence level of this conclusion.

The procedure is repeated with the resulting conflict set R6. After executing R6, the system returns to step 1, and finding no applicable rules, it stops. The final fact set is shown in Fig. 11.19. Note that this diagnostic procedure has found two potential diagnoses: a break in the fuel line (confidence level 0.65) and mixture too rich (confidence level 0.70).

Gas gauge works
Starter turns engine but it fails to start
Gas gauge is not on empty
Can smell gas
Suspect no fuel reaching engine
Suspect there is no spark
Suspect too much fuel is reaching engine
Break in fuel line (0.65)
Mixture too rich (0.7)

FIG. 11.19 Final resulting database of known facts.

The previous example is intended merely to illustrate the application of AI to automotive diagnosis and repair. To perform diagnosis on a specific car using an expert system, the service technician identifies all the relevant features to the service technician's terminal including, of course, the engine type. After connecting the data link from the onboard diagnostic system to the terminal, the diagnosis can begin. The terminal can ask the service technician to perform specific tasks that are required to complete the diagnosis, for example, starting or stopping the engine.

The expert system is an interactive program and, as such, has many interesting features. For example, when the expert system requests that the service technician perform some specific task, he/she can ask the expert system why he or she should do this, or why the system asked the question. The expert system then explains the motivation for the task, much the way a human expert would do if he or she were guiding the service technician. An expert system is frequently formulated on rules of thumb that have been acquired through years of experience by human experts. It often benefits the service technician in his or her task to have requests for tasks explained in terms of both these rules and the experience base that has led to the development of the expert system.

The general science of expert systems is so broad that it cannot be covered in this book. The interested reader can contact any good engineering library for further material in this exciting area. In addition, the SAE has many publications covering the application of expert systems to automotive diagnosis.

From time to time, automotive maintenance problems will occur that are outside the scope of the expertise incorporated in the expert system. In these cases, an automotive diagnostic system needs to be supplemented by direct contact of the service technician with human experts. Automobile manufacturers all have technical assistance available to service technicians via internal connections or email.

Vehicle off-board diagnostic systems (whether they are expert systems or not) continue to be developed and refined as experience is gained with the various systems, as the diagnostic database expands, and as additional software is written. The evolution of such diagnostic systems may be heading in the direction of fully automated, rapid, and efficient diagnoses of problems in cars equipped with modern digital control systems.

CHAPTER

AUTONOMOUS VEHICLES 12

CHAPTER OUTLINE

Automatic Parallel Parking System 575
Autonomous Vehicle Block Diagram 581

Autonomous vehicles, also known as "self-driving vehicles," represent an extreme application of electronics to vehicles. Such vehicles are controlled during motion by a computer along with various electronic subsystems and components rather than by a human driver. At the time of this writing, they are in a research and development stage. There are multiple levels of autonomy as classified by government agencies (e.g., USDOT) and by the Society of Automotive Engineers (SAE). The primary inputs to a vehicle by a human driver include steering, braking, throttle, and transmission mode select. However, to drive a vehicle, the driver must continuously monitor the environment visually and react to the conditions. This means maintaining the vehicle in an appropriate lane on a road for a given trip and reacting correctly to all road signs and signals along a given trip. It is vitally important that he/she react to changes in the pathway that require decision-making such as traffic, pedestrians, and any objects in the path. Typically, the driver must decide whether it is necessary to stop or possibly steer around other objects in the pathway. In addition, the driver should also attempt to make a prediction about an object or a pedestrian that is moving such that in the short time ahead, it is probable that the object/pedestrian will require action by the driver. For an autonomous vehicle to operate safely, it must have the same type of decision-making capability described above (and other decisions not discussed).

In addition to the humanlike decision capability, an autonomous vehicle requires a number of subsystems that are automatic. Some of these subsystems have been implemented in commercially available vehicles for a relatively long time. Many of these are described elsewhere in this book. For example, vehicle emission control and antilock braking are automated subsystems that require no driver control and have been around for decades.

More recent than these examples are automatic braking, enhanced stability, traction control and automatic parking capabilities that will be a part of an autonomous vehicle. These examples are explained in the vehicle motion chapter.

However, before discussing the various subsystems employed in autonomous vehicles, it is perhaps helpful to consider classification of such vehicles in accordance with the division between automatic

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-12-810434-7.00012-0
Copyright © 2017 Elsevier Inc. All rights reserved.

control and driver intervention. One such classification has been suggested for the United States by the National Highway Traffic Safety Administration (NHTSA) and is as follows:

- Level 0—the driver completely controls the vehicle at all times.
- Level 1—individual vehicle controls are automated, such as electronic stability control or automatic braking.
- Level 2—at least two controls can be automated in unison, such as adaptive cruise control (ACC) in combination with lane keeping.
- Level 3—the driver can fully cede control of all safety-critical functions in certain conditions. The car senses when conditions require the driver to retake control and provides a "sufficiently comfortable transition time" for the driver to do so.
- Level 4—the vehicle performs all safety-critical functions for the entire trip, with the driver not expected to control the vehicle at any time. As this vehicle would control all functions from start to stop, including all parking functions, it could include unoccupied cars.

The SAE is an important organization that has consistently promoted new automotive technology. It also forms groups of engineers, technicians, and legal experts for creating and publishing standards and recommended practices. This organization has presented a classification system for autonomous vehicles that has six levels, depending on the level of human driver intervention and environmental observations and supervisory decision-making. These six levels of classification are the following:

- Level 0—automated system has no vehicle control but may issue warnings.
- Level 1—driver must be ready to take control at anytime. Automated system may include features such as ACC, parking assistance with automated steering, and lane keeping assistance (LKA) type II in any combination.
- Level 2—the driver is obliged to detect objects and events and respond if the automated system fails to respond properly. The automated system executes accelerating, braking, and steering. The automated system can deactivate immediately on takeover by the driver.
- Lever 3—within known, limited environments (such as freeways), the driver can safely turn their attention away from driving tasks.
- Level 4—the automated system can control the vehicle in all but a few environments such as severe weather. The driver must enable the automated system only when it is safe to do so. When enabled, driver attention is not required.
- Level 5—other than setting the destination and starting the system, no human intervention is required. The automatic system can drive to any location where it is legal to drive.

It is important to note that essentially all of the electronic subsystem hardware and control software necessary to implement an autonomous vehicle at any of the above levels exists as of the time of this writing. The major development required for the higher levels listed above is the algorithms required to process sensor data, make decisions, and regulate the various subsystems. However, before proceeding with a discussion of this software aspect of autonomous vehicles, it is necessary to explain the one electronic subsystem that has not been discussed elsewhere in this book. That subsystem is automatic steering. Such a system exists in certain production vehicles at the time of this writing and is an automatic parallel parking system (APPS), which is presented next. The main feature of APPS as it relates to autonomous vehicles is computer-controlled automatic steering of the vehicle.

AUTOMATIC PARALLEL PARKING SYSTEM

Although it is not an SAE level 5 autonomous vehicle, any vehicle equipped with APPS has certain components and features of level 5 autonomous vehicle. These include electronically controlled steering and vehicle surrounding environment sensing. In operation, an APPS requires the driver to position the car ahead of the intended parking space with the rear bumper slightly ahead of the outer rear wheel of the car that is parked immediately in front of the space. Fig. 12.1 depicts a simplified APPS vehicle configuration.

The vehicle equipped with the APPS has a number of sensors S_n of Fig. 12.1 that can provide data to a computer that can assess the parking configuration and can also provide data from which warnings about obstacles in the path of the parking space can be given to the driver. During the APPS operation, vehicle steering (i.e., deflection of angle δ_F of Fig. 12.1) is accomplished by an electromechanical actuator denoted A_F in Fig. 12.1. The input to A_F is an electrical signal denoted e_F in Fig. 12.1. This signal is generated by an electronic control unit (ECU) control system (APPS-ECU), as explained later in this section. When in this approximate position, the driver can activate the system by first placing the car in reverse and then selecting an APPS button (typically a button displayed on the flat panel display/touch screen).

Once activated, the computer uses the sensor data to compute an input to the electronic steering that will cause the vehicle to follow a path that will place it in the parking spot. This maneuver is essentially a lane change maneuver with the vehicle moving in reverse. Contemporary vehicles (often series hybrids excepted) are virtually all equipped with automatic transmissions. With the engine running and the transmission in reverse, the torque coupled to the drive axles causes the vehicle to move at a relatively low speed. In some APPS-equipped vehicles, the driver must operate the braked manually to stop the car before it strikes the car at the rear of the space. However, any vehicle equipped with an automatic braking or a stability enhancement system (see Chapter 10), brakes can be automatically applied when the parking maneuver is at the intended (computed) stopping point.

The APPS-ECU consists of several functional subsystems implemented typically by calculation in a microcontroller. One portion of the ECU contains software that performs an analysis of the parking

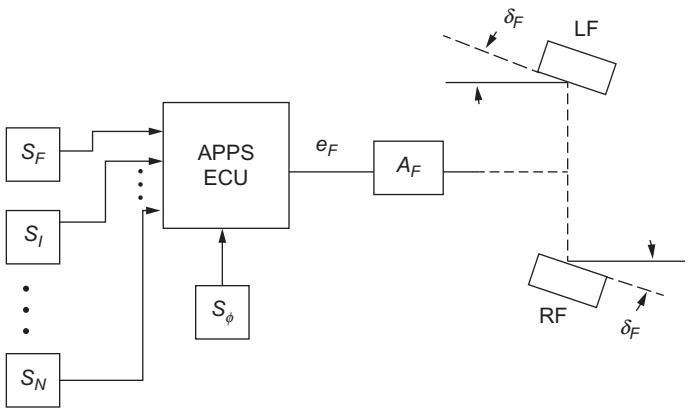


FIG. 12.1 Automatic parallel parking lock diagram.

environment based on data from the environmental surveillance sensors (see Chapters 5 and 10) and on algorithms from which the path to be followed during APPS operation are computed. The output of this section of the ECU is a command steering angle δ_{Fc} that causes the vehicle to follow the desired path to the final parking position. Another portion of the APPS-ECU is the control portion as explained later in this section of the chapter. This exemplary APPS control is a closed-loop control system with a sensor S_F for measuring the actual front-wheel steering angle δ_F. The details of the control are explained later.

The actual steering input for any vehicle is manufacturer-/model-specific. However, as an illustration of a representative steering input, we consider an example that can be called a steering doublet (borrowing a term from aircraft control input). This so-called doublet steering involves an initial steering input δ_F that causes the vehicle to move in the intended direction. Midway through the maneuver, the opposite steering angle is commanded. A model for this parallel parking maneuver with the steering doublet for an example vehicle is presented next.

In Fig. 12.1, the sensors are denoted S_F and S_n with $n = 1,2,\cdots,N$. The sensor labeled S_F measures the front-wheel steering angle δ_F. The block labeled A_F is an actuator that mechanically deflects the front wheels (right front (RF) and left front (LF)) in response to the electrical output of the computer e_F. There are numerous sensor configurations that are capable of generating an analog or digital output signal that is proportional to δ_F as described in Chapter 5 on Sensors and Actuators. For example, a simple rotary position sensor attached to the steering wheel axis provides sufficient information to calculate the front wheel δ_F. An exemplary actuator A_F is discussed in the electronic steering section of Chapter 7 on motion control. All sensors and the actuator are connected to the digital control system of APPS-ECU as depicted in Fig. 12.1. The algorithms for controlling the steering angle δ_F for the APPS maneuver are discussed later in this section.

The remaining sensors S_n provide the data required to measure and assess the parking space. Such sensors can either be ultrasonic, video camera, microwave radar, or lidar-type devices. Radar-type sensors are discussed in the chapter on Vehicle Safety and operate by measuring the roundtrip time from the radar antenna to reflecting objects. Ultrasonic sensors operate in the same general way of measuring distances via the roundtrip time from an ultrasonic source to a reflecting object. These sensors in combination with a yaw angle sensor S_ϕ can yield vehicle position in the parking coordinates x_p, y_p.

As explained above in the exemplary APPS, the computer calculates the steering doublet. A closed-loop control can be implemented in the computer to assure that the actual δ_F is controlled to be the computed steering doublet that is now denoted δ_{Fc}. A simplified block diagram depicts the functional closed-loop steering as given in Fig. 12.2.

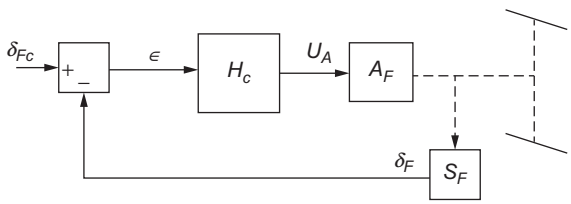


FIG. 12.2 Block diagram of closed-loop control for exemplary APPS.

The chapter on vehicle motion control includes a section on electronic steering assist. In this section, it is shown that vehicle motion with steering input at a constant (relatively low) speed u_o can be modeled with a two-dimensional state-vector equation with the state-vector x given by

$$x = [v, r]^T$$

where r = yaw rate about the vehicle z-axis through the CG and v = vehicle velocity component along the vehicle transverse coordinates.

The state-variable equation has the form

$$\dot{x} = Ax + Bu$$

where the matrices A and B are given in Eq. (7.110) of Chapter 7 and where $u = \delta_F$ = front-wheel steering input.

The parameters in these matrices are defined in Chapter 7. The vehicle motion during parking is given by the coordinates $x_p y_p$ where x_p is parallel to the curb and y_p is orthogonal to the curve.

These coordinate positions are given by the time integrals of the velocity components that are denoted $\dot{x}_p\, \dot{y}_p$:

$$x_p(t) = x_{po} + \int_d^t \dot{x}_p dt$$

$$y_p(t) = y_{po} + \int_o^t \dot{y}_p dt$$

where $x_{po} y_{po}$ is the starting point.

The components of velocity $\dot{x}_p\, \dot{y}_p$ are given by

$$\dot{x}_p = u_o \cos\phi$$

$$\dot{y}_p = u_o \sin\phi$$

where ϕ = instantaneous vehicle heading in $x_p\, y_p$ coordinates and u_o = vehicle speed.

The heading angle ϕ is found by integrating the yaw rate r:

$$\phi = \int_o^t r(t) dt$$

where the origin of ϕ (i.e., $\phi = 0$) is along the x_p axis. For reverse parallel parking, the vehicle speed $u_o < 0$.

Since the dynamics are relatively slow, the control block (H_c) could be a simple proportional control algorithm such that the electrical signal to the actuator (U_A) is given by

$$\begin{aligned} U_A &= K_p \in \\ &= K_p(\delta_{FC} - \delta_F) \end{aligned}$$

However, in order to avoid any bias/drift (possible in proportional control), the control algorithm could be proportional-integral (PI as explained in Appendix A) for which U_A is given by

$$U_A = K_p \in + K_I \int \in dt$$

where $\in = \delta_{FC} - \delta_F$.

In any event, the sensors S_n and S_ϕ provide sufficient data for the computer to monitor the parking system performance in terms of the conformity of measured to computed vehicle position during the parking maneuver. Once the surveillance sensors have provided sufficient data to model the parking space in the present example control, the ECU calculates the desired $\delta_{FC}(x_p, y_p)$ and uses this as an input to the control.

The transfer function for the control block $H_c(s)$ for an assumed PI control law is given by

$$\begin{aligned} H_c(s) &= \frac{U_A(s)}{\in(s)} \\ &= \frac{K_p s + K_I}{s} \end{aligned}$$

A representative model for the actuator is a first-order differential equation as given by

$$\dot{\delta}_F + \frac{\delta_F}{\tau_A} = U_A$$

where τ_A = actuator time constant.

For APPS applications, τ_A can be of the order of 1 sec.

The transfer function $H_A(s)$ for the actuator is given by

$$\begin{aligned} H_A(s) &= \frac{\delta_F(s)}{U_A(s)} \\ &= \frac{1}{s + \dfrac{1}{\tau_A}} \end{aligned}$$

The forward-path transfer function $H_f(s)$ is given by

$$\begin{aligned} H_f(s) &= \frac{\delta_F(s)}{\in(s)} \\ &= H_c(s) H_A(s) H_{SF}(s) \end{aligned} \tag{12.1}$$

For convenience, the sensor transfer function $H_{SF}(s)$ is taken to be unity:

$$H_{SF}(s) = 1$$

With this assumption, the forward transfer function is given by

$$H_f(s) = \left(K_p + \frac{K_I}{s}\right)\left(\frac{1}{s + \dfrac{1}{\tau_A}}\right)$$

As explained in Appendix A, the closed-loop transfer function for the APPS is given by

$$\begin{aligned} H_{C\ell}(s) &= \frac{\delta_F(s)}{\delta_{FC}(s)} \\ &= \frac{H_f(s)}{I + H_f(s)} \end{aligned} \tag{12.2}$$

In the illustrative APPS, the ECU is a discrete-time system. As explained in Appendix B, the system closed-loop transfer function must be found using z-transforms rather than a Laplace transforms yielding a closed-loop transfer function $H_{C\ell}(z)$ which is given by

$$H_{C\ell}(z) = \frac{H_f(z)}{1 + H_f(z)}$$

For illustrative purposes, it is assumed that the ECU output is digital and requires a D/A converter with a zero-order hold (ZOH) to generate the analog signal $\bar{U}_A$ that drives the actuator. As explained in Appendix B, the z-transform of $H_f(s)$ with the D/A and ZOH is given by

$$H_f(z) = \left(1 - z^{-1}\right) \mathcal{Z}\left[\frac{H_f(s)}{s}\right] \tag{12.3}$$

$$H_f(z) = \left(1 - z^{-1}\right) \mathcal{Z}\left[\frac{K_p}{s\left(s + \dfrac{1}{\tau_A}\right)} + \frac{K_I}{s^2\left(s + \dfrac{1}{\tau_A}\right)}\right] \tag{12.4}$$

Using partial fraction expansions and z-transforms of the resulting terms from the table of z-transforms near the end of Appendix B, it can be shown that

$$H_f(z) = \frac{\left[K_p\tau_A(1 - z_A) + K_I\tau_A^2\left(\dfrac{T}{\tau_A} + (z_A - 1)\right)\right]z - K_p\tau_A(1 - z_A) + K_I\tau_A^2\left(1 - z_A\left(1 + \dfrac{T}{\tau_A}\right)\right)}{z^2 - (1 + z_a)z + z_A} \tag{12.5}$$

where T = sample period.

$$z_A = e^{-T/\tau_A}$$

The closed-loop transfer function computed from Eqs. (12.1), (12.2) can be used to evaluate the performance and stability of the APPS.

The implementation of the APPS control once K_p and K_I have been determined to meet system requirements involve the following expression to calculate the sequence of control outputs $\{U_n\}$ from which the analog signal $\bar{u}$ is generated is given by

$$U_n = K_p \in_n + K_I T \sum_{k=1}^{K} \frac{\in_{n-k} + \in_{n-(k+1)}}{2}$$

where T = sample period, K = number of error samples for calculating the integral part of the control, and where $\in_k = \delta_{FC}(k) - \delta_F(k)$.

The algorithms for implementing this control law are straightforward (particularly using AUTOSAR).

As an illustration of the parallel parking maneuver of a vehicle with an APPS, a simulation was run. For this simulation, the coefficient in the A and B matrices are those given in the set of values listed in 122 of Chapter 7 on vehicle motion control. These are the same set of coefficients used in the 4WS lane change simulation. However, in this case, $u_o = -3$ ft/s. The steering doublet and the track of the vehicle CG and the point on the side of the vehicle along a line orthogonal to its longitudinal axis through the

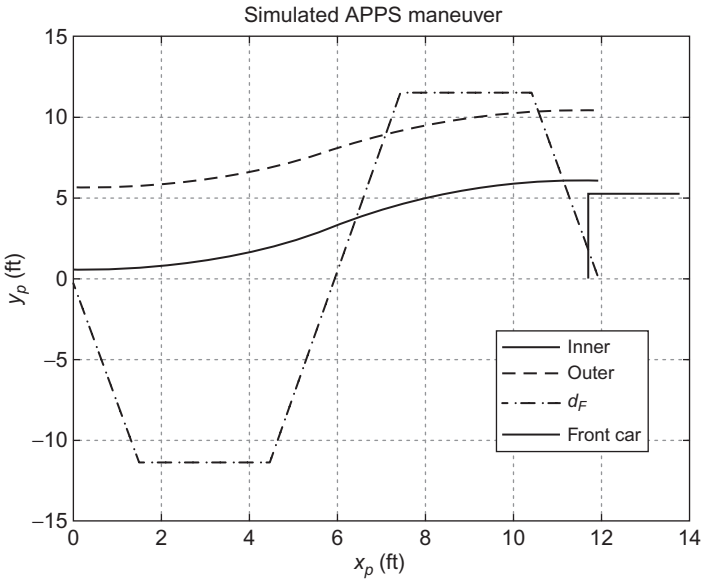


FIG. 12.3 Simulation result of example APPS maneuver.

CG are depicted in Fig. 12.3. This figure depicts the motion of the vehicle (x_{pr} y_{pr}) during the simulated APPS maneuver. The paths of inner and outer wheels are plotted. The coordinates of the inner and outer rear wheels are computed by standard coordinate transformation as explained in Appendix E.

The vehicle begins at the point $x_p = 12$ ft and moves to the point $x_p = 0$. The rearmost point of the car that is at the front of the space is at $x_p = 11.5$ ft, $y_p = 5.5$ ft so the parking car can be seen to avoid contact with the forward car. The most forward point of the car at the rear of the space is at $x_p = -3$ ft. After stopping at $x_p = 0$, the rearmost part of the parking car is at $x_p = -2.5$ ft. It should be noted that the simulation depicted in Fig. 12.3 is not optimized for parking in a "tight spot." It is possible to park in a shorter space (horizontally) than depicted by changing certain parameters (e.g., u_o and maximum amplitude of δ_F). The control of the vehicle motion via δ_F as the regulated variable depends on known cornering stiffness parameters that, in general, are not always known. Furthermore, an alternate control scheme can be based on control of the angle ϕ. For such a control strategy, the vehicle motion is not dependent on cornering stiffness parameters. This figure is only intended to illustrate the type of automatic steering found in autonomous vehicle, so the implied assumption of known C_F and C_R is not a fundamental limitation on the illustrative example model and simulation of APPS.

In this illustrative example APPS, the steering doublet begins at $\delta_F = 0$. It increases linearly to a maximum value $\delta_{F\max}$ and remains there for a period and then changes linearly to $-\delta_{F\max}$ and remains at this value for the same duration as at the positive maximum. It then increases to $\delta_F = 0$ at point $x_p = 0$ where the vehicle stops. A vehicle control input that has the type of symmetry used in this simulation is termed a doublet.

AUTONOMOUS VEHICLE BLOCK DIAGRAM

The above discussion of APPS in combination with other electronic subsystems discussed elsewhere in the book permits the presentation of a block diagram of a representative autonomous vehicle such as is given in Fig. 12.4.

The block in Fig. 12.4 labeled "vision sensors" is a generic representation for many different possible configurations of optical sensors. The fundamental role of these sensors is to give 360 degrees field of view surrounding the vehicle. For example, this sensor subsystem can consist of numerous solid-state cameras. Each consists of an assembly containing one or more focusing lenses and a planar array of photo sensors as described in Chapter 10 on vehicle safety and occupant protection. The number of cameras and the image resolution (i.e., number of pixels) is system-specific, but it must have the resolution required for the image detection system implemented in the computer to identify the objects it is intended to detect (e.g., lane markings). The operation of the cameras and pattern recognition algorithms are explained in the sections of Chapter 10 under the subjects of automatic braking and blind spot detection. The configuration and theory of operation of certain classes of digital cameras having potential surveillance application are given in Chapter 5.

In addition to image detection with the camera signals, the range to the various objects is measured using the radar systems and/or the lidar (optical equivalent to radar) sensors. The processing of these sensor signals is done in the computer. In the exemplary system, the radar systems include a relatively short-range (within a few meters of the car exterior surfaces) and long-range radar. In the latter radar system, the range is selected such that there is sufficient time for processing the range and direction to an object at highway speeds for decisions to be made and any required action (e.g., steering or braking) to be accomplished. The long-range radar is primarily used for forward and rearward object detection. The forward viewing systems are employed in actions such as collision avoidance. The rearward directed camera data and radar/lidar measurements can provide warnings for a rear collision due to an approaching vehicle.

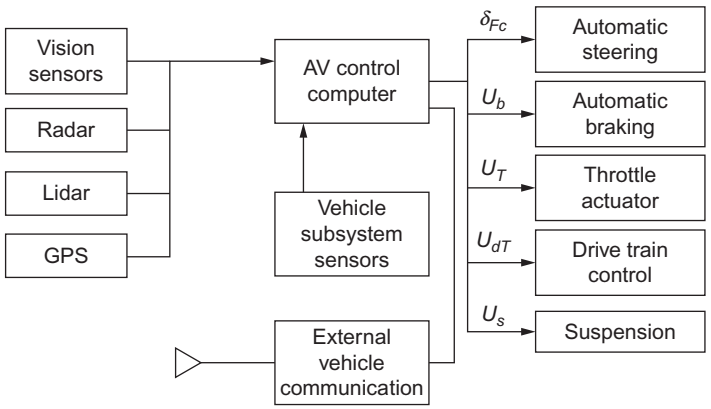


FIG. 12.4 Representative autonomous vehicle block diagram.

The short-range radar is mostly directed to the vehicle sides. This has already been described in the section on blind spot detection. However, it can also be employed in APPS systems. In addition, it is useful for pedestrian detection.

The azimuthal resolution of any radar system is improved with increasing aperture of the radar antenna. For a multiantenna system that is typical of automotive radar, the effective aperture is given by the separation of the outermost antennas. In addition, the distance measuring resolution is determined by "sharpness" of the transmitted radar pulse and by the pulse duration. The pulse "sharpness" is equivalent to the pulse rise time. As explained in Appendix A, the radar system bandwidth is essentially inversely proportional to the pulse rise time and its duration. Thus, the resolution in distance to object measurement is an increasing function of the radar system bandwidth. Depending on the power level of the radar system, this bandwidth may well be set by government regulations (e.g., by FCC in the United States).

Another important sensor system in an autonomous vehicle is the GPS position measuring system. The theory of GPS is explained in Chapter 9 on vehicular communication. At the highest levels of autonomy, the autonomous vehicle will follow a preprogrammed route that is stored in the computer before trip begins. The route will be along positions on electronic maps that are also stored in computer memory. These electronic maps are explained in the section of the chapter on instrumentation devoted to vehicular navigation. The vehicle speed is limited to the local speed limit but is chosen by computer-based decision on the safe speed that may be well less than the speed limit depending on the driving environment (e.g., traffic, pedestrians, and unprogrammed objects).

If there were no other vehicles, objects, and pedestrians in the lane of the road being traveled, then, in principle, the autonomous vehicle could travel at the legal speed limit along the entire route slowing for curves that are marked for a recommended speed, stopping for traffic lights that are red or physical stop signs. However, such a trip is highly unlikely. Rather, the autonomous vehicle will respond to a continuously changing environment and avoiding collision with any other vehicle or object.

A block diagram of an illustrative control system for an automatic steering system is depicted in Fig. 12.5. In this figure, the control input ϕ_c to the system is the set point or command input to the control system. For the autonomous vehicle, this control system regulates δ_F such that the instantaneous vehicle heading ϕ tracks the command input ϕ_c. The theory of the operation of this exemplary automatic steering control system is explained, quantitatively, later in this chapter.

The steering system requires an electromechanical actuator to turn the front wheels. There are many potential actuators including some that are based on classical power steering boost actuators. If these

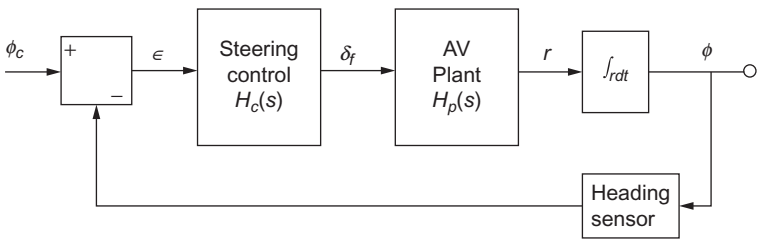


FIG. 12.5 Simplified autonomous vehicle steering control block diagram.

are employed, they must provide sufficient steering torque to turn the wheels without driver input. One important constant on such an actuator is that it must have sufficient torque to maintain δ_F at the required angle for a constant rate turn. This torque must overcome the normal steering restoring torque (called aligning torque) that tends to return δ_F to 0. This aligning torque is produced by the vehicle weight on the front wheels in combination with the front tire pneumatic trail and the orientation of the axis about which the steering wheels rotate. This alignment is the camber and castor angles of this axis.

This constraint prohibits the use of ordinary electric motors such as D-C or induction/synchronous motors. Such motors only produce torque while the armature/or equivalent structure is rotating. The chapter on sensors and actuators explains the operation and develops analytic models for various motors.

On the other hand, a stepper motor (also explained in the Sensors and Actuators chapter) has the capability of being controlled digitally to a given angular position. A steeper motor connected to an appropriate valve (e.g., rotary valve) can be used to regulate the pressure in a pneumatic or hydraulic cylinder from an accumulator. The combination of these components can form a steering actuator in which the commanded angular position of the stepper motor (θ_s) regulates the force from the hydraulic/pneumatic cylinder through mechanical linkage to control steering angle $\delta_F(\theta_s)$. There are other components and configurations capable of forming an electromechanical steering actuator. Other actuators and subsystems that are a required part of any autonomous vehicle include throttle actuator and cruise control subsystem, antilock brake systems with appropriate actuators, and drive train control systems are discussed in the chapter on vehicle motion control.

Although the autonomous vehicle hardware is sufficiently developed, the algorithms for controlling the various hardware subsystems are in the process of development at the time of this writing. These algorithms must perform a variety of relatively complex tasks. For example, as outlined above, there must be one or more algorithms for processing the sensor signals. Some aspects of this task have already been discussed in the chapter on Vehicle Safety for blind spot detection and automatic braking for collision avoidance. However, for an autonomous vehicle, the algorithms for these tasks must be combined with decision-making algorithms, particularly for level 5 vehicles.

Algorithm(s) should also be available to perform a task similar to the human driver for anticipating a condition that is likely to occur based on the instantaneous environment such as a pedestrian approaching the street not at a corner. This could include a typical present-day situation in which a pedestrian is using a cell phone and not obviously attentive to approaching vehicular traffic and could continue across the street without looking for traffic. Similarly, another vehicle approaching a stop on a cross street at an excessive speed such that it might not stop. It is also not uncommon for bicycle riders to ignore stop signs on roads with relatively light traffic and drivers need to be aware that a sudden stop might be required to avoid a collision with a bicycle.

Once the software has reached the decision to perform an action, it must generate the appropriate control signal to the affected subsystem(s). Such computation is far more straightforward than the algorithms required for environmental monitoring and decision-making. Another relatively routine algorithms (or set of algorithms) are those involved in tracking the desired route as described qualitatively earlier in this chapter. The autonomous vehicle automatic steering involves the vehicle following a path determined by various sensors and algorithms in the AV computer that provides the command input to the automatic steering system. This command input can be generated in response to the vehicle surveillance sensors. For example, a lane tracking set of algorithms exists in certain contemporary

vehicles and can be combined with data from electronic maps to select the roads being traveled. The surveillance system is explained in some detail in Chapter 11, but any lane tracking algorithms are vehicle model-specific. However, we illustrate the generation of the automatic steering command input with a hypothetical (and theoretical) example. This exemplary steering control input involves monitoring vehicle instantaneous position (e.g., via GPS), comparing with the intended position and heading (e.g., from electronic maps). The data in these maps are compared with the GPS-measured position such that the instantaneous vehicle velocity vector $(\bar{v})$ is tangent to the intended course. The direction ϕ of the vector $\bar{v}$ provides a command input ϕ_c to a control system that regulates front steering angle δ_F such that the vehicle velocity vector is tangent to the intended contour of the path that the vehicle must follow for correct navigation. It is possible to illustrate the control of an autonomous vehicle along a predetermined route that is essentially a classical "tracking" control problem. This illustration of autonomous vehicle tracking involves the vehicle traveling on a highway curve with a relatively large radius (R). For such a curve, the roll moment, roll angle ϕ_R, and lateral weight transfer are sufficiently small that the dynamic model is adequately given in terms of a two-dimensional state-vector $\bar{x}$:

$$\bar{x} = [v_y, r]$$

where v_y = vehicle lateral velocity component and r = yaw rate.

The state-vector dynamic model for this highly simplified example is given in Chapter 7 Eq. (7.110) which, for convenience, are repeated below:

$$\dot{\bar{x}} = A\bar{x} + Bu$$

$$\bar{y} = C\bar{x}$$

where

$$A = \begin{bmatrix} -2\dfrac{(C_F + C_R)}{Mu_o} & -2\dfrac{(aC_F - bC_R)}{Mu_o} - u_o \\ -2\dfrac{(aC_F - bC_R)}{I_{zz}u_o} & -2\dfrac{(a^2C_F + b^2C_R)}{I_{zz}u_o} \end{bmatrix}$$

$$B = \begin{bmatrix} \dfrac{2C_F}{M} \\ \dfrac{2aC_F}{I_{zz}} \end{bmatrix}$$

$$C = [0, 1]$$

with the output matrix C as given above the output $\bar{y}$ is a scalar y given by

$$y = r = \frac{d\phi}{dt}$$

The input to the plant $u = \delta_F$.

All parameters in this state-variable problem are defined in association with Eq. (7.110) from Chapter 7. The functional block diagram for the steering control is depicted in Fig. 12.5. In this figure, the input to the control system is the desired angle of the velocity vector $\bar{v}$ that is denoted ϕ_c. The actual angle is denoted ϕ and is given by

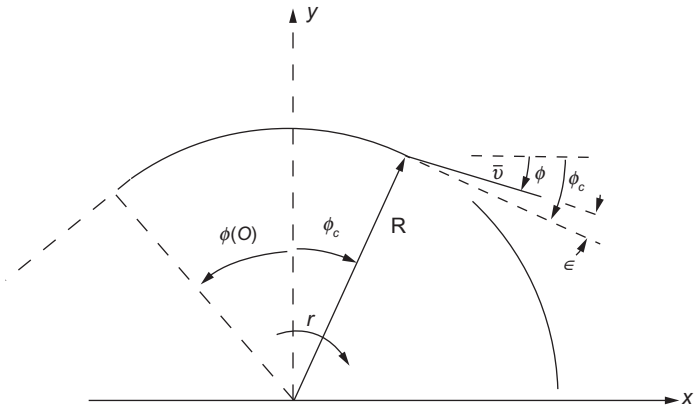


FIG. 12.6 Geometry of autonomous vehicle traveling along a curve of radius R.

$$\phi(t) = \phi(O) + \int_o^t r(\tau)d\tau \tag{12.6}$$

where $\phi(O)$ is the angle of the velocity vector when the vehicle enters the curve.

In the exemplary implementation of this steering control, the desired angle $\phi_c(t)$ is computed by the autonomous vehicle control computer from the electronic map data. Fig. 12.6 depicts the geometry of the route as contained in an electronic map (see Chapter 8 on instrumentation).

In this figure, it is assumed that the Cartesian coordinate system from the electronic map for the portion of the route along the curve is denoted x,y with the origin arbitrarily chosen to be at the center of the circular arc of radius R. The velocity vector angle ϕ is measured from a line parallel to the x-axis with positive being in the CW sense. In this figure, the tangent to the curve that is the desired velocity vector angle and is denoted ϕ_c is measured relative to the x-axis. The actual velocity vector $\bar{v}$ depicted in Fig. 12.6 is at an angle ϕ that yields an error input $\in$ to the steering control of Fig. 12.5. For illustrative purposes only, the magnitude of the error as depicted in Fig. 12.5 is exaggerated. The vehicle speed $u_o = \|\bar{v}\|$.

In any autonomous vehicle, the computer that controls the motion would have the capability of computing the angle ϕ_c such that $\bar{v}$ is tangent to the circular arc. Also depicted in this figure is the instantaneous yaw rate r where

$$r = \frac{d\phi}{dt} \tag{12.7}$$

By taking the Laplace transfer function of Eq. (12.7), the function $\phi(s)$ is given by

$$\phi(s) = \frac{r(s)}{s}$$

In Fig. 12.6, the vehicle dynamics (plant) are represented by the transfer function $H_p(s)$ where

$$\begin{aligned} H_p(s) &= \frac{r(s)}{\delta_F(s)} \\ &= C(sI - A)^{-1}B \end{aligned} \tag{12.8}$$

The derivation of this transfer function from state-variable models is explained in Appendix A.

The transfer function for the control block that is denoted $H_c(s)$ is given by

$$H_c(s) = \frac{\delta_F(s)}{\in(s)}$$

where $\in = \phi_c - \phi$.

This control transfer function can be one of several available to the system designer as explained in Appendix A including proportional (P), PI, proportional differential (PD), or proportional-integral-differential (PID). The control law for each of these is given sequentially as follows:

$$\text{P}: \quad \delta_F = K_p \in$$

$$\text{PI}: \quad \delta_F = K_p + K_I \int \in dt$$

$$\text{PID}: \quad \delta_F = K_p + K_I \int \in dt + K_D \frac{d\in}{dt}$$

$$\text{PD}: \quad \delta_F = K_p + K_D \frac{d\in}{dt}$$

The corresponding control transfer functions are given by the Laplace transform of the above equations. The most general case takes the form of the PID since various coefficients can be set = 0 for other control laws:

$$\text{PID}: \quad H_C(s) = K_p + \frac{K_I}{s} + K_D s$$

In the illustrative example, the error ϵ upon which the control variable is computed is based on ϕ. In this case, the transfer function from δ_F to output ϕ which is denoted $H_T(s)$ is obtained by combining Eqs. (12.7), (12.8) and is given by

$$H_T(s) = \frac{\phi(s)}{\delta_F(s)}$$

$$H_T(s) = \frac{H_p(s)}{s}$$

$$= \frac{C(sI - A)^{-1}B}{s}$$

As explained in Appendix A, the closed-loop transfer function $H_{C\ell}(s)$ for the steering control system is given by

$$H_{C\ell}(s) = \frac{\phi(s)}{\phi_C(s)}$$

It is further shown in Appendix A that $H_{C\ell}$ is given by

$$H_{C\ell}(s) = \frac{H_T(s)H_C(s)}{1 + H_T(s)H_C(s)H_s(s)}$$

where $H_S(s)$ = sensor transfer function for measuring ϕ. One such method of measuring heading angle ϕ is by integrating the output of a yaw rate sensor as shown in Eq. (12.6) above. An example of a relatively inexpensive angle rate sensor is given in Chapter 5. By orienting an angle rate sensor with its sensitive axis along the vehicle, z-axis yields vehicle yaw rate r.

In a typical configuration, $H_s(s)$ has sufficiently fast dynamics that it can be approximated by

$$H_S(s) \simeq 1$$

In addition, Appendix A describes the influence of the parameter choices for K_p, K_I, and K_D on the dynamic response of a control system. In addition, the stability of the steering control system, which is critically important for autonomous vehicle safety, is evaluated by various standard techniques including root locus and Bode plots. It is left as an exercise to evaluate an example autonomous vehicle steering system for a vehicle with given parameters.

A simulation of this tracking maneuver was run using the models from Chapter 7, Eq. (7.110), for A and B. In this simulation, the vehicle is traveling along a straight road at heading angle $\phi = 0$ at speed $u_o = 30$ m/s until time $t = 5$. At this point, the road intersects an arc of a circle of radius $R = 790$ m for the interval $0 \leq \phi \leq \pi/2$. The heading angle as computed by the navigation system varies linearly with time, while the vehicle continues to travel at speed u_o such that the heading rate of change ϕ_c is given by

$$\phi_c = \frac{u_o(t-5)}{R} \qquad 5 \leq t \leq 5 + \frac{\pi R}{2u_o}$$

In this simulation of the vehicle motion along the curve, the control is a PI type. The control transfer function $H_c(s)$ is given by

$$H_c(s) = \frac{K_p s + K_I}{s}$$

As an illustration of the performance of the heading control, the gain parameters were intentionally chosen to be low and are given by

$$K_p = 0.04$$
$$K_I = 0.02$$

Fig. 12.7 is a plot of the command heading $\phi_c(t)$ and the actual heading $\phi(t)$.

This figure illustrates a small initial lag in the heading due to the finite dynamic response with the artificially low K_p and K_I. However, the true heading is the desired heading after a brief transient period and a discontinuous heading of a road is extremely unlikely in practice. It is left as an exercise for the interested reader to perform the same simulation with larger $H_c(s)$ coefficients. The transient response at the beginning of the curve decays more quickly to 0 as K_p and K_I are increased.

A typical route followed by a vehicle (autonomous or not) is generally more complicated than the above example. However, this example illustrates the autonomous vehicle tracking problem and identifies the computational components involved including determining the instantaneous velocity vector for the vehicle to follow the preprogrammed route and calculating the inputs to vehicle subsystems involved in determining the vehicle actual path. However, it must be emphasized that, as described

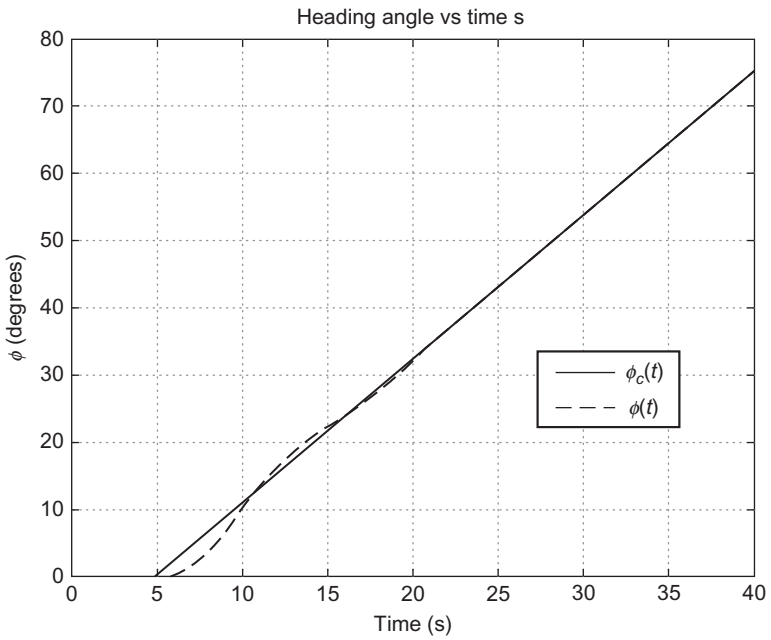


FIG. 12.7 Plot of command heading and actual heading.

above, the computer must primarily monitor the environment and continuously make decisions associated with changes in the commanded velocity vector required for safe vehicle operation.

The steering control system is only one of the hardware systems associated with an autonomous vehicle. However, it is exemplary of another important safety-related issue. The failure or degradation in performance of the steering control and other systems could render the vehicle unsafe. As is common in aircraft systems that affect safety hardware redundancy can minimize or eliminate the dangers involved in such a failure/degradation. It should be a requirement in autonomous vehicles to have hardware redundancy in the various control subsystems.

The concept of redundancy involves having a replacement component or subsystem onboard the vehicle that can take over the subsystem function when a failure/degradation has occurred. However, the redundant system by itself is not sufficient to assure vehicle safety. There must be a method of detecting failure/degradation in the operating subsystem before it is replaced with the redundant component or subsystem. Chapter 11 presents and explains a method of reliably detecting and identifying a failure in a component that is incorporated in an electronic system (e.g., automatic steering).

One of the methods employed in aircraft (e.g., those equipped with a flight management system (FMS)) is to have triple redundancy in at least a portion of the system or double redundancy in components (e.g., elevator or rudder actuator). A supervisory control system monitors the three subsystems (for triple redundancy) and compares the data from all three. If two of these subsystems have identical (or near identical) behavior and the third differs, it is the one that differs from the other two that is deemed to have

failed. If the differing subsystem is in use, it is replaced with one of the other two. This method of failure detection is not the only method capable of identifying a malfunctioning subsystem. However, in automotive applications, triple redundancy of control systems might not be economically viable.

It is a generally accepted doctrine that hardware redundancy is a necessary component of an autonomous vehicle. This is particularly the case for safety-related systems or subsystems (e.g., automatic steering systems). The normal hardware redundancy is to have a duplicate component installed such that when a failure occurs in the primary component, the control system will switch to the backup redundant component.

However, in certain vehicle configurations, there is another way to achieve redundancy via alternate hardware that can perform the same function of the system as the failed component. As an illustration of the use of alternate hardware to provide redundancy in system performance, the automatic steering system is an example. For the illustration of this topic of alternate hardware redundancy, it is assumed that the steering actuator in an automatic steering system has failed. This actuator applies a moment to the wheels that steer the vehicle (normally the front wheels). Automatic steering is employed in the higher levels of autonomous vehicles. Some level 5 autonomous vehicles are completely driverless and may not even be equipped with a steering wheel and the associated steering mechanism. However, even with a vehicle having a driver, failure of the steering actuator during certain maneuvers (e.g., following relatively short radius curve) may require steering action at a time interval that is too short for correct and safe driver action.

An alternative redundant backup for vehicle directional control to the presence of a duplicate steering actuator is to have a control system that takes over directional control via regulating differential (right/left) braking. Such a system can individually apply brakes to separate wheels on opposite sides of the vehicle or can apply brakes to the wheel on the inside of the curve of intended vehicle motion and drive force to the opposite wheel. The drive force can be directly applied to the corresponding wheel electric motor of a series hybrid or electric vehicle. It could also apply drive force to the desired wheel via traction control depending on vehicle configuration.

In any such alternate redundant directional control, this control is only to be activated on a temporary basis either until a driver takes steering control or until the vehicle can be maneuvered under automatic directional control until it reaches a safe stop (e.g., in a safe road shoulder area).

The vehicle directional control via differential brake/drive torque can be demonstrated using the lateral equations of motion for a vehicle that were presented in Chapter 7. For the purposes of this directional control, the equations can be simplified by assuming that the lateral motions involve negligible roll dynamics and for which $\dot{p} \cong 0$ and $p \simeq 0$ where

$$p = \frac{d\phi_R}{dt}$$

$$\phi_n = \text{roll angle}$$

For this simplifying assumption, the dynamic equations are given by Eqs. (7.113), (7.114) from Chapter 7 with $p = 0$, $\dot{p} = 0$ and with $\phi_R = 0$:

$$M\dot{v} + u_o r = F_{yF} + F_{yR}$$

and

$$I_{zz}\dot{r} = aF_{yF} - bF_{yR} + \delta F_b c$$

where

$$F_{yF} = 2C_F\left[\delta_F - \frac{v + ar}{u_o}\right]$$

$$F_{yR} = -2C_R\left(\frac{v + br}{u_o}\right)$$

c = lateral distance between wheels, v = lateral velocity component, and r = yaw rate.

The variable and parameter definition in these equations are given in Chapter 7 in Fig. 7.32. The variable δF_b is the difference between the wheel brake/drive force on the left/right rear wheels and is given by

$$\delta F_b = F_{bo} - F_{bi}$$

F_{bo} = brake/drive force on outer wheel of curve and F_{bi} = brake force on inner wheel of curve.

There can be circumstances involving a vehicle under automatic steering control in which the vehicle must momentarily maintain a constant speed u_o. For example, an autonomous vehicle could be traveling in a platoon of autonomous vehicles in heavy traffic on an inner lane. The vehicle with the failed steering actuator must signal for a turn that will enable it to pass through an adjacent lane when there is a safe opening in the traffic. For the affected vehicle to follow the road contour, the directional control required via δF_b must be accomplished initially with the following:

$$F_{bi} + F_{bo} = 0 \rightarrow u_o = \text{constant}$$

$$F_{bo} > 0$$

$$\delta F_b \simeq 2F_{bo}$$

Once the affected vehicle is in a lane for which deceleration (eventually to a safe stop) is possible, then

$$F_{bi} < 0$$

$$F_{bo} < 0$$

$$\dot{u}_o < 0$$

In the absence of the torque applied by the steering actuator, there is a moment applied to the front wheels due to the aligning torque. A dynamic model for the front-wheel steering angle δ_F is given by

$$J_z\dot{V}_F = \left(-t_p F_{\mathrm{YR}} + t_b\frac{c\,\delta F_b}{\ell}\right)\Big/ 2$$

$$V_F = \dot{\delta}_F$$

t_p = pneumatic trail of the front wheels, t_b = effective moment arm for the force due to $\dot{\delta}_F$ acting on each wheel, and ℓ = a+b (wheel base)where J_z = moment of inertia of each front wheel assembly.

The four equations above can be put in a state-variable model of the form

$$\dot{x} = Ax + Bu$$

where

$$x = \text{state vector}$$
$$= [v, r, V_F, \delta_F]$$

and where the matrices are given by

$$A = \begin{bmatrix} \dfrac{-(C_1+C_2)}{Mu_0} & -\left[u_o + \dfrac{aC_1 - bC_2}{Mu_o}\right] & 0 & \dfrac{C_1}{M} \\ \dfrac{-(aC_1 - bC_2)}{I_{zz}u_0} & -\left[\dfrac{a^2C_1 + b^2C_2}{I_{zz}u_0}\right] & 0 & \dfrac{aC_1}{I_{zz}} \\ \dfrac{t_pC_1}{J_z} & \dfrac{t_paC_1}{J_z} & 0 & \dfrac{-t_pC_1}{J_z} \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

where $C_1 = 2C_F$, $C_2 = 2C_R$, and $u_o =$ vehicle speed:

$$B = [0, c/I_{zz}, t_b c/(\ell J_z), 0]$$

A simulation of the maneuver in which the vehicle moves at steady speed u_o on a straight highway until it has reached a lane or the road shoulder to the right side of its initial lane where it can safety slow to a stop. The parameters for this simulation are those given in Eq. (7.122) supplemented with $J_z = 14.2$, $t_b = 0.202$, $t_p = 0.08$, $c = 1.6$. In this simulation, the differential braking causes the vehicle to turn initially to the right and then to the left so that it is traveling in the direction of the road. After reaching this safe stopping lane, $\delta F_b = 0$ and $F_{bo} = F_{bi} < 0$ causing the vehicle to decelerate to a stop. The differential braking versus time is given in Fig. 12.8.

In this figure, it is assumed that at $t = 0$, the pathway for the affected vehicle is cleared for a lane change maneuver to the right. Turning right is caused by $\delta F_b > 0$. The steering input is in the form of

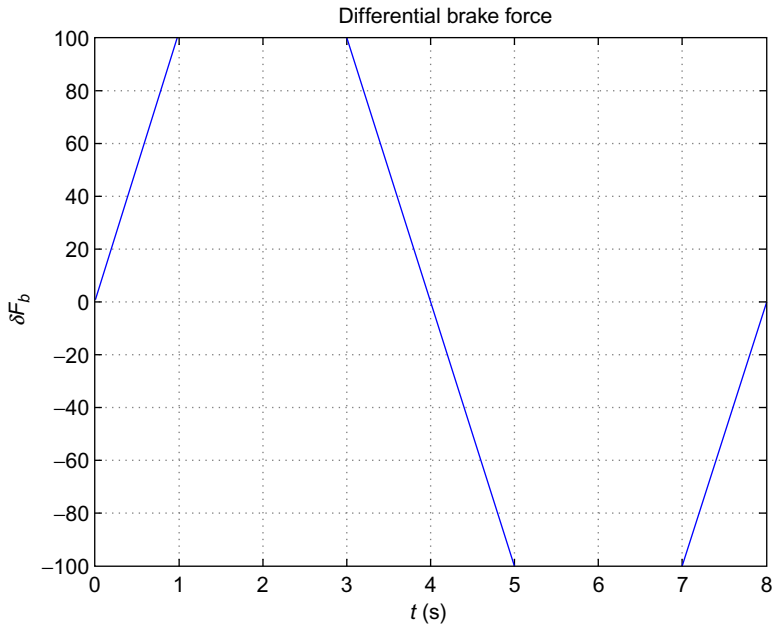


FIG. 12.8 Plot of $\delta F_b(t)$.

what can properly be called a "doublet" in which the differential brake changes polarity at $t=4$ s and turns the vehicle back to the left from $t=4$ to $t=8$ s.

The vehicle maneuver is computed from the yaw rate (r) output of the simulation, which is the second component of the state-vector. The yaw rate is the time derivative of the vehicle heading angle ϕ measured from the axis of the straight road:

$$r = \dot{\phi}$$

The vehicle instantaneous heading θ at time θ is given by

$$\phi = \int_o^t r(\tau) d\tau$$

The vehicle maneuver is given by its position $[x(t), y(t)]$ where x is the coordinate in the direction of the road and any y is the transverse coordinate. These coordinates are computed by

$$x(t) = \int_o^t u_o \cos[\phi(\tau)] d\tau$$

$$y(t) = \int_o^t u_o \sin[\phi(\tau)] d\tau$$

In the example simulation, the vehicle has reached a lane where it can decelerate at $t=8$ s and at which time $\delta F_b = 0$ and $\phi = 0$. The brakes are then applied equally on both sides, and the vehicle deceleration is given by

$$\ddot{x} = \frac{F_{bo} + F_{bi}}{M}$$
$$= \frac{-2|F_{bi}|}{M}$$

The position of the vehicle during this maneuver is given in Fig. 12.9.

The simulation example of redundant backup for a failed steering actuator has been presented in a simplified version to illustrate technology that can, in practice, be applied to move complex maneuvers for this exemplary hardware redundancy. However, not all failed components can have redundancy via alternate hardware. For such components, particularly in safety-related systems, a duplicate of the component should be installed in the vehicle. In addition, the autonomous vehicle control system must have the capability to switch to the redundant, backup component when failure of the primary component has been detected. Such detection is possible via model-based failure detection as explained in Chapter 11.

It is generally agreed among automotive manufacturers and government regulators that significant vehicular communication capability is required in addition to GPS and cell phones. One such communication system would require an infrastructure (similar to that in position for cell phones) involving multiple transceivers with necessary antennas for vehicle to infrastructure V2I (or V2V) communication (see Chapter 9) of safety-related information. For example, an unanticipated object in one or more lanes of a road (e.g., a stalled vehicle) when detected by a vehicle encountering it should involve communicating a report of this possible obstacle to other vehicles approaching it. Such information could be relayed to the other vehicles via the V2I infrastructure. Alternatively, there could be a direct vehicle-to-vehicle (V2V) communication system.

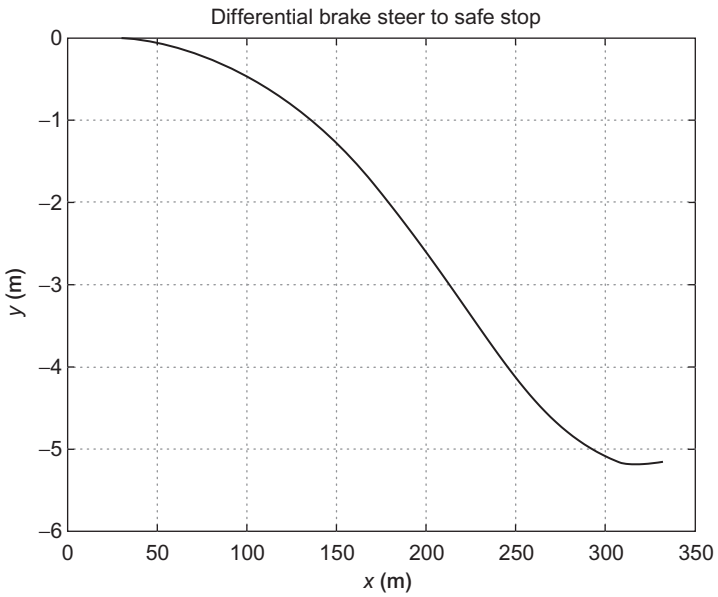


FIG. 12.9 Vehicle maneuver via differential braking.

In addition, there are great safety implications in vehicles reporting adverse weather such as sleet/ice or heavy rain such a report when received by vehicles approaching the affected area could after driving (e.g., reduce speed) patterns to maximize safety. Such adverse weather reporting has been active in aviation for decades. The report of adverse weather including icing conditions or relatively high turbulence is issued verbally by flight crew to the air traffic control system. Such reports are termed PIREPS (short for pilot reports). Perhaps the autonomous vehicle reporting would be termed drivereps. In the case of autonomous vehicles, the drivereps would be transmitted V2V or V2I automatically by the vehicle communication system.

In addition to safety-related information, the proposed V2I infrastructure has the potential to relay information about traffic conditions that can adversely affect travel times. An autonomous vehicle in conjunction with this infrastructure and with its stored electronic maps can select an alternate route.

One of the other issues that affect the widespread introduction of level 5 autonomous vehicles is the introduction of regulations. Such regulations will almost certainly be required to be adopted by federal (as opposed to individual state) governmental agencies. However, the regulations that will have to be made will require a thorough demonstration experimentally that level 5 autonomous vehicles can react safely to any possible safety-related event in the vehicle environment.

At the time of the writing of this book, the various issues involved in the broad utilization of autonomous vehicles including software development, hardware redundancy regulations, and infrastructure are still under development. However, the successful demonstration of experimental autonomous vehicles is indicative of a potentially bright future for this technology.

APPENDIX

THE SYSTEMS APPROACH TO CONTROL AND INSTRUMENTATION

A

OVERVIEW

This book discusses the application of electronics in automobiles from the standpoint of electronic systems and subsystems. In a sense, the systems approach to describing automotive electronics is a way of organizing the subject into its component parts based on functional groups. This appendix will lay the foundation for this discussion by explaining the concepts of a system and a subsystem and how such systems function and interact with one another. The means for characterizing the performance of any system will be explained so that the reader will understand some of the relative benefits and limitations of automotive electronic systems. This appendix will explain, generally, what a system is and, more precisely, what an electronic system is. In addition, basic concepts of electronic systems that are applicable to all automotive electronic systems, such as structure (architecture) and quantitative performance analysis principles, will be discussed. In the general field of electronic systems (including automotive systems), there are three major categories of function, including control, measurement, and communication.

Two major classes of electronic systems —analog or continuous time (this appendix) and digital or discrete time (Appendix B)—will be explained. In most cases, it is theoretically possible to implement a given electronic system as either an analog or a digital system. The relatively low cost of digital electronics coupled with the high performance achievable relative to analog electronics has led modern automotive electronic system designers to choose digital rather than analog realizations for new systems.

CONCEPT OF A SYSTEM

A *system* is a collection of components that function together to perform a specific task. Various systems are encountered in everyday life. It is a common practice to refer to the bones of the human body as the skeletal system. The collection of highways linking the country's population centers is known as the interstate freeway system.

Electronic systems are similar in the sense that they consist of collections of electronic and electrical parts interconnected in such a way as to perform a specific function. The components of an electronic system include transistors, diodes, resistors, and capacitors, as well as standard electrical parts such as switches and connectors, among others. All these components are interconnected with individual

wires or with printed circuit boards. In addition, many automotive electronic systems incorporate specialized components known as *sensors* or *actuators* that enable the electronic system to interface with the appropriate automotive mechanical systems. Systems can often be broken down into subsystems. The subsystems also consist of a number of individual parts.

Any electronic system can be described at various levels of abstraction, from a pictorial description or a schematic drawing at the lowest level to a block diagram at the highest level. For the purposes of this appendix, this higher-level abstraction is preferable. At this level, each functional subsystem is characterized by inputs, outputs, and the relationship between input and output. Normally, only the system designer or maintenance technician would be concerned with the detailed schematics and the internal workings of the system. Furthermore, the only practical way to cover the vast range of automotive electronic systems is to limit our discussion to this so-called system level of abstraction in this appendix. It is important for the reader to realize that there are typically many different circuit configurations capable of performing a given function.

BLOCK DIAGRAM REPRESENTATION OF A SYSTEM

At the level of abstraction appropriate for the present discussion, a block diagram will represent the electronic system. Depending on whether a given electronic system application is to (a) control, (b) measure, or (c) communicate, it will have one of the three block diagram configurations shown in Fig. A.1. The designer of a system often begins with a block diagram, in which major components are represented as blocks.

In block diagram architecture, each functional component or subsystem is represented by an appropriately labeled block. The inputs and outputs for each block are identified. In electronic systems, these input and output variables are electrical signals, except for the system input and system output. One benefit of this approach is that the subsystem operation can be described by functional relationships between input and output. There is no need to describe the operation of individual transistors and components within the blocks at this block diagram level.

In the performance analysis of an existing system or in the design of a new one, the system or subsystem is represented by a mathematical model that is derived from its physical configuration. Normally, this model is derived from known models of each of its constituent parts, that is, its basic physics. Initially, this appendix will consider components, subsystems, and system blocks that can be represented by a linear mathematical model. Later in this appendix, the treatment of nonlinearities is discussed.

For a block having input $x(t)$ and output $y(t)$ that can be represented by a linear model, the model is of the form of a differential equation (A.1):

$$a_o y + a_1 \frac{dy}{dt} + \cdots a_n \frac{d^n y}{dt^n} = b_o x + b_1 \frac{dx}{dt} \cdots b_m \frac{d^m x}{dt^m} \tag{A.1}$$

Typically, $n \geq m$ and such a system block or component is said to be of order n. Analysis of this block is accomplished by calculating its output $y(t)$ for an arbitrary (but physically realizable) input $x(t)$. The performance of such a block in an automotive system normally involves finding its response to certain physically meaningful inputs. Such analysis is explained later in this appendix.

Fig. A.1A depicts the architecture or configuration for a control application electronic system. In such a system, control of a physical subsystem (called the *plant*) occurs by regulating some physical variable (or variables) through an actuator. An actuator is an energy-conversion device having an electrical input and an output of the physical form required to vary the plant (e.g., mechanical energy) as

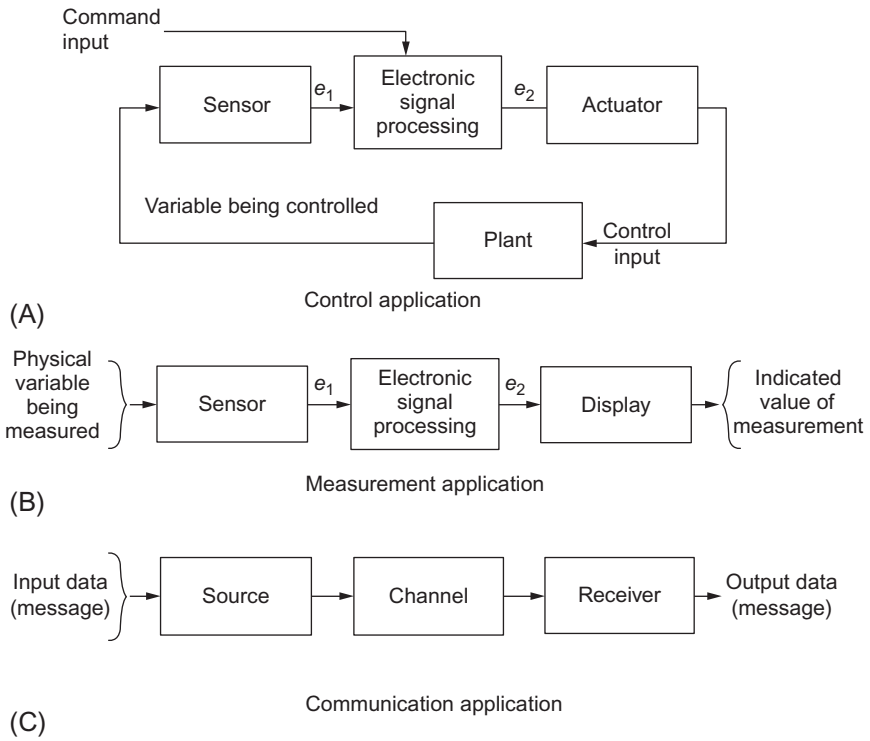


FIG. A.1 Electronic system block diagram variable being controlled. (A) Control system block diagram, (B) measurement system block diagram, and (C) communication system block diagram.

required to perform the desired system function. Thus, an actuator has an electrical input and an output that may be mechanical, pneumatic, hydraulic, chemical, or so forth. The plant being controlled varies in response to changes in the actuator output. The control is determined by electronic signal processing based on the measurement of some variable (or variables) by a sensor in relationship to a command input by the operator of the system (i.e., by the driver in an automotive application). Both sensors and actuators with automotive applications are explained in Chapter 5.

In an electronic control system, the output of the sensor is always an electrical signal (denoted e_1 in Fig. A.1A). The input is the desired value of the physical variable in the plant being controlled. The electronic signal processing generates an output electrical signal (denoted e_2 in Fig. A.1A) that operates the actuator. The signal processing is designed to achieve the desired control of the plant in relation to the variable being measured by the sensor. The operation of such a control system is described later in this appendix. At this point, we are interested only in describing the control-system architecture. An explanation of electronic control is presented later in this appendix with appropriate analytic models and analysis.

The architecture for electronic measurement (also known as instrumentation) is similar to that for a control system in the sense that both structures incorporate a sensor and electronic signal processing. However, instead of an actuator, the measurement architecture incorporates a display device. A display

is an electromechanical or electro-optical device capable of presenting numerical values to the user (driver). In automotive electronic measurement, the display is sometimes simply a fixed message rather than a numeric display. Nevertheless, the architecture is as shown in Fig. A.1B. It should be noted that both control and instrumentation electronic systems use one or more sensors and electronic signal processing.

Fig. A.1C depicts a block diagram for a communication system. In such a system, data or messages are sent from a source to a receiver over a communication channel. This particular architecture is sufficiently general that it can accommodate all communication systems from ordinary car radios to digital data buses between multiple electronic systems on cars and extravehicular communication. Communication systems are described in detail in Chapter 9.

ANALOG (CONTINUOUS TIME) SYSTEMS

Modern automotive digital electronic systems have virtually completely replaced analog systems. Whereas digital systems are represented by discrete-time models, analog systems are represented by continuous-time models having a form such as is given in Eq. (A.1). Normally, automotive electronic systems incorporate components (e.g., sensors and actuators) that are best characterized by continuous-time models. Typically, only the electronic portion is best characterized by discrete-time models. Furthermore, even the digital electronics can be represented by an equivalent continuous-time model, which can be converted to a discrete-time equivalent readily. Consequently, this discussion begins with a brief overview of linear continuous-time system theory. The discrete-time system theory is reviewed in Appendix B.

LINEAR SYSTEM THEORY: CONTINUOUS TIME

The performance of a continuous-time block (i.e., component/system) is found from the solution to the differential equation (A.1) for a specific input. One straightforward method of solving this equation is to take the Laplace transform of each term. The Laplace transform (also denoted $x(s) = \mathcal{L}[x(t)]$) of the input is denoted $x(s)$ and is defined as following the linear integral transform of its time-domain representation:

$$x(s) = \int_{o}^{\infty} e^{-st} x(t)dt + x(t)|_{t=0^-} \tag{A.2}$$

where $s = \sigma + j\omega$ = complex frequency
and where

$$j = \sqrt{-1} \tag{A.3}$$

Similarly, the Laplace transform of the block output is denoted $y(s)$ and is given by

$$y(s) = \int_{o}^{\infty} e^{-st} y(t)dt + y(t)|_{t=0^-} \tag{A.4}$$

where $t = 0^-$ means the limit as $t \to 0$ from the left. This restriction on $x(t)$ allows the Laplace transform to be taken with a discontinuity in $x(t)$ at $t = 0$ (e.g., step input).

The differential equation model for a given continuous-time block includes time derivatives of the input and output. The Laplace transform of the time derivative of order m of a variable (e.g., the input) is given by

$$\int_{o}^{\infty} e^{-st}\frac{d^m x}{dt^m}dt = s^m x(s) \quad m = 1,2,\ldots \tag{A.5}$$

For any practical application of the Laplace transform, the initial conditions for both input and output are zero,

$$x(t)|_{t=o} = 0, \quad y(t)|_{t=o} = 0$$

the Laplace transform of the differential equation (Eq. A.1) for the block yields

$$\left[a_0 + a_1 s + a_2 s^2 \ldots a_n s^n\right] y(s) = \left[b_0 + b_1 s + \cdots b_m s^m\right] x(s) \tag{A.6}$$

It is conventional for the purpose of conducting analysis for continuous-time systems to define the transfer function ($H(s)$) for each block (Eq. A.7):

$$\begin{aligned} H(s) &= \frac{y(s)}{x(s)} \\ &= \frac{b_0 + b_1 s + \cdots b_m s^m}{a_0 + a_1 s + \cdots a_n s^n} \end{aligned} \tag{A.7}$$

The transfer-function concept is highly useful for continuous-time linear system analysis since the transfer function for any such system made up of a cascade connection of K blocks (e.g., as depicted in Fig. A.2) is the product of the transfer functions of the individual blocks.

Denoting the transfer function of the kth block $H_k(s)$ and for the complete system $H(s)$, the latter is given by

$$H(s) = \prod_{k=1}^{K} H_k(s) \tag{A.8}$$

An alternate, highly useful, formulation of the transfer function is based on the roots of equations formed from its numerator and denominator polynomials. The roots z_j of the numerator polynomial ($P_N(s)$) are the m solutions to the equation:

$$P_N(s) = b_0 + b_1 s + \cdots b_m s^m = 0 \tag{A.9}$$

where $P_N(z_j) = 0 \quad j = 1,2,\ldots,m.$

The roots z_j are called the zeros of the transfer function. Similarly, the roots p_i of the denominator polynomial ($P_D(s)$) are the n solutions to the Eq. (A.10):

$$P_D(s) = a_0 + a_1 s + \cdots a_n s^n = 0 \tag{A.10}$$

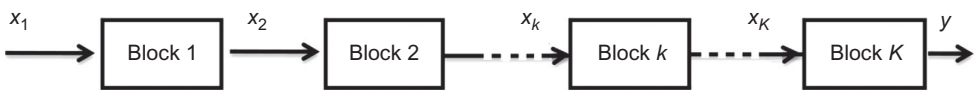


FIG. A.2 System cascade connection block diagram.

where $P_D(p_i) = 0 \quad i = 1,2,\ldots,n$.

The roots p_i are called the poles of the transfer function. For a system that is stable, all poles and zeros have negative real parts (i.e., $\sigma < 0$), or equivalently, all poles and zeros of a stable system lie in the left half of the complex s-plane. The dynamic response of the block to any input is determined by its poles and zeros.

The alternate form for $H(s)$ in terms of its poles and zeros is given by

$$H(s) = \frac{b_m \prod_{j=1}^{m} (s - z_j)}{a_n \prod_{i=1}^{n} (s - p_i)} \tag{A.11}$$

The time-domain response of a continuous-time block to any given input is given by the inverse Laplace transform of $Y(s)$. One method of computing this inverse Laplace transform uses the residue theorem of complex analysis. The block output $Y(s)$ is given as (Eq. A.12)

$$Y(s) = H(s)X(s) \tag{A.12}$$

The residue theorem expresses the output time domain in terms of the poles and zeros of the product $H(s)X(s)$ and includes the poles and zeros of $H(s)$ and any zeros and poles of the input:

$$Y(s) = K \frac{\prod_{j=1}^{M} (s - z_j)}{\prod_{i=1}^{N} (s - p_i)} \tag{A.13}$$

where $K = \dfrac{b_m'}{a_n'}$ and b_m' is the coefficient of the highest power of s in the numerator of $Y(s)$; a_n' is the coefficient of the highest power of s in the denominator of $Y(s)$.

Assuming that all of the poles are distinct (*i.e.*, $p_{j \neq} p_k$ unless $j = k$), the time-domain output is given by

$$y(t) = \sum_{k=1}^{N} (s - p_k) H(s) X(s) \Big|_{s \to p_k} e^{p_k t} \tag{A.14}$$

In evaluating this expression, the pole at $s = p_k$ is canceled by the term $(s - p_k)$ in the product of $H(s)$ $X(s)$. That is, each pole of the product contributes an exponential term to the output.

The above formula for calculating the output time domain is, in fact, the inverse Laplace transform of $y(s)$ denoted

$$y(t) = \mathcal{L}^{-1}[Y(s)] \tag{A.15}$$

The formula given above for calculating the inverse transform is known as the *residue theorem*.

The inverse Laplace transform of a system transfer function $H(s)$ is known as its impulse response. It is denoted $h(t)$ and is given by the inverse Laplace transform of $H(s)$:

$$h(t) = \mathcal{L}^{-1}[H(s)] \tag{A.16}$$

It is the response of a linear system to a unit impulse. The output of a linear system to an input $x(t)$ can be found from its impulse response by the so-called convolution theorem, which is given as follows:

$$y(t) = \int_{-\infty}^{\infty} h(\tau)x(t-\tau)d\tau \tag{A.17}$$

For any causal, stable system, the impulse response is zero for negative argument:

$$h(\tau) = 0 \quad \tau \leq 0 \tag{A.18}$$

One of the important inputs for assessing the performance of a block is its response to a unit step input:

$$\begin{aligned} x(t) &= 0 \quad t < 0 \\ &= 1 \quad t \geq 0 \end{aligned}$$

It is straightforward to show that the Laplace transform of this step input is given by

$$x(s) = \frac{1}{s}$$

That is, this step input contributes one pole to the product $H(s)X(s)$ at $s = 0$.

FIRST-ORDER SYSTEM

As an example of this type of analysis, we consider the step response of a first-order block whose model is given by

$$a_0 y + a_1 \frac{dy}{dt} = b_0 x \tag{A.19}$$

The transfer function for this block is given by

$$H(s) = \frac{b_0}{a_0 + a_1 s} = \frac{\dfrac{b_0}{a_1}}{s + \dfrac{a_0}{a_1}} \tag{A.20}$$

This transfer function has a single pole where $p = -\dfrac{a_0}{a_1}$.

The block output $Y(s)$ for a unit step input has poles at $s = 0$ and $s = -\dfrac{a_0}{a_1}$.

The time response of this block to a unit step is given by

$$y(t) = \frac{b_0}{a_1}\left[1 - \exp\left(-\frac{a_0 t}{a_1}\right)\right] \tag{A.21}$$

It is common practice to characterize the step response of a first-order system in terms of the reciprocal of its pole, denoted τ here:

$$\tau = \frac{a_1}{a_0}$$

This parameter has dimensions of time (second) and is called the first-order time constant for the system. Continuing with the present example, Fig. A.3 depicts the unit step response of a first-order system in which $b_0 = 3.3$, $a_0 = 1$, and $a_1 = 0.55$.

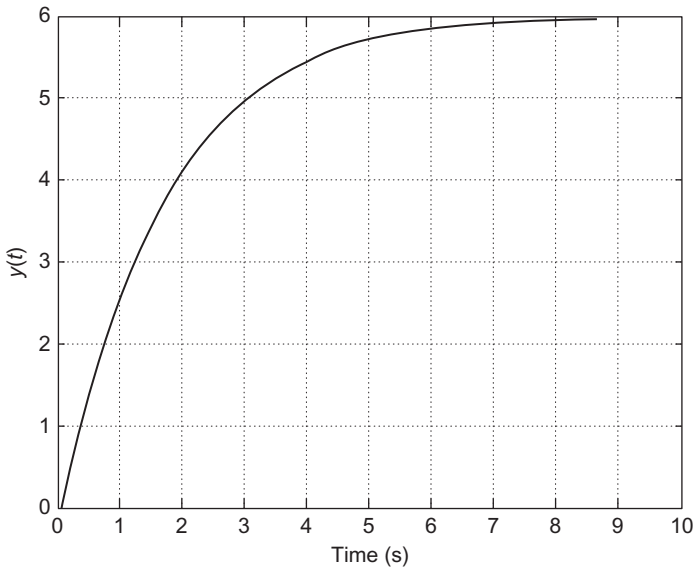


FIG. A.3 Unit step response of a first-order system.

A second-order system has two poles that are both either negative real numbers or complex conjugate pair with negative real parts. This latter case will have a step response of the form of a sinusoid of exponentially decreasing amplitude. A second-order system transfer-function step-response example is presented later in this appendix. For higher than second order, no simple intuitive description of the response is possible. Such higher-order systems are encountered as examples in several chapters in this book.

The dynamic response of any linear continuous-time automotive system (regardless of its physical form, e.g., electronic and mechanical) is found in the same procedure as given above. That is, a mathematical model of the system of the form of Eq. (A.1) is developed. It should be noted before proceeding that, depending upon the variables chosen for the model, one or more of the initial formulations may contain time integrals of certain variables. That equation can be reduced to the form of Eq. (A.1) by differentiating all terms with respect to time. Alternatively, in the second step of taking the Laplace transform of the equation, the Laplace transform of the integral of a variable, for example,

$$\int_0^t x(t')\,dt'$$

(with assumed zero initial condition) is given by

$$\int_0^\infty e^{-st}\left(\int_0^t x\ (t')\ dt'\right)dt = \frac{x(s)}{s} \tag{A.22}$$

The next step in the analysis process is to form the transfer function for each block in the system. The transfer function for the entire system is the product of the transfer function for each block in any cascade connection. Of course, not all automotive systems have a simple cascade connection topology. Rather, automatic control systems have a topology that involves connecting the output of some block to the input of a previous block as will be explained later.

Fortunately, there are computer simulation application programs (e.g., MATLAB/SIMULINK) that can find the solution to the system differential equation for any of the practically useful inputs for system analysis. However, any computer simulation program requires that each block in the system be modeled as accurately as possible. One of the primary benefits to computer simulation for system performance analysis is that these programs can handle nonlinearities. It is shown via example in various chapters of this book that computer simulation of system performance also permits optimization of any given design configuration by selectively varying certain key parameters. Ultimately, however, the value of design/analysis through computer simulation is dependent on the accuracy of the model for each component. To support the computer simulation approach to system design/analysis, much of the chapters of this book are devoted to modeling both automotive components and electronic system blocks and conducting performance analysis via simulation.

SECOND-ORDER SYSTEM

As an example of the performance analysis of a second-order continuous-time linear system, we consider a spring/mass/damper subsystem as depicted in Fig. A.4.

This example is a highly oversimplified lowest approximation to an automotive suspension system at one wheel (i.e., quarter car). A much more realistic model for the quarter-car suspension is presented in Chapter 7. In this figure, a mass (M_u) representing the wheel/tire/brake assembly (called the unsprung mass) that is attached via spring having spring rate K and viscous damper (e.g., shock absorber) having damping parameter D to an inertial reference frame denoted M_s. A time-varying force $F(t)$ is

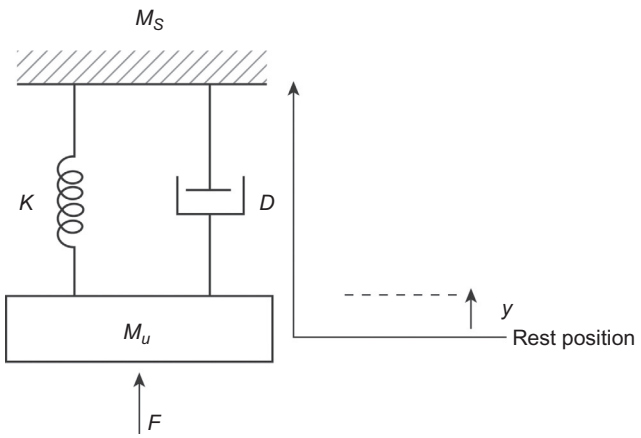


FIG. A.4 Example of second-order system configuration.

applied vertically to mass M_u. The instantaneous displacement of mass M_u relative to its position for $F=0$ is denoted $y(t)$.

The model for the dynamic response of $y(t)$ to force $F(t)$ is found by setting the sum of all forces in the y-direction to 0 (Eq. A.23):

$$F - M_u\ddot{y} - D\dot{y} - Ky = 0 \tag{A.23}$$

where

$$\dot{y} = \frac{dy}{dt}$$

$$\ddot{y} = \frac{d^2y}{dt^2}$$

The second term on the left-hand side of the equation is the inertial force associated with acceleration of mass M_u. The third term is the force due to the viscous damper, and the last term is the force due to the spring displacement. Here and throughout this book, the dot over a variable is a common notation for its derivative with respect to time. This differential equation can readily be put in a form similar to our standard model of Eq. (A.1):

$$M_u\ddot{y} + D\dot{y} + Ky = F \tag{A.24}$$

It is common a practice when dealing with second-order blocks such as this to rewrite the equation by dividing it by M_u and introducing the following parameters:

$$\omega_0 = \sqrt{\frac{K}{M_u}} = \text{natural frequency}$$

$$\zeta = \frac{D}{2\omega_0 M_u} = \text{damping ratio}$$

That standard-form differential equation with a notation simplification for the input is

$$\ddot{y} + 2\zeta\omega_0\dot{y} + \omega_0^2 y = \frac{F}{M_u} = f(t) \tag{A.25}$$

The solution to this differential equation is most readily found using the Laplace transform method explained earlier in this appendix. Assuming for convenience that the system is initially at rest (i.e., $F(0)=0$ and $\delta y(0)=0$), the Laplace transform of the differential equation is given by

$$\left(s^2 + 2\zeta\omega_0 s + {\omega_0}^2\right)y(s) = \frac{F(s)}{M_u} = f(s) \tag{A.26}$$

The operational transfer function is given by (Eq. A.27)

$$H(s) = \frac{y(s)}{F(s)} = \frac{1/M_u}{s^2 + 2\zeta\omega_0 s + {\omega_0}^2} \tag{A.27}$$

The response of the example second-order system to an arbitrary input $F(t)$ can be found by first finding $F(s)$ and then taking the inverse Laplace transform of $y(s)$ (e.g., by the method of residues) to obtain $y(t)$. However, in practice, solutions to the differential equations derived for systems/subsystems/components are done using computer simulation tools (e.g., MATLAB/SIMULINK). These simulation

tools permit analysis of system response for systems that have nonlinear models and are of essentially arbitrary second order.

It is beyond the scope of this work to fully explain the MATLAB/SIMULINK tools, which are, in any event, covered very well by the accompanying documentation. Rather, the purpose here is to continue with the dynamic analysis of the second-order example system. For time-domain analysis using MATLAB/SIMULINK, it is helpful to rewrite the original differential equation with the input in the simplified notation of $f(t)$ in the form

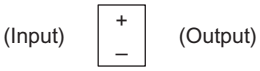
$$\ddot{y} = f(t) - \frac{D\dot{y}}{M_u} - \frac{Ky}{M_u}$$

Simulation programs such as SIMULINK incorporate blocks from a library of standard blocks that are connected together to form a block diagram of the complete system such that the necessary operations to solve the equation are performed in the correct sequence. One of the most important operations in solving any differential equation is integration with respect to time. The block, which performs the time integral of its input, is depicted in the SIMULINK library by the symbol of

That is, the operational transfer function of a time integral is $1/s$, which is implied by this library block. The input to this block is on its left side, and its output is on the right.

Multiplication by a constant K is depicted by the symbol

The sum (or difference) of two or more variables is depicted by the block



The number of $\pm$ signs on the block is chosen by the user to be the number of variables being summed with the correct signs for the variables in the equation being solved. Various inputs to the system (i.e., $f(t)$) are available from the SIMULINK library, including step, sinusoid, signal generator, random process, and an arbitrary user created input function that is stored in a file that is represented by a block in the SIMULINK block diagram. It is helpful in interpreting the SIMULINK block diagram for the example second-order system to note the following relationships:

$$\dot{y} = \int \ddot{y}\,dt$$

$$y = \int \dot{y}\,dt$$

These relationships are implied by the MATLAB/SIMULINK system depicted in Fig. A.5.

This figure is a SIMULINK block diagram for finding the unit step response of the example second-order system. In this figure, the input $f(t)$ is a unit step, and the output of the summing block is $\ddot{y}$. The inputs to this block are the components of the right-hand side of the second-order differential equation above. The first integrator output is $\dot{y}$, and the second integrator output is y. The first gain block is a constant multiplier of value D/M_u, and the second is a constant of value K/M_u. The block labeled "To

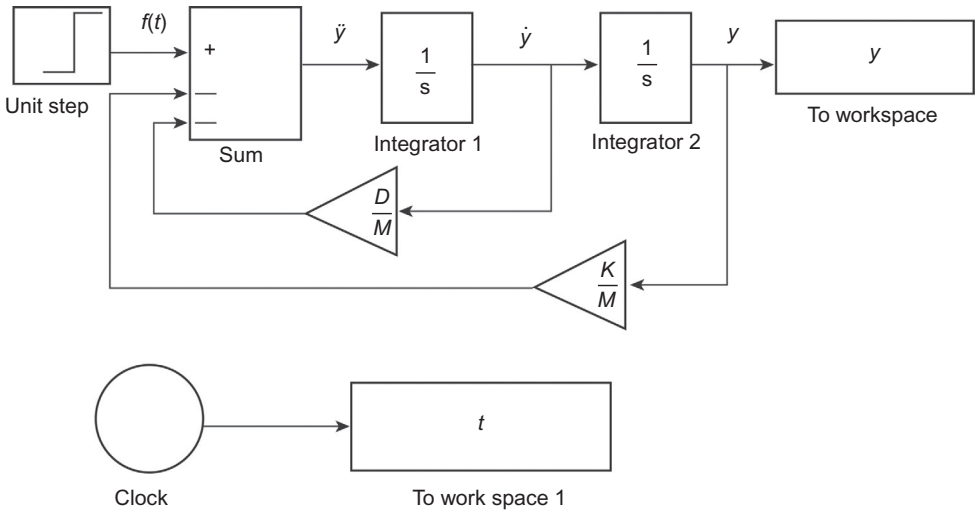


FIG. A.5 MATLAB/SIMULINK second-order system.

Workspace" with label y is a file that is created by the simulation output. Below the main simulation block diagram is a second file "To Workspace 1," which creates a file of time that is synchronous with the output y and contains simulation time t.

To further illustrate the use of this SIMULINK model, the following specific parameters were arbitrarily chosen:

$$M_u = 1.7$$

$$K = 50$$

$$D = 5.83$$

The normalized input $f(t)$ is a unit step at $t = 0.5$ sec, that is,

$$f(t) = 0 \quad t < 0.5$$

$$= 1 \quad t \geq 0.5$$

Fig. A.6 is a graph of $y(t)$ depicting the motion of the mass M_u in response to a step in $F = M_u f(t)$.

In various chapters of this book, we will present examples of the dynamic response of automotive systems/subsystems with models that are much more accurate in representing the actual physical devices.

STEADY-STATE SINUSOIDAL FREQUENCY RESPONSE OF A SYSTEM

Another important input in conducting performance analysis of a system is the sinusoidal function. The input in this case is of the form $x(t) = A\cos(\omega t) + B\sin(\omega t)$, where A and B can be varied to evaluate certain system responses (including setting either variable to zero). There is an important identity from

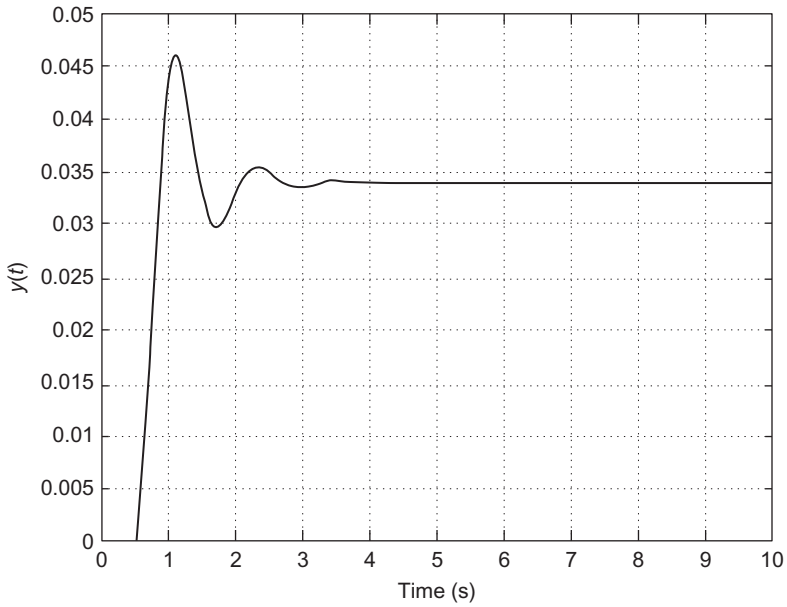


FIG. A.6 Unit step response of example second-order system.

complex analysis that simplifies the calculation of the sinusoidal frequency response of any system, which is given by

$$e^{j\omega t} = \cos(\omega t) + j\sin(\omega t) \tag{A.28}$$

This identity can also be expressed by the following relations:

$$\cos(\omega t) = \mathrm{Re}\left[e^{j\omega t}\right] \tag{A.29}$$

$$\sin(\omega t) = \mathrm{Im}\left[e^{j\omega t}\right]$$

where Re( ) is the real and Im( ) is the imaginary component of the argument and where $\omega = 2\pi f$ (f = natural frequency Hz).

For any stable, linear system, its response to a sinusoidal input consists of two parts, transient response and steady-state sinusoidal (SSS) response. The transient response is the dynamic output of the system to the initial application of the sinusoid. Each component of this transient response for a stable system decays exponentially to zero. Following the period in which the transient decays to zero, the remaining output is a SSS output. It is the SSS response that is the goal of the system analysis with sinusoidal input

$$x(t) = \mathrm{Re}\left[X(j\omega)e^{j\omega t}\right]$$

The SSS frequency response system output (after all transients have decayed to zero) is of the form

$$y(t) = \mathrm{Re}\left[Y(j\omega)e^{j\omega t}\right]$$

The SSS frequency response, which is denoted $H(j\omega)$, is defined as

$$H(j\omega) = \frac{Y(j\omega)}{X(j\omega)} \tag{A.30}$$

This SSS is a complex function of $j\omega$. It is identical to the transfer function along the imaginary axis of the complex frequency plane or an s-plane. Thus, the SSS is obtained by replacing s in the transfer function with $(j\omega)$:

$$H(j\omega) = H(s)|_{s=j\omega} \tag{A.31}$$

Since it is a complex-valued function, it can be expressed in the form of an absolute value or magnitude ($|H(j\omega)|$) and a phase angle ϕ:

$$H(j\omega) = |H(j\omega)|e^{j\varphi(\omega)} \tag{A.32}$$

The magnitude of $H(j\omega)$ is the ratio of the amplitude of its SSS response to the amplitude of the sinusoidal input. The phase $\phi(\omega)$ is the phase of the output to the input sinusoid.

The SSS is useful for evaluating the fidelity of the system response over the spectrum of the input. Ideally, a system block should have a constant amplitude and phase over the entire frequency content of its input. In practice, any physically realizable block has a response that varies with frequency ω. Often, the system designer can choose design parameters to achieve acceptable performance over a required frequency range. The range of frequencies over which such acceptable performance is achieved is termed its "bandwidth." The same computer simulation software used to evaluate step response is capable of calculating and plotting the SSS frequency response. Normally, this is presented in a "Bode" plot format in which the magnitude and phase are given in the form

$$\text{magnitude}: 20\log|H(j\omega)| \text{vs. } \log(\omega)$$

$$\text{phase}: \phi(\omega) \text{ vs.} \log(\omega)$$

This Bode plot can be evaluated by the software from the same mathematical model for the block as is used to calculate its step response.

STATE VARIABLE FORMULATION OF MODELS

Often, in the process of modeling physical systems (such as automotive systems or subsystems), it is possible to write a set of first-order linear differential equations for a set of N variables. This set of variables is denoted

$$x_n; \quad n = 1,2,\cdots,N$$

The differential equations are written in the form

$$\dot{x}_m = \sum_{n=1}^{N} A_{mn}x_n + \sum_{k=1}^{K} B_{mk}u_k \tag{A.33}$$

where u_k $(k = 1,2,\cdots,K)$ (inputs to the system)
and where A_{mn} and B_{mk} are constants for the given physical system. A unique solution for each independent variable for any given known input set is possible provided that there are N-independent

equations of the above form. For this type of model, the set of equations can be written in matrix form

$$\begin{aligned}\dot{x}_1 &= A_{11}x_1 + \cdots A_{1N}x_n + B_{11}u_1 \cdots B_{1k}u_k \\ &\vdots \\ \dot{x}_N &= A_{N1}x_1 \cdots A_{NN}x_n + B_{N1}u_1 \cdots B_{Nk}u_k\end{aligned} \tag{A.34}$$

An N-dimensional vector x is then defined as

$$x = [x_1, x_2, \cdots, x_N]^T$$

where $T \rightarrow$ transpose. The variables x_n in this formulation are known as "state variables." Similarly, the input is defined as the K-dimensional vector:

$$u = [u_1, \cdots, u_K]^T$$

These equations are written in a standard "state-variable model" form (Eq. A.35):

$$\dot{x} = Ax + Bu \tag{A.35}$$

where, for any real physical system, the dimensionality is given by the following set:

$$\begin{aligned}x &\in R^N \\ u &\in R^K \\ A &\in R^{N\times N} \\ B &\in R^{N\times K}\end{aligned}$$

The desired output variables for the system

$$y_j = \sum_{n=1}^{N} C_{jn}x_n + \sum_{k=1}^{K} D_{jk}u_k \quad j = 1,2,\cdots,J \tag{A.36}$$

In this formulation, the possibility that a set of input variables contributes to various outputs is shown by the second term on the right-hand side of the above equation. This set of output equations can also be put in matrix form in terms of the output vector:

$$\begin{aligned}y &= [y_1, y_2, \cdots, y_J]^T \\ y &= Cx + Du\end{aligned} \tag{A.37}$$

$$\begin{aligned}y &\in R^J \\ \text{where } C &\in R^{J\times N} \\ D &\in R^{J\times K}\end{aligned}$$

The complete model for the system in standard state-variable form is given by

$$\begin{aligned}\dot{x} &= Ax + Bu \\ y &= Cx + Du\end{aligned} \tag{A.38}$$

It should be noted that the direct coupling of the system input (u) to the output (y) is not a frequent occurrence. The majority of state variable models have $D = 0$. This term is automatically suppressed in all models encountered in this book except for the battery equivalent circuit at the end of this Appendix.

This system of equations is solved (in closed form) by taking Laplace transforms of the equations. For the present discussion, we assume zero initial conditions (i.e., $x_n(0) = 0 \forall n$). The system of N first-order differential equations becomes a system of N algebraic equations in the complex variable s:

$$\begin{aligned} sx(s) &= Ax(s) + Bu(s) \\ y(s) &= Cx(s) + Du(s) \end{aligned} \tag{A.39}$$

The first of these equations can be rewritten in the form

$$(sI - A)x(s) = Bu(s) \tag{A.40}$$

where $I = N$-dimensional identity matrix (i.e., all diagonal elements are 1 and off-diagonal elements are 0) and $I \in R^{N \times N}$.

This equation can be solved for $x(s)$ by multiplying both sides by the inverse of the matrix $(sI - A)$, which is denoted $(sI - A)^{-1}$:

$$x(s) = (sI - A)^{-1}Bu(s) \tag{A.41}$$

The desired output $y(s)$ is given by

$$\begin{aligned} y(s) &= C(sI - A)^{-1}Bu(s) + Du(s) \\ &= H(s)\,u(s) \end{aligned}$$

The output equation is actually a set of J equations for variables y_j:

$$y_j(s) = \sum_{k=1}^{K} H_{jk}(s)u_k(s) \quad j = 1,2,\ldots,J \tag{A.42}$$

The response of any output variable y_j to any single input u_k is given by the operational transfer function $H_{jk}(s)$:

$$y_j = H_{jk}(s)u(s) \quad k = 1,2,\ldots,K$$

The corresponding time-domain variable $y_j(t)$ can be found by taking the inverse Laplace transform or equivalently using the residue theorem method. The state-variable formulation is used throughout this book with respect to automotive electronic systems.

The numerical solution to the state-variable equation (A.38) is frequently found via simulation for a given input $u(t)$ using analytic techniques (e.g., MATLAB/SIMULINK). It is possible to interpret the solution to $y(t)$ and choose the optimal parameters for the design (i.e., coefficients in A and B) to achieve a desired system performance.

CONTROL THEORY

Electronic control systems have many applications in modern automobiles. We present here a general theory of linear control systems that is useful for explaining and understanding those encountered in various chapters.

There are two major categories of control systems: open-loop (or feedforward) and closed-loop (or feedback) systems. There are many automotive examples of each, as we will show in various chapters throughout this book. The architecture of an open-loop system is given in the block diagram of Fig. A.7.

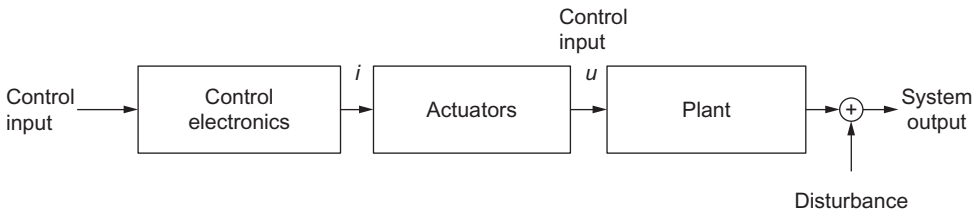


FIG. A.7 Open-loop system configuration.

OPEN-LOOP CONTROL

The components of an open-loop controller include the electronic controller, which has an output to an actuator. The actuator, in turn, regulates the plant being controlled in accordance with the desired relationship between the command input and the value of the controlled variable in the plant. Many examples of open-loop control are encountered in automotive electronic systems, such as fuel control in certain operating modes. An open-loop control system never compares actual output with the desired value.

In the open-loop control system of Fig. A.7, the command input is sent to the electronic controller, which performs a control operation on the input to generate an intermediate electrical signal (denoted *i* in Fig. A.7). This electrical signal is the input to the actuator that generates a control input (denoted *u* in Fig. A.7) to the plant that, in turn, regulates the plant output to the desired value. This type of control is called open-loop control because the output of the system is never compared with the command input to evaluate control-system performance at regulating the output to the desired input.

The operation of the plant is directly regulated by the actuator (which might simply be an electrical motor). Chapter 5 presents a discussion of various actuators used in automotive electronic control systems. The system output may also be affected by external disturbances that are not an inherent part of the plant but are the result of the operating environment. There are many disturbances occurring in automotive electronic systems as discussed in relationship to specific examples.

One of the principal drawbacks to the open-loop controller is its inability to compensate for changes that might occur in the controller or the plant or for any disturbances or due to environmental changes. This defect is eliminated in a closed-loop control system, in which the actual system output is compared with the desired output value in accordance with the input. Of course, a measurement must be made of the plant output in such a system, and this requires measurement instrumentation (discussed later in this appendix), that is often simply a sensor as explained in Chapter 5.

CLOSED-LOOP CONTROL

In its simplest form (which can be expanded to cover very complex systems), the block diagram of a so-called electronic feedback-control system is depicted in Fig. A.8. In this configuration, the control system is intended to regulate the output (y) of a system or subsystem called the "plant." Normally, the goal of this control system is to have the output equal numerically to the system input (x) often called the reference input. Wherever a difference (called error $\in$) between the output and reference input is nonzero, the control subsystem or compensator (an electronic subsystem) generates a variable (u) that

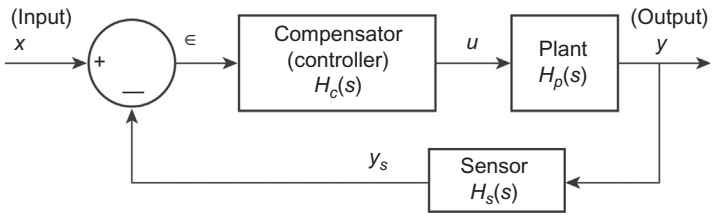


FIG. A.8 Closed-loop system configuration.

causes the plant input to change in such a way as to reduce the error toward zero. The configuration of Fig. A.8 is called a feedback-control system because a measurement of the plant output via a sensor is "fed back" to the input. The topology of the system is such that the signal path back to the input forms a loop (i.e., a closed loop). The sensor has an electrical output that is given here as its output voltage (y_s).

In order that the control variable u can cause a change in the plant output, there must be a component (i.e., an actuator) that receives this electrical input and causes the plant to change its state. Typically, this component is electromechanical in nature. The response of the plant to this electrical input is normally called its "open-loop" response.

The goal of the present discussion is to develop models for the feedback-control system by which its dynamic performance can be analyzed. The performance of the control system is influenced by its system component dynamics. Normally, the control-system designer can optimize closed-loop system performance in some sense by proper design of the compensator/controller. In modern automotive electronic control systems, the compensator is implemented by a microprocessor or microcontroller as explained in various chapters. In such cases, the compensator operation is determined by the program(s) running its microcontroller. However, at this point, it is useful to characterize the entire closed-loop control system as a continuous-time system. Appendix B discusses discrete-time models and digital control techniques.

It is assumed for the present that the models for the various components are known and that each can be characterized by a transfer function. These models are given by the following:

$$\text{error} \quad \in = x - y_s \tag{A.43}$$

$$\text{plant} \quad y(s) = H_p(s)u(s) \tag{A.44}$$

$$\text{compensator} \quad u(s) = H_c(s) \in (s) \tag{A.45}$$

$$\text{sensor} \quad y_S(s) = H_S(s)y(s) \tag{A.46}$$

Combining the models for the components, the following model can be written for the closed-loop system:

$$y(s) = \left(\frac{H_p(s)H_c(s)}{1 + H_s(s)H_p(s)H_c(s)} \right) x(s) \tag{A.47}$$

For convenience (and without serious loss of generality), the sensor transfer function is taken to be unity (i.e., $H_s(s) = 1$) yielding the most familiar form of the closed-loop transfer function $H_{c\ell}(s)$, which is defined as follows:

$$H_{c\ell}(s) = \frac{y(s)}{x(s)}$$

$$H_{c\ell}(s) = \frac{H_p(s)H_c(s)}{1 + H_p(s)H_c(s)} \tag{A.48}$$

It is convenient for simplifying the notation for $H_{c\ell}(s)$ to define the so-called forward-path transfer function that is denoted $H_F(s)$ and is given by

$$H_F(s) = H_p(s)H_c(s)$$

Using the definition of $H_F(s)$, the closed-loop transfer function with a sensor transfer function $H_s(s)$ is given by

$$H_{c\ell}(s) = \frac{H_F(s)}{1 + H_F(s)H_s(s)}$$

Although there is a large class of compensator configuration, there are three main types that have been in widespread use as outlined below:

1. Proportional $u = K_p \in$
2. Proportional-integral (PI) $u = K_p \in + K_I \int \in dt$
3. Proportional-integral-differential (PID) $u = K_p \in + K_I \int \in dt + K_D \frac{d \in}{dt}$

The transfer functions for these three types are

1. P $H_c(s) = K_p$
2. PI $H_c(s) = K_p + \frac{K_I}{s}$
3. PID $H_c(s) = K_p + \frac{K_I}{s} + K_D s$

where K_P is the proportional gain, K_I the integral gain, and K_D the differential gain.

The closed-loop dynamic response is influenced by the type of control via the compensator. Generally, the compensator (or controller) performs a transformation on the error signal to satisfy performance requirements for certain criteria, including

1. transient response characteristics,
2. steady-state errors,
3. disturbance rejection,
4. sensitivity to plant parameter changes over time or with environmental parameter changes (e.g., temperature).

In addition to these criteria, it is also necessary that the closed-loop system be stable in the sense that a bounded input produces a bounded output. By contrast, an unstable system has an output that grows continuously regardless of input until some (typically nonlinear) limit is reached in one of its components or subsystems. The output of such a system at its limit is said to be in saturation.

We consider first the criterion of transient response. The transient response for most closed-loop systems is best represented by its response to a unit step (i.e., its step response). Assuming that the

closed-loop system is stable, it is possible to make several general remarks about the relative step response for various compensator transfer functions. A proportional controller (i.e., $H_c(s) = K_p$) has a steady-state error where this error varies inversely with proportional gain K_p.

Alternatively, a PI compensator that has a transfer function

$$H_c(s) = K_p + \frac{K_I}{s} \tag{A.49}$$

and has a steady-state error of zero

$$\underset{t\to\infty}{Lim} \in (t) = 0$$

provided that the system is stable.

However, for many plants, the addition of an integral term in the compensator component can reduce stability of the closed-loop system relative to that for a proportional-only compensator for sufficiently large integral gain K_I.

The addition of a derivative term to the compensator resulting in a PID controller can improve the transient response in certain respects relative to a PI controller. Typically, it can increase the initial rate of change of the output $\left(\frac{dy}{dt}|_{t=0}\right)$, although it may do so with an overshoot of the final intended value, depending upon the associated gain K_D. The relative benefits of these types of controllers depend upon the particular application and the other system criteria.

We illustrate the influence of the compensator on transient response with the second-order system (see Fig. A.4) introduced above consisting of a mass that is connected to an inertial reference frame by a parallel spring and viscous damper (i.e., a highly simplified model for a suspension system). Acting on this mass is a force $F(t)$ that changes its vertical position $y(t)$. It was shown above that the transfer function of this was shown to be

$$\frac{y(s)}{F(s)} = \frac{1/M_u}{s^2 + 2\zeta\omega_0 s + \omega_0^2} \tag{A.50}$$

where the parameters ω_0 and ζ are given above in the discussion of a second-order system. A closed-loop control system is to be formed in which the vertical position is to be regulated to the reference x by the force that is generated by an actuator. The actuator model is

$$F = K_a u$$

where K_a is the actuator constant and u is the control signal from the compensator. We consider, initially, a proportional compensator having the following model:

$$u = K_p \in$$

Where

$$\in = \text{error}$$

$$= x - y$$

The open-loop transfer function of this plant $H_p(s) = y(s)/u(s)$ is given by

$$H_p = \frac{K_a/M_u}{s^2 + 2\zeta\omega_0 s + \omega_0^2} \tag{A.51}$$

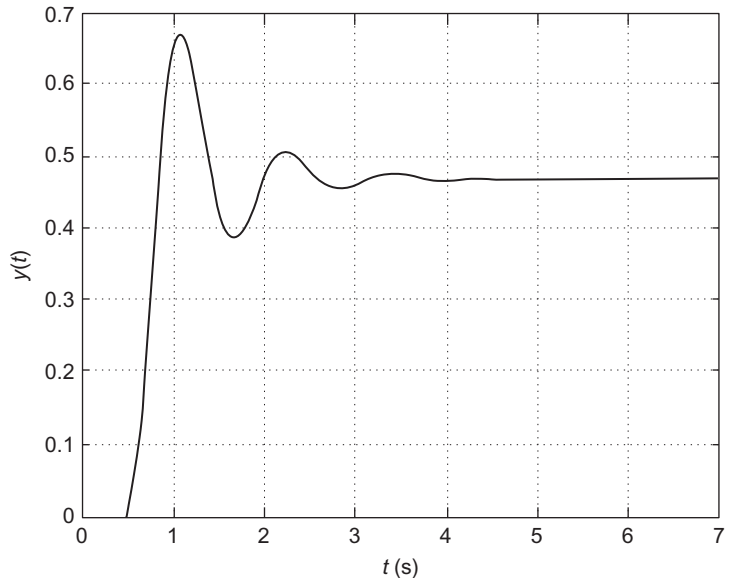


FIG. A.9 Unit step response of example open-loop system.

In order to better understand the performance of a closed-loop system, it is helpful to consider the open-loop response. We illustrate with the second-order system as the plant. We assume that the actuator has a gain $K_a = 23.5$. Using the parameters from the example second-order system, the step response to unit step input $u(t)$ at $t = 0.5$ sec is shown in Fig. A.9.

The closed-loop transfer function (assuming that $H_s(s) = 1$) is given by

$$H_{c\ell} = \frac{y(s)}{x(s)} \tag{A.52}$$

$$= \frac{K_p H_p(s)}{1 + K_p H_p(s)} \tag{A.53}$$

$$\frac{K_a K_p}{M_u \left[s^2 + 2\zeta\omega_0 s + \omega_0{}^2 + \dfrac{K_a K_p}{M_u} \right]} \tag{A.54}$$

Using different parameters ($K_a = 50$, $K = 53.3$, $D = 17.6$, and $M_u = 1.71$) than the parameters from the earlier example, it is possible to evaluate the step response via simulation using MATLAB/SIMULINK. Fig. A.10 is a plot of the unit step response to a command input step for two values of the proportional gain. The practical automotive application for this example could be a commanded change in the vehicle height above the ground in an electronically controlled suspension system (see chapter on motion control). This plot shows that the steady-state unit step (at 0.5 sec)-response error varies inversely with proportional gain. Note, however, that the damping of the closed-loop system is decreased with increasing gain (K_p) as is shown by the greater overshoot of $y(t)$ for $K_p = 10$ relative to that for $K_p = 5$.

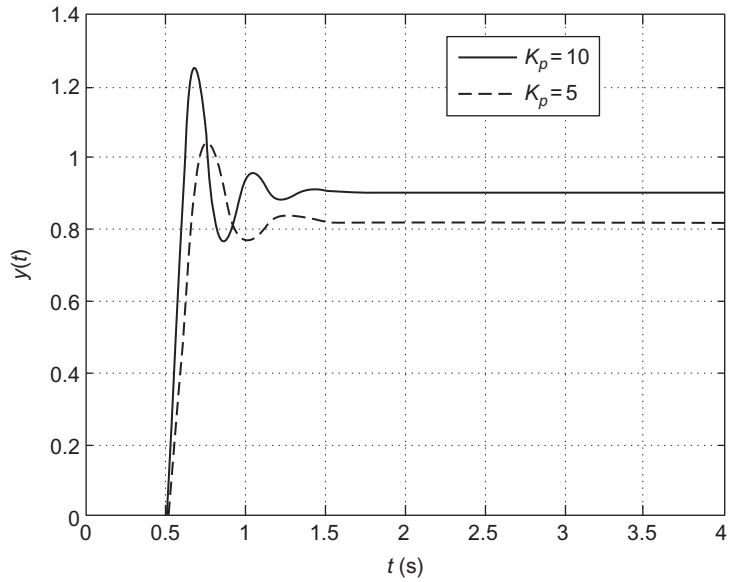


FIG. A.10 Response of proportional feedback-control system.

Consider next the step response of a PI controller. An integral term $\left(K_I \int \in dt\right)$ was added to the controller of the previous example. The closed-loop transfer function for this system is given by

$$H_{c\ell}(s) = \frac{K_a\left(K_p s + K_I\right)/M_u}{\left(s^2 + 2\zeta\omega_0 s + \omega_0^2\right) s + \left(K_p s + K_I\right)\dfrac{K_a}{M_u}} \tag{A.55}$$

Using the same system parameters as in the proportional control, the response to a unit step at $t = 0.5$ sec can be found using MATLAB/SIMULINK. Generally speaking, the addition of an integral term in the controller forces the steady-state error toward zero. Fig. A.11 is a plot of the unit step-response error for $K_p = 5$ and $K_I = 15$. Note also that the overshoot (about 25%) is greater than the proportional controller (about 12.5%) for the same K_p. However, the steady-state error asymptotically approaches zero as predicted above.

A further improvement to the dynamic response of a control system is possible by including a derivative term:

$$K_D \frac{d \in}{dt}$$

such that the compensator transfer function is given by

$$H_c(s) = K_p + \frac{K_I}{s} + K_D s \tag{A.56}$$

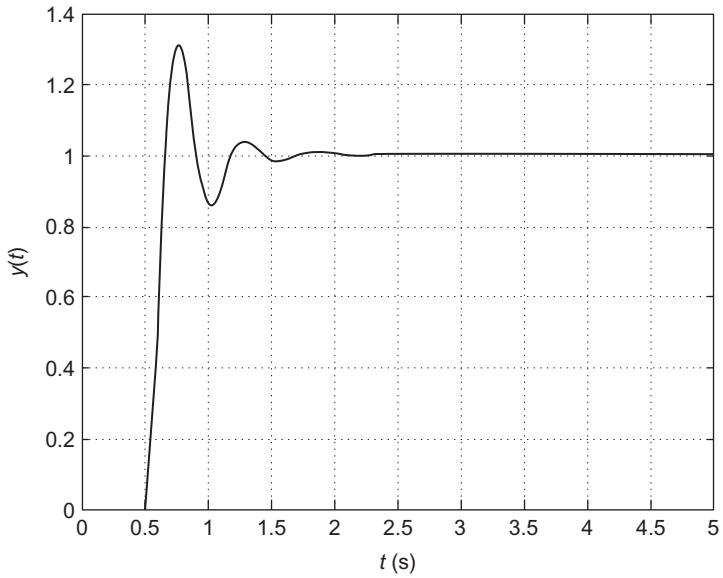


FIG. A.11 Response of proportional-integral feedback-control system.

Fig. A.12 is a plot of the unit step (at $t=0.5$ sec) response of the PID closed-loop system having the same second-order system for the plant as for the P and PI controls. In the simulation for the figure, $K_p=5$, $K_I=20$, and $K_D=0.3$.

This closed-loop system has the zero steady-state error properties of PI closed-loop system but has very little overshoot and very rapid response to any dynamic input. The above examples clearly show that PID control has the potential for excellent closed-loop dynamic response. It is left as an exercise for the interested reader to show that the closed-loop system is close to critical damping (i.e., the fastest rise time without overshoot) for $K_D=0.5$, $K_p=5$, and $K_I=15$.

Caution must be exercised by the control-system design in choosing the type of controller and the gains. Certain values of these gains can adversely affect the closed-loop system relative to both its open-loop response or other gain choices. In the extreme case, the controller, if poorly designed, can yield an unstable closed-loop system as will be discussed in the next section of this appendix.

STABILITY OF CONTROL SYSTEM

One of the most important issues concerning the practical utility of any control system is its stability. Simply put, a control system is stable if a bounded input results in a bounded output. What this means effectively is that a system will not "run away" on its own. In most cases, the output of an unstable system will grow in amplitude until some physical limitation ceases its growth.

However, the stability of a closed-loop control system does not necessarily require that the plant being controlled is, itself, stable. In fact, a control system can, in certain cases, stabilize an unstable plant such as in the case of many modern fighter aircraft (e.g., F-16). However, the application of a stabilizing control system for an unstable plant is rare in automotive applications.

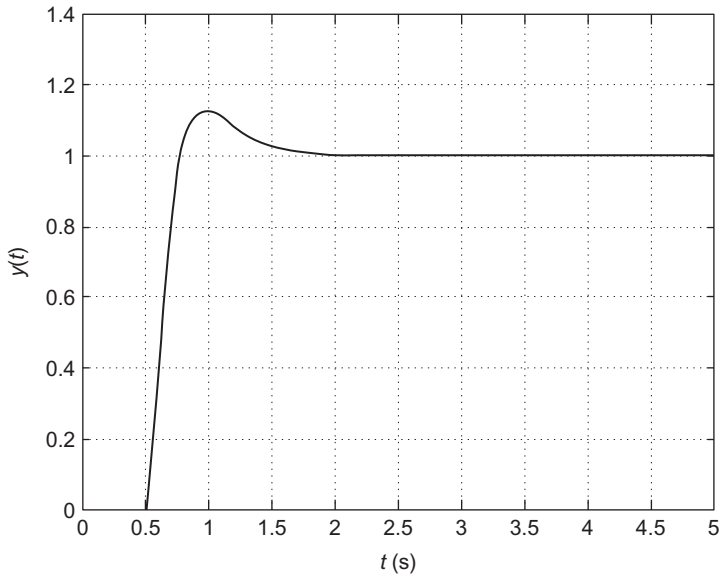


FIG. A.12 Response of PID control system.

Modern control methodology offers the system designer many important tools for assessing the stability of a control system. Commercially available software provides a control-system designer the capability of rapidly assessing the stability of a candidate control system provided a relatively robust linear mathematical model is available for the system. It is beyond the scope of this book to cover all techniques for evaluating control-system stability. However, we present one of the important techniques considered here called the root-locus technique. It is important in any control-system application and especially so in automotive systems to have some margin of stability to insure that the system remains stable even if some system parameters change with time over the vehicle lifetime. One way of evaluating the robustness of a control system to parameter variations is through an analysis technique called gain and phase margin (PM), which is also explained below.

ROOT-LOCUS TECHNIQUES

We begin with a brief survey of root-locus techniques. A root locus of a control system is a plot of the poles of its closed-loop transfer function as some system parameter is varied. It has been shown that the closed-loop transfer function for a plant having open-loop transfer function $H_p(s)$ being controlled by a controller having transfer function $H_c(s)$ and assuming an ideal sensor (i.e., $H_s(s) = 1$) is given by

$$H_{cl} = \frac{H_c(s)H_p(s)}{1 + H_c(s)H_p(s)} \tag{A.57}$$

The poles of this transfer function are the zeros (in the complex s-plane) of the function:

$$1 + H_c(s)H_p(s) = 0 \tag{A.58}$$

In order to assess the influence of a parameter K on system stability, the above equation must be rewritten in the following form:

$$1 + KG(s) = 0 \tag{A.59}$$

This expression is known as the characteristic equation for the closed-loop system. The root locus for this system is the locus in the complex s-plane of the zeros of the equation as a function of the parameter K. Once in this form, the root locus is readily obtained using the MATLAB rlocus [$G(s)$] function. In practice, it is convenient to write $G(s)$ as a ratio of functions $N(s)$, numerator polynomial, and $D(s)$, denominator polynomial:

$$\begin{aligned} 1 + KN(s)/D(s) &= 0 \\ \text{or } \; D(s) + KN(s) &= 0 \end{aligned} \tag{A.60}$$

The root-locus function finds and plots the roots of this equation on a complex-valued polar graph. For the system to remain stable, none of the roots can be in the right half of this complex-plane plot. Any system having one or more roots on the imaginary axis are neutrally stable, a condition that cannot be tolerated in any automotive system that can affect vehicle stability or occupant safety.

As an example of the use of root locus, we consider the second-order system plant depicted in Fig. A.4 and having transfer function given by Eq. (A.51) with a PID controller. The transfer function for this controller is given by

$$H_c(s) = K_p + \frac{K_I}{s} + K_D s \tag{A.61}$$

which can be rewritten in the form

$$H_c(s) = \frac{K_p}{s}\left(\frac{K_D}{K_p}s^2 + s + \frac{K_I}{K_p}\right) \tag{A.62}$$

The parameter K in the general formula given above for root locus is taken to be K_p. In this case, it is possible to examine closed-loop stability as K is varied while keeping K_I/K_p and K_D/K_p fixed. It can be shown for this system that the $G(s)$ needed for the root locus is given by

$$G(s) = \frac{\left[\left(\frac{K_D}{K_p}s^2 + s + \frac{K_I}{K_p}\right)\right]}{s\left(s^2 + 2\zeta\omega_0 s + \omega_0^2\right)}\frac{K_a}{M_u} \tag{A.63}$$

The root locus for the system for variations in the parameter K with fixed ratios

$$\frac{K_D}{K_p} = 0.05$$

$$\frac{K_I}{K_p} = 1.5$$

is given in Fig. A.13.

The closed-loop transfer function for this example is a cubic function of s so that there are three roots to the characteristic equation. The root loci are the paths indicated by the solid curves beginning at the roots for $K = 0$ indicated in the figure by asterisks. As the gain increases, the roots move from these initial values as shown. In this root-locus plot, the dashed straight lines from the origin are loci of

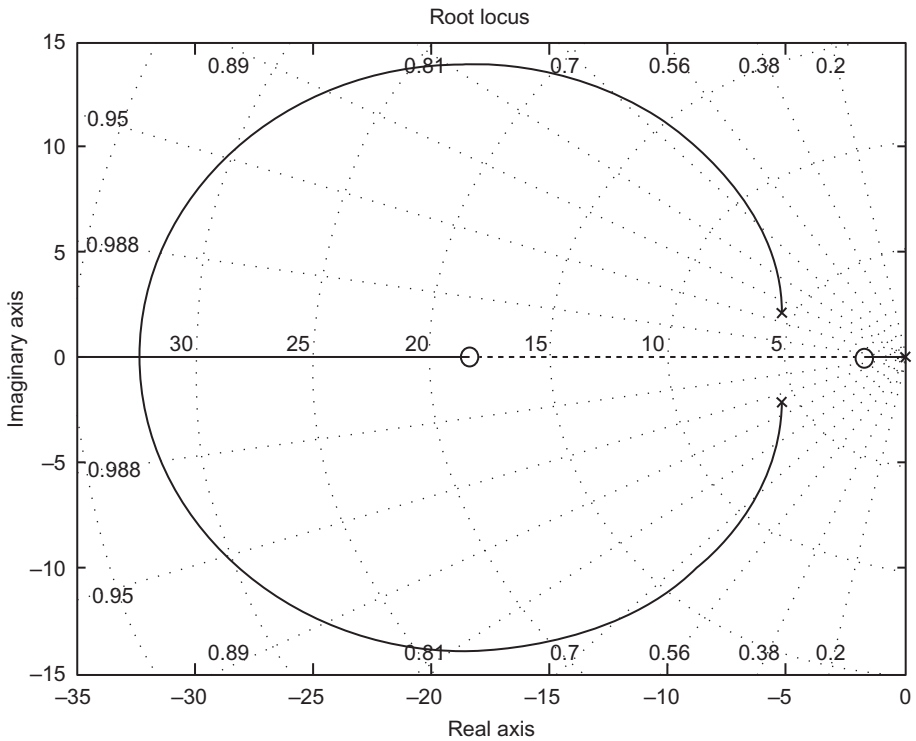


FIG. A.13 Root-locus plot, for example, PID control system.

constant pole damping ratio, and the dashed circles about the origin represent constant magnitude of poles. This figure shows that the closed-loop poles remain in the left half of the complex s-plane $\forall K_p$ for which the system is stable. The closed-loop dynamic response of this system can be altered (within certain limits) by suitable gain choices.

The root-locus technique can be used to assist the design of the controller particularly if the specifications for the closed-loop system performance include placing the corresponding poles in certain regions of the complex plane. For example, the damping ratio or step-response overshoot might be specified; the gains can be chosen such that the closed-loop system has poles in locations that satisfy the requirements.

As explained at the beginning of this section, there are many techniques in addition to root locus for assessing the stability of a linear closed-loop system. It is beyond the scope of this book to cover all these other techniques. Rather, there are many excellent texts that cover these subjects in great detail.

ROBUSTNESS OF CONTROL-SYSTEM STABILITY

However, another important issue in the stability of a closed-loop system is the robustness of stability to system parameter changes. Such changes occur in practice because typically the linear models used to design or to conduct performance analyzes of a closed-loop system are linearized approximations

to a nonlinear model for the actual system in the neighborhood of an operating point. Changes in the operating point normally require changes to the parameters of the linearized approximation to the actual model. A closed-loop system whose controller was designed/optimized at one operating point and found to be stable there may not be stable at other operating points. One important method of assessing the relative stability of a closed-loop system is based upon an evaluation of the characteristic equation (i.e., Eq. A.59) for a SSS excitation. For any given set of controller gains, the characteristic equation can be written as

$$1 + L(s) = 0 \tag{A.64}$$

where

$$L(s) = H_c(s)H_p(s) \tag{A.65}$$

and where $L(s)$ is called the loop gain. For SSS excitation, the characteristic equation is given by

$$1 + L(j\omega) = -1 \tag{A.66}$$

Instability occurs whenever

$$L(j\omega) = -1 \tag{A.67}$$

or $|L(j\omega)| = 1$
and $\angle L(j\omega) = -180^{\circ}$

That is, both the magnitude and phase conditions must be satisfied for closed-loop instability.

The relative stability of a closed-loop system is found in terms of a pair of frequencies defined as the gain crossover frequency (ω_G) and the phase crossover frequency (ω_p) where

$$\log |L(j\omega_G| = 0 \tag{A.68}$$

$$\angle L(j\omega_p) = -180^{\circ}$$

The relative stability of a closed-loop system is expressed by two quantities, (1) gain margin (GM) and (2) PM. The GM is defined as

$$\text{GM} = -20 \log_{10} |L(j\omega_p)| \tag{A.69}$$

It is the amount of gain (in dB) that must be added to the system at the phase crossover frequency for the magnitude $L(j\omega_p)$ to be unity. The PM is defined as

$$\text{PM} = \angle L(j\omega_G) + 180^{\circ} \tag{A.70}$$

The gain and PM for any closed-loop system configuration can readily be found from the Bode plot of the loop gain $L(j\omega)$. We illustrate gain and PM using the example second-order system with PID controller. For this illustration, the gain parameters are picked to be $K_p = 1.0$, $K_I = 1.5$, and $K_D = 0.05$. These gains have the same ratios $\dfrac{K_D}{K_p}$ and $\dfrac{K_I}{K_p}$ as in the r-locus example. The Bode plot for this example is given in Fig. A.14.

The gain crossover (i.e., loop gain = 0 dB) in this example is at about 2 rad/s. There is no phase crossover as the phase never goes more negative than about $-110°$. This system has an infinite GM and a PM of about 110°. Any system with these GM and PM is highly stable and will remain stable with respect to relatively large system parameter variations. It can be seen from Fig. A.24 that this set of gains yields the largest PM. If these gains are all increased by a factor of 10, the gain crossover

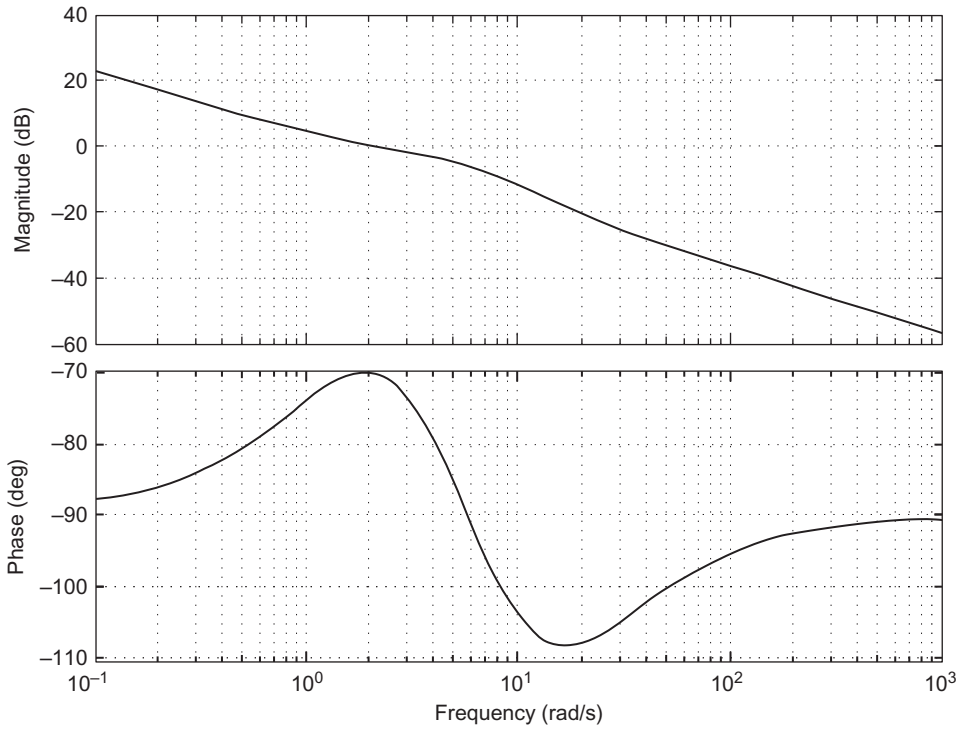


FIG. A.14 Bode plot, for example, PID control system.

frequency is approximately 2 rad/sec. This set of gain gives nearly the smallest PM (i.e., 70°) that can be shown as an exercise for the interested reader.

In the design of a control system, a general rule of thumb is that the margins should satisfy the following:

$$\mathrm{GM} \gtrsim 10\mathrm{db}$$

$$\mathrm{PM} \gtrsim 40^{o}$$

These margins are normally sufficient to provide robust stability with respect to system parameter changes. In the event that the system plant model is obtained by linearizing a nonlinear model for certain operating regions, then the GM and PM probably should be larger than the rule of thumb given above.

CLOSED-LOOP LIMIT-CYCLE CONTROL

Another type of control that is used in automotive applications is limit-cycle control. Limit-cycle control is a type of feedback control that monitors the system's output and responds only when the output goes beyond preset limits. Limit-cycle controllers often are used to control plants with nonlinear or complicated transfer functions.

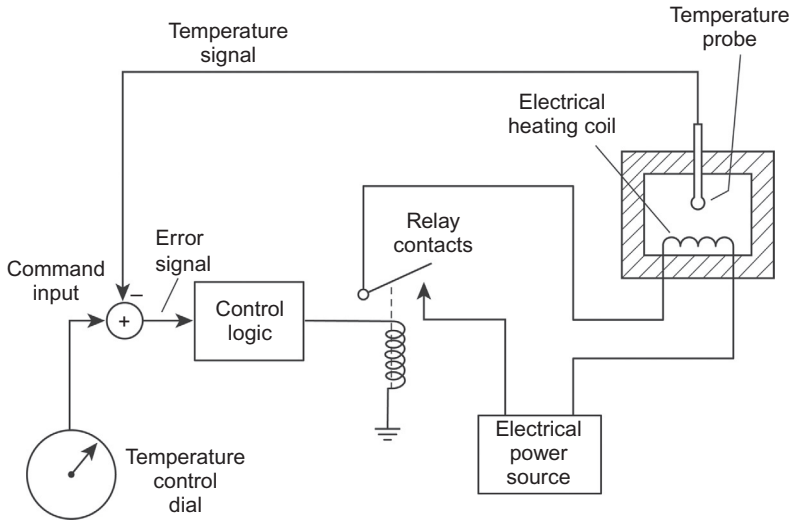


FIG. A.15 Example of limit-cycle control system.

Limit-cycle control responds only when the error is outside a pair of limits. An example of a limit-cycle controller is the temperature-controlled oven depicted in Fig. A.15.

The temperature inside the oven is controlled by the length of time the heating coil is energized. The temperature of the oven is measured with a temperature probe, and the corresponding electrical signal is fed back to the command to obtain an error signal. The control electronics checks the error signal against the temperature control dial to determine if one of the following two conditions exists:

1. Oven temperature is below minimum setting of command input.
2. Oven temperature is above maximum setting of command input.

The control electronics responds to error condition 1 by closing the relay contacts to energize the heating element. This causes the temperature in the oven to increase until the temperature rises above a maximum limit, producing error condition 2. In this case, the control electronics opens the relay contacts, and the heat is turned off. The oven gradually cools until condition 1 again occurs and the cycle repeats. The oven temperature varies between the upper and lower limit, and the variations can be graphed as a function of time, as shown qualitatively in Fig. A.16.

The amplitude of the temperature variations, called the *differential*, can be decreased by reducing the difference between the maximum and minimum temperature limits that are set in the controller. As the limits get closer together, the temperature cycles more rapidly (frequency increases) to hold the actual temperature deviations closer to the desired constant temperature than for larger limits. Thus, the limit-cycle controller controls the system to maintain an average value close to the command input yet cycles above and below the desired value. This type of controller has gained popularity due to its simplicity, low cost, and ease of application. Fuel control, one of the most important automotive electronic control systems, is, at least partially, a limit-cycle control system (see Chapters 4 and 6).

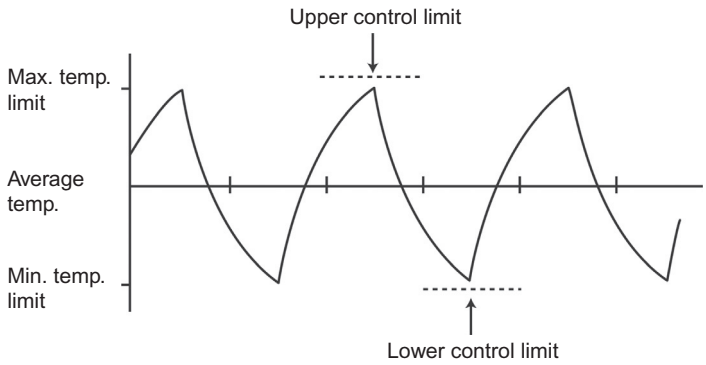


FIG. A.16 Frequency response for limit-cycle control system.

A limit-cycle control system is not linear and is, consequently, not amenable to analysis by the linear techniques described above. Rather, the performance of any given limit-cycle control system is best accomplished via simulation.

INSTRUMENTATION

An instrument (or instrumentation system) is a device for measuring some specific quantity. Automotive instruments have traditionally been mechanical, pneumatic, hydraulic, electrical, or combinations of these. However, modern automotive instrumentation is largely electronic. These electronic instruments or instrumentation systems are used to measure a variety of physical quantities, including the following:

1. Vehicle speed
2. Total distance traveled
3. Engine angular speed (rpm)
4. Fuel quantity and/or flow rate
5. Oil pressure
6. Engine coolant temperature
7. Alternator charging current and/or voltage
8. Tire pressure
9. Estimated range to empty fuel tank

In addition to providing the driver with measurements of important variables, measurements of variables are sometimes made to assist in the diagnosis of problems with various subsystems. A typical automotive digital electronic system monitors measurements of certain variables to assess whether or not they fall within an allowed band. In the event that a variable is out of tolerance, a warning error message is stored in memory. If this out-of-bound variable is capable of affecting the normal vehicle operation, a warning message (e.g., “check engine”) is displayed to the driver. This diagnostic application of instrumentation is discussed in detail in Chapter 11.

For an understanding of measurement instrumentation, it is helpful to review a definition of measurement. Automotive instrumentation systems, whether electronic, mechanical, or a combination of both, measure a physical quantity and provide a numerical value report of that measurement to the driver (or sometimes to the maintenance technician).

MEASUREMENT

A *measurement* is defined as a numerical comparison of an unknown magnitude of a given physical quantity to a standard magnitude of the same physical quantity. In this sense, the result of a measurement is normally a numerical value expressing the indicated value of the measurement as a multiple of the appropriate standard. However, other display devices are possible in which simple messages are given. For example, it is a common practice not to provide a display of measured values for engine oil pressure or coolant temperature. Warning lamps are activated by the electronic instrumentation system whenever oil pressure is too low or coolant temperature is too high.

ISSUES

In any measurement made with any instrument, there are several important issues, including the following:

1. Standards
2. Precision
3. Calibration
4. Accuracy
5. Errors
6. Reliability

Each of these issues has an important impact on the performance of the instrumentation.

The *standard* magnitudes of the physical variables measured by any instrument are maintained by the National Institute of Standards and Technology in the United States. These standard magnitudes and the fundamental relationships between physical variables determine the units for each physical quantity. Contemporary vehicles often use metric standards known as the meter, kilogram, and second (MKS) system along with some English units (e.g., mph).

The *precision* of any instrument is related to the number of significant figures that is readable from the display device. The greater the number of significant figures displayed, the greater the precision of the instrument.

Calibration is the act of setting the parameters of an instrument such that the indicated value conforms to the true value of the quantity being measured.

The *accuracy* of any measurement is the conformity of the indicated value to the true value of the quantity being measured. *Error* is defined as the difference between true and indicated values. Hence, accuracy and error vary inversely. The required accuracy for automotive electronic systems varies with application, as is shown in various chapters. In general, those instruments used solely for driver information (e.g., fuel quantity) might have lower accuracy requirements than those used for applications such as engine control or diagnosis.

The errors in any measurement are generally classified as either systematic or random. *Systematic* errors result from known variations and imperfections in instrument performance, for which corrections can be made if desired. There are many sources of systematic error, including limited dynamic response to rapidly changing variables, temperature variations in calibration, and loading. Since virtually any component in an instrument is potentially susceptible to temperature variations, great care must be exercised in instrument design to minimize temperature variations in calibration. As is shown in Chapter 8, some automotive instruments have relatively low precision and accuracy requirements, so that temperature variations in calibration are negligible. *Random* errors are essentially random fluctuations in indicated value for the measurement. Most random measurement errors result from noise from various sources as explained later in this appendix.

SYSTEMATIC ERRORS

One example of a systematic error is known as *loading* errors, which are due to the energy extracted by an instrument when making a measurement. Whenever the energy extracted from a system under measurement is not negligible, the extracted energy causes a change in the quantity being measured. Wherever possible, an instrument is designed to minimize such loading effects. The idea of loading error can be illustrated by the simple example of an electrical measurement, as illustrated in Fig. A.17.

A voltmeter M having resistance R_m measures the voltage across resistance R. The correct voltage (v_c) is given by

$$v_c = V\left(\frac{R}{R+R_1}\right) \tag{A.71}$$

However, the measured voltage v_m is given by

$$v_m = V\left(\frac{R_p}{R_p+R_1}\right) \tag{A.72}$$

where R_p is the parallel combination of R and R_m:

$$R_p = \frac{RR_m}{R+R_m} \tag{A.73}$$

Loading is minimized by increasing the meter resistance R_m to the largest possible value. For conditions where R_m approaches infinite resistance, R_p approaches resistance R, and v_m approaches the

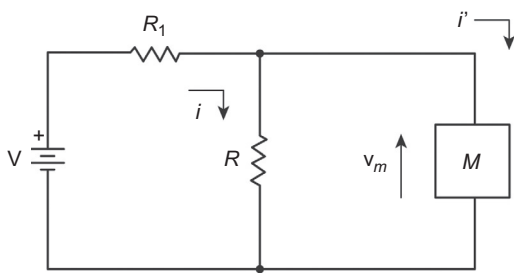


FIG. A.17 Illustration of loading error voltmeter.

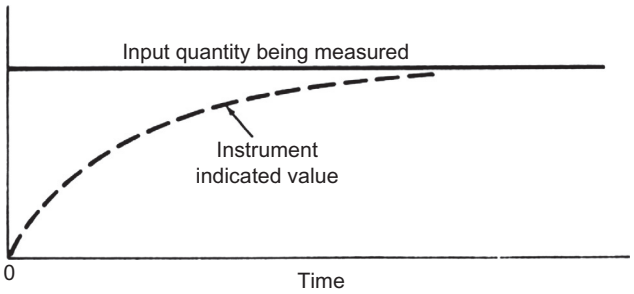


FIG. A.18 Illustration of instrument dynamic response error.

correct voltage. Loading is similarly minimized in the measurement of any quantity by minimizing extracted energy. Normally, loading is negligible in modern instrumentation.

Another significant systematic error source is the *dynamic response* of the instrument. Any instrument has a limited response rate to very rapidly changing input, as illustrated in Fig. A.18. In this illustration, an input quantity to the instrument changes abruptly at some time. The instrument begins responding, but cannot instantaneously change and produce the new value. After a transient period, the indicated value approaches the correct reading (presuming correct instrument calibration). The dynamic response of an instrument to rapidly changing input quantity varies inversely with its bandwidth as illustrated earlier in this appendix with respect to the step response of a first order in system for which the response time as characterized by τ is inversely proportional to the pole (a_o/a_1). It is left as an exercise for the interested ready to show that the frequency band (called bandwidth) for which the first-order system response remains close to the dc response is inversely proportional to the time constant τ.

In many automotive instrumentation applications, the bandwidth is purposely reduced to avoid rapid fluctuations in readings. For example, the type of sensor used for fuel-quantity measurements actually measures the height of fuel in the tank with a small float. As the car moves, the fuel sloshes in the tank, causing the sensor reading to fluctuate randomly about its mean value. The signal processing associated with this sensor is actually a low-pass filter such as is explained later in this appendix and has an extremely low bandwidth so that only the average reading of the fuel quantity is displayed, thereby eliminating the undesirable fluctuations in fuel-quantity measurements that would occur if the bandwidth were not restricted. This example is quantitatively examined in Chapter 8.

The *reliability* of an instrumentation system refers to its ability to perform its designed function accurately and continuously whenever required, under unfavorable conditions, and for a reasonable amount of time. Reliability must be designed into the system by using adequate design margins and quality components that operate both over the desired temperature range and under the applicable environmental conditions.

BASIC MEASUREMENT SYSTEM

The basic block diagram for an electronic instrumentation system has been given in Fig. A.1B. That is, each system has three basic components: sensor, signal processing, and display. Essentially, all electronic measurement systems incorporated in automobiles have this basic structure regardless of

the physical variable being measured, the type of display being used, or whether the signal processing is digital or analog.

Understanding automotive electronic instrumentation systems is facilitated by consideration of some fundamental characteristics of the three functional components. Again, it should be noted that automotive electronic systems are essentially digital rather than analog realization. Modeling and analysis of digital electronic systems are in terms of discrete time. Such modeling/analysis is discussed in Appendix B. However, instrument systems often incorporate analog (continuous time) sensors. Consequently, for the remainder of this appendix, all variables are expressed in terms of continuous-time models.

SENSOR

A *sensor* is a device that converts energy from the form of the measurement variable to an electrical signal. A large portion of Chapter 5 is devoted to explaining, modeling, and conducting performance analyses of various sensors. An ideal analog sensor generates an output voltage that is proportional to the quantity q being measured:

$$v_s = K_s q \tag{A.74}$$

where K_s is the sensor calibration constant.

By way of illustration, consider a typical automotive sensor—the throttle position sensor. The quantity being measured is the angle (θ) of the throttle plate relative to closed throttle. Assuming for the sake of illustration that the throttle angle varies from 0 to $\theta_{\max}$ and the voltage varies from 0 to 5 V, the sensor calibration constant K_s is

$$K_s = \frac{5}{\theta_{\max}}$$

Alternatively, a sensor can have a digital output, making it directly compatible with digital signal processing. For such sensors, the output is an electrical equivalent of a numerical value, using a binary number system as explained in Chapter 2. Fig. A.19 illustrates the output for such a sensor. There are N output leads, each of which can have one of two possible voltages, representing a 0 or 1. In such an arrangement, 2^N possible numerical values can be represented. For automotive applications, N has traditionally ranged from 8 to 16, corresponding to a range of 256–65,536 numerical values, although N is larger than 16 in most contemporary digital systems. Digital instruments belong to the class of discrete-time systems, which are discussed in detail in Chapter 8.

Of course, a sensor is susceptible to error just as in any system or system component. Potential systematic error sources include loading, finite dynamic response, calibration shift, and nonlinear behavior. Often, it is possible to compensate for these and other types of errors in the electronic signal-processing unit of the instrument. If a sensor has limited bandwidth, it will introduce errors when measuring rapidly changing input quantities. Fig. A.20 illustrates such dynamic errors for an analog sensor measuring an input that abruptly changes between two values (this type of input is said to have a *square-wave* waveform). Fig. A.20A depicts a square-wave input to the sensor. Fig. A.20B illustrates the response that the sensor will have if its bandwidth is too small. Note that the output does not respond to the instantaneous input changes. Rather, its output changes gradually, slowly approaching the correct value. The response of a first-order sensor is quantitatively given earlier in this appendix (e.g., see Fig. A.3).

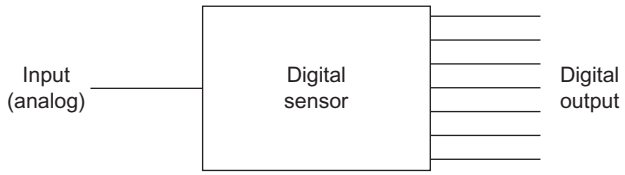


FIG. A.19 Analog input and digital output sensor.

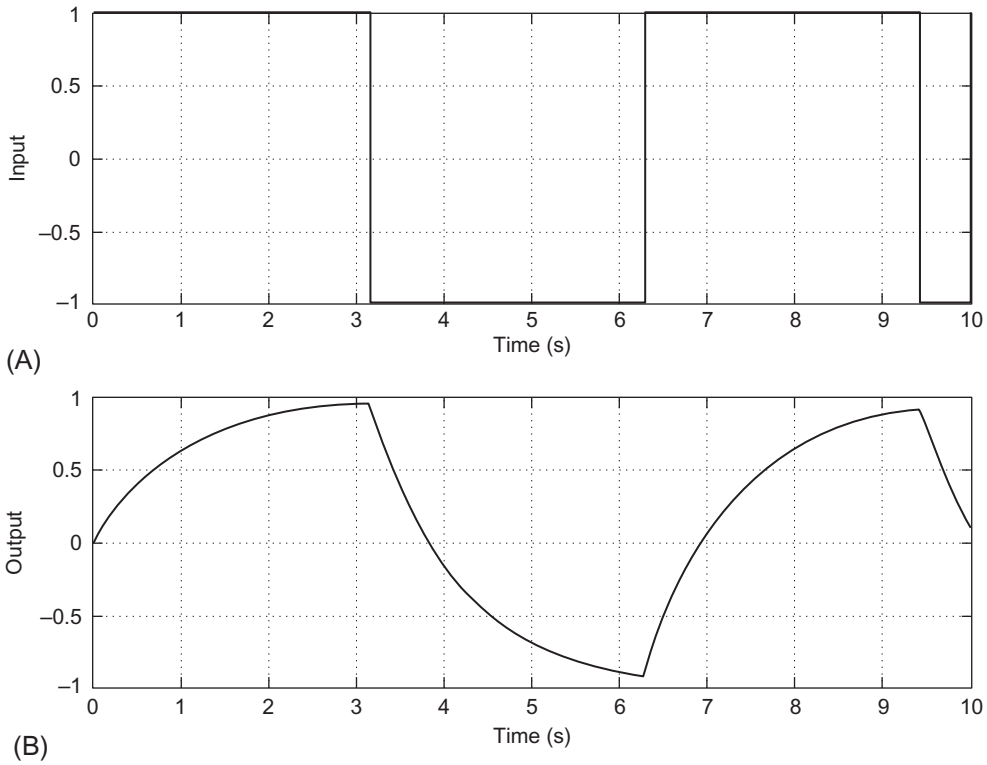


FIG. A.20 Instrument square-wave dynamic response error. (A) Instrument exemplary input and (B) illustrative instrument response (e.g., for first-order instrument).

An ideal sensor has a *linear transfer characteristic* (or transfer function), as shown in Fig. A.21A. However, often the sensor output voltage is a nonlinear function of the quantity being measured (i.e., $v_o(q)$ is nonlinear). Signal processing can be used to linearize the output signal so that it will appear as if the sensor has a linear transfer characteristic, as shown in the dashed curve of Fig. A.21B. Sometimes, a nonlinear sensor may provide satisfactory operation without linearization if it is operated in a particular "nearly" linear region of its transfer characteristic (Fig. A.21B). Moreover, with digital signal processing, a simple calculation can be used to "correct" any nonlinearities of a given sensor, yielding a correct value of the variable being measured. This signal processing would perform the nonlinear

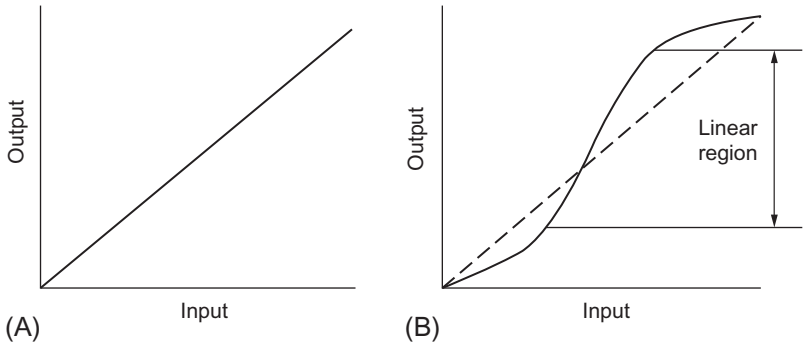


FIG. A.21 Linearization of nonlinear sensor. (A) Linear and (B) nonlinear.

correction by suitable calculation on the data from the sensor output. Such type of correction calculation is best done with digital instruments, which are discussed in Appendix B.

RANDOM ERRORS

Random sensor errors can occur due to external noise sources (i.e., random fluctuations in the quantity being measured) or due to internal noise sources. Random errors generated internally in sensors are caused primarily by internal electrical noise. Internal electrical noise can be caused by molecular vibrations due to heat (thermal noise) or random electron movement in semiconductors (shot noise). In certain cases, a sensor may respond to quantities other than the quantity being measured. For example, the output voltage of a sensor that measures a given physical quantity may include random error in the form of an electrical noise. Any random component of the quantity being measured is termed "process noise" (e.g., road roughness induced random acceleration in an accelerometer). The sensor output due to process noise is in the form of an electrical random process.

Electrical noise is a random process, which can only be meaningfully modeled statistically. Typically, sensor electrical noise voltage v_n is essentially a stationary random process meaning its statistics are time invariant. For most noise sources encountered in automotive electronic sensors, the amplitude statistics are given by the Gaussian probability density function $p(v_n)$:

$$p(v_n) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{v_n}{\sigma}\right)^2} \tag{A.75}$$

where σ is the standard deviation of v_n.

The spectral statistics are given by the so-called power spectral density $W(f)$. The power spectral density essentially models the distribution of the power per unit bandwidth versus frequency f for the noise source. The power spectral density can be determined from a sample $v_T(t)$ of $v_n(t)$ where

$$\begin{aligned} v_T(t) &= v_n(t) \quad 0 \le t \le T \\ &= 0 \quad \text{elsewhere} \end{aligned} \tag{A.76}$$

The spectrum of $v_T(t)$ is given formally by its Fourier transform $V_T(j\omega)$:

$$V_T(j\omega) = \int_{-\omega}^{\infty} e^{-j\omega t} v_T(t)\,dt \tag{A.77}$$

The power spectral density is given (with sufficient accuracy) by the following:

$$W(f) = \underset{T\to\infty}{Lim}\left(\frac{|V_T(j2\pi f)|^2}{T}\right) \tag{A.78}$$

where $f = \omega/2\pi$

Any practical sensor has finite dynamic response. Depending on the origin of the noise in the sensor, a potential noise model for such a sensor is shown in Fig. A.22. For this figure and model, it is assumed that the noise is associated with the quantity being measured (i.e., so-called process noise).

In this figure, $H_s(j\omega)$ is the SSS frequency response for the sensor that expresses its dynamic response to the noise random process. This frequency response can be found using the linear system modeling given earlier in this appendix. The "white noise source" is an ideal noise source having a constant (W_o) power spectral density (W_w):

$$W_w(f) = W_o \quad \forall f \tag{A.79}$$

The power spectral density of v_n (i.e., $W_n(f)$) is given by

$$W_n(f) = W_o|H_s(j\,2\pi f)|^2 \tag{A.80}$$

In this noise model, the amplitude (W_o) of the basic noise source depends upon the physical origin of the noise. For example, noise is generated in any resistance R at absolute temperature T_o (i.e., thermal noise) that has power spectral density given by

$$W_o = 4kT_oR$$

where k is the Boltzmann constant. Similar models exist for electronic noise that is generated by a current flowing through a semiconductor junction. The "amplitude" of the sensor output noise is best represented by its rms value $\tilde{v}_n$, which is given by

$$\tilde{v}_n = \left[\int_o^\infty W_n(f)df\right]^{\frac{1}{2}} \tag{A.81}$$

$$= \left[W_o\int_o^\infty |H_s(j2\pi f)|^2 df\right]^{\frac{1}{2}}$$

Thus, the noise amplitude of any sensor having a noise model depicted in Fig. A.22 is proportional to the sensor bandwidth.

On the other hand, noise can be generated at any point in a sensor configuration (including output resistance). In this case, the sensor output power spectral density may well occupy a spectral bandwidth

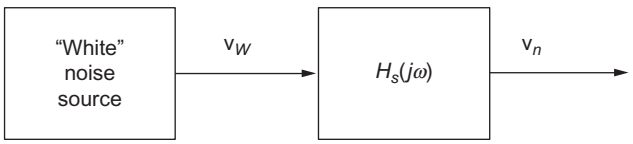


FIG. A.22 Sensor noise model.

that is large compared with the sensor bandwidth or the desired spectrum of the quantity being measured. In this case, the noise amplitude may be reduced by signal processing. Such signal processing takes the form of a filter. As will be shown in the next section of this appendix, a filter can be designed that leaves the signal component of sensor output essentially unchanged yet reduces the rms noise voltage. Such filtering improves the signal-to-noise ratio, which is always desirable in any measurement.

To be useful for measurement purposes, an electronic instrumentation system must somehow make the results of measurement available to the user. This is done through the display, which yields numerical values to the user. As in other aspects of electronic systems, the display can be analog or digital. Both types of displays are described in detail in Chapter 8. As stated earlier, in automotive applications, a "display" is often just a warning (e.g., lamp) to the driver of an out-of-tolerance value for a given variable or parameter.

Automotive display devices, which are either analog or digital, provide a visual indication of the measurements made by the sensors. Actuators convert electrical inputs to an action such as a mechanical movement. Displays, like sensors, are energy-conversion devices. They have bandwidth, dynamic range, and calibration characteristics and, therefore, have the same types of errors as do sensors. As with sensors, many of the shortcomings of display devices can be reduced or eliminated through the imaginative use of signal processing.

SIGNAL PROCESSING

Instrumentation signal processing, as defined earlier, is any operation that is performed on signals traveling between the sensor and the display. Signal processing converts the sensor signal to an electrical signal that is suitable to drive the display. In addition, it can increase the accuracy, reliability, or readability of the measurement. Signal processing can make a nonlinear sensor appear linear, or it can smooth a sensor's frequency response. Signal processing can be used to perform unit conversions such as converting from miles per hour to kilometers per hour. It can perform display formatting (such as scaling and shifting a temperature sensor's output so that it can be displayed on the engine temperature gauge either in centigrade or in Fahrenheit) or process signals in a way that reduces the effects of random system errors.

Signal processing can be accomplished with either a digital or an analog subsystem. Contemporary vehicles employ digital instrumentation that means that the electronic signal processing is accomplished with a digital computer.

FILTERING

One of the most important signal-processing operations in instrumentation is filtering. As explained above, filtering can improve the signal/noise, which enhances measurement accuracy. In the next section, we discuss filter types and design methods applicable in instrumentation.

In linear continuous-time instruments, electronic signal processing of the electrical output of a sensor can perform many types of operations including (1) arithmetic, (2) integration, and (3) differentiation and filtering. Although not implemented in modern automotive electronic systems, a continuous-time model can be used during the design process to determine the optimum signal-processing operation. Then, the continuous-time operation is converted to a discrete-time model for implementation in a digital system as explained in Appendix B.

In order to understand this process, it is, perhaps, helpful to consider the design process for a continuous-time filter (e.g., one that is developed for working with an analog sensor). This design process can be illustrated with the design of an analog filter. Filters are generally classified in terms of their so-called passbands and stopbands. A passband is a range of frequencies over which the filter has relatively low-attenuation (e.g., approximately <3 dB) characteristics, and a stopband is a range of frequencies over which the attenuation is relatively large (e.g., many tens of dB).

The filter itself is characterized by its complex frequency transfer function ($H(s)$) or its SSS frequency response ($H(j\omega)$). For example, a low-pass filter has a passband from 0 through some cutoff frequency (ω_p) at which point

$$\left|H\left(j\omega_p\right)\right|^2 = \frac{1}{2}|H(j0)|^2 \tag{A.82}$$

It has a stopband with relatively high attenuation for $\omega > \omega_s$ where ω_s is the lower edge of the stopband and $\omega_s > \omega_p$. A high-pass filter is similar to the low-pass filter with the two bands interchanged. A band-pass filter has a passband between a pair of corner frequencies (ω_{p1} ω_{p2}) and a pair of stopbands defined:

$$\begin{aligned} &\text{passband} \quad \omega_{p1} \le \omega \le \omega_{p2} \\ &\text{stopbands} \quad \omega < \omega_{s1} \text{ and } \omega > \omega_{s2} \end{aligned}$$

A bandstop filter has a single stopband between frequencies ω_1 and ω_2 and two passbands outside this range:

$$\begin{aligned} &\text{passband} \quad \omega < \omega_{p1} \quad \omega > \omega_{p2} \\ &\text{stopbands} \quad \omega_{s2} < \omega < \omega_{s2} \end{aligned}$$

Any practical, physically realizable filter has a region of transition between any passband and an adjacent stopband, the slope of which, with respect to frequency, is determined by the order of the filter transfer function.

FILTER-DESIGN TECHNIQUES

Filter design begins with a low-pass prototype function normalized frequency ($\Omega = \omega/\omega_c$) where ω_c is the passband corner frequency. The other three filter types are derived by a linear transformation involving complex frequency s as explained below.

Filters are designed and classified by the function $F(\Omega)$ from which they are derived. For example, the so-called Butterworth filter is derived from the function

$$F(\Omega) = \frac{1}{1+\Omega^{2n}} \tag{A.83}$$

where n is the order of the filter. Butterworth filters are characterized by maximally flat passbands. This function is taken to be the squared magnitude of the sinusoidal frequency response of the desired filter:

$$F(\Omega) = |H(j\Omega)|^2 \tag{A.84}$$

$$= H(S)H(-S)\Big|_{S=j\frac{\omega}{\omega_c}} \tag{A.85}$$

where S is a normalized complex frequency given by $S = s/\omega_c$. To find the transfer function ($H(s)$), the substitution $\Omega^2 \rightarrow -S^2$ is made and the resulting function factored to find the roots of the numerator and denominator. For the example of Butterworth filter, these roots lie along the unit circle in the normalized complex s-plane. These roots are equally spaced, and except for odd n where a single root exists at $S = -1$, they occur in complex conjugate pairs. The roots of the denominator (i.e., poles at $S = P_m$) in the left-half s-plane are determined and the transfer function is given by

$$H(S) = \frac{1}{\prod_{m=1}^{n}(S - P_m)} \tag{A.86}$$

For example, the third-order (i.e., $n = 3$) Butterworth low-pass filter prototype is

$$= H(S) = \frac{1}{(S+1)(S^2+S+1)} \tag{A.87}$$

The transfer function is then unnormalized by replacing S with s/ω_c.

The magnitude and phase of a third-order Butterworth filter are given in Fig. A.23 plot for $H(s)$. Note that 20 log $|H(j1)|^2$ that is equivalent to

$$|H(j\Omega)|_{\Omega=1} = \frac{1}{\sqrt{2}}$$

The normalized frequency $\Omega = 1$ is the corner frequency of this filter in normalized frequency. The steepness of the drop from passband to stopband increases with increasing filter order n. The slope in the transition from the passband is $-20\ n$ dB/decade.

The other major filter types are based upon polynomials in normalized frequency. The two commonest of these are Chebyshev or Cauer. The Chebyshev filter prototypes are derived from the following polynomials:

$$F\left(\Omega^2 = \in^2 C_n^2(\Omega)\right. \tag{A.88}$$

where $C_n(\Omega) = \cos\left(n\cos^{-1}(\Omega)\right) \quad |\Omega| \leq 1$

$$= \cosh\left(n\cosh^{-1}(\Omega)\right) \quad |\Omega| > 1$$

and $\in$ is a parameter determined by the passband "ripple." Cauer filter prototypes are derived from Jacobi elliptical function of Ω and are beyond the scope of the present text. However, any standard filter-design reference will supply design parameter tables. Also, filter design is readily accomplished using MATLAB or other design software.

The design of a high-pass filter normalized prototype by replacing S with $1/S$. Similar linear transformations of S are available for bandpass or bandstop filters, but the transformation is dependent upon the relative values for ω_1 and ω_2. Fortunately, modern software such as MATLAB has the capability of calculating the transfer functions of any of the filters discussed.

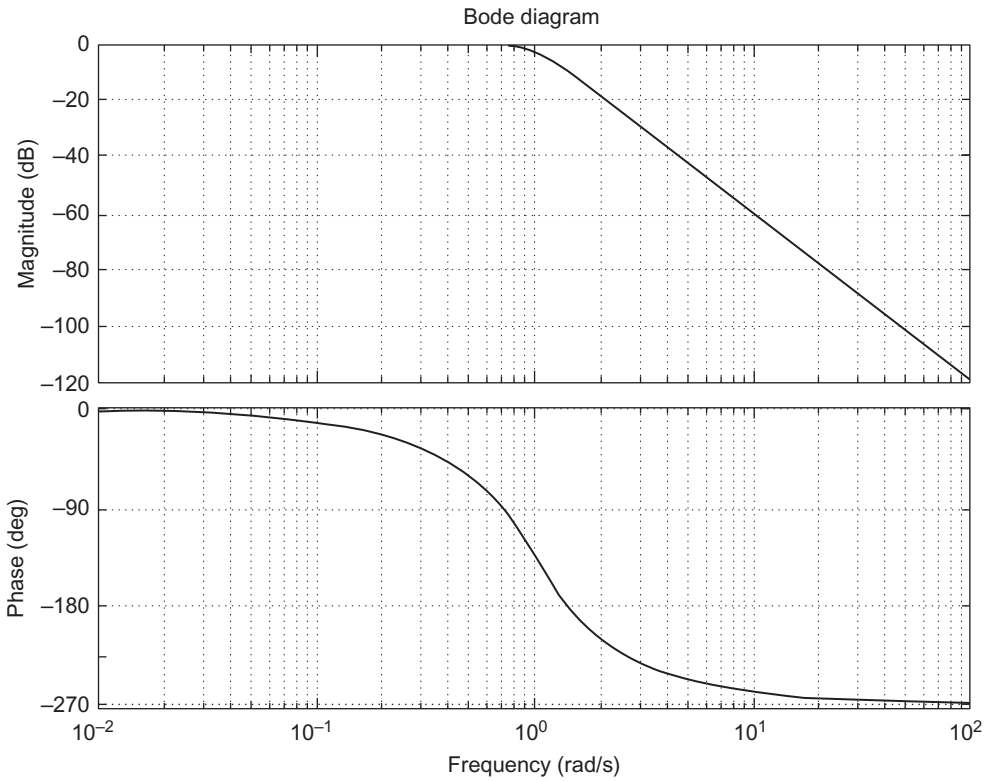


FIG. A.23 Magnitude and phase frequency response of third-order Butterworth filter.

It is clear from the above discussion of filter-design techniques that, in principle, it is possible to design a filter for the measurement of a given quantity that optimizes the signal/noise. Such an optimum design is based on a priori information about the spectrum of the measured quantity. It is normally not possible (nor is it necessary) to know the exact spectrum of this quantity. However, it is often possible to be able to determine upper and lower bounds of this "signal spectrum." The signal-processing filter passband can be selected to enclose these spectral band limits. Noise suppression occurs in the associated stopbands for the optimal filter.

In automotive electronic instrumentation, the sensor often measures a mechanical variable. The dynamic model for the associated mechanical system is often known with great accuracy, thereby allowing the "signal" spectral bounds to be closely estimated. In such cases, the signal-processing filter optimization can readily proceed.

It would be possible to present circuit implementation of continuous-time filter circuits using cascade connections of operational amplifiers. However, in contemporary vehicles, such filters are implemented in digital systems. Appendix B derives algorithms for filter implementation that are applicable in contemporary vehicular electronic system.

VEHICLE STORAGE BATTERY EQUIVALENT CIRCUIT

An equivalent circuit for any electrical or electronic circuit located within a system/subsystem or component is a model for the circuit based on ideal circuit components in the form of a circuit. The voltages and currents throughout the actual circuit can be computed using the models for the idealized components as introduced in Chapter 1 including resistors, capacitors, inductors, and ideal constant voltage and current sources. Examples of equivalent circuits have been presented in various chapters in this book. Normally, the topology of an equivalent circuit closely resembles that of the actual circuit of the system.

At this point, it is significant to illustrate the equivalent circuit concept with the example of the equivalent circuit for a vehicular storage battery. The equivalent circuit topology for a battery with the associated electrical load depends on the battery configuration including its chemistry (e.g., lead acid, Li ion, etc.), the mode of operation (e.g., charging, discharging, or static), the magnitude of the flowing current, and very significantly on its thermal environment. A complete equivalent circuit is nonlinear in both the electrical and thermal models. However, a battery equivalent circuit is highly significant for engineering vehicle electrical systems particularly for relatively large currents (e.g., powering a motor in a hybrid or electric vehicle).

It is, perhaps, instructive to illustrate a battery equivalent circuit in its simplest form. We consider the example of a battery at a constant temperature that is delivering power to a resistive load as depicted in Fig. A.24. In this figure, a switch, denoted $S(t)$, can, for example, be periodically closed for an interval and then opened for an interval such as would be involved in the DC voltage changing circuit of Fig. 6.36.

The equivalent circuit of Fig. A.24 includes an ideal voltage source V_{bo} connected as shown to ideal circuit components R_o, R_1, C_1, and R_L. The current that flows from the battery is denoted I and the voltages at either side of the parallel R_1 C_1 combination are denoted V_1 and V_L. The battery terminals are the points labeled + and −.

Numerically, the capacitance C_1 is extremely large compared to values for typical electronic or electrical circuit applications. Furthermore, the resistances are in the milliohm range. The dynamic response of the battery to a time-varying load is illustrative of the use of the equivalent circuit for performance analysis as well as for the design of vehicular electrical systems that are powered by the battery.

The dynamic model analysis for the equivalent circuit of Fig. A.24 begins with the model for the switch closed and with the current I, which is the sum of the currents through R_1 and C_1 and is given by:

$$I = \frac{V_1 - V_L}{R_1} + C_1 \frac{d(V_1 - V_L)}{dt}$$

where

$$V_1 = V_{bo} - IR_o$$

$$V_L = IR_L$$

The battery open circuit voltage is an ideal voltage source that varies with a parameter called the state of change (SOC), which is defined in terms of the integral of the current flowing from the battery.

$$SOC = 1 - \frac{1}{C(T)} \int_o^t I(\tau) dt$$

where $C(T)$ is the so-called battery capacity that is a function of temperature T as well as other parameters that are not being considered in the present, simplified model. An approximate model for the voltage V_{bo} is given by:

$$V_{bo} = V_{bF} - K_b(T)(1 - SOC)$$

where

$$V_{bF} = V_{bo} \quad \text{for } SOC = 1$$

and where $K_b(T)$ is another battery-dependent parameter. Substituting the equations for V_1 and V_L into the current equation yields the following

$$I = \frac{V_{bo} - I(R_o + R_L)}{R_1} + C_1 \left[\frac{dV_{bo}}{dt} - (R_o + R_L) \frac{dI}{dt} \right]$$

The above equation can be rewritten to yield

$$\dot{I} - \frac{V_{bo}}{(R_o + R_L)} = \frac{V_{bo}}{(R_o + R_L)\tau_1} - \frac{I}{r\tau_1} \tag{A.89}$$

where

$$\tau_1 = R_1 C_1$$
$$r = \frac{R_o + R_L}{R_o + R_L + R_1}$$

Eq. (A.89) can be written in the form of a transfer function model by taking the Laplace transform of Eq. (A.89)

$$I(s) = \frac{V_{bo}(s)}{R_L + R_o} H(s)$$

where

$$H(s) = \frac{s + \dfrac{1}{\tau_1}}{s + \dfrac{1}{r\tau_1}}$$

The battery terminal voltage $V_L = IR_L$ is given by

$$V_L(s) = \frac{V_{bo}(s) R_L}{R_L + R_o} H(s)$$

Assuming that the switch is closed at $t = 0$ and, for illustrative purposes only, that the current remains sufficiently small that SOC remains at approximately 1, the battery voltage $V_{bo}(t)$ is a step input that is given by

$$V_{bo}(t) = 0 \quad t < 0$$
$$= V \quad 0 \le t$$

The terminal voltage as a function of time can be found by taking the inverse Laplace transform of $V_L(s)$:

$$V_L(s) = \frac{V_o R_L}{R_o + R_L} \frac{H(s)}{s}$$
$$V_L(t) = \mathcal{L}^{-1} V_L(s) \tag{A.90}$$
$$= \frac{V_o R_L}{R_o + R_L} \left[1 + (r-1) \left(1 - e^{\frac{-t}{r\tau_1}} \right) \right]$$

For the purpose of designing systems that are powered by the vehicle storage battery, it is often useful to model the system in a state variable form due to the coupling between the battery current and other variables in the system being powered by it (e.g., for a traditional DC engine starter motor). Eq. (A.89) can be written in a single dimensional state variable model with a simple substitution for the state variable x

$$x = I - u$$

where

$$u = \frac{V_{bo}}{R_o + R_1}$$

The standard state variable equation is

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

where

$$A = \frac{1}{r\tau_1}$$
$$B = \frac{(r-1)}{r\tau_1}$$
$$C = 1$$
$$D = 1$$

$D = 1$. The solution to this state variable equation for $I(t)$ with a step input u gives the same solution for V_L as the inverse Laplace solution of Eq. (A.90) by multiplying y by $V_{bo}R_L/(R_L + R_o)$

$$V_L = \frac{R_L V_{bo}}{R_L + R_o} y$$

The present illustrative example of vehicle battery equivalent circuit is not representative of a practical battery equivalent circuit. The literature contains multiple equivalent circuit models for practical battery modeling. A practical equivalent circuit model for any given vehicle battery has separate models for charge and for discharge modes as well as detailed thermal models. The influence of *SOC* on open

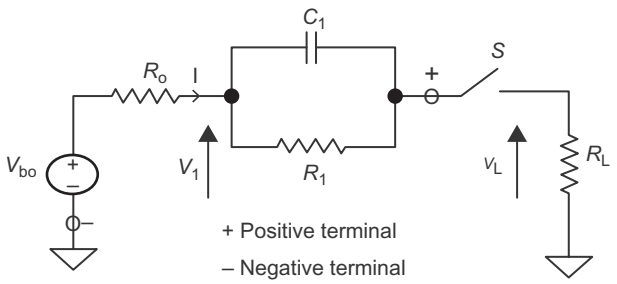


FIG. A.24 Vehicle battery equivalent circuit.

circuit voltage is well discussed in the literature. A typical practical battery equivalent circuit will have a series connection of multiple parallel R and C combinations of the form depicted by R_1 and C_1 in Fig. A.24. The design and performance analyses of vehicle electrical systems requiring accurate battery models can be implemented with the practical equivalent circuit using methods of calculating transfer functions and/or state variable models as illustrated above with the simplified exemplary equivalent circuit.

This appendix has reviewed some basic principles of continuous-time system theory that are applicable through the various chapters of the book. Specific applications of this theory are found in nearly all automotive electronic systems. However, as explained earlier, modern automotive electronics are digital and are modeled and analyzed using discrete-time methods. Appendix B reviews basic principles of such discrete-time system modeling/analysis/design.

APPENDIX

DISCRETE TIME SYSTEMS THEORY B

As explained in Appendix A, automotive electronic control and instrumentation systems (as well as virtually all other electrical systems) are implemented with digital electronics as at least some component or subsystem. Digital controllers and/or signal processing subsystems incorporate one or more microprocessors or microcontrollers, each having a stored program to run the system. Such systems are fundamentally discrete time systems.

However, automotive electronic systems also incorporate analog or continuous time components (e.g., sensors and actuators). In order for the digital subsystem to perform its intended operation, it has, for its input/output variables, numerical values of the continuous input/output at discrete times (t_k where $k = 1, 2, \ldots$). The time between successive input/output values must be sufficient for the digital system to perform all operations on the input to generate an output.

Although it is not necessary, most discrete time systems use periodic times to represent input/output; that is, the kth discrete time is given by

$$t_k = kT_s \qquad k = 1, 2, 3, \cdots$$

where T_s is the sample period. The configuration for a discrete time system with an embedded digital system and an analog destination component is depicted in Fig. B.1.

In this figure, the source has a continuous time electrical signal $\mathrm{v}(t)$ that could, for example, be a sensor output. The interface electronics-labeled A/D converter (which is modeled later in this appendix) generates a sequence of numerical values called samples at each discrete time or "sample period" t_k:

$$\mathrm{v}_k = \mathrm{v}(t_k)$$

These samples must be in a format that can be input to the digital system. The input to the digital system at t_k is denoted x_k, which is a digital (N bit) numerical value equal to v_k; that is, the sampled variable v_k becomes a binary number x_k. The digital system generates an output y_n associated with input sample x_n (as well as previous samples depending on the operations performed). Although the destination component might be a display device that can display the desired output numerical value, it may also be an actuator requiring a continuous time electrical signal $y(t)$. We assume here that the destination component (e.g., a display or actuator) requires a continuous time electrical input. This continuous time electrical signal is generated from the output y_n via an output interface D/A converter. A system that is partly continuous time along with one or more sampling operations is called a sampled data system or a discrete time system.

It is important when explaining such systems for either design or performance analysis to develop appropriate models for mixed continuous and discrete time systems. For the purpose of developing such models, it is helpful to discuss initially only linear, time-invariant systems. In chapters that are

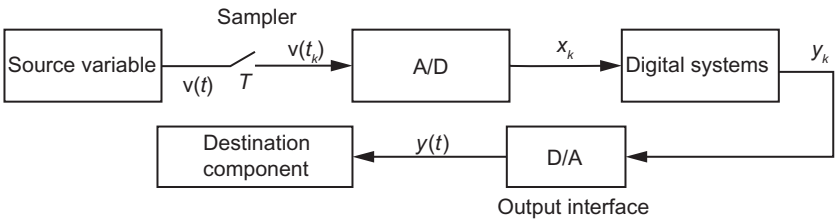


FIG. B.1 Discrete time system configuration.

concerned with specific automotive systems, we deal with nonlinearities as required to explain the particular system.

As shown in the Appendix A, a linear time-invariant continuous time system is characterized by an nth order differential equation with constant coefficients. The linear time-invariant discrete time system is characterized by a model in the format of difference equations. One commonly used model for calculating the output y_n of a discrete time system is in the form of a recursive model:

$$y_n = \sum_{k=0}^{K} a_k x_{n-k} - \sum_{\ell=1}^{L} b_\ell y_{n-\ell} \tag{B.1}$$

The dynamic response of such a system is determined by the coefficients a_k and b_ℓ.

It is shown in Appendix A that a continuous time system is usefully characterized by its transfer system ($H(s)$), which is obtained from the Laplace transforms of its input and output. A similar procedure is very useful for conducting performance analysis or design of a discrete time digital system.

For a discrete time system, the transform of a sequence x_n that is analogous to the Laplace transform for a continuous time system is the $\mathcal{Z}$-transform $X(z)$ defined by

$$\begin{aligned} X(z) &= \mathcal{Z}(x_n) \\ X(z) &= \sum_{n=-\infty}^{\infty} x_n z^{-n} \end{aligned} \tag{B.2}$$

We illustrate with an example in which x_n is defined as

$$\begin{aligned} x_n &= c^n \quad n \geq 0 \\ &= 0 \quad n \leq 0 \end{aligned} \tag{B.3}$$

The $\mathcal{Z}$-transform is given by

$$\begin{aligned} X(z) &= \sum_{n=0}^{\infty} c^n z^{-n} \\ &= \sum_{n=0}^{\infty} \left(cz^{-1}\right)^n \end{aligned} \tag{B.4}$$

where $z =$ complex variable analogous to s for the Laplace transform. This latter sum is a geometric series that converges if $\left|cz^{-1}\right| \leq 1$ or $|z| \geq |c|$ to the closed form result

$$X(z) = \frac{1}{1 - cz^{-1}} \quad |z| \geq |c| \tag{B.5}$$

There are several important elementary properties of the $\mathcal{Z}$-transform that are important in the analysis of discrete time digital systems that are summarized without proof below:

1. Linearity:

$$\mathcal{Z}[ax_n + by_n] = aX(z) + bY(z) \tag{B.6}$$

2. Time shift by an integer k

$$\mathcal{Z}[x_{n+k}] = z^k X(z) \tag{B.7}$$

3. Convolution: let

$$W_n = \sum_{k=-\infty}^{\infty} x_k y_{n-k} = \sum_{n=-\infty}^{\infty} x_{n-k} y_k \tag{B.8}$$

then

$$W(z) = X(z)Y(z)$$

As in the case of a continuous time system for which the inverse Laplace transform can be found, there is an inverse $\mathcal{Z}$-transform of $X(z)$ yielding the sequence $\{x_n\}$denoted as $\mathcal{Z}^{-1}[X(z)] = \{x_n\}$, which is given by the following contour integral in the complex z-plane:

$$x_n = \frac{1}{2\pi j}\oint_C X(z) z^n \frac{dz}{z} \tag{B.9}$$

where the contour C is chosen in a region of the complex z-plane for which the series converges.

It is assumed that $Y(z)$ is the $\mathcal{Z}$-transform of a sequence y_n that is bounded as $n \to \pm\infty$. In this case (which is the case of practical significance in any automotive electronic system), the unit circle in the complex z-plane (i.e., $|z| = 1$) forms the boundary of the region of convergence of $Y(z)$. All poles of $Y(z)$ lie inside the unit circle, and $Y(z)$ is analytic for $|z| > 1$. The inverse $\mathcal{Z}$-transform of $Y(z)$ is a single-sided sequence $\{y_n\}$ where

$$y_n = 0 \quad n < 0$$

In this case (of practical interest), the contour C is the unit circle (i.e., $C \to |z| = 1$).

In practice, the inverse $\mathcal{Z}$-transform of a function of z (e.g., $Y(z)$) is normally computed from a partial fraction expansion of $Y(z)$ about its poles z_k:

$$\begin{aligned} Y(z) &= \sum_{j=1}^{n} \frac{a_j}{z - z_j} \\ &= \sum_{j=1}^{n} \frac{a_j z^{-1}}{1 - z_j z^{-1}} \end{aligned} \tag{B.10}$$

where $a_j =$ residue at pole z_j.

The residue theorem was explained in Appendix A and applies equally to the partial fraction expansion of functions of complex variable z.

Each of these terms can be rewritten in the form of a Taylor series for each pole provided $|z| > |z_j|$:

$$\frac{a_j z^{-1}}{1 - z_j z^{-1}} = a_j \sum_{m=0}^{\infty} z_j^m z^{-(m+1)} \qquad j = 1, 2, \cdots, n \tag{B.11}$$

Replacing the summation index m with $k - 1$ and beginning the series sum with $k = 1$ yield an expression for each partial fraction of the same form as $Y(z)$. Combining terms of like power yields the following expression for $Y(z)$:

$$Y(z) = \sum_{k=1}^{\infty} \left[\sum_{j=1}^{n} \left(a_j z_j^{k-1} \right) \right] z^{-k} \tag{B.12}$$

Comparing like powers of z of Eq. (B.12) with the definition of $Y(z)$ yields the inverse $\mathcal{Z}$-transform of $Y(z)$, which is the sequence $\{y_k\}$ where

$$y_k = \sum_{j=1}^{n} a_j z_j^{k-1} \tag{B.13}$$

DIGITAL SUBSYSTEM

Before proceeding with the discussion of complete sampled data systems, it is, perhaps, worthwhile to discuss certain basic characteristics of the digital subsystem shown in Fig. B.1. Once again, assuming linear time invariance for this component, it has already been explained that its model is generally of the recursive form

$$y_n = \sum_{k=0}^{K} a_k x_{n-k} - \sum_{k=1}^{L} b_k y_{n-k} \tag{B.14}$$

Such a subsystem is typically called a digital filter, regardless of its specific function in the larger system. We proceed with the approach to the design/analysis of the digital filter by first determining its digital transfer function. This can be computed directly from the $\mathcal{Z}$-transform of the above model:

$$Y(z) = \mathcal{Z}(y_n) = \sum_{n=-\infty}^{\infty} y_n z^{-n} \tag{B.15}$$

$$= \sum_{n=-\infty}^{\infty} \left[\sum_{k=0}^{K} a_k x_{n-k} - \sum_{k=1}^{L} b_k y_{n-k} \right] z^{-n} \tag{B.16}$$

Using the shift property, it can be shown that

$$Y(z) = \left[\sum_{k=0}^{K} a_k z^{-k} \right] X(z) - \left[\sum_{k=1}^{L} b_k z^{-k} \right] Y(z) \tag{B.17}$$

which can be rewritten in the form

$$Y(z) = H(z) X(z) \tag{B.18}$$

The function $H(z)$ is the digital transfer function of the digital filter and is given by

$$H(z) = \frac{Y(z)}{X(z)} \tag{B.19}$$

$$= \frac{\sum_{k=0}^{K} a_k z^{-k}}{1 + \sum_{k=1}^{L} b_k z^{-k}} \tag{B.20}$$

The design procedures presented later in this appendix permit the calculation of the digital transfer function to be computed. From this transfer function, the filter coefficients can be obtained from the corresponding power of z^{-1}.

As in the case of a continuous time filter, the response to a unit impulse for the filter is the digital filter impulse response. For such an input, its $\mathcal{Z}$-transform $X(z)=1$ and the output $Y(z)=H(z)$. The inverse $\mathcal{Z}$-transform of $H(z)$ is the sequence $\{h_n\}$, where components are given by

$$h_n=\frac{1}{2\pi j}\oint_C H(z)z^n\frac{dz}{z} \tag{B.21}$$

where the contour C is the unit circle $|z|=1$. Any physically realizable filter requires no future inputs (i.e., any input prior to x_n) to generate y_n, and the filter is said to be causal; that is to say, $h_n=0$ for $n\leq 0$. For a filter having the property

$$\underset{n\to\infty}{Lim}h_n=0$$

the filter is assured to be stable.

A filter that has all $b_k=0$ is called nonrecursive since it uses no previously calculated outputs to yield the most recent output y_n. Such a filter is also said to have a finite impulse response since

$$\begin{aligned} h_n &= a_n \quad 0\leq n\leq K \\ &= 0 \quad \text{elsewhere} \end{aligned}$$

A recursive filter has at least one nonzero b_k coefficient. Such a filter has an infinite impulse response.

SINUSOIDAL FREQUENCY RESPONSE

One of the most important inputs for assessing system performance is the sinusoid. For an understanding of the sinusoidal frequency response of a digital filter, it is necessary to have the $\mathcal{Z}$-transform of a sampled sinusoidal signal having frequency ω sampled at period $t_n=nT$. The input sequence x_n is given by

$$\begin{aligned} x_n &= A\sin(\Omega n) \quad t\geq 0 \quad n=0,1,2,\ldots \\ &= 0 \quad t<0 \end{aligned}$$

where $\Omega=\omega T$. The sinusoid can be rewritten as

$$\sin(\Omega n)=\left[e^{j\Omega n}-e^{-j\Omega n}\right]/2j$$

The $\mathcal{Z}$-transform of $\{x_n\}$, which is denoted $X(z)$, is given by

$$X(z)=\frac{A}{2j}\left[\sum_{n=0}^{\infty}\left(e^{j\Omega}z^{-1}\right)^n-\sum_{n=0}^{\infty}\left(e^{-j\Omega}z^{-1}\right)^n\right] \tag{B.22}$$

Both series converge yielding the following expression for $X(z)$:

$$\begin{aligned} X(z) &= \frac{A}{2j}\left[\frac{1}{(1-e^{j\Omega}z^{-1})}-\frac{1}{(1-e^{-j\Omega}z^{-1})}\right] \\ &= \frac{A\sin(\Omega)z^{-1}}{(1-e^{j\Omega}z^{-1})(1-e^{-j\Omega}z^{-1})} \end{aligned} \tag{B.23}$$

The filter output $Y(z)$ is given by

$$\begin{aligned} Y(z) &= H(z)X(z) \\ &= \frac{A\sin(\Omega)z^{-1}H(z)}{(1-e^{j\Omega}z^{-1})(1-e^{-j\Omega}z^{-1})} \end{aligned} \tag{B.24}$$

where $H(z)$ = transfer function for the digital filter. By partial fraction expansion, the filter output is given by

$$Y(z) = \frac{AH(e^{j\Omega})}{2j(1-e^{j\Omega}z^{-1})} - \frac{AH(e^{-j\Omega})}{2j(1-e^{-j\Omega}z^{-1})} + \sum_{k=1}^{K}\frac{\alpha_k z^{-1}}{(1-\beta_k z^{-1})} \tag{B.25}$$

where the latter sum terms (involving poles β_k) are due to poles of $H(z)$. The steady-state sinusoidal frequency response corresponds to the limiting value of $Y(z)$ for $n \to \infty$. The operation performed by this digital filter is determined by the filter coefficients a_k and b_k. Powerful methods have been developed permitting a designer to determine these filter coefficients such that the filter performs the operation required of it to meet the objectives of the sampled data systems. Many examples are presented in various chapters of this book dealing with specific automotive subsystems. The designer chooses filter coefficients to obtain the required system performance. The terms, due to the poles of $H(z)$, all asymptotically approach zero for $n \to \infty$. The remaining first two terms in the above expression represent the digital filter steady-state sinusoidal frequency response $Y_{ss}(z)$, which can be written in the form

$$Y_{ss}(z) = A\left\{\frac{z^{-1}\sin(\Omega)[H(e^{j\Omega}) + H(e^{-j\Omega})] - j[1 - z^{-1}\cos(\Omega)[H(e^{j\Omega}) - H(e^{-j\Omega})]]}{2[1 - 2\cos(\Omega)z^{-1} + z^{-2}]}\right\} \tag{B.26}$$

The inverse $\mathcal{Z}$-transform of $Y_{ss}(z)$ can be shown to be (using the table of transforms, Table B.1)

$$\begin{aligned} y_n &= A\left\{\frac{[H(e^{j\Omega}) + H(e^{-j\Omega})]}{2}\sin(n\Omega) + \frac{[H(e^{j\Omega}) - H(e^{-j\Omega})]}{2j}\cos(n\Omega)\right\} \\ &= A\left|H\left(e^{j\Omega}\right)\right|\sin[n\Omega + \phi(j\Omega)] \end{aligned} \tag{B.27}$$

where $\phi = \angle H\left(e^{j\Omega}\right)$ = (phase angle of $H(j\Omega)$).

The steady-state sinusoidal frequency response of a digital filter having a transfer function $H(z)$ is a sinusoid of the same frequency scaled in amplitude by $H(e^{j\Omega})$ and having a phase $\phi(j\Omega)$ given by $\angle H\left(e^{j\Omega}\right)$. Thus, the behavior of $H(z)$ on the unit circle $z = e^{j\Omega}$ gives the frequency response characteristics for $-\pi \le \Omega \le \pi$, where Ω is the digital frequency.

We consider now digital filtering of analog signals for any system employing analog, continuous time components along with the digital filter (e.g., sensor and actuator or display). The configuration for this process is shown in Fig. B.2, which depicts a subset of the components of Fig. B.1 focusing here on the components associated with digital filtering of an analog input $x(t)$ to yield an analog output $y(t)$.

The first component is called an analog-to-digital converter (A/D). The A/D converter samples the input periodically (with period T) and prepares the sampled signal x_k in a form that can be input to the computer; that is, the A/D quantizes the sample x_k and codes it in a binary or similar computer usable form. The computer, under program control, calculates the numerical value of the filter output y_k. The final component called a digital-to-analog (D/A) converter receives the output from the digital

Table B.1 Table of Transforms

LaPlace transform	Time function	z-Transform
1	Unit impulse δ	1
$\frac{1}{s}$	Unit step $u_s(t)$	$\frac{z}{z-1}$
$\frac{1}{1-e^{-Ts}}$	$\delta_r(t)=\sum_{n=0}^{\infty}\delta(t-nT)$	$\frac{z}{z-1}$
$\frac{1}{s^2}$	t	$\frac{T_z}{(z-1)^2}$
$\frac{1}{s^3}$	$\frac{t^2}{2}$	$\frac{T^2z(z+1)}{2(z-1)^3}$
$\frac{1}{s^{n+1}}$	$\frac{t^n}{n!}$	$\lim_{\alpha\to 0}\frac{(-1)^n}{n!}\frac{\partial^n}{\partial\alpha^n}\left[\frac{z}{z-e^{-\alpha T}}\right]$
$\frac{1}{s+\alpha}$	e^{-at}	$\frac{z}{z-e^{\alpha T}}$
$\frac{1}{(s+\alpha)^2}$	te^{-at}	$\frac{Tze^{-\alpha T}}{(z-e^{-\alpha T})^2}$
$\frac{\alpha}{s(s+\alpha)}$	$1-e^{-at}$	$\frac{(1-e^{-\alpha T})z}{(z-1)(z-e^{-\alpha T})}$
$\frac{\omega}{s^2+\omega^2}$	$\sin\omega t$	$\frac{z\sin\omega T}{z^2-2z\cos\omega T+1}$
$\frac{\omega}{(s+\alpha)^2+\omega^2}$	$e^{-at}\sin\omega t$	$\frac{ze^{\alpha T}\sin\omega T}{z^2-2ze^{-\alpha T}\cos\omega T+e^{-2\alpha T}}$
$\frac{s}{s^2+\omega^2}$	$\cos\omega t$	$\frac{z(z-\cos\omega T)}{z^2-2z\cos\omega T+1}$
$\frac{s+\alpha}{(s+a)^2+\omega^2}$	$e^{-at}\cos\omega t$	$\frac{z^2-ze^{-\alpha T}\cos\omega T)}{z^2-2ze^{-\alpha T}\cos\omega T+e^{-2\alpha T}}$

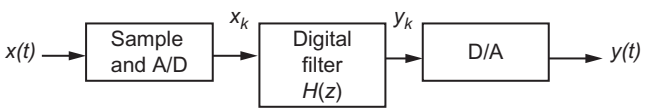


FIG. B.2 Digital filtering of analog signal.

filter computer and reconstructs a continuous time signal $y(t)$ such that samples of $y(t)$ at t_k are as close as possible, within the capabilities of the computer and the A/D converter, to being samples of $y(t)$:

$$y_k \cong y(t_k)$$

The limitations placed on these approximations are discussed in various chapters in this book.

It is worthwhile here to present some important aspects of the sampled analog signal. It is clear that there is a loss of information during the sampling process since the sampled signal only represents the analog signal at discrete times t_k. This loss of information is mitigated somewhat by the conceptual installation of a reconstruction device, which most commonly is a zero-order hold (ZOH) (explained in detail later in this appendix with exemplary circuit implementation in Chapter 2). This device essentially clamps the output signal to the value of the latest sample. Although the actual sampled data

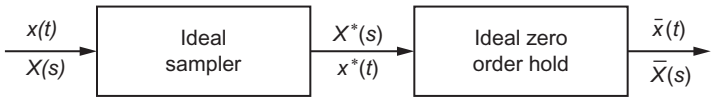


FIG. B.3 Ideal sampler configuration.

system incorporates an A/D converter, for analysis purposes, it is convenient to represent this system as depicted in Fig. B.3

The output of the sample and hold can be represented by the following model:

$$\begin{aligned}\bar{x}(t) = x(0)[u(t) - u(t-T)] + x(T)[u(t-T) - u(t-2T)] \\ + x(2T)[u(t-2T) - u(t-3T)]\cdots\end{aligned} \tag{B.28}$$

where $u(t)$ is a unit step. Taking the Laplace transform of $\bar{x}(t)$ yields $\bar{X}(s)$, which can be shown to be

$$\begin{aligned}\bar{X}(s) &= \frac{(1-e^{-Ts})}{s}\left[\sum_{n=0}^{\infty} x(nT)e^{-nTs}\right] \\ &= \frac{(1-e^{-Ts})}{s}X^*(s)\end{aligned} \tag{B.29}$$

The first factor is effectively the transfer function of the ZOH and the second $X^*(s)$, which is called the starred transform, is defined as

$$X^*(s) = \sum_{n=0}^{\infty} x(nT)e^{-nTs} \tag{B.30}$$

The starred transform of any variable is the $\mathcal{Z}$-transform with z replaced by e^{sT} as expressed below:

$$X^*(s) = X(z)|_{z=e^{sT}}$$

It should be emphasized at this point that $X^*(s)$ is a fictitious signal introduced solely for analysis purposes. The fundamental problem in modeling sampled data systems using starred transforms is that the ideal sampler does not have a transfer function relating its input to its output. The inverse Laplace transform of $X^*(s)$ is denoted $x^*(t)$ and is given by

$$x^*(t) = x(0)\delta t + x(T)\delta(t-T) + \cdots x(nT)\delta(t-nT)\cdots \tag{B.31}$$

where $\delta(t)$ is the ideal impulse function. This expression for $x^*(t)$ is equivalent to the output of an ideal sampler as depicted in Fig. B.3.

The starred transform can also be rewritten in the form

$$\begin{aligned}X^*(s) = \frac{1}{T}[&X(s) + X(s+j\omega_s) + \cdots X(s+jn\omega_s)\cdots \\ &+ X(s-j\omega_s) + X(s-2j\omega_s) + \cdots X(s-jn\omega_s) + \cdots \\ &+ \frac{x(0)}{2}\Big]\end{aligned} \tag{B.32}$$

where $\omega_s = 2\pi/T$. This result indicates that the Laplace transform is periodic in sample radian frequency.

For $s = j\omega$, the starred transform $X^*(j\omega)$ is the spectrum of the ideal sampled signal. This spectrum is a periodic repetition of the spectrum of the input signal. In theory, the original signal could be reconstructed with an ideal low-pass filter having frequency response $H(j\omega)$ given by

$$H(j\omega) = 1 \quad -\frac{\omega_s}{2} < \omega < \frac{\omega_s}{2}$$
$$= 0 \quad \text{elsewhere}$$

provided that the input spectrum is confined to the ideal filter pass band. Any signal exceeding this band cannot be even theoretically reconstructed without errors due to the overlap of adjacent repetitions of the original signal spectrum. This input spectrum restriction is known as the sampling theorem, and errors that occur when the limit is violated are known as aliasing errors. The sampling theorem requires that the sampling frequency ($F_s = 1/T$) be at least twice the highest frequency component in the signal being sampled to avoid aliasing errors.

We consider first the design of a digital filter to achieve the desired operation based upon a continuous time (analog) prototype. In this case, the desired continuous time linear transfer function $H(s)$ is known. Conversion to the corresponding digital filter transfer function $H(z)$ yields the filter coefficients a_k and b_k necessary to perform the filtering numerically. There are numerous techniques for converting from $H(s)$ to $H(z)$ that yield very close approximations to the desired $H(s)$. Fortunately, there is software available to accomplish this task. For example, MATLAB has a range of functions that give the filter coefficients directly from parameters entered (e.g., sampling frequency, filter type, and pass- and stop-band edge frequencies). The MATLAB function butter creates an output of a_k and b_k for the digital transfer function $H(z)$ of the form given in Eq. (B.20) based on a Butterworth analog prototype. The design of digital filters from analog prototypes requires that the analog frequencies be normalized to the cutoff frequency ω_c of a low-pass prototype. Normalized analog frequency is denoted ω_n and is given by $\omega_n = \omega/\omega_c$, where ω_c is the 3 db corner frequency. It requires inputs m = filter order and normalized cutoff frequency ω_{nc} where $0 \leq \omega_{nc} \leq 1$ and where $\omega_n = 1$ corresponds to $F_s/2$. The digital corner frequency is related to the analog corner frequency ω_c by the following relationship:

$$\Omega_c = \omega_c T \tag{B.33}$$

The sampling frequency must be selected such that the highest input frequency $\omega_{\max}$ satisfies

$$\omega_{\max} \leq \frac{\pi}{T} \tag{B.34}$$

to avoid aliasing errors as described above. There are numerous design procedures for finding the transfer function for a digital filter from a continuous time equivalent analog filter. In any such procedure, the sampling frequency $F_s = 1/T$ choice is influenced by the spectrum of the input analog signal $X(\omega)$. To avoid aliasing errors, the digital frequency for any analog frequency must fall within the band $-\pi \leq \Omega \leq \pi$. One such procedure utilizes a linear one-to-one mapping of normalized analog frequency band $0 \leq \omega_n \leq \infty$ into the digital frequency band $0 \leq \Omega \leq \pi$. This transformation is given by

$$\omega_n \tan\left(\frac{\Omega_c}{2}\right) = \tan\left(\frac{\Omega}{2}\right) \tag{B.35}$$

where the digital corner frequency is given by

$$\Omega_c = \frac{2\pi f_c}{F_s} \tag{B.36}$$

and where f_c is the actual desired corner frequency (in Hz) and F_s is the sampling frequency. The normalized analog corner frequency is $\omega_n = 1$. Note that aliasing errors are avoided since all analog frequencies map to the required digital frequency board.

The transformation from analog-to-digital transfer functions is found by replacing $s = j\omega_n$ and $z = e^{j\Omega}$ in the linear mapping transformation, which yields

$$s = j\cot\left(\frac{\Omega_c}{2}\right)\frac{\sin\left(\frac{\Omega}{2}\right)}{\cos\left(\frac{\Omega}{2}\right)}\Bigg|_{e^{j\Omega}=z} \tag{B.37}$$

$$= \cot\left(\frac{\Omega_c}{2}\right)\left[\frac{e^{j\Omega/2} - e^{-j\Omega/2}}{e^{j\Omega/2} + e^{j\Omega/2}}\right]\Bigg|_{e^{j\Omega}=z} \tag{B.38}$$

$$= c\left(\frac{z-1}{z+1}\right) \tag{B.39}$$

where $c = \cot\left(\frac{\Omega_c}{2}\right)$.

The digital transfer function $H(z)$ is given in terms of analog transfer function that is denoted as $H^a(S)$:

$$H(z) = H^a(S)\Big|_{S=\frac{c(z-1)}{z+1}}$$

$$H(z) = H^a\left[\frac{c(z-1)}{z+1}\right] \tag{B.40}$$

where $S = \dfrac{s}{\omega_c}$ = normalized complex frequency.

As an example of this linear transformation method for finding $H(z)$, we consider a third-order Butterworth normalized frequency prototype. This prototype Butterworth filter has analog transfer function given by (see Appendix A)

$$H_a(S) = \frac{1}{(S+1)(S^2+S+1)} \tag{B.41}$$

It can be shown that the digital transfer function $H(z)$ is given by

$$H(z) = \frac{(z+1)^3}{K_1(z+z_1)(z^2+\alpha z+\beta)} \tag{B.42}$$

where

$$K_1 = (c+1)\left(c^2+c+1\right) \tag{B.43}$$

$$z_1 = \frac{(1-c)}{1+c} \tag{B.44}$$

$$\alpha = \frac{2(1-c^2)}{(c^2+c+1}} \tag{B.45}$$

$$\beta = \frac{(c^2-c+1)}{c^2+c+1} \tag{B.46}$$

The coefficients of the recursive algorithm for this digital filter are found by rewriting the expression for $H(z)$ as a ratio of polynomials in powers of z^{-1} as given below:

$$H(z) = \frac{1+3z^{-1}+3z^{-2}+z^{-3}}{K_1[1+(z_1+\alpha)z^{-1}+(\alpha z_1+\beta)z^{-2}+z_1\beta z^{-3}]} \tag{B.47}$$

The recursive filter equation for this example is found by selecting the coefficients (a_k and b_k) using the previously given digital transfer function model

$$H(z) = \frac{\sum_{k=0}^{K} a_k z^{-k}}{1+\sum_{k=1}^{L} b_k z^{-k}} \tag{B.48}$$

where, for this example, $K=L=3$. Thus, we can write the recursive filter model:

$$\begin{aligned} y_n = \{&x_n + 3x_{n-1} + 3x_{n-2} + x_{n-3} \\ &- [(z_1+\alpha)y_{n-1} + (\alpha z_1+\beta)y_{n-2} + z_1\beta y_{n-3}]\}/K_1 \end{aligned} \tag{B.49}$$

As an example of this type of digital filter design, we present a digital version of the third-order Butterworth filter presented in Appendix A. In this example, let the sample frequency be $F_s = 10$ kHz and the corner frequency $f_c = 2$ kHz. The digital corner frequency Ω_c is given by

$$\Omega_c = \frac{2\pi f_c}{F_s} = 1.2566 \tag{B.50}$$

The parameters of $H(z)$ are given by

$$\begin{aligned} c &= \cot\left(\frac{\Omega_c}{2}\right) = 1.3764 \\ K_1 &= (c+1)\left(c^2+c+1\right) = 10.149 \\ z_1 &= (1-c)/(1+c) = -0.1584 \\ \alpha &= 2\left(1-c^2\right)/\left(c^2+c+1\right) = -0.4189 \\ \beta &= \left(c^2-c+1\right)/\left(c^2+c+1\right) = 0.3554 \end{aligned} \tag{B.51}$$

The digital sinusoidal frequency response ($H(e^{j\Omega})$) for this example is given in Fig. B.4.

The magnitude squared of the response at the digital corner frequency is

$$\left|H\left(e^{j\Omega_c}\right)\right|^2 = \frac{1}{2} \tag{B.52}$$

which corresponds to the response of the analog filter presented in Fig. A.23 for the normalized analog corner frequency of $\omega_{nc} = 1$.

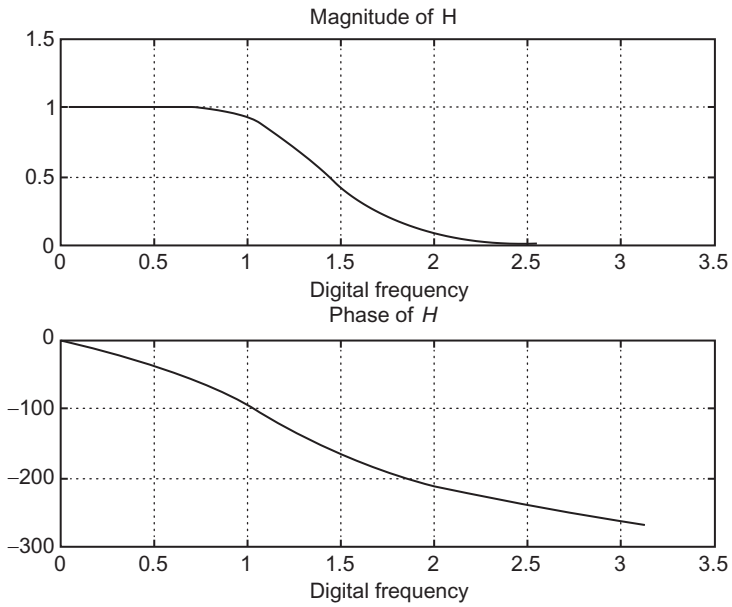


FIG. B.4 Digital sinusoidal frequency response of third-order Butterworth filter.

DISCRETE TIME CONTROL SYSTEM

The previous section of this appendix, which involves digital filtering of analog signals (in which the output signal is sent to a display device), would be applied in the design of an instrument system. For control applications, the digital "filter" output would be sent to an actuator that is associated with the plant being controlled. For design/analysis procedures, the plant model should include the actuator dynamic model.

Normally, the actuator is an analog device requiring a continuous time electrical input signal. In this case, the output of the digital controller (filter) must be converted to analog form via the D/A converter. For analytic purposes, this digital/analog conversion is taken to include a ZOH.

The operation of and model for the input sample and A/D process have already been explained. The D/A process at the output of the discrete time digital control system using a ZOH is best described from its idealized model. Variations from the ideal to the practical system can be minimized by design. A circuit configuration for the ZOH is given in Chapter 2. The ZOH is actually an analog circuit that receives input pulses of amplitude $\bar{u}_n$. These pulses can be modeled as ideal impulses and, in practice, are created by a D/A converter from the output sequences $\{u_n\}$ of the digital controller. These pulses are generated at times $t_n = nT$ where T is the period of the input sampler. Apart from a small time delay during which the digital filter performs its operation, these output pulses are synchronous with the input samples to the A/D converter.

A simple approximate model for the ZOH output $\bar{u}(t)$ is given by

$$\bar{u}(t) = \bar{u}_n \quad t_n \le t < t_n + T \tag{B.53}$$

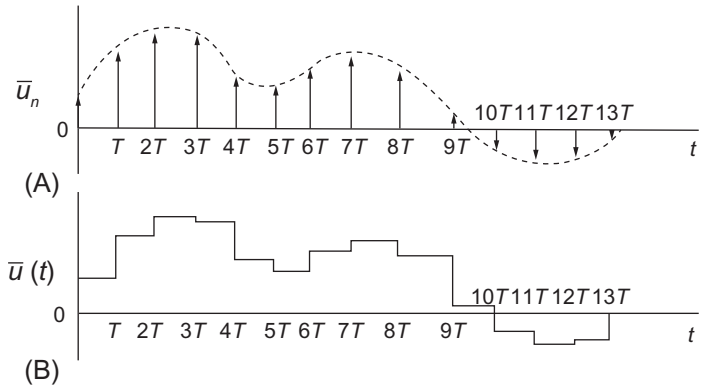


FIG. B.5 Illustration of sample and hold signals. (A) Samples of output pulses and (B) zero-order hold (ZOH) output.

that is, an ideal ZOH receives pulses at times t_n and holds the output at the value $\bar{u}_n$ for one sample period. Fig. B.5 illustrates this process in which the bar over the variable signifies an analog signal. In actual practice, the voltage pulses $\bar{u}_n$ are of finite duration and of an amplitude given by $\bar{u}_n = u_n$ where u_n is the numerical value of the digital system output.

The ZOH output yields a piecewise continuous function $\bar{u}(t)$ of the corresponding continuous control signal $u(t)$ that would be generated by the analog (continuous time) prototype system from which the discrete time system was developed. The closeness of the approximation $(\bar{u}(t) \cong u(t))$ is influenced by the sample period relative to the system dynamics and the precision of computation of the digital discrete time system (e.g., number of bits in the digital data) as explained in Chapter 3.

In a control system application, the ZOH output drives the plant actuator, which, in turn, drives the plant dynamics. The configuration for an open-loop system utilizing a digital controller is depicted in Fig. B.6.

The model for the digital controller is

$$u(z) = H_c(z)X(z) \tag{B.54}$$

In terms of the starred transfer function, this model, with the substitution $z = e^{st}$, can be written as

$$u^*(s) = H^*(s)X^*(s) \tag{B.55}$$

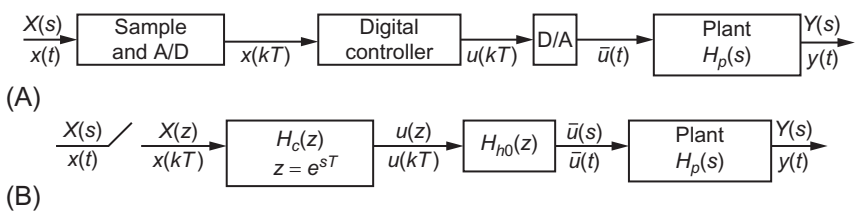


FIG. B.6 Open-loop discrete time control system. (A) Open-loop system with digital control and (B) model for open-loop system.

The A/D and D/A converters, controller, and plant can be combined to yield the output $Y(s)$

$$Y(s) = H_p(s)\bar{u}(s) \tag{B.56}$$

$$= H_p(s)\left[\frac{1-e^{-sT}}{s}\right]u^*(s) \tag{B.57}$$

The $\mathcal{Z}$-transform of the output $Y(z)$ is given by

$$Y(z) = \mathcal{Z}\left[\frac{(1-e^{-sT})}{s}H_p(s)\right]u(z) \tag{B.58}$$

$$= \mathcal{Z}\left[\frac{(1-e^{-sT})}{s}H_p(s)\right]H_c(z)X(z) \tag{B.59}$$

$$= (1-z^{-1})\,\mathcal{Z}\left[\frac{(H_p(s))}{s}\right]H_c(z)X(z) \tag{B.60}$$

Eq. (B.60) is obtained from Eq. (B.59) from the time shift property of the $\mathcal{Z}$-transform. The $\mathcal{Z}$-transform for the combination ZOH and plant is known as the pulse transfer function and is normally found from tables. A sample of $\mathcal{Z}$-transforms is given in Table B.1. The time domain system output at times t_k is found by computing the inverse $\mathcal{Z}$-transform of $Y(z)$:

$$y(kT) = \mathcal{Z}^{-1}[Y(z)] \tag{B.61}$$

As an example of the analysis of an open-loop system having a digital controller, we consider a simple first-order plant having an analog transfer function given by

$$H_p(s) = \frac{1}{s+1}$$

We further assume a simple PD controller having the following difference equation:

$$u(kt) = K_p(kT) + K_D\left\{\frac{x(kT) - x[(k-1)T]}{T}\right\} \tag{B.62}$$

For the purposes of illustrating this procedure, we make the numerical simplification $K_p = 1$ and $(K_D/T) = 1$ yielding the following control algorithm:

$$u(kT) = 2x(kT) - x[(k-1)T] \tag{B.63}$$

The controller digital transfer function $H_c(z)$ can be found by taking the $\mathcal{Z}$-transform of the time domain model

$$\begin{aligned} H_c(z) &= 2 - z^{-1} \\ &= \frac{2z-1}{z} \end{aligned} \tag{B.64}$$

The $\mathcal{Z}$-transform of the combined ZOH/plant is given by

$$\mathcal{Z}\left[\frac{1-e^{-Ts}}{s(s+1)}\right] = \frac{1-e^{-T}}{z-e^{-T}} \tag{B.65}$$

as found from the transform tables (Table B.1) given later in this appendix.

The dynamic response for this example system is characterized by its response to a unit step

$$\begin{aligned} x(t) &= 1 \quad t \geq 0 \\ &= 0 \quad t < 0 \end{aligned}$$

The $\mathcal{Z}$-transform for this input $x(z)$ is given by

$$X(z) = \frac{z}{z-1} \tag{B.66}$$

The controller output $u(z)$ is given by

$$\begin{aligned} u(z) &= \left(\frac{2z-1}{z}\right)\left(\frac{z}{z-1}\right) \\ &= \frac{2z-1}{z-1} \end{aligned} \tag{B.67}$$

The output of this example $Y(z)$ is given by

$$\begin{aligned} Y(z) &= \left(\frac{2z-1}{z}\right)\left(\frac{1-e^{-T}}{z-e^{-T}}\right)\frac{z}{z-1} \\ &= \frac{(2z-1)(1-e^{-T})}{(z-e^{-T})(z-1)} \end{aligned} \tag{B.68}$$

It can be shown using partial fraction expansion and the table of transforms with time shift property that the time domain system output (which is $\mathcal{Z}^{-1}[Y(z)]$) is given by

$$y_k = \left[1 + \left(1 - 2e^{-T}\right)e^{-(k-1)T}\right]\, u(k-1) \tag{B.69}$$

The continuous time output is given by a smooth curve connecting all of the sampled points at times t_k. Similar procedures can be followed for analyzing the dynamic response of other open-loop systems involving other plants and controllers.

The reader will have noticed that special techniques are required to find transfer functions for sampled data systems, because no transfer function exists for an ideal sampler. This transfer function issue is also found in closed-loop systems, which we consider next.

CLOSED LOOP CONTROL

We consider first the model for a closed-loop system shown in Fig. B.7 in which, for notational simplicity, the ZOH, controller, and plant are represented by a single transfer function $H(s)$.

In this figure, a reference input $R(s)$ is the desired value for the system output $Y(s)$. An error signal $E(s)$ is obtained, which is given by

$$E(s) = R(s) - Y(s) \tag{B.70}$$

It is assumed that the output is measured via a sensor having transfer function $H_s(s)$ such that the error signal is given by

$$E(s) = R(s) - H_s(s)Y(s) \tag{B.71}$$

Since the input to the combined plant is starred (i.e., sampled), the system output is given by

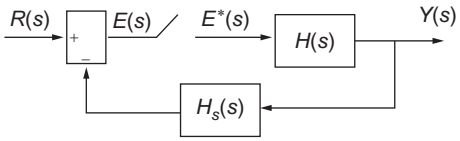


FIG. B.7 Simplified block diagram of a closed-loop system.

$$Y(s) = H(s)E^*(s) \tag{B.72}$$

The error is, then, given by

$$E^*(s) = R(s) - H(s)H_s(s)E^*(s) \tag{B.73}$$

The starred error $E^*(s)$ is given by

$$E^*(s) = \sum_{n=0}^{\infty} e(nT)e^{-nTs} \tag{B.74}$$

Taking the starred transform of both sides of the equation yields

$$E^*(s) = R^*(s) - \overline{HH_s}^*(s)E^*(s) \tag{B.75}$$

where the bar over the product indicates that the product is taken before the transform. Solving the above equation for $E^*(s)$ yields

$$E^*(s) = \frac{R^*(s)}{1 + \overline{HH_s}^*(s)} \tag{B.76}$$

The $\mathcal{Z}$-transform for this expression is found by replacing e^{Ts} with z:

$$E(z) = \frac{R(z)}{1 + \overline{HH_s}(z)} \tag{B.77}$$

The system output is given by

$$\begin{aligned} Y(z) &= H(z)E(z) \\ &= \frac{H(z)R(z)}{1 + \overline{HH_s}(z)} \end{aligned} \tag{B.78}$$

The closed-loop transfer function $H_{c\ell}(z)$ is given by

$$\begin{aligned} H_{c\ell} &= \frac{Y(z)}{R(z)} \\ &= \frac{H(z)}{1 + \overline{HH_s}(z)} \end{aligned} \tag{B.79}$$

The time domain output at the sampling instants is given by the inverse $\mathcal{Z}$-transform of $Y(z)$

$$y(nT) = \mathcal{Z}^{-1}[Y(z)] \tag{B.80}$$

Unfortunately, this result gives no direct information of $Y(t)$ at time other than $t_n = nT$.

In principle, the output at all times (i.e., $y(t)$) could be found from the analog transfer function model

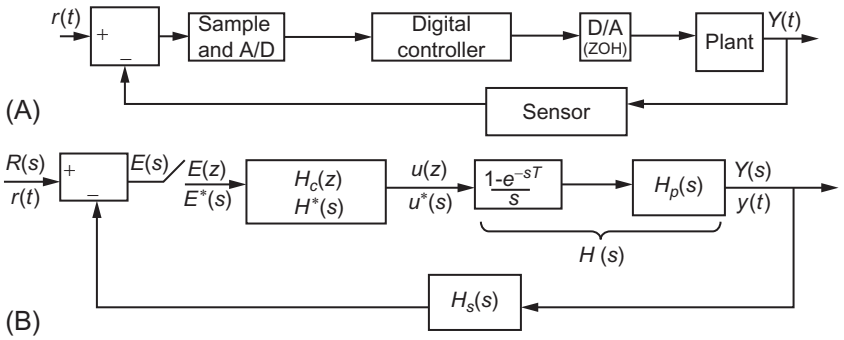


FIG. B.8 Discrete time closed-loop system. (A) Closed-loop system configuration and (B) model for closed-loop system.

$$Y(s) = \frac{H(s)R^*(s)}{1 + \overline{HH_s}^*(s)} \tag{B.81}$$

However, in practice, the difficulties in analysis using this analog model generally lead to the digital transfer function approach.

We consider next the block diagram of a closed-loop system in which all of the major components are explicitly depicted and given in Fig. B.8.

In this figure, the independent variable for each component as it is modeled is depicted. Starred transforms are also depicted. The error $E(s)$ is given by

$$\begin{aligned} E(s) &= R(s) - H(s)H_s(s)H_c{}^*(s)E^*(s) \\ Y(s) &= H(s)H_c{}^*(s)E^*(s) \end{aligned} \tag{B.82}$$

where $H(s) = \left(\dfrac{1 - e^{-sT}}{s}\right) H_p(s)$

and where $H_p(s) =$ plant transfer function.

Finding the starred transform of both sides of the first equation yields

$$E^*(s) = R^*(s) - \overline{HH_s}^*(s)H_c{}^*(s)E^*(s) \tag{B.83}$$

Solving for $E^*(s)$ yields

$$E^*(s) = \frac{R^*(s)}{1 + H_c{}^*(s)\overline{HH_s}^*(s)} \tag{B.84}$$

From this expression, the z-transfer function can be found:

$$E(z) = \frac{R(z)}{1 + H_c(z)\overline{HH_s}(z)} \tag{B.85}$$

and the output is found by taking the $\mathcal{Z}$-transform of $Y(s)$ which is given by

$$Y(z) = \frac{H_c(z)H(z)R(z)}{1 + H_c(z)\overline{HH_s}(z)} \tag{B.86}$$

The closed-loop transfer function $H_{c\ell}(z)$ is given by

$$\begin{aligned} H_{c\ell}(z) &= \frac{Y(z)}{R(z)} \\ &= \frac{H_c(z)H(z)}{1 + H_c(z)\overline{HH_s}(z)} \end{aligned} \tag{B.87}$$

For a stable control system, the poles of $H_{c\ell}(z)$ must have magnitudes <1. The procedures for developing $H_{c\ell}(z)$ from the various components in the continuous model (which are functions of s) are illustrated in the next section of this appendix.

It is a common practice in developing a closed-loop control system to work with continuous time models. These models can yield an analog controller transfer function using methods discussed in the previous chapter (e.g., P-, PI-, and PID-type control) with gain optimization to satisfy requirements as closely as possible. Many methods have been discussed in Appendix A (e.g., the use of root locus techniques to select gains that place closed-loop poles where desired). Stability analysis can also be done along with phase compensation through lead/lag networks. Once the optimized analog transfer function for the controller is found, the corresponding digital transfer function $H(z)$ can be obtained by the methods given above. For example, a digital controller system might be of the form of a PID control law. For this case, the control output at time t_n (i.e., u_n) could, for example, be given by the following function of the error ($\in_n$) (see Eq. B.92)

$$u_n = K_p \in_n + \frac{K_D}{T}[\in_n - \in_{n-1}] + K_I T \sum_{k=1}^{K} \frac{\in_{n-k} + \in_{n-(k+1)}}{2} \tag{B.88}$$

where the third term above represents a discrete time approximation to the integral of the error. The control transfer function can be found by taking the $\mathcal{Z}$-transform of the control law, making use of the time shift property. For the above control law, the control variable $u(z)$ is given by

$$u(z) = \left[K_p + \frac{K_D}{T}\left(1 - z^{-1}\right) + K_I T \sum_{k=1}^{K} \frac{z^{-k} + z^{-(k+1)}}{2} \right] E(z) \tag{B.89}$$

The control transfer function $H_c(z)$ is then given by

$$H_c(z) = \frac{u(z)}{E(z)} \tag{B.90}$$

where $E_z = \mathcal{Z}\{\in_{\boldsymbol{k}}\}$ and where

$$H_c(z) = K_p + \frac{K_D}{T}\left(1 - z^{-1}\right) + K_I T \sum_{k=1}^{K} \frac{z^{-k} + z^{-(k+1)}}{2} \tag{B.91}$$

EXAMPLE DISCRETE TIME CONTROL SYSTEM

We illustrate the above discrete time control methodology with a specific example. In this example, we avoid using the fictional starred transforms that were introduced simply to explain the theory of discrete time systems. Fig. B.9 is a block diagram of a plant having transfer function $H_p(s)$ and a discrete time digital control system.

This system controls the plant output variable $y(t)$ in response to the command input $x(t)$. The error signal $\in(t)$ is given by

$$\epsilon(t) = x(t) - y(t) \tag{B.92}$$

where an ideal sensor (i.e., $H_s(s) = 1$) is assumed that provides the feedback signal. The sample and A/D converter provide discrete time samples $\in_k$ of $\in$ at times $t_k = kT$:

$$\epsilon_k = \epsilon(t_k) \quad k = 1, 2, \cdots \tag{B.93}$$

where T is the sample period. The digital control generates control signal u_k in accordance with the desired control algorithm. It is assumed that the digital controller has a D/A converter such that u_k are voltage pulses that are sent to the ZOH. A ZOH and filter convert the samples u_k to a piecewise continuous time control signal $\bar{u}(t)$, which operates the actuator that drives the plant. For the present example, the plant is represented by a continuous time model that has transfer function (that was used for an example plant in Appendix A) $H_p(s)$:

$$H_p = \frac{K_a}{s(s + p_1)} \tag{B.94}$$

where p_1 is the first-order pole and K_a is the actuator constant.

In order to reduce computational complexity, a proportional-only controller having proportional gain K_p and relatively simple plant model are assumed. However, the following procedure is followed regardless of controller and plant complexity.

The forward path transfer function $H_F(s)$ of the continuous time system is given by

$$H_F = K_p \left(\frac{1 - e^{-Ts}}{s} \right) H_p(s) \tag{B.95}$$

The $\mathcal{Z}$-transform $H_F(z)$ is given by

$$H_F(z) = \mathcal{Z}\left\{ K_p \left(\frac{1 - e^{-Ts}}{s} \right) H_p(s) \right\} \tag{B.96}$$

$$= K_p \left(1 - z^{-1}\right) \mathcal{Z}\left[\frac{H_p(s)}{s} \right] \tag{B.97}$$

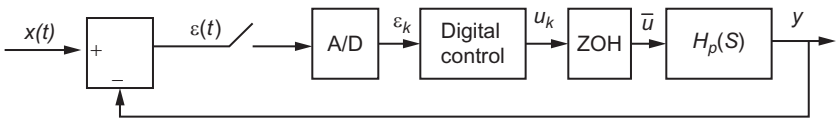


FIG. B.9 Block diagram of example discrete time closed-loop control system.

From the time shift property of the $\mathcal{Z}$-transform, $e^{-Ts} \rightarrow z^{-1}$.

In order to evaluate the $\mathcal{Z}$-transform of $H_p(s)/s$ (using transform tables), specific values are chosen for the following parameters:

$$\begin{aligned} K_a &= 4500 \\ p_1 &= 351.2 \\ T &= 0.001\,s \\ K_p &= 14.5 \end{aligned}$$

The procedure for obtaining the $\mathcal{Z}$-transform of the function $H_p(s)/s$ is to represent it in a partial fraction expansion of the form

$$\frac{H_p(s)}{s} = \frac{\alpha_1}{s+p_1} + \frac{\alpha_2}{s+p_2} + \cdots \frac{\alpha_n}{s+p_n} \tag{B.98}$$

However, in the present example, the function $H_p(s)/s$ has a double pole at $s=0$. In general, for a pole of multiplicity r at $s=-s_r$, the partial fraction expansion of $H_p(s)/s$ becomes

$$\frac{H_p(s)}{s} = \frac{A_r}{(s+s_r)^r} + \frac{A_{r-1}}{(s+s_r)^{r-1}} + \cdots \frac{A_1}{s+s_r} + \sum_{j=1}^{J} \frac{\alpha_j}{s+s_j} \tag{B.99}$$

where

$$A_r = \left[(s+s_r)^r \frac{H_p(s)}{s} \right] \Bigg|_{s=-s_r} \tag{B.100}$$

$$A_{r-k} = \frac{1}{(r-k)!} \frac{d^k}{ds^k} \left[(s+s_r)^k \left(\frac{H_P(s)}{s} \right) \right] \Bigg|_{s=-sr} \tag{B.101}$$

and where the poles s_j $(j=1,2,\cdots,J)$ are simple poles.

The $\mathcal{Z}$-transform of each term can be found from the tables once the sample period T has been determined. For example, the $\mathcal{Z}$-transform of each simple pole is given by

$$\mathcal{Z} = \left[\frac{\alpha_j}{s+s_j} \right] = \frac{\alpha_j z}{z - e^{-s_j T}} \tag{B.102}$$

Table B.1 also gives the $\mathcal{Z}$-transform for terms associated with multiple poles.

The present example discrete time control system involves the $\mathcal{Z}$-transform of the following:

$$\frac{H_p(s)}{s} = \frac{K_a}{s^2(s+p_1)} \tag{B.103}$$

The partial fraction expansion of this function yields three terms given by

$$\frac{K_a}{s^2(s+p_1)} = \frac{K_a}{p_1 s^2} - \frac{K_a}{p_1^2 s} + \frac{K_a}{p_1^2(s+p_1)} \tag{B.104}$$

The $\mathcal{Z}$-transform for each of the terms above can be determined using the Table of Transforms in this appendix. Then the individual terms can be combined to yield the desired $H_F(z)$.

The $\mathcal{Z}$-transform of $H_F(z)$ (Eq. B.97) is evaluated to be

$$H_F(s) = \frac{0.029z + 0.0257}{z^2 - 1.7038z + 0.7038} \tag{B.105}$$

The closed-loop transfer function $H_{c\ell}(z)$ is given by

$$H_{c\ell}(z) = \frac{Y(z)}{X(z)} \tag{B.106}$$

$$H_{c\ell}(z) = \frac{H_F(z)}{1 + H_F(z)} \tag{B.107}$$

It is left as an exercise for the reader to show that $H_{c\ell}$ is given by

$$H_{c\ell}(z) = \frac{0.029z + 0.0257}{z^2 - 1.675z + 0.7297} \tag{B.108}$$

This closed-loop control system is stable since its poles (z_2 and z_3 below) satisfy $|z| < 1$, thereby assuring stability of $H_{c\ell}(z)$

The dynamic response of the closed-loop discrete time system is illustrated by its response to a unit step. The $\mathcal{Z}$-transform of the unit step $X(z)$ is given by

$$X(z) = \frac{z}{z - 1} \tag{B.109}$$

The system output $Y(z)$ is given by

$$Y(z) = H_{c\ell}(z)X(z) \tag{B.110}$$

$$= \frac{z(0.029z + 0.0257)}{(z - 1)(z^2 - 1.675z + 0.7297)} \tag{B.111}$$

The three poles of $Y(z)$ are given by

$$\left.\begin{aligned} z_1 &= 1 \quad \text{(pole of input step } X(z)\text{)} \\ z_2 &= 0.8374 - 0.1681i \\ z_3 &= 0.8374 + 0.1681i \end{aligned}\right\} \text{ poles of } H_{c\ell}(z)$$

The output sequence $\{y_k\}$ can be found using the procedures described earlier beginning with a partial fraction expansion of $Y(z)$:

$$Y(z) = \frac{\alpha_1}{z - z_1} + \frac{\alpha_2}{z - z_2} + \frac{\alpha_3}{z - z_3} \tag{B.112}$$

$$= \frac{\alpha_1 z^{-1}}{1 - z_1 z^{-1}} + \frac{\alpha_2 z^{-1}}{z - z_2 z^{-1}} + \frac{\alpha_3 z^{-1}}{z - z_3 z^{-1}} \tag{B.113}$$

where $\alpha_1 = 1.0000$, $\alpha_2 = -0.4855 + 0.2486i$, and $\alpha_3 = -0.4855 - 0.2486i$.

The second and third partial fractions are analytic outside the unit circle in the complex z-plane since the poles have magnitudes satisfying

$$|z_j| < 1 \quad j = 2,3$$

Thus, the sequence $\{y_k\}=0$ for $k<0$ as expected. The partial fractions can be rewritten in a Taylor series in powers of z^{-1}:

$$\frac{\alpha_j z^{-1}}{1-z_j z^{-1}}=\alpha_j\sum_{m=0}^{\infty} z_j^m z^{-(m+1)} \tag{B.114}$$

By replacing $m+1=k$ and summing over k beginning with $k=1$, the partial fraction can be written in the form of a $\mathcal{Z}$-transform, and the coefficients of z^{-k} become y_k. The output sequence terms are given by

$$y_k=\sum_{j=1}^{3}\alpha_j z_j^{k-1} \qquad k=1,2,3,\cdots \tag{B.115}$$

The output $y(t)$ is given only at the sample periods t_k:

$$y(t)\big|_{t=t_k}=y_k \tag{B.116}$$

Thus, the system continuous time output is only correctly given at these sample times. However, if the sample period (T) is sufficiently short, these samples represent $y(t)$ with enough accuracy for most practical circumstances. However, if the sample period is increased, the accuracy of the sampled output is degraded. This point is demonstrated in Fig. B.10, which plots the output $y(t_k)$ for $T=0.002$ sec (i.e., 2 msec) and for the continuous time step response of $H_p(s)$ using the same parameters as given above. The solid curve in Fig. B.10 is a superposition of the continuous time $y(t)$ and y_k for $T=0.001$ sec. Note

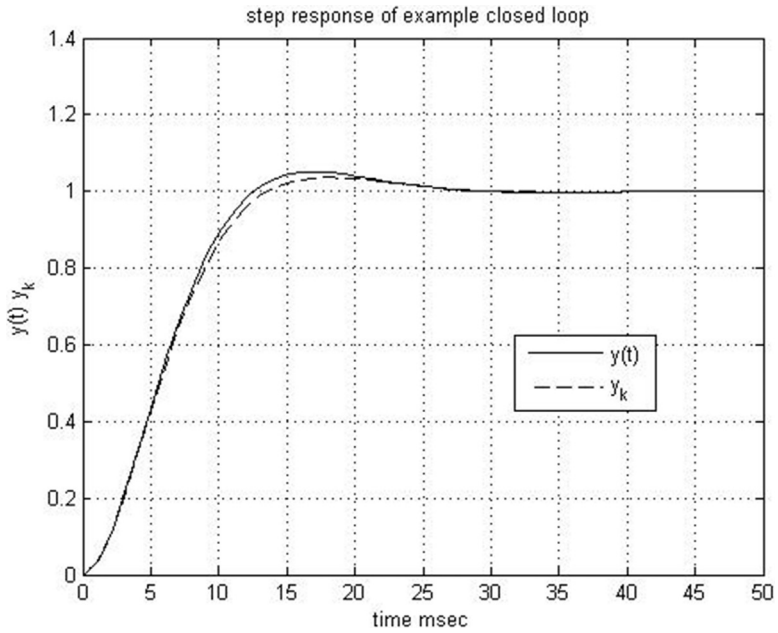


FIG. B.10 Unit step response of example discrete time closed-loop control system.

from Fig. B.10 that the proportional gain K_p is sufficient to ensure that the asymptotic error is negligible.

The output for the 2 msec sample period agrees very closely with the output of the corresponding continuous time output. However, for the longer sample period, the errors can become quite large.

The same procedure is used for finding the closed-loop transfer function and dynamic response sequence $\{y_k\}$ regardless of the complexity of the plant and controller transfer function complexity and the input waveform $x(t)$. Finding the $\mathcal{Z}$-transform of the discrete time forward path is normally best accomplished with respect to published tables of $\mathcal{Z}$-transforms (also available online). The closed-loop transfer function is found, and the output $Y(z)$ for any given input $X(z)$ is given by

$$Y(z) = H_{c\ell}(z)X(z) \tag{B.117}$$

The time sequence $\{y_k\}$ is, then, found by expanding $Y(z)$ in a partial fraction model. Each term in the partial fraction model can be written as a Taylor series in powers of z^{-k}. The coefficients of all multiples of z^{-k}, when combined, yield the time sequence $\{y_k\}$, which yields a sampled version of the continuous time system output $y(t)$.

SUMMARY

This appendix has presented the general systems theory for discrete time digital systems. Specific applications in automotive electronic systems are presented throughout this book. The same basic procedures presented in this appendix apply to each of these exemplary systems.

APPENDIX

DYNAMICS IN MOVING COORDINATE SYSTEMS

C

This appendix derives the equations of motion of a point in a coordinate system that translates and rotates relative to a reference coordinate system. This set of equations is applied to the vehicular angular rate sensor (ARS) presented in Chapter 5. We begin with a generic system and demonstrate the origin in the motion of the so-called Coriolis acceleration. The general dynamic model then is applied to the specific configuration of the ARS.

The coordinate systems and the dynamic motion of a point in the moving coordinate system are depicted in Fig. C.1.

The reference or "fixed" coordinate system's denoted X,Y,Z. The moving coordinate system is denoted x,y,z, and the origin of this system is located at vector position $\bar{R}$ in the X,Y,Z coordinates. The moving coordinate system is translating at velocity $\dot{\bar{R}}$ and rotating about an axis at angular velocity vector denoted $\overline{\omega}$. The direction of $\overline{\omega}$ is the axis about which the x,y,z coordinate system is rotating, and the rate of rotation is given by

$$\|\overline{\omega}\|$$

The point whose motion is being modeled, which is denoted P in Fig. C.1, is located at vector position $\bar{p}$ in the x,y,z coordinate system. This same point is located at vector position $\bar{r}$ in the X,Y,Z coordinate system. By vector addition, $\bar{r}$ is given by

$$\bar{r} = \bar{R} + \bar{p}$$

Fig. C.1 also depicted unit vectors $\hat{x},\hat{y},\hat{z}$ in the moving coordinate system and unit vectors $\hat{X},\hat{Y},\hat{Z}$ in the fixed coordinate system such that $\bar{r}$ is given by

$$\bar{r} = X\hat{X} + Y\hat{Y} + Z\hat{Z} + x\hat{x} + y\hat{y} + z\hat{z}$$

The calculation of the motion is done via time derivatives of the variables that are denoted by an over dot. The vector velocity of the point P is the time derivative of $\bar{r}$, which is given by

$$\dot{\bar{r}} = \dot{\bar{R}} + \dot{x}\hat{x} + \dot{y}\hat{y} + \dot{z}\hat{z} + x\dot{\hat{x}} + y\dot{\hat{y}} + z\dot{\hat{z}}$$

The time derivatives of the unit vectors are nonzero because the moving coordinate system has a rotation expressed by $\overline{\omega}$. These time derivatives are related to the angular velocity vector $\overline{\omega}$ as given by

$$\begin{aligned}\dot{\hat{x}} &= \overline{\omega} \times \hat{x}\\ \dot{\hat{y}} &= \overline{\omega} \times \hat{y}\\ \dot{\hat{z}} &= \overline{\omega} \times \hat{z}\end{aligned}$$

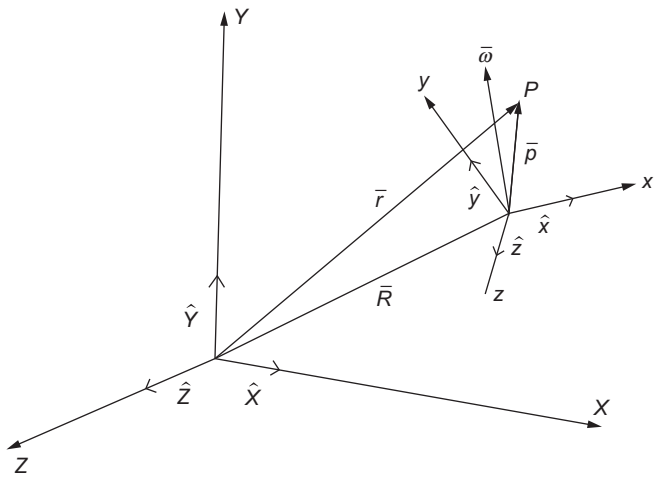


FIG. C.1 Coordinate systems for derivation of Coriolis acceleration.

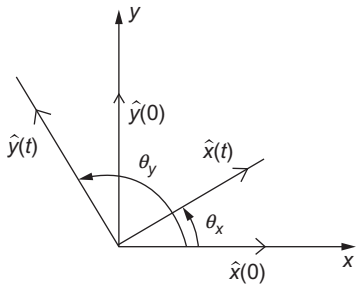


FIG. C.2 Illustration of $\dot{\hat{x}}$, $\dot{\hat{y}}$.

The general proof of these relationships is left as an exercise for the interested reader. However, the first two equations can be illustrated by assuming rotation is about the z-axis such that $\overline{\omega} = \omega\hat{z}$. Fig. C.2 illustrates the motion of $\hat{x}$ and $\hat{y}$ for a constant angular velocity ω.

In Fig. C.2, the angles are given by

$$\theta_x = \omega t \text{ and } \theta_y = \omega t + \frac{\pi}{2}$$

For the assumed constant motion $\omega =$ constant, the unit vector $\hat{x}(t)$ and $\hat{y}(t)$ relative to an arbitrary starting orientation $\hat{x}(0)$, $\hat{y}(0)$ are given by

$$\hat{x}(t) = \cos(\omega t)\hat{x}(0) + \sin(\omega t)\hat{y}(0)$$

$$\hat{y}(t) = \cos\left(\omega t + \frac{\pi}{2}\right)\hat{x}(0) + \sin\left(\omega t + \frac{\pi}{2}\right)\hat{y}(0)$$

The derivatives are given by

$$\dot{\hat{x}} = -\omega sin(\omega t)\hat{x}(0) + \omega\cos(\omega t)\hat{y}(0)$$

$$\dot{\hat{y}} = -\omega\sin\left(\omega t + \frac{\pi}{2}\right)\hat{x}(0) + \omega\cos\left(\omega t + \frac{\pi}{2}\right)\hat{y}(0)$$

The vector products are given by

$$\begin{aligned}\overline{\omega}\times\hat{x} &= \omega\cos(\omega t)\,\hat{y}(0) - \omega\sin(\omega t)\hat{x}(0)\\ &= \dot{\hat{x}}\\ \overline{\omega}\times\hat{y} &= \omega\cos\left(\omega t + \frac{\pi}{2}\right)\hat{y}(0) - \omega\sin\left(\omega t + \frac{\pi}{2}\right)\hat{x}(0)\\ &= \dot{\hat{y}}\end{aligned}$$

A similar derivation can be made for any arbitrary $\overline{\omega}$ vector. Substituting the general relationships for the time derivatives of the unit vectors yields the following equation for $\dot{\bar{r}}$:

$$\dot{\bar{r}} = \dot{\bar{R}} + \dot{\bar{p}}_r + \overline{\omega}\times\bar{p}$$

where $\dot{\bar{p}}_r = \dot{x}\hat{x} + \dot{y}\hat{y} + \dot{z}\hat{z} =$ relative translational velocity of P in x,y,z.

The acceleration of the point P is computed by taking the second time derivative of $\bar{r}$ and is denoted $\ddot{\bar{r}}$

$$\begin{aligned}\ddot{\bar{r}} &= \ddot{\bar{R}} + \ddot{\bar{p}}\\ &= \ddot{\bar{R}} + \ddot{\bar{p}}_r + \frac{d}{dt}(\overline{\omega}\times\bar{p})\\ &= \ddot{\bar{R}} + \ddot{x}\hat{x} + \ddot{y}\hat{y} + \ddot{z}\hat{z} + \dot{x}\dot{\hat{x}} + \dot{y}\dot{\hat{y}} + \dot{z}\dot{\hat{z}}\\ &\quad + \dot{\overline{\omega}}\times\bar{p} + \overline{\omega}\times(\dot{x}\hat{x} + \dot{y}\hat{y} + \dot{z}\hat{z})\\ &\quad + \overline{\omega}\times\left(x\dot{\hat{x}} + y\dot{\hat{y}} + z\dot{\hat{z}}\right)\end{aligned}$$

Simplifying the expression for $\ddot{\bar{r}}$ including the substitution of the vector products for the unit vector time derivatives yields

$$\ddot{\bar{r}} = \ddot{\bar{R}} + \overline{\omega}\times(\overline{\omega}\times\bar{p}) + \dot{\overline{\omega}}\times\bar{p} + \ddot{\bar{p}}_r + 2\overline{\omega}\times\dot{\bar{p}}_r$$

The first three terms of this expression are the absolute acceleration of a point moving with the x,y,z coordinates. The fourth term is the acceleration of a point relative to the x,y,z coordinate system. The final term is the Coriolis acceleration.

If there is a mass m located at P, the equation of motion for this mass gives the force $\bar{F}$ on m and is given by

$$\begin{aligned}\bar{F} &= m\ddot{\bar{r}}\\ &= m\ddot{\bar{R}} + m[\overline{\omega}\times(\overline{\omega}\times\bar{p})] + m\dot{\overline{\omega}}\times\bar{p} + m\ddot{\bar{p}}_r + 2m\overline{\omega}\times\dot{\bar{p}}_r\end{aligned}$$

It should be noted that the Coriolis force (i.e., the last term) is in a direction orthogonal to $\overline{\omega}$ and $\dot{\bar{p}}_r$.

This equation of motion can be applied to the ARS described in Chapter 5 and depicted in Fig. 5.29. With reference to Fig. C.1, the variable $\bar{r}$ is the vector position of the differential portion of the tine T_{TR} of width dz. The point P in the derivation of the equation of motion is the geometric center of the differential length (dz). The vector $\bar{R}$ in the equation of motion above is the constant vector $\bar{z}$ in this figure. The vector $\bar{p}$ in the equation of motion becomes $\bar{x}_T$ in the ARS such that

$$\bar{p} = x_t \hat{x}$$

where x_t = position of the differential

The acceleration of the point P in the ARS is a special case version of the general case equation of motion such that the acceleration of the point P in the ARS is given by

$$\ddot{\bar{r}} = \ddot{\bar{x}}_T + \bar{\Omega} \times (\bar{\Omega} \times \bar{x}_T) + \dot{\bar{\Omega}} \times \bar{x}_T + 2\bar{\Omega} \times \dot{\bar{x}}_T$$

The equation of motion is given by the inertial acceleration force

$$\bar{F} = dm\ddot{\bar{r}}$$

where $dm = \rho_Q w t_s dz$ = mass of differential element (see Eq. 5.69) in Chapter 5).

This equation of motion is presented in Chapter 5 and is the basis for explaining the ARS theory of operation in measuring the angular rate $\bar{\Omega}$ of the structure about its z-axis. One of the important aspects of this theory is the out-of-plane direction of the Coriolis component of the force vector $\bar{F}$.

APPENDIX

FDI FEEDBACK MATRIX DESIGN

D

This appendix presents the algorithm for the design of the feedback matrix in the failure detection and identification (FDI) method of model-based diagnostics in Chapter 11. The feedback matrix determines the output error residual vector and, in particular, its direction in the output vector space due to failure or degradation in the performance of the component for which the FDI is designed. In Chapter 11, a model for the system in which the failed/failing component is incorporated is repeated below = 6p

$$\dot{x} = Ax + Bu + f_i$$

where f_i = failure event vector

The FDI is based on the state estimator $\hat{x}$, which is given by the solution to the following equation:

$$\dot{\hat{x}} = A\hat{x} + Bu + D(y - \hat{y})$$

where $y = Cx$

$$\hat{y} = C\hat{x}$$

An algorithm for designing the D matrix is outlined below in which the inputs are the failure event vector f_i and λ_d, which is the eigenvalue for the detection space. For an N-dimensional state vector, the design requires N eigenvalues to be chosen, one of which is λ_d. The remaining eigenvalues are incorporated in a matrix that is denoted P and is part of the design algorithm.

$$P = [\lambda_d \lambda_2 \lambda_3 \cdots \lambda_N]^T$$

These eigenvalues determine the dynamic response of the state estimator and must be negative or, if complex, must be in complex conjugate pairs with negative real parts. Normally, the choice for λ_n is for all to be negative real numbers. With these parameters and inputs, the algorithm consists of the following steps:

a) $C_f = Cf_i$

b) $C_p = \left(C_f^T C_f\right)^{-1} C_f^T$

c) $C_s = I(N) - C_f C_p$ where $I(N) = N$-dimensional identity matrix

d) $C_i = C_s C$

e) $D_d = (-\lambda_d f_i + A f_i) C_P$

f) $G_p = A - D_d C$

g) $K = \text{place}\left(G_p^T, C_i^T, P\right)$

h) $D_P = K^T C_s$

i) $D = D_d + D_P$

In this algorithm, the function "place" is a MATLAB function for assigning eigenvalues to a vector space.

As an illustration of the FDI method of detecting actuator component failures using FDI method, an example is presented next of a vehicle equipped with an automatic steering system. The application of automatic steering is explained in Chapter 12 with respect to various levels of autonomous vehicles. It is applied in contemporary vehicles for automatic lane following. In this system, the front wheel steering is implemented with an electromechanical actuator. The details of the electromechanical actuators of various types are explained in Chapter 5 and are not significant for the purposes of explaining the FDI operation. It is shown in Chapter 11 that the failure event vector for an electromechanical actuator is given by

$$f_i = B\delta K_a \mathrm{v}_s$$

The magnitude of the degradation in actuator performance can be represented by a degradation index d given by

$$d = \delta K_a / K_a$$

The direction of f_i for this actuator failure is in the direction of B. Any scalar multiple of B can be used as the event vector for designing the D matrix.

For the illustrative example of the FDI, the model for the plant is obtained from Eqs. (7.117) through (7.120) of Chapter 7 except that E has only the first column for the present example. Using the vehicle parameters given in association with these models for a vehicle traveling at 30 m/s, the A and B matrices are given by

$$A = \begin{bmatrix} -4.0449 & -30.2981 & 2.0702 & 30.6074 \\ -0.1821 & -6.5380 & 0.0809 & 1.1956 \\ 3.2428 & 0.4836 & -8.6259 & -127.5307 \\ 0 & 0 & 1.0000 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 50.2223 \\ 64.3993 \\ -42.5928 \\ 0 \end{bmatrix}$$

The system input $u = \delta_F$, where δ_F = front wheel steering angle.

It is assumed for the present example that all state variables are measured with separate sensors such that the C matrix is a rank 4 identity matrix. The detection space eigenvalue $\lambda_d = -2$. The remaining three eigenvalues are given in the P matrix below.

$$P = [\lambda_d, -3, -4, -5]^T$$

Following the algorithm given above, the D matrix for this steering actuator FDI example is given by

$$D = \begin{bmatrix} 0.6341 & -32.4952 & 1.9072 & 30.5519 \\ -0.2407 & -4.1956 & 0.5294 & 1.1646 \\ 2.3618 & 1.8252 & -5.6362 & -127.6151 \\ -0.0485 & -0.0219 & 0.9097 & 3.9889 \end{bmatrix}$$

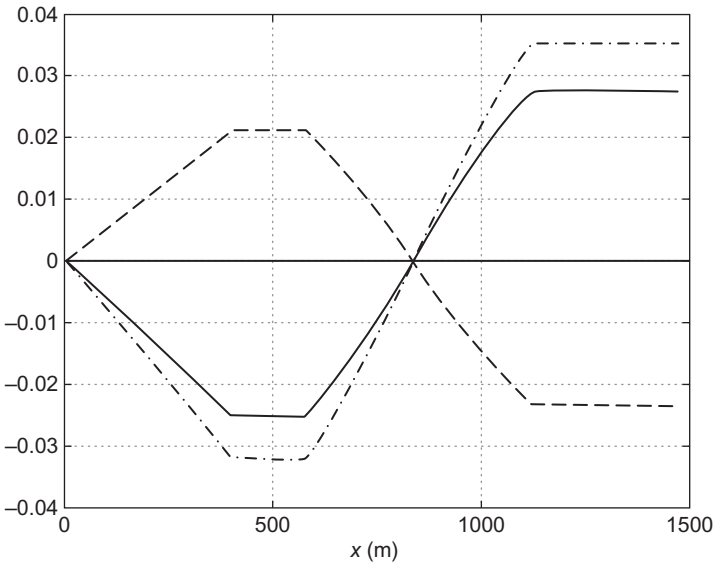


FIG. D.1 Error residual components for steering actuator failure.

In addition to the simulation results presented in Chapter 11, it is illustrative to plot the error vector components versus time for a vehicle involved in a maneuver following an S curve on a road. Fig. D.1 presents the four components of $e(f_i)$ for a 20% calibration reduction along this maneuver.

The error residual vector has four components as given below

$$e = [e_1, e_2, e_3, e_4]^T$$

In Fig. D.1, the e_1 component is the solid line that deviates from 0. The component e_2 is the $-$ curve. The e_3 component is the $-\cdot$ curve. The e_4 component is 0. The angle $\delta\phi$ relative to the theoretical angle for the steering actuator failure is $<10^{-6}$ radians along the entire contour. The degradation index at time sample t_k is denoted d_k and is given by

$$d = \|GE(k)\|/(\|B\| \cdot |\delta_F(k)|) \quad |\delta_F| > 0$$

where $G = A^k - DC$.

For the example above with a 20% calibration reduction, $d_k = 0.2$.

The actual steering input follows a curve identical to each component but with a peak amplitude of $\sim$3 degrees.

Fig. D.2 is a plot of the contour of the maneuver followed by the vehicle in this simulation.

These two plots demonstrate the ability to detect steering actuator degradation in calibration along a maneuver involving significant changes in the input to the system.

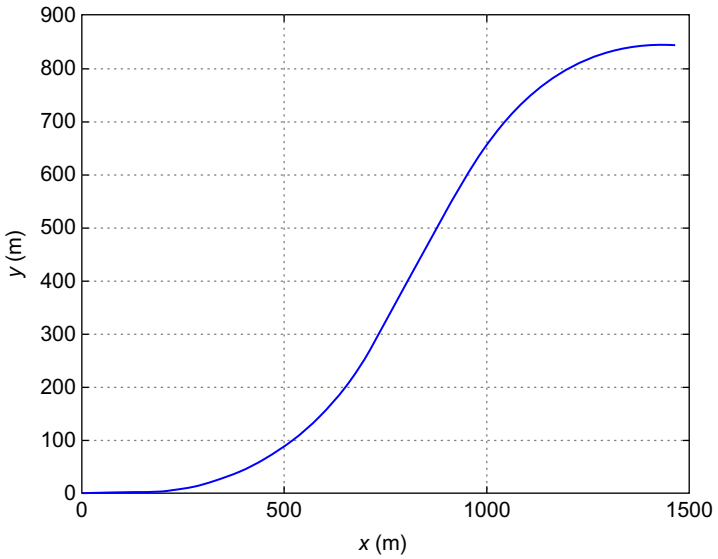


FIG. D.2 Simulation maneuver.

COORDINATE TRANSFORMATION E

The transformation of coordinates occurs frequently in vehicular electronic systems for such applications as navigation, position location, and modeling of vehicle motion. Although a transformation of coordinates is applicable to any type of coordinate system (e.g., Cartesian, cylindrical, and spherical), the process is, perhaps, most readily understandable with Cartesian coordinates. For the purpose of explaining such a transformation of coordinates, a representative configuration is depicted in Fig. E.1.

In Fig. E.1, the two coordinate systems are x,y,z and x',y',z', respectively. Also depicted in this figure is a set of coordinates x_T,y_T,z_T that are parallel, respectively, to x,y,z and have the same origin as x',y',z'. The origins of this latter pair of coordinates are translated relative to the origin of x,y,z coordinates by a translation vector $\bar{T}$, which is given by the coordinates of the origins of x',y',z' that are denoted as x_o,y_o,z_o from which is given by

$$\bar{T} = [x_o y_o z_o]^T \tag{E.1}$$

Also depicted in Fig. E.1 is a point that is specified in x',y',z' by vector $\bar{X}'_P$. The components of $\bar{X}'_P$ in primed coordinates are denoted x'_P,y'_P,z'_P with which $\bar{X}'_P$ is given by

$$\bar{X}'_P = \left[x'_P, y'_P, z'_P\right]^T \tag{E.2}$$

The vector position of P in x,y,z is given by $\bar{X}_P$ where

$$\bar{X}_P = [x_P, y_P, z_P]^T$$

and where x_P,y_P,z_P are the coordinate components of P in x,y,z coordinates.

The transformation of coordinate from $\bar{X}'_P$ to $\bar{X}_P$ is given by

$$\bar{X}_P = \bar{T} + R\bar{X}'_P$$

where R is a matrix that transforms to a vector in the intermediate coordinates $\bar{X}_{TP}$where

$$\begin{aligned}\bar{X}_{TP} &= R\bar{X}'_P \\ &= [x_{TP} y_{TP} z_{TP}]^T\end{aligned}$$

This latter transformation can be achieved by representing the orientation of $\bar{X}_{TP}$ as a set of three rotations about coordinate axes through a set of angles that are called "Euler Angles," which are denoted as ϕ,θ,ψ. The matrix R can be represented by the product of three matrices:

$$R = R_1(\psi)R_2(\theta)R_3(\phi)$$

The matrix R and its three components can be achieved by a sequence of rotations in the following order:

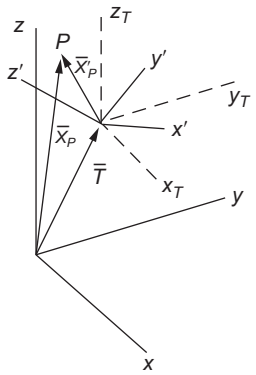


FIG. E.1 Illustration of coordinate transformation.

1) rotation about the z' axis of angle ψ
2) rotation about the transformed y axis from step 1 of angle θ
3) rotation about the transformed x axis from step 2 of angle ϕ

The components of R are given below:

$$R_1(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R_2(\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \tag{E.3}$$

$$R_3(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix}$$

The final vector $\bar{X}_P$ is obtained with the following transformation:

$$\bar{X}_P = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} + R_1(\psi)R_2(\theta)R_3(\phi) \begin{bmatrix} x'_p \\ y'_p \\ z'_p \end{bmatrix} \tag{E.4}$$

As an illustration of coordinate transformation, the coordinates of the rear wheels during an APPS maneuver are computed relative to the vehicle CG position (x_p, y_p) in a Cartesian coordinate system x,y in which the x axis is parallel to the curb and the y coordinate is orthogonal to the curb. Fig. E.2 depicts the associated coordinates. This illustration of coordinate transformation involves rotation about the z axis to an intermediate coordinate's system. This rotation has a two-dimensional rotation matrix with a single angle variable denoted as ϕ in Fig. E.2. The origin of this set of intermediate coordinates is at the

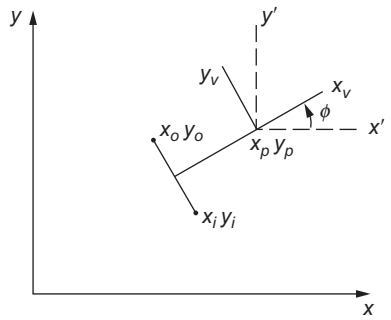


FIG. E.2 Coordinates of APPS vehicle rear wheels.

vehicle CG. The transformation of the coordinates of the inner and outer rear wheels to the x,y coordinate system involves the translation of the vehicle CG to the (x,y) coordinates as presented later.

In Fig. E.2, the vehicle body coordinates are denoted (x_v,y_v) in which the origin is at the vehicle CG and x_v is the longitudinal axis. The vehicle is depicted partway through an APPS maneuver with the vehicle axis at an angle ϕ relative to the x axis (see Chapter 12 for the explanation of the angle ϕ). The intermediate coordinate system (x',y') is depicted for which the origin is at the vehicle CG and in which x',y' are parallel, respectively, to x and y.

In Fig. E.2, the inner and outer wheel coordinates (in x,y) are denoted x_i,y_i and x_oy_o, respectively. In the vehicle coordinate system, these inner and outer wheel coordinates are denoted x_{iv},y_{iv} and x_{ov},y_{ov} are given by

$$x_{iv} = -b \quad y_{i2} = -\frac{C}{2}$$
$$x_{ov} = -b \quad y_{ov} = \frac{C}{2}$$

where $c =$ lateral distance between rear wheels

In vector notation, the two wheel positions in vehicle coordinates are given by $\bar{X}_{iv}$ and $\bar{X}_{ov}$:

$$\begin{aligned} \bar{X}_{iv} &= [x_{iv}y_{iv}]^T \\ \bar{X}_{ov} &= [x_{ov}y_{ov}]^T \end{aligned} \tag{E.5}$$

Similarly, the vector position in x',y' are denoted $\bar{X}_i'$ and $\bar{X}_o$:

$$\begin{aligned} \bar{X}_i' &= [x_i', y_i']^T \\ \bar{X}_o &= [x_o', x_o']^T \end{aligned} \tag{E.6}$$

The transformations from vehicle to intermediate coordinates are given by

$$\bar{X}_i' = R(\phi)\bar{X}_{iv} \tag{E.7}$$

$$\bar{X}_o' = R(\phi)\bar{X}_{ov} \tag{E.8}$$

where

$$R(\phi) = \begin{bmatrix} \cos\phi & -\sin\phi \\ -\sin\phi & \cos\phi \end{bmatrix}$$

The transformation from the intermediate coordinates to x,y coordinates is by means of a translational vector $\bar{X}_T$. In vector notation, the position of the inner and outer wheels in x,y is denoted $\bar{X}_i$ on $\bar{X}_o$ and is given by

$$\begin{aligned}\bar{X}_i &= R(\phi)\bar{X}_{iv} + T \\ \bar{X}_o &= R(\phi)\bar{X}_{ov} + T\end{aligned} \tag{E.9}$$

where T is the vector position of the origin of $x_v y_v$ in x,y coordinates and is given by

$$T = \begin{bmatrix} x_p \\ y_p \end{bmatrix}$$

The transformation of Eq. (E.9) was used to calculate the coordinates of the inner and outer wheels in the section of Chapter 12 devoted to the example APPS maneuver.

One of the important coordinate transformations, particularly for navigation and for displaying electronic maps as explained in Chapter 8, is to transform from Cartesian coordinates to geodetic. In the latter, a point in geodetic coordinates, which are spherical coordinates, is given by radial distances ρ_E from the origin (which in geodetic coordinates is the center of the earth) and the angular position θ_E along a great circle through the axis of rotation measured from the equatorial plane. The third coordinate is the angular position ϕ_E on a circle formed by the intersection with an earth-centered sphere that passes through the point and a plane that is orthogonal to the axis of rotation and that also contains the point being represented. The origin of angle ϕ_E is a point on the circle that is on a plane through the earth axis of rotation and Greenwich. The angle θ_E is called the latitude, and the angle ϕ_E is called the longitude of the point, and with these angular coordinates, the point can be displayed at a set of map coordinates. The radial position ρ_E is normally given by the elevation of the point above or below an earth-centered sphere that represents "mean sea level" (MSL). However, for land-vehicle navigation in which the map is displayed on the flat panel display (see Chapter 8), the elevation need not be given since it is a unique function of θ_E and ϕ_E determined by the earth terrain.

On the other hand, there are some experimental cars that have the capability of converting to small aircraft. Such vehicles are commonly called "flying cars." For such vehicles, the elevation is important.

The representation of vehicular position in the map or geodetic coordinates is computed from the position in an ECEF Cartesian coordinate system that has z axis along the earth axis of rotation. In the ECEF coordinate system, the vehicle position is given by x_E,y_E,z_E. The same point in geodetic spherical coordinates is given by (ρ_E,θ_E,ϕ_E). The relationship between these two coordinate representations of a point (e.g., vehicle position) is given by

$$\begin{aligned} z_E &= \rho_E \sin\theta_E \\ x_E &= \rho_E \cos\theta_E \cos\phi_E \\ y_E &= -\rho_E \cos\theta_E \sin\phi_E \end{aligned}$$

where the minus sign in y_E results from the ECEF being a right-handed Cartesian coordinate system.

The point being represented can be put in the ECEF coordinate from any arbitrarily oriented coordinate system using the transformation given in Eq. (E.4). An example of the combined transformation of a point (or vector) from an arbitrary Cartesian coordinate system to geodetic is presented in Appendix F with respect to converting a calculated vehicle position (using GPS) to geodetic (map) coordinates.

APPENDIX

GPS THEORY

F

In this appendix, the determination of the vehicle position from satellite pseudorange measurements is explained first by trilateration then by the far superior method of a Kalman filter. Trilateration as a method of calculating the position of a point in a three-dimensional Cartesian coordinate system has existed long before GPS and is not practical for the precision required for GPS navigational applications. Nevertheless, it is presented here in a highly simplified illustration of position determination from distance measurements to three or more points of known location.

To illustrate the contrast between trilateration and the Kalman filter approach to vehicle position estimation, a very simplified geometry is assumed (solely for illustrative purposes). An earth-fixed (EF) Cartesian coordinate system is chosen in which the z axis passes through the center of the earth and the origin of the x,y coordinate. The x,y plane is tangent to the earth of an arbitrary point near the start of the vehicle motion. A simplified solution to vehicle position calculation by the trilateration method involves calculations in these coordinate systems. For convenience, the origin of the z axis is taken as the point of tangency of the lateral coordinate to the earth. The notation for the coordinates of any point in an ECEF coordinate system is x_E,y_E,z_E. The vehicle position in this EF coordinate system is denoted (x,y,z). The transformation from any point x,y,z in the EF coordinates to ECEF coordinates x_E,y_E,z_E is explained in Appendix E and in general involves a rotation matrix R and a translation vector. It is also assumed that there are four satellites moving in the same direction at an angle θ_s to the x axis at orbital speed in a plane parallel to the x,y plane of the EF coordinate system. This assumption involves errors in the z coordinate of the satellite that are sufficiently small if the four satellites are roughly overhead of the vehicle and if the duration of the simplified illustration is sufficiently small.

One of the simplest solutions to the trilateration calculation of vehicle position involves another coordinate system that is denoted x_s,y_s,z_s in which the x_s axis passes through the position of satellites 1 and 2 with the origin at satellite 1 position. For computational simplicity, the ECEF coordinate system x is taken as parallel to x_s axis. Conversion of vehicle position to any other coordinate system involves a simple transformation using matrices for rotation and translation as explained in Appendix E.

The position of satellite n in the EF coordinate system is denoted x_n,y_n,z_n ($n = 1,2,3,4$). Each satellite transmits its position and the time of transmission t_k. The discrete time positions of satellite n and the vehicle at t_k are denoted:

Satellite n position:

$$\begin{aligned} x_n(k) &= x_n(t_k) \\ y_n(k) &= y_n(t_k) \quad n = 1,2,3,4 \\ z_n(k) &= z_n(t_k) \quad k = 0,1,2\cdots \end{aligned} \tag{F.1}$$

Vehicle position:

$$x(k) = x(t_k)$$
$$y(k) = y(t_k)$$
$$z(k) = z(t_k)$$

For computational simplicity, it is assumed that the vehicle moves in the x,y plane such that $z(k) = 0 \forall k$ and that the satellites move in a plane of constant z coordinates. The assumed motion of the satellites is such that the x distance between satellite 1 and satellite 2 is constant and is denoted as $\delta x_{12} = (x_2 - x_1)$.

With the assumed coordinate system and geometry, the distances between the vehicle and satellites 1 and 2 at time t_k are given by

$$\begin{aligned} R_1(k) &= \left[x_s^2(k) + y_s^2(k) + z_1^2(k)\right]^{\frac{1}{2}} \\ R_2(k) &= \left[(x_s(k) - \delta x_{12})^2 + y_s^2(k) + z_2^2(k)\right]^{\frac{1}{2}} \end{aligned} \tag{F.2}$$

where x_s, y_s are the x,y coordinates of the vehicle in the satellite-1-based coordinate system.

$$x_s(k) = x(k) - x_1(k) \quad \text{and} \quad y_s(k) = y(k) - y_1(k)$$

In the simplified assumed geometry, the y and z coordinates are given by $y_1(k) = y_2(k)$ and $z_1(k) = z_2(k)$. By squaring each equation and subtracting, it can be shown that

$$x_s(k) = \frac{R_1^2(k) - R_2^2(k) + \delta x_{12}^2}{2\delta x_{12}} \tag{F.3}$$

The x position of the vehicle in the EF coordinate system is obtained by a translation of coordinates:

$$x(k) = x_s(k) + x_1(k) \tag{F.4}$$

The position of the third satellite in the x_s,y_s,z_s coordinate system is given by $(\delta x_{13}, \delta y_{13}, 0)$ where δx_{13} is the x coordinate and δy_{13} is the y coordinate of satellite 3 in the x_s,y_s,z_s coordinate system. The distance between satellite 3 and the vehicle is given by

$$R_3(k) = \left[(x_s - \delta x_{13})^2 + (y_s - \delta y_{13})^2 + z_3^2\right]^{\frac{1}{2}} \tag{F.5}$$

where, by assumption of the simplified geometry, $z_3(k) = z_1(k)$.

By combining equations for R_1^2 and R_3^2, it can be shown that y_s is given by

$$y_s(k) = \frac{R_1^2(k) - R_3^2(k) + \delta x_{13}^2 + \delta y_{13}^2}{2\delta y_{13}} - \frac{\delta x_{13} x_s(k)}{\delta y_{13}} \tag{F.6}$$

The $y(k)$ position of the vehicle in EF coordinates is given by

$$y(k) = y_s(k) + y_1(k) \tag{F.7}$$

The vehicle vector position $\bar{P}(k)$ in the EF coordinate system is given by

$$\bar{P}(k) = [x(k), y(k), z(k)]^T \tag{F.8}$$

This vector position can be transformed to map coordinates using the methods explained in Appendix E. However, we defer a discussion of this transformation until after the Kalman filter method of estimating vehicle position computation is finished.

In Chapter 9, it was explained that the GPS estimate of position is computed using a Kalman filter. The Kalman filter is based on the signal model for $X(k)$ and the measurement model $z(k)$ that are presented in Chapter 9 and repeated below:

$$\bar{X}(k+1) = F\bar{X}(k) + \bar{w}(k) \quad \text{(signal)}$$

$$\bar{z}(k) = H(k)\bar{X}(k) + \bar{n}(k) \quad \text{(measurement)}$$

All variables in these two models are defined in Chapter 9. In Chapter 9, the estimate of the state vector $\hat{X}(k)$ is computed recursively as given below:

$$\hat{X}(k+1) = (F - K(k)H(k))\,\hat{X}(k) + K(k)\delta R(k) \tag{F.9}$$

We consider next the procedure for computing the Kalman filter gain $K(k)$. For the purpose of simplifying the development of Kalman filter theory, the same EF Cartesian coordinate system assumed for the discussion of trilateration is assumed for the Kalman signal and measurement model. The Kalman gain matrix is computed from the following equation:

$$K(k) = F\Pi(k)H^T(k)\left(H(k)\Pi(k)H^T(k) + \Sigma\right)^{-1} \tag{F.10}$$

where

$$\Sigma = E\left[\bar{n}(k)n^T(k)\right] \tag{F.11}$$

In this appendix, the notation E[·] denotes the expected value of the quantity in the parentheses. Very often, in practice, the measurement random errors n_n are independent, stationary white Gaussian random processes. In this case, the matrix Σ is given by

$$\sum = \begin{bmatrix} \sigma_1{}^2 & 0 & 0 & \cdots & 0 \\ 0 & \sigma_2{}^2 & \cdots & \cdots & \cdots \\ 0 & 0 & \sigma_3{}^2 & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 0 & \sigma_N{}^2 \end{bmatrix} \forall k \tag{F.12}$$

where σ_n = standard deviation of n_n.

The matrix Π is termed the error covariance matrix and is defined in terms of the expected value of the error between $X(k)$ and $\hat{X}(k)$:

$$\Pi = E\left[\left(\hat{X}(k) - \bar{X}(k)\right)\left(\hat{X}(k) - \bar{X}(k)\right)^T\right] \tag{F.13}$$

The Π matrix which for stationary random errors is independent of k is the solution to the following Riccati equation:

$$\Pi = F\left[\Pi - \Pi H^T(k)\left(H(k)\Pi H^T(k) + \Sigma\right)^{-1}H(k)\Pi\right]F^T + GQG^T \tag{F.14}$$

where

$$Q = E\left[w(k)w^T(k)\right] \tag{F.15}$$

and where $w(k)$ is the process noise as explained in Chapter 9. Algorithms for solving the Riccati equation above are available in numerous references. The F, H, G, and Q matrices are presented in Chapter 9.

The reader will note that the Kalman filter structure and gain ($K(k)$) calculations are based on statistical parameters of measurement and process noise or random processes. Measurements of the random error/noise made up to time t_R can be used to estimate the required parameters. In addition, it should be noted that the direction cosine coefficients $c_{ni}(k)$ in the $H(k)$ matrix are defined in terms of the vehicle time position ($x(k)$, $y(k)$, $z(k)$) that is not known. However, in practice, these coefficients are estimated with little error by using the previous estimate of the relevant component. For example, c_{ni} is given approximately by

$$c_{ni} \cong \frac{\hat{X}_i(k-1) - \hat{X}_{ni}(k-1)}{R_n(k-1)}$$

In this estimate of direction cosines, the subscript i refers to the component number of the X vector (i.e., $i = 1 \rightarrow x$) and the subscript n is the satellite number. Similar calculations for all other components of $H(k)$ based upon $\hat{X}(k)$ values will yield very close estimates of $c_{ni}(k)$ ($i = 1,2,3$) and yield sufficiently accurate numerical values for $H(k)$ components for useful GPS navigation.

There are many practical factors affecting the accuracy of GPS navigation as described in Chapter 9 including satellite vector position at the time of transmission. In the above model, the vector position $X_n(k)$ is taken as the true satellite position in the x,y,z coordinate system. Errors in the satellite position (i.e., ephemeris errors) are included as part of the measurement error ($n_n(k)$) for $R_n(k)$. In Chapter 9, the method of correcting ephemeris errors is explained.

The superior performance of the Kalman filter estimate of vehicle position relative to trilateration can be illustrated via simulation. In this simulation, the initial vector positions of the four satellites in EF coordinates are given by vectors X_n $n = 1,2,3,4$ with units of miles:

$$X_1 = [200, 600, 810]^T$$
$$X_2 = [500, 300, 810]^T$$
$$X_3 = [100, 200, 810]^T$$
$$X_4 = [640, 300, 810]^T$$

It should be noted that these vector positions are not representative of actual GPS satellites. Rather, they are chosen for computational convenience to illustrate the principles involved in both GPS/Kalman and trilateration estimates of vehicle position. In this simplified simulation based on the geometry presented above for trilateration, all four satellites have velocity components in units of miles/s given by

$$v_{sx} = 3.91\,\text{mi/s}$$
$$v_{sy} = 2.25\,\text{mi/s}$$
$$v_{sz} = 0$$

The speed associated with these components is essentially mean orbital speed (16,180 mph) at the 4800 mi, average orbital distance from earth center for the illustrative simulation parameters. The vehicle is traveling along a curved road initially at 70 mph gradually slowing to about 62 mph.

The random error associated with pseudorange measurements using a filtered white noise source in MATLAB/Simulink software. The pseudorange measurements $R_n(k)$ were simulated by adding a sample of the random error $n(k)$ to the actual calculated true range as given below:

$$R_n(k) = \left[(x_n(k) - x(k))^2 + (y_n(k) - y(k))^2 + z_n^2(k)\right]^{\frac{1}{2}} + n(k + k_n) \quad n = 1,2,3,4 \tag{F.16}$$

where the parameters k_n are a shift in the sample point of the random error such that all simulated pseudorange measurements have independent, uncorrelated random errors. However, since the simulated random error is a stationary random process, the standard deviation of the sampled errors all have the same value (i.e., that of the simulation noise source) σ_n. The matrix Σ in the algebraic Riccati equation is a diagonal matrix with all nonzero elements σ_n^2.

The process noise $w(k)$ in the signal model presented in Chapter 9 is constructed using another simulated random error such that the vehicle speed has random fluctuations. The Q matrix is computed from this random error and is a diagonal matrix with the fourth and fifth element as the standard deviation of the simulated x and y speed component random fluctuations, respectively. All other elements in Q are O in this somewhat simplified simulation model.

In the GPS simulation, the Riccati equation is solved for matrix $\Pi(k)$ using the dare function in MATLAB. Once this matrix is computed, the Kalman gain is found using Eq. (F.10). The recursive calculation of the estimate of the state vector $\hat{X}(k)$ is found using Eq. (F.9). With each new estimate (i.e., for each k), the direction cosines c_{ni} are found using the vehicle position components and the computed satellite positions $X_n(k)$ as well as the pseudorange measurements $R_n(k)$. The above sequence of steps is repeated for each recursive estimate of $\hat{X}(k)$. The vehicle position estimate $\hat{P}(k)$ was obtained from the state vector estimate using output matrix C presented in Chapter 9 where $\hat{P}(k)$ is the estimate of $\bar{P}(k)$.

A computer program was written in MATLAB (i.e., an m-file) to perform the sequence of steps discussed above and calculate $\hat{X}(k)$ and the vehicle position estimate $\hat{P}(k)$ in EF coordinates. With $\hat{P}(k)$ denoting the true vehicle position (in EF) coordinates, the error in estimating vehicle position $e(k)$ is given by the distance between these two vectors:

$$e(k) = \left\| \bar{P}(k) - \hat{P}(k) \right\|$$

The m-file also creates a plot of the vector positions of $x(k), y(k)$, $\hat{x}(k)$, $\hat{y}(k)$ from the Kalman filter and the calculated positions from trilateration. This plot is presented in Fig. 9.19 in Chapter 9 in which the true vehicle positions are represented by a solid line and the Kalman estimates by a dashed line with the trilateration solutions depicted by + symbols for each k. A computation of vehicle position using trilateration with zero measurement error can be shown to yield true vehicle position with zero error. In addition, the m-file creates a plot of the error in the Kalman estimates $e(k)$. This latter plot that is also presented in Chapter 9 shows the very small but nonzero remaining random error component of $e(k)$.

In presenting the plot of $\hat{x}(k)$, $\hat{y}(k)$ in a vehicle navigation display, these positions can be readily converted to map coordinates by a coordinate transformation matrix as explained in Appendix E. This coordinate transformation applies equally well to the GPS/Kalman filter estimate as to the trilateration estimate of vehicle position.

The Kalman filter model for vehicle position navigation is based on a Cartesian coordinate system. However, for navigational purposes, the vehicle position must be represented in map coordinates that are most conveniently expressed as latitude and longitude. Furthermore, satellite position is also based on latitude, longitude, and elevation or radial distance from the earth center. As an illustration of coordinate transformation between Kalman filters, GPS coordinates, and map coordinates, the Cartesian coordinate system from which the map coordinates are computed is chosen as an ECEF coordinate system with the origin at earth center; the z axis is the axis of earth's rotation with positive through the North Pole as explained in Appendix E.

The corresponding Cartesian coordinates for a vehicle in ECEF coordinates are denoted x_E, y_E, z_E and are related to its geodetic coordinates ρ_v, θ_v, ϕ_v as given below:

$$z_E = \rho_v \sin(\theta_v)$$
$$x_E = \rho_v \cos(\theta_v)\cos(\phi_v)$$
$$y_E = \rho_v \cos(\theta_v)\sin(\phi_v)$$

For GPS navigation with the Kalman filter method explained above, the initial estimated vehicle position is used to begin the recursive estimated state vector $\hat{X}(k)$.

The details of the procedure for converting from EF Cartesian to map coordinates $\rho(k), \theta(k), \phi(k)$ from vehicle estimated position vector $\hat{P}(k)$ is explained in Appendix E. For the purpose of illustration of converting from $\hat{P}(k)$ in the x,y,z EF coordinate system to map coordinates, the vector position of the vehicle in ECEF coordinates in which z_E is along the earth axis of rotation and x_E, y_E are in the equatorial plane is computed first. The estimated vector position $\hat{P}(k)$ in ECEF coordinates is denoted $\hat{P}_E(k)$. Using the methods of Appendix E, $\hat{P}_E(k)$ can be obtained with the following transformation:

$$\hat{P}_E(k) = R\hat{P}(k) + T$$

where R is a matrix of rotations via Euler angles that transforms $\hat{P}_E(k)$ to an intermediate set of coordinates in which each coordinate component is parallel to the corresponding ECEF component and $\bar{T}$ is the location of the origin of the EF coordinate system in ECEF coordinates. The map coordinates that are assumed for this example to be geodetic are computed from $\hat{P}_E$ as follows:

$$\hat{\rho}_E(k) = \left\|\hat{P}_E(k)\right\|$$
$$\hat{\phi}(k) = \tan^{-1}[-\hat{y}_E(k)/\hat{x}_E(k)]$$
$$\hat{\theta}(k) = \sin^{-1}[\hat{z}_E(k)/\hat{\rho}_E(k)]$$

The vehicle position $\hat{\theta}(k), \hat{\phi}(k)$ is then presented on a flat panel display (see Chapter on vehicle instrumentation). A digital map is available in any vehicle instrumentation capable of displaying the vehicle position on an electronic flat panel display. The map displays roads and streets in the vicinity of the vehicle estimated position with a user selectable scale. A more detailed discussion of digital maps and display than given here is presented in Chapter 8 on vehicle instrumentation.

Index

Note: Page numbers followed by *f* indicate figures, and *t* indicate tables.

A

Acceleration, 278–279
  enrichment, 288–289
  hard, 322–323
  sensor, 244–247, 245*f*
Accelerometer, 393–395, 508–510
Accumulator register, CPU, 98
Active suspension system, 379
Actuator and Sensor, 9, 595–596
  automotive control system applications
    airflow rate sensor, 186–190
    electronic engine control system, 184–185, 185*f*
    engine crankshaft angular position sensor, 194–195
    Hall-effect position sensor, 205–208
    magnetic reluctance position sensor, 195–205
    optical crankshaft position sensor, 208–210
    pressure measurements, 191–194
    variables in engine control, 185–186
  automotive engine control, 247–254
  duty-cycle-controlled throttle, 359
  electric motor, 258–268
  stepper motor-based, 360–362, 361*f*
  throttle, 356–359, 358*f*
  vacuum-operated, 362–364, 363*f*
Address bus (AB), 94
Advanced cruise control (ACC), 344, 364–368
  configuration, 365*f*
  on hill with long downgrade, 367, 367*f*
Aerospace angular rate sensors, 223
AI. *See* Artificial intelligence (AI)
Airbag safety device, 505–512, 509*f*
  accelerometer-based system, 508, 508*f*
  deployment system, 505–506, 506*f*, 507*t*
Airflow rate sensor, 186–190
Air mass measurement, 173–175
  speed-density method, 173–175, 174*f*
Air suspension system, 396
Alpha-numeric display, 431–433
  BCD to seven-segment display decoder, 434*f*
  decoder truth table, 433*f*
  instrumentation system, 432*f*
  seven-segment digital display, 432*f*
ALU. *See* Arithmetic and logic unit (ALU)
Amplifier
  FET, 50–52, 50*f*, 52*f*
  grounded source, 50
  linear, 51
  noninverting, 56
  operational, 53
  summing mode, 56–66
Analog comparator, 57, 58*f*
Analog multiplexer, 66*f*
Analog signal, digital filtering of, 647*f*
Analog system, 598
Analog to digital (A/D) converter, 413–414
AND
  gate, 69*f*, 70, 107
  operation, 109–110, 111*f*
  subroutine, assembly language, 113*f*
Angular rate sensor (ARS), 223–226, 665, 667–668
  configuration, 224*f*
  demodulator, 226*f*
  sensing tines, 225–226
Angular speed sensor, 203–204
Antilock braking system (ABS), 368–377, 368*f*, 374*f*, 376*f*
  configuration, 369, 370*f*
  exemplary variation in friction coefficients with slip, 370–371, 371*f*
  SMO, 373–374
  wheel torque *vs.* slip under, 372
APPS. *See* Automatic parallel parking system (APPS)
Arithmetic and logic unit (ALU), 86–87, 86*f*, 100
Arithmetic logic functions, 87*t*
ARS. *See* Angular rate sensor (ARS)
Artificial intelligence (AI), 567–568, 572
Assembly language, 109
  mnemonics, 110*t*
  programming and function in, 111
  AND subroutine, 113*f*
Audio communication, 482
Automatic collision avoidance system (ACAS), 19
Automatic parallel parking system (APPS), 574–580, 575–576*f*, 580*f*
  APPS-ECU, 575–576
  closed-loop control, 575–576, 576*f*, 578
  state-variable equation, 577
  transfer function, 578
Automatic steering, 20–21
Automatic transmission control, 323–329
Automotive control system applications
  sensors and actuators
    airflow rate sensor, 186–190

Automotive control system applications *(Continued)*
electronic engine control system, 184–185, 185*f*
engine crankshaft angular position sensor, 194–195
Hall-effect position sensor, 205–208
magnetic reluctance position sensor, 195–205
optical crankshaft position sensor, 208–210
pressure measurements, 191–194
variables in engine control, 185–186
Automotive electronic system, 641
digital, 598
Automotive engine control actuators, 247–254
exhaust gas recirculation actuator, 253–254
fuel injection, 251–253
solenoid, 248, 248*f*, 250
Automotive instrumentation, 409–410
alpha-numeric display, 431–433, 432–434*f*
block diagram, 410, 410*f*
computer-based instrumentation, 412*f*, 419
coolant temperature measurement, 452–454, 453*f*
electro optic displays, 423–424
evolution of, 410
FDP
block diagram, 438*f*, 440*f*
digital maps, 442–443
instrument clusters, 434–441
pictorial display capability of, 441
solid-state array-type display, 435*f*
touch screen, 443–447, 444–445*f*
fuel quantity measurement, 447–452, 448–449*f*, 451–452*f*
galvanometer-type display, 420–423, 420–422*f*
input and output signal conversion, 413–414, 413–414*f*
multiplexing, 415–416, 415*f*
multirate sampling, 416–419, 418*f*
light-emitting diode, 424–425, 424*f*
liquid-crystal display, 426–428, 426*f*, 428*f*
oil pressure measurement, 454–455, 455*f*
signal processing, 410–411
transmissive LCD, 428–429
trip information system, 457–460, 458*f*
vacuum-fluorescent display, 429–431, 430–431*f*
vehicle speed measurement, 456–457, 456*f*
Automotive Open System Architecture (AUTOSAR), 108–109, 133–134
Automotive system, microcomputer applications in, 122–124
Autonomous vehicles, 573
APPS, 575–580, 575–576*f*, 580*f*
block diagram, 581–593, 581–582*f*, 591*f*
command and actual heading, 588*f*
NHTSA, 573–574
SAE, 574
traveling along curve of radius, 585*f*
vehicle maneuver via differential braking, 593*f*
Autosar, 133–134
Avalanche photodiodes, 228

B

Backlighting, 429, 434–437
Band theory of electrons, 26
Base-emitter junction, of transistor, 39
Base pulse duration, 283
Battery Equivalent Circuit, 638–639
BCD to seven-segment display decoder, 434*f*
Bidirectional switch, 64–66, 65*f*
Binary number system, 68
Bipolar junction transistor (BJT), NPN, 39
Bipolar transistor, 38
Birefringent material, 427
BJT. *See* Bipolar junction transistor (BJT)
Blind spot detection (BSD), 18, 512–515
Bluetooth, 489
Boolean algebra, 70–71, 71*t*
Bottom dead center (BDC), 255
Brake-specific fuel consumption (BSFC), 154
Bridge circuit, 187–188, 188*f*
Brushless DC motor, 9, 266–268
BSD. *See* Blind spot detection (BSD)
BSFC. *See* Brake-specific fuel consumption (BSFC)
Butterworth filter, 451, 633–634, 635*f*, 652*f*

C

Calibration, engine performance, 156
Camber angle, 398–399
Cam lobe (CL), 256
control, electronic fuel control system, 167–169
Camshaft, 195–196
phasing, variable valve timing control, 297
Camshaft position sensor (CPS), 165
CAN. *See* Controller area network (CAN)
Capacitance, 443
mutual, 444–446, 444*f*
units of, 444
Capacitor, 3
configuration and circuit symbol, 3*f*
CAS. *See* Collision avoidance system (CAS)
Caster angle, 398–399
Catalytic converter, 162*f*
exhaust, 161–164
oxidizing, 161
CCD. *See* Charge-coupled device (CCD)
CDMA. *See* Code division multiple access (CDMA)
Charge-coupled device (CCD), 230–231
channel configuration, 231
channel stops, 231
horizontal, 234

output amplifier, 234*f*
scanning column, 231–232, 232*f*
scanning voltages, 233*f*
Chassis dynamometer, 137–138, 138*f*
Clean Air Act, 533
Closed-loop control, 322, 344, 655–658, 656*f*
automatic parallel parking system, 575–576, 576*f*, 578
Closed-loop ignition timing, 312–317
Closed-loop limit-cycle control, 622–624, 623–624*f*
CMOS technology, 64
CNN. *See* Convolutional neural network (CNN)
Code division multiple access (CDMA), 17, 483–486, 488–490
Collision avoidance system (CAS), 515–521, 516*f*, 518*f*
Communication, application of diodes, 37
Comparator, 57
analog, 57, 58*f*
Computer-based instrumentation, 412*f*, 419
Concierge service, 482, 503
Condition code (CC) register, CPU, 98–99, 98*f*
Connection master, 480–481
Control bus (CB), 94
Control channel, 481
Controller area network (CAN), 464–472
arbitration on, 472
block diagram, 467, 468*f*
bus transceiver, 467–468
CAN_H and CAN_L bus wire, 465
electronic circuits, 469–471
local interconnect network, 472–474, 473*f*
transceiver circuitry, 470*f*
voltage
levels, 465, 466*f*
waveforms, 468, 469*f*
Control modes, for fuel control
acceleration, 278–279, 288–289
automatic transmission control, 323–329
closed-loop control, 284–288, 322
correction factor, 283
deceleration, 278–279, 289, 323
differential and traction control, 329–331
engine control configuration, 279–280
engine crank, 281, 321
engine start, 278
engine warm-up, 281–283, 321–322
hard acceleration, 322–323
hybrid electric vehicle powertrain control, 331–341
idle mode, 279
idle speed control, 289–291
integral-like term, 285–286
open-loop control, 283, 322
open-loop mode, 278
torque converter lock-up control, 329
Control system stability, 617–622
robustness of, 620–622
root-locus techniques, 618–620, 620*f*
Control theory, 610–617
closed-loop control, 611–617, 612*f*
open-loop control, 611, 611*f*, 615*f*
Convolutional neural network (CNN), 515
Coolant temperature, measurement, 452–454, 453*f*
Coordinate transformation, 673–675, 674–675*f*
Coriolis acceleration, 225, 665, 666*f*
CPU registers, 97–100, 97*f*
accumulator register, 98
condition code register, 98–99, 98*f*
Crankshaft angular position sensor, 194–195
angular speed sensor, 203–204
disadvantage, 202
eight-cylinder engine, 203
ferromagnetism, 197–198, 197*f*
initial magnetization curve, 198
magnetic field intensity vector, 196–197
magnetic flux density vector, 196–197
magnetic reluctance position sensor, 195–205
static engine timing, 202
zero-crossing point, 202–203
Crash detection algorithm, 511–512
Critical speed, 525–527
Cruise control
advanced, 364–368, 365*f*, 367*f*
analog configuration, 363*f*
block diagram, 348*f*
brake circuit, 551, 554*f*
configuration, 347, 347*f*
digital, 351–353, 351*f*, 354*f*, 359*f*
discrete-time, 353
flowchart, 553*f*
hardware implementation issues, 354–356, 355–356*f*
linearized equation of motion, 347
microprocessor-based, 359–360, 362
performance, 345
proportional integral control strategy, 348–350
qualities, 350–351
speed performance, 350, 350*f*
stepper motor-based actuator electronics, 360–362, 361*f*
throttle actuator, 356–359, 358*f*
traditional, 344–345
vacuum-operated actuator, 362–364, 363*f*
Curve-fitting algorithm, 445
Cyclic redundancy check (CRC), 476

D

DAB. *See* Digital audio broadcasting (DAB)
D/A converter, 414
Damping
  critical, 349–350
  ratio, 349–350
  in suspension system, 378, 391
Data bus (DB), 94
Data link connector (DLC), 465
DC motors, brushless, 266–268
DC-to-DC converter, 339
Deceleration, 278–279
  and idle, 323
  leaning, 289
Decimation process, 417–419
Decoder circuit, 429, 432–433, 437
Demultiplexer (DEMUX), 65–66, 415, 418*f*
D flip-flop (DFF), 79–80, 79*f*
Diagnostic fault codes, 543–555
  electronic control system diagnostics, 534–536, 535*f*
  FDI system, 539–543
  flowchart, 547*f*, 549–550*f*
  misfire detection system, 536–537, 555–567
  model-based diagnostics, 539–543, 542*f*
  model-based actuator failure detection, 538–539
  off-board diagnosis, 534–536, 545, 564, 572
  onboard diagnostics, 534, 536–538, 543, 548–551, 555
  sample, 544–545*t*
  switch test sequence, 551–552, 552*f*
  two-digit diagnostic codes, 551
Diagnostic scan tool, 535, 535*f*
Differential control mode, 329–331
Differential pressure sensor (DPS), 275, 295
Digital audio broadcasting (DAB), 18, 479, 490, 493
Digital circuit, 66–68, 67*f*
Digital computer, 91
Digital controllers, 641
Digital cruise control, 351–353, 354*f*
  block diagram, 351*f*
  change in set speed, 354*f*
Digital engine control, 272–274
  binary fuel injection, 277
  components, 275*f*
  features, 274–277
  suboptimal fuel injection, 275–276
Digital filter, 126–128
  of analog signal, 647*f*
Digital integrated circuits, 86–87
Digital maps, 442–443
Digital signal processing (DSP), 411, 414–415, 449
Digital signal processor (DSP), 228–229
Digital subsystem, 644–645
Digital transfer function model, 651
Digital video camera, 229–235
  CCD, 230–231
  configuration, 229, 230*f*
  interline, 231
  photosensitive capacitor configuration, 231*f*
Diode, 27–29
  communications applications of, 37
  laser, 34–35, 35*f*
  light generating, 34
  PIN, 34
  p-n diode, 28*f*
  solid-state, 27
  transfer characteristics, 29*f*
  zener, 29, 30*f*
Direct fuel injection (DFI), 306–307
Discrete Fourier transform (DFT), 491
Discrete time control system, 652–655
  closed-loop, 657*f*, 662*f*
  example, 659–663, 659*f*
  open-loop, 653*f*
Discrete time idle speed control, 291–294, 291*f*
Discrete time system, 642*f*
Display devices, 419–420
  alpha-numeric display, 431–433, 432–434*f*
  electro optic displays, 423–424
  FPD (*see* Flat panel display (FDP))
  galvanometer-type display, 420–423, 420–422*f*
  light-emitting diode, 424–425, 424*f*
  liquid-crystal display, 426–428, 426*f*, 428*f*
  transmissive LCD, 428–429
  vacuum-fluorescent display, 429–431, 430–431*f*
Display RAM, 439–440
Distributed computing, 462
Doping, 27
  silicon, 24–27
Doppler shift, 35
Doublet steering. *See* Steering doublet
Driver electronics, 360
  for cruise control, 362
  stepper motor, 360
Drive wheels (DWs), 332
Duty cycle, 358–359, 362
Duty-cycle-controlled throttle actuator, 359
Dynamic analytic models, 13

E

Earth-centered, Earth-fixed (ECEF) Cartesian coordinates, 496, 499
Earth fixed (EF) Cartesian coordinate system, 677
ECEF Cartesian coordinates. *See* Earth-centered, Earth-fixed (ECEF) Cartesian coordinates

ECU. *See* Engine control unit (ECU)
EGO. *See* Exhaust gas oxygen (EGO)
Electric motor (EM), 332
  actuator, 258–268
    brushless DC motors, 266–268
    two-phase induction motor, 263–266
Electric vehicle (EV), in HEV, 341
Electromagnetic theory, 25, 198, 201
Electronic control system diagnostics, 534–536, 535*f*
Electronic engine control system, 142–149, 142–143*f*
  engine functions and control diagram, 144*f*
  federal government test procedures, 137–142, 139*f*
    additional cost incentive, 141–142
    fuel economy requirements, 140–141
    meeting the requirements, 141
    role of electronics, 141–142
  inputs to controller, 144–145, 145*f*
  intake manifold pressure analysis, 172–176, 172–173*f*
  motivation for, 136
  output from controller, 145–146, 145*f*
Electronic fuel control system, 164–172, 164*f*
  CL control, 167–169
  CL operation, 169–172
  closed-loop, 168*f*
  engine control sequence, 166
  frequency and deviation of fuel controller, 170–172
  OL control, 167
  waveforms in closed-loop, 170*f*
Electronic ignition, 181–182, 182*f*
  control, 309–318
    closed-loop ignition timing, 312–317
    spark advance correction scheme, 317–318
Electronic instrumentation system, 624–627
  measurement, 625, 627–632
    issues, 625–626
    random errors, 630–632
    sensor, 628–630, 629–630*f*
    signal processing, 632
    systematic errors, 626–627, 626–627*f*
Electronic safety-related systems
  airbag safety device, 505–512, 506*f*, 507*t*, 508–509*f*
  blind spot detection, 512–515
  CAS, 515–521, 516*f*, 518*f*
  EVS, 524–531, 527*f*, 530*f*
  LDM, 521
  TPWS, 522–523, 524*f*
Electronic steering control, 398–401, 399–401*f*
Electronic suspension system, 377–397, 378*f*, 380*f*
  active, 379
  classes of, 378–379
  configuration, 397, 397*f*
  electronic steering control, 398–401, 399–401*f*
  normal force variation *vs.* frequency, 383*f*, 384
  parameters, 392*t*
  purpose of, 377–378
  QCM, 387, 387*f*, 389
  second-order differential equation, 382
  semiactive, 378–379
  strut damping, 391, 394–395, 394*f*
  variable damping via variable strut fluid viscosity, 395–396
  variable spring rate, 396–397
Electronic system
  analog (continuous time) systems, 598
  block diagram, 596–598, 597*f*
  concept of system, 595–598
Electro optic displays, 423–424
  LCD, 426–429, 426*f*, 428*f*
  LED, 424–425, 424*f*
  VFD, 429–431, 430–431*f*
Electro optics, 30–35
Emissions, exhaust, 136–137
Engine angular speed sensor, 203–204
Engine control configuration, 279–280
Engine control unit (ECU), 546, 548
Engine crank, 281, 321
Engine crankshaft angular position sensor, 194–195
Engine mapping, engine performance, 157
Engine overall efficiency, 156
Engine performance terms
  calibration, 156
  effect
    of air/fuel ratio on performance, 157–158, 158*f*
    of EGR on performance, 159–160, 160*f*
    of spark timing on performance, 158–159, 159*f*
  engine mapping, 157
  engine overall efficiency, 156
  fuel consumption, 154–156
  power, 153–154
  torque, 150–152, 151–152*f*
Engine position sensor (EPS), 165
Engine started mode, 278
Engine warm-up, 281–283, 321–322
Enhanced stability system (ESS), 377
Enhanced vehicle stability (EVS), 524–531
  algebraic equation, 526
  block diagram, 530, 530*f*
  rotational motion equation, 525
  steering coefficient, 526
  tire slip angle, 528–529
  translational motion equation, 525
  vertical forces on vehicle, 527*f*
  yaw rate, 524–527, 530
Environmental Protection Agency (EPA), 534
Ephemeris errors, GPS, 500–501

Estimated time of arrival (ETA), 459
Evasive steering, 520
EVS. *See* Enhanced vehicle stability (EVS)
Exemplary circuits, for logic gates, 71–75
Exhaust catalytic converters, 161–164
Exhaust emissions, 136–137
Exhaust equivalence ratio, 285–286
Exhaust gas oxygen (EGO)
  concentration, 168–169
  sensor, 166, 168–169, 215–220
    characteristics, 217
    closed-loop, 168*f*, 219–220, 284
    commercial sensor, 218*f*
    engine warm-up, 281
    fuel control using, 219–220
    improvements, 220
    open-loop control mode, 219–220, 278
    switching time, 217–220, 217*f*
    switching transients, 218*f*
    temperature dependence of, 219
    TiO_2, 215
    voltage, 284–286
    ZrO_2, 215–216
Exhaust gas recirculation (EGR), 176, 185, 253–254
  control, 294–296, 296*f*
  on performance, 159–160, 160*f*
Expert system, 536
  in automotive diagnosis, 567–572, 571*f*
  benefits, 567
  database of known facts, 571, 571–572*f*
  developing tools for mainframes, 570*t*
  environment, 569*f*
  IF-THEN rules, 568–570
  procedure, 569*f*

F

Failure detection and identification (FDI) system, 539–543, 669–670
Fast inverse Fourier transform, 491–492
Fault indication lamp (FIL), 537, 543, 546
FDI system. *See* Failure detection and identification (FDI) system
FDP. *See* Flat panel display (FDP)
Feedback control, sensor for
  exhaust gas oxygen sensor, 215–220
  oxygen sensor improvements, 220
Ferromagnetism, 197, 197*f*
Field-effect transistor (FET), 45–47
  amplifier, 50–52, 50*f*, 52*f*
  circuit symbol, 46*f*
  configuration, 46*f*
  depletion mode, 49–50
  inverter circuit, 72*f*
  NAND gate FET circuit, 73*f*
  N-channel enhancement, 48–49*f*
  in switch mode, 63*f*
  theory, 47–52, 47*f*
  transconductance for, 52
FIL. *See* Fault indication lamp (FIL)
Filtering, 632–639
Flat panel display (FDP), 15, 410, 462, 513
  block diagram, 438*f*, 440*f*
  digital maps, 442–443
  instrument clusters, 434–441
  pictorial display capability of, 441
  raster-type scan, 436–438
  representative pixel drive circuits, 437*f*
  solid-state array-type display, 435*f*
  touch screen, 443–447, 444–445*f*
Flex fuel, 308–309
Flex-fuel sensor (FFS), 235–247, 309
  acceleration sensor, 244–247
  capacitance, 237–239
    oscillator methods of measuring, 239–244
  charged conductor, 237*f*
  circuit for measuring, 239*f*
  configuration, 237, 237*f*
  electric field intensity vector, 238
  equivalent circuit, 243, 243*f*
  microprocessor-based control system, 242
  555 timer circuit, 239–240, 240*f*
FlexRay IVN, 474–478
  bus voltage states, 476*f*
  normal mode, 474–475
  standby mode, 474–475
  topology, 475*f*
  transceiver circuit, 477–478, 477–478*f*
Floating-point operations (FLOPS), 92
Flux density, 201
Flywheel, 194, 195*f*
4-Bit adder circuit, 77*f*
Fourier transform
  fast inverse, 491–492
  IDFT, 491
Four-stroke engine operation principle, 146–149, 147*f*, 149*f*
Four-wheel steering (FWS), 401–408, 402*f*, 407*f*
FPD. *See* Flat panel display (FPD)
Free electrons, 26
Frequency hopping (FH), 17
  spread spectrum, 489–490
Frequency-mixing circuit, 37, 37*f*
Frequency-to-voltage converter, 566
Fuel control, control modes for

acceleration, 278–279, 288–289
automatic transmission control, 323–329
closed-loop control, 284–288, 322
correction factor, 283
deceleration, 278–279, 289, 323
differential and traction control, 329–331
engine control configuration, 279–280
engine crank, 281, 321
engine start, 278
engine warm-up, 281–283, 321–322
hard acceleration, 322–323
hybrid electric vehicle powertrain control, 331–341
idle mode, 279
idle speed control, 289–291
integral-like term, 285–286
open-loop control, 283, 322
open-loop mode, 278
torque converter lock-up control, 329
Fuel controller, frequency and deviation of, 170–172
Fuel economy, 137
requirements, 140–141
Fuel gauge
electromechanical system, 419
sloshing effect, 449
Fuel injection
configuration, 251–252, 252*f*
fuel injector signal, 251–253
Fuel injector (FI), 156, 165, 185
Fuel quantity
filtered *vs.* unfiltered, 452*f*
filtering fuel sensor signal, 449*f*
measurement, 447–452, 448–449*f*, 451–452*f*
sensor configuration, 448*f*
short-term time average, 449
Simulink model, 451*f*
FWS. *See* Four-wheel steering (FWS)

G

Gallium arsenide phosphide (GaAsP), 424–425
Galvanometer-type display, 420–423
circuit diagram, 422*f*
configuration, 420*f*
magnetic field configuration, 421*f*
Gasoline, 136
Gateway, 464
Geometric dilution of position (GDOP), 501–502
Glass cockpits, 15
Global positioning system (GPS), 17, 493–503
clock errors, 500–501
control configuration, 501*f*
ephemeris errors, 500–501
error feet *vs.* time, 500*f*
Kalman filter, 495–500
pseudorange model, 493–497, 500–501
structure, 500–503
vehicle position, 499*f*
Global position satellite (GPS), 15, 441
Grounded-emitter configuration, 40, 40*f*
Grounded source amplifier, 50

H

Hall-effect position sensor, 205–208, 206*f*
shielded-field sensor, 208, 209*f*
Heated exhaust gas oxygen (HEGO) sensor, 220, 220*f*, 537
diagnosing fault in, 547*f*
ECU, 546
measurement, 548–551
operation, 548
proper switching test, 550*f*
voltage, 546, 548, 549*f*
HEV. *See* Hybrid electric vehicle (HEV)
High-voltage bus (HVB), 333
Hybrid electric vehicle (HEV), 11, 332*f*
with mechanical coupler, 333*f*
powertrain control, 331–341
Hydraulic locking, 256–257
Hydrocarbons, 136

I

Ideal sampler configuration, 648*f*
Idle mode, 279
Idle speed control (ISC), 176–181, 179–181*f*, 289–291, 290*f*
continuous-time model, 292
discrete time, 291–294, 291*f*
Ignition control module (IGN), 165
Ignition system, 268–270
coil operations, 269–270
dwell period, 269
engine control unit, 269
secondary voltage, 270
subsystem, 268, 269*f*
Ignition timing, 11
closed-loop, 312–317
Index of refraction, 501–502
Inductor, 4–5
configuration and circuit symbol, 4*f*
Inference engine, 568
Instrumentation control system (ICS), 456
Instrument panel (IP), 410, 412, 441
Intake manifold pressure analysis, 172–176, 172–173*f*
air mass measurement, 173–175
speed-density method, 173–175, 174*f*
influence of valve system on volumetric efficiency, 175–176

Integrated circuit (IC), 52–53, 86
Integrated engine control system
  automatic system adjustment, 319–320
  evaporative emissions canister purge, 319
  secondary air management, 319
  system diagnosis, 320–321
Internal combustion engine (ICE), HEV and, 331–341
Interpolation, 282
In-vehicle network (IVN), 14–16, 274, 412, 462–464, 463*f*
  CAN, 464–472
    arbitration on, 472
    block diagram, 467, 468*f*
    bus transceiver, 467–468
    electronic circuits, 469–471
    local interconnect network, 472–474, 473*f*
    transceiver circuitry, 470*f*
    voltage levels, 465, 466*f*
    voltage waveforms, 468, 469*f*
  FlexRay, 474–478, 475–478*f*
  MOST, 478–481, 479–480*f*
  physical link, 463
Inverse discrete Fourier transform (IDFT), 491
Inverse fast Fourier transform (IFFT), 491–492
Inverter circuit, FET, 72*f*
ISC. *See* Idle speed control (ISC)
Isotropic homogeneous material (IHM), 236
IVN. *See* In-vehicle network (IVN)

J

JK flip-flop, 78–79, 78*f*

K

Kalman filter, 495–500, 677, 679–681
Knocking, 312–317, 314–315*f*
Knock sensors, 221–223

L

Lambda sensor, 215–220
Lane departure monitoring (LDM), 521
Laplace transform, 598, 642, 647*t*, 648
Laser
  diode, 34–35, 35*f*
  light, 35
  solid-state, 34
LCD. *See* Liquid-crystal display (LCD)
LDM. *See* Lane departure monitoring (LDM)
LED. *See* Light-emitting diode (LED)
Light detection and ranging (LIDAR) system, 227–229
Light-emitting diode (LED), 34, 208–210, 424–425, 424*f*
  MOST, 479, 479*f*
Light generating diode, 34
Light, laser, 35
Limit-cycle controller, 284–286, 288*f*
LIN. *See* Local interconnect network (LIN)
Linear amplifier, 51
Linear system theory
  continuous time, 598–606, 599*f*
  first-order system, 601–603, 602*f*
  second-order system, 603–606, 603*f*, 606–607*f*
Liquid-crystal display (LCD), 426–428, 428*f*
  construction, 426*f*
  disadvantage, 428
  thin-film-transistor, 423–424, 429, 434–435
  transflexive, 429
  transmissive, 428–429
Local interconnect network (LIN), 472–474, 473*f*
Logic circuits, 69–77
  combination, 75–77, 76*f*
  with memory, 77
    D flip-flop, 79–80, 79*f*
    JK flip-flop, 78–79, 78*f*
    R-S flip-flop, 77–78, 78*f*
Logic functions, 109–110
Logic gates, 69*f*
  exemplary circuits for, 71–75
Lorentz force, 207
Low-frequency transmitter (LFT), 523
Low-pass filter (LPF), 36, 187, 373, 414, 449, 566
  output voltage, 226
  sinusoidal frequency response, 226
Low-voltage bus (LVB) voltage, 337
LPF. *See* Low-pass filter (LPF)

M

MAF sensor. *See* Mass airflow rate (MAF) sensor
Magnetic field intensity vector, 196–197
Magnetic flux density
  electric motor, 260
  vector, 196–197, 200, 208
Magnetic reluctance position sensor, 195–205
Magnetorheological (MR) fluid, 395–396
Magnetostriction, 221–223
Mainframe computers *vs.* microcomputers, 92
Manifold absolute pressure (MAP), 172–173, 191
  sensing resistors, 191, 192*f*
  strain gauge MAP sensor, 191–194
  Wheatstone bridge, 191–192, 193*f*
Masking technique, 112
Mass airflow (MAF) rate sensor, 186
  bridge circuit, 187–188, 188*f*
  configuration, 188
  dynamic response, 189

hot-wire anemometer, 187
Laplace methods of analysis, 190
output voltage, 190, 190*f*
Mass airflow (MAF) sensor, digital engine control, 275–276
Matched filter (MF), 511, 517, 519
MATLAB function, 649
MATLAB/SIMULINK
simulation, 286–287
system, 605, 606*f*
Mean best torque (MBT), 313
Media-oriented system transport (MOST), 478–481
functional block diagram, 480–481, 480*f*
LED, 479, 479*f*
receive circuit, 479–480, 480*f*
Metal-oxide semiconductor (MOS), 46–47
Microcomputer, 90, 123*f*, 125*f*
applications in automotive systems, 122–124, 130–132, 130*f*
buffer configuration, 107–108, 108*f*
closed-loop control system, 128
digital computer, 91
feedback control system, 128–130, 129*f*
hardware, 114
analog-to-digital converter, 118–120, 118–119*f*
CPU, 114–115
digital-to-analog converter, 116–118, 117*f*
interrupts, 121–122
I/O parallel interface, 115–116
polling, 121
RAM, 115
ROM, 115
sampling, 120–121, 120*f*
vectored interrupts, 122
instrumentation applications of, 124–128
limit-cycle controller, 128
vs. mainframe computers, 92
multivariable and multiple task system, 132–133
operations, 94
addressing peripherals, 96–97
buses, 94, 95*f*
memory-read/write, 94–96, 95*f*
timing, 96
parts of computer, 91, 92*f*
programs, 93
reading instructions, 100–106
branch instruction, 104, 104*f*
decode bytes, 103*f*
decode subsystem, 103*f*
initialization, 102
jump instruction, 105
jump-to-subroutine instruction, 105–106, 106*f*
operation codes, 102, 103*f*
program counter, 102–103
return-from-subroutine instruction, 106, 107*f*
table lookup, 130–132, 130–131*f*
tasks, 93–94
Microcontroller, 360, 398, 406
Microprocessor (MPU), 87–88, 88*f*, 133
architecture, 100, 101*f*
based cruise control, 359–360, 362
Mild hybrids, 331
Misfire detection system, 536–537, 555
model-based, 555–567
Model-based diagnostics, 539–543, 542*f*
Model-based sensor failure detection, 538–539
MOS. *See* Metal-oxide semiconductor (MOS)
MOST. *See* Media-oriented system transport (MOST)
Motor-driven pump, 377, 396–397
MPU. *See* Microprocessor (MPU)
Multidrop topology, 474, 475*f*
Multiplexer (MUX), 65, 66*f*
Multiplexing, 415–416, 415*f*, 464
CDMA, 483–486, 488–490
TDMA, 476, 483
Multirate sampling, 416–419, 418*f*
Mutual capacitance method, 444–446, 444*f*

N

NAND gate FET circuit, 73*f*
National Highway Traffic Safety Administration (NHTSA), 511, 573–574
Network interface controller (NIC), 478–481
Network master, 480–481
Noninverting amplifier, 56
Nonuniformity index, 562*f*, 563, 566
NOR logic, 74*f*
NOT gate, 70–71
NPN
phototransistor, 210
transistor, 37
amplifier circuit, 38*f*
bipolar junction, 39
current and voltages for, 41*f*
digital circuit, 67*f*
grounded-emitter configuration and voltages, 39*f*
grounded emitter NPN transistor amplifier, 43*f*
Nyquist sampling rate, 482–483

O

OBD. *See* On-board diagnostics (OBD)
OC. *See* Oxidizing catalyst (OC)
OEM, 4–5
OFDM. *See* Orthogonal frequency-division multiplexing (OFDM)

Off-board diagnosis, 534–536, 545, 564, 572
Oil pressure, measurement, 454–455, 455*f*, 543
On-board diagnostics (OBD), 19, 534, 536–538, 543, 548–551, 555
Open-loop (OL)
  control, 322
    electronic fuel control system, 167
    fuel, 172
  mode, 278
Operational amplifiers (OP-AMPS), 53
  circuits, 53*f*, 55–56*f*
  use of feedback in, 54–56
Optical sensor
  photoconductive, 32
  photodiode, 33*f*
Optics, electro, 30–35
Optoelectronics, 423
OR gate, 69*f*, 70
Orthogonal frequency-division multiplexing (OFDM), 18, 490, 493
Oxidizing catalyst (OC), 161, 162*f*
  conversion efficiency *vs.* temperature, 161, 162*f*
Oxidizing catalytic converter, 161
Oxygen sensor improvements, 220

P

Parallel hybrid, 332–333, 332*f*
Phased array method, 519
Phase-locked loop (PLL), 58–60, 58*f*, 566
Phase-shift keying (PSK), 486–488, 487*f*
Phosphor emits light, 429
Photoconductor, 31–32, 31*f*
  optical sensors, 32
Photodiode, 32–34
  characteristic curves, 33*f*
  optical sensor circuit, 33*f*
Photons, 30
Phototransistor, 210
PID
  control law, 298
  controller, 129–130
Piezoresistivity, 191
PIN diode, 34
Pitch, 343
Planetary gears, 326–327
PLL. *See* Phase-locked loop (PLL)
p-n diode, 28*f*
Pneumatic springs, 396–397
p-n junction, 27–29
  optical sensor, 32
PNP transistor, 37
Portable scan diagnostic tool (PSDT), 535–536, 546, 548, 551
Potentiometer, 211, 447
  movable contact, 213
  rotary potentiometer, 212, 213*f*
  schematic circuit, 211*f*
Power, engine performance, 153–154
Power master, 480–481
Power steering, 13–14, 400–401, 400–401*f*
Powertrain, 272
Pressure measurement, 191–194
Programming languages, 108–109
  assembly language, 109, 110*t*
    programming and function, 111
  logic functions, 109–110
  masking, 112
  SHIFT, 110–111, 112*f*
  SHIFT and AND, 113
  use of subroutines, 113–114
PSDT. *See* Portable scan diagnostic tool (PSDT)
PSK. *See* Phase-shift keying (PSK)

Q

Quadrature-phase-shift keying (QPSK), 486–488, 487*f*
Quarter car model (QCM), 387, 387*f*, 389

R

Radar system, 519, 581–582
  detecting an overtaking vehicle, 514
  measurement, 514
  transmitted and received signals, 517–518, 518*f*
RAM, 438
Random access display, 438, 439*f*
Raster-type FDP, 436–438
Read-only memory (ROM), 360, 419
Rectifier circuit, 35–47, 36*f*
Rectifier waveform, 36*f*
Recursive algorithm, 411, 449
Redundancy, concept of, 588–589
Refractive index, 501–502
Reluctance sensor, 201, 203
Remanent magnetization, 198
Reset-set (R-S) flip-flop, 77–78, 78*f*
Resistor, 2
  circuit symbol and model, 3*f*
Reverse magnetostriction, 221
Riccati equation, 679, 681
Ride, driver/passenger standpoint, 378
Ring gear, 326–327
Root-locus techniques, 618–620, 620*f*
Root-mean-squared (RMS) value, 391, 394–395

S

SAE. *See* Society of Automotive Engineers (SAE)
SAE J-2284-3, 465, 467
Sample and zero-order hold circuits, 60–62, 61*f*
Satellite vehicle communication, 490–493
SBDT. *See* Service bay diagnostic tool (SBDT)
Secondary air management, 319
Self-driving vehicles. *See* Autonomous vehicles
Semiactive suspension system, 378–379
Semiconductor
  current conduction in, 25*f*
  devices, 24–29
  n-type, p-type, 26–27
Sensor, 8–9, 629*f*, 630*f*, 595–596, 628–630. *See also specific types of sensor*
  angular rate sensor, 223–226
  automotive control system applications
    airflow rate sensor, 186–190
    electronic engine control system, 184–185, 185*f*
    engine crankshaft angular position sensor, 194–195
    Hall-effect position sensor, 205–208
    magnetic reluctance position sensor, 195–205
    optical crankshaft position sensor, 208–210
    pressure measurements, 191–194
    variables in engine control, 185–186
  for feedback control
    exhaust gas oxygen sensor, 215–220
    oxygen sensor improvements, 220
  knock sensors, 221–223
  noise model, 631*f*
  photoconductive optical, 32
  p-n junction optical, 32
  temperature sensors, 213
  terminal voltage, 201
  throttle angle sensor, 211–213
  typical coolant sensor, 214
Sequential sampling, 418*f*
Series hybrid vehicle (SHV), 332, 332*f*
Service bay diagnostic tool (SBDT), 535–536, 553–555
Seven-segment display, 432, 432*f*, 434*f*
Shielded-field sensor, 208, 209*f*
Shift register, 84–86, 85*f*
Shift schedule, 329
Shock absorber damping, 378–379, 393
Short-range wireless communications, 488–490
Signal processing, 410–411, 508–509, 515
  algorithm, 419
  complexity, 508
  computer-based, 449
  for crash sensing, 510
  to D/A converter, 414
  digital, 411, 414–415, 449
  linear operation, 411
Signal-to-noise ratio (SNR), 519
Silicon, doping, 24–27
Simulink simulation model, fuel quantity, 451*f*
Sinusoidal frequency response, 645–651
Sliding mode observer (SMO), 373–374, 564–566
Sloshing effect, fuel gauge, 449
Small-signal linear incremental transistor model, 44
SMO. *See* Sliding mode observer (SMO)
Society of Automotive Engineers (SAE), 464, 543–544, 573–574
Solenoid, 248, 248*f*, 250, 329
Solid-state devices. *See* Semiconductor, devices
Solid-state diode, 27
Solid-state laser, 34
Spark advance (SA), 310–311
  correction scheme, 317–318
Speech recognition, 503
Speed-density method, air mass measurement, 173–175
Speed sensor, 353–356, 355*f*
Spread spectrum
  CDMA, 483, 485, 489–490
  frequency-hopping, 489–490
  technique, 483, 489–490
Sprung mass, 377–379, 384
  dynamic models, 388
  forces acting on, 380, 382
  in frequency region, 384
  RMS value, 391
Star topology, 474, 475*f*
State variable formulation of models, 608–610
Status register, CPU, 98–99, 98*f*
Steady-state operating motor speed, 336
Steady-state sinusoidal (SSS) frequency response of system, 606–608
Steering coefficient, 526
Steering control system, 586–588
Steering doublet, 576
Stepper motor, 9, 268, 289–290, 583
Stepper motor-based actuator, 356–357, 360–362, 361*f*
Stimulated emission, 34
Strain gauge MAP sensor, 191–194
Strut damping, 391, 394–395, 394*f*
Summing mode amplifier, 56–66, 56*f*
Sun gear, 326–327
Suspension system, 377–397, 378*f*, 380*f*
  active, 379
  classes of, 378–379
  configuration, 397, 397*f*
  electronic steering control, 398–401, 399–401*f*
  normal force variation *vs*. frequency, 383*f*, 384
  parameters, 392*t*
  purpose of, 377–378
  QCM, 387, 387*f*, 389

Suspension system *(Continued)*
second-order differential equation, 382
semiactive, 378–379
strut damping, 391, 394–395, 394*f*
variable damping via variable strut fluid viscosity, 395–396
variable spring rate, 396–397
Synchronous counter, 83*f*
register circuits, 83–84, 84*f*

T

TC. *See* Turbocharger (TC)
TDM. *See* Time-domain multiplexing (TDM)
TDMA. *See* Time-division multiplexing access (TDMA)
Temperature-compensating resistance, 189
Temperature sensors, 213
Thermistor, 452
Thin-film-transistor liquid-crystal display (TFT-LCD), 423–424, 429, 434–435
Three-way catalyst (TWC), 162–164, 163*f*
Throttle actuator, 356–359, 358*f*
Throttle angle sensor, 211–213
Throttle body fuel injectors (TBFIs), 277–278
Throttle position sensor (TPS), 185
Time-division multiplexing access (TDMA), 17, 476, 483
Time-domain multiplexing (TDM), 65, 415
Timed sequential port fuel injection (TSPFI), 278
Timer circuit, 80–82, 80*f*
Timing master, 480–481
Timing sensor, for ignition and fuel delivery, 204–205
Tire pressure monitoring system (TPWS), 522–523, 524*f*
Tire slip
angle, 528–529
controller, 377
Top dead center (TDC), 194, 255
Torque
engine performance, 150–152, 151–152*f*
restoring, 398–399, 582–583
Torque converter (TC), 327
lock-up control, 329
Torque-converter-locking clutch (TCC), 329
Touch screen, 443–447, 444–445*f*
TPWS. *See* Tire pressure monitoring system (TPWS)
Traction control, 329–331
Transceiver circuit
CAN, 470*f*
FlexRay IVN, 477–478, 477–478*f*
Transfer function
APPS, 578
closed-loop, 353
control block, 578, 586
definition, 389
discrete-time control system, 352
forward-path, 578
frequency response, 389, 390*f*
galvanometer, 422–423
plant for zero disturbance, 347
relating braking force to pitch angle, 382
for roll dynamics, 385
z-operational, 352
Transflexive LCD, 429
Transistor, 37–45. *See also* Field-effect transistor (FET)
base-emitter junction of, 39
bipolar, 38
field-effect, 45–47, 46*f*
NPN, 37
amplifier circuit, 38*f*
PNP, 37
schematic symbols, 37, 38*f*
Transmit/receive (TR) switch, 516–517
Trigger pulse, 445–446
Trilateration process, 494–495, 498–500
Trip information system, 457–460, 458*f*
Turbine shaft, 303
Turbocharger (TC), 302–306
closed-loop control system, 305–306
configuration, 303–304, 304*f*
first-order model, 305
subsystem, 306*f*
waste gate, 305
TWC. *See* Three-way catalyst (TWC)
Twisted nematic liquid crystal, 426
Two-phase induction motor, 263–266
Typical coolant sensor, 214

U

Understeer coefficient. *See* Steering coefficient
Unsprung mass, 377–379
dynamic models, 388
for low damping, 384

V

Vacuum-fluorescent display (VFD), 429–431
brightness control range for, 431*f*
configuration, 430*f*
Vacuum-operated actuator, 357, 362–364, 363*f*
throttle actuator, 358*f*, 364
Valence band, 26
Valve system, on volumetric efficiency, 175–176
Variable reluctance sensor, 201, 202*f*
equivalent circuit for, 202–203, 203*f*
Variable spring rate, 396–397
Variable valve phasing (VVP), 10, 148, 296
Bode plot, 299, 299*f*
configuration, 298

discrete-time model, 299
dynamic response, 298
mechanism, 255, 257–258
physical configuration, 297, 297*f*
PID control law, 298
Variable valve timing (VVT), 10, 175, 254–258
control, 296–301
Vehicle communication, 461–462
GPS, 493–503
control configuration, 501*f*
error feet *vs.* time, 500*f*
pseudorange model, 493–497, 500–501
structure, 500–503
vehicle position, 499*f*
IVN, 462–464, 463*f*
CAN, 464–472, 466*f*, 468–470*f*, 473*f*
FlexRay, 474–478, 475–478*f*
local interconnect network, 472–474, 473*f*
MOST, 478–481, 479–480*f*
QPSR, 487–488, 487*f*
satellite, 490–493
vehicle-to-cellular infrastructure, 482–486, 484*f*, 486*f*
vehicle to infrastructure communication, 481–482, 503–504
wireless communications, short-range, 488–490
Vehicle dynamic motion, 343–344, 349, 378, 387
Vehicle motion control, 343–344
antilock braking system, 368–377, 368*f*, 370–372*f*, 374*f*, 376*f*
cruise control
advanced, 364–368, 365*f*, 367*f*
analog configuration, 363*f*
block diagram, 348*f*
configuration, 347*f*
digital, 351–353, 351*f*, 354*f*, 359*f*
hardware implementation issues, 354–356, 355–356*f*
speed performance, 350*f*
stepper motor-based actuator electronics, 360–362, 361*f*
throttle actuator, 356–359, 358*f*
vacuum-operated actuator, 362–364, 363*f*
electronic suspension system, 377–397, 378*f*, 380*f*
active, 379
classes of, 378–379
configuration, 397, 397*f*
electronic steering control, 398–401, 399–401*f*
normal force variation *vs.* frequency, 383*f*, 384
parameters, 392*t*
purpose of, 377–378
QCM, 387, 387*f*, 389
second-order differential equation, 382
semiactive, 378–379
strut damping, 391, 394–395, 394*f*
variable damping via variable strut fluid viscosity, 395–396
variable spring rate, 396–397
four-wheel steering, 401–408, 402*f*, 407*f*
Vehicle speed, measurement, 456–457, 456*f*
Vehicle status sensors, 412–413
Vehicle-to-cellular infrastructure, 482–486, 484*f*, 486*f*
Vehicle to infrastructure communication, 481–482
safety aspects of, 503–504
Vehicle to infrastructure (V2I/V2X) communication, 592–593
Vehicle-to-satellite infrastructure, 482
Vehicle-to-vehicle (V2V) communication system, 592–593
VFD. *See* Vacuum-fluorescent display (VFD)
Video communication, 482
Vision sensors, 581
VVT. *See* Variable valve timing (VVT)

W

Walsh code, 483, 485
Waste gate, 305
Wheatstone bridge, 191–192, 193*f*
Wide open throttle (WOT), 328–329
Wireless communication, short-range, 488–490

X

XOR, 484
circuit equivalent of, 484, 484*f*

Y

Yaw, 343
rate, 524–527, 530

Z

ZCD. *See* Zero-crossing detector (ZCD)
Zener diode, 29, 30*f*
Zero-crossing detector (ZCD), 58, 81*f*, 567
Zero-order hold (ZOH), 298
circuit, 60–64, 63*f*
Zirconium dioxide (ZrO_2), 215–216
ZOH. *See* Zero-order hold (ZOH)
Z-transform, 642
elementary properties of, 642–643
inverse, 643, 645