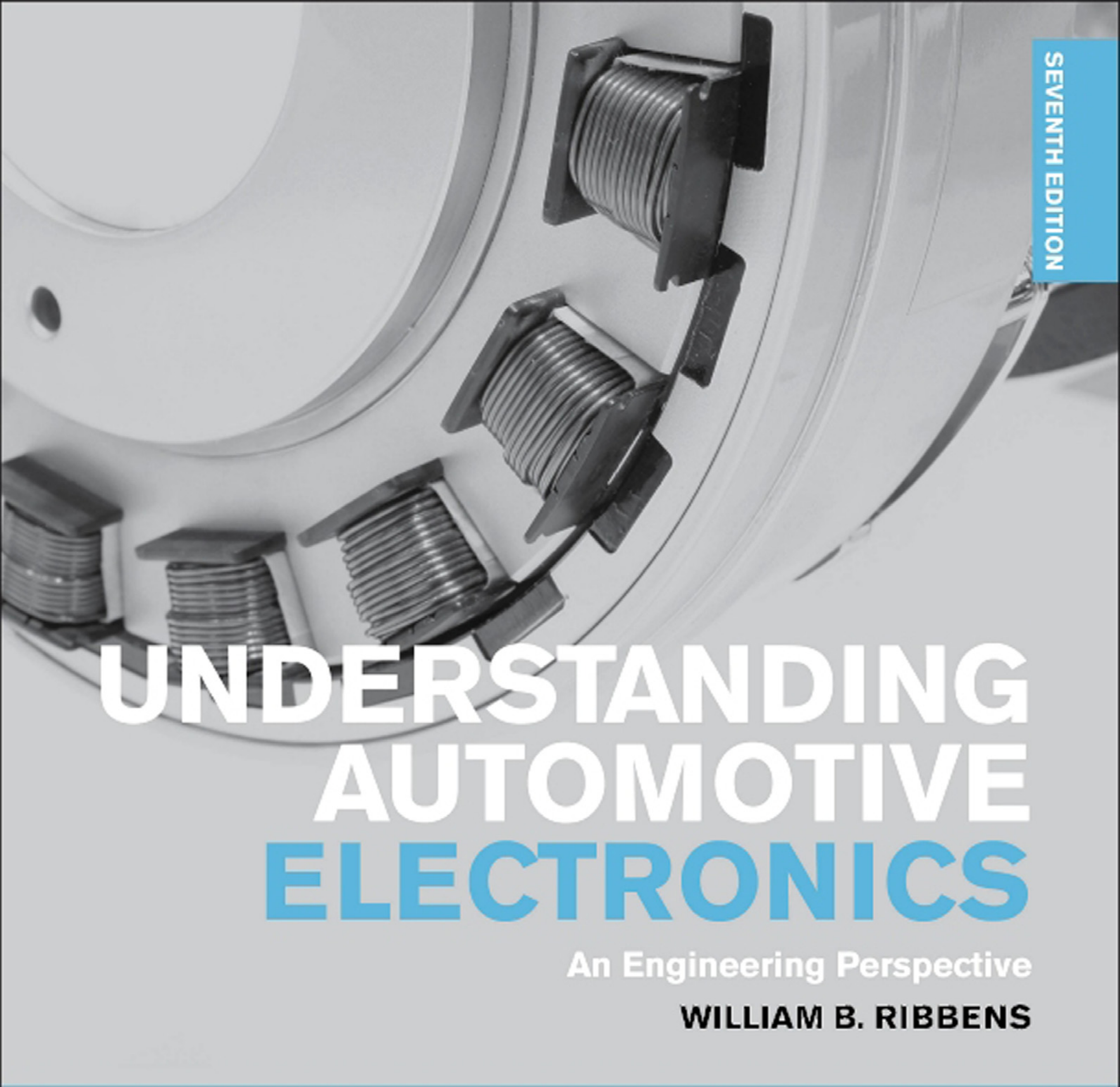
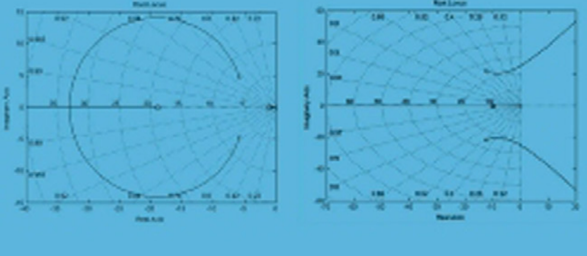


SEVENTH EDITION

UNDERSTANDING AUTOMOTIVE ELECTRONICS

An Engineering Perspective

WILLIAM B. RIBBENS



Understanding Automotive Electronics

An Engineering Perspective

I would like to thank my wife 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.

Understanding Automotive Electronics

An Engineering Perspective

Seventh edition

William Ribbens

ELSEVIER

AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO

Butterworth−Heinemann is an imprint of Elsevier

Butterworth-Heinemann is an imprint of Elsevier
225 Wyman Street, Waltham, MA 02451, USA
The Boulevard, Langford Lane, Kidlington, Oxford, 0X5 1GB, UK

Sixth edition 2003
Seventh edtion 2012

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
Application submitted

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

ISBN: 978-0-08-097097-4

Typeset by: TNQ Books and Journals, Chennai, India

Printed in the United States of America
13 10 9 8 7 6 5 4 3 2 1

Contents

Preface

The present edition of this book is an extensive revision of the previous editions based upon suggestions of reviewers. The present edition has a strong engineering focus including analytical models and quantitative performance analysis of electronic components/ subsystems and systems found in contemporary automobiles. However, the largely qualitative explanations of automotive electronic systems from previous editions has been retained and, in some cases, expanded wherever it has been possible to do so.

It has been the intention in writing this book to make it accessible to readers who have not had the formal training in physical sciences and mathematics (as well as those who have) to understand the functional operation of automotive electronic systems. The author recommends that such individuals lacking the science background skip over the mathematical portions of the text and concentrate on the qualitative discussion and verbal explanations preceding and following the mathematical sections of the book.

It is hoped that the mathematical models and performance analyses presented throughout this book will be informative to readers with backgrounds in the sciences, engineering and/or mathematics. There may even be engineers/scientists working within the automotive industry for whom a detailed discussion of electronic subsystems and components will be useful.

The first chapter in previous editions presented an overview of basic functional components of an automobile. This chapter has been removed, although a shortened version of the explanation of certain automotive components and subsystems (e.g., engine, drivetrain, braking, steering etc.) is presented as deemed necessary in the new chapters of this edition.

There are some topics in the present edition of this book that were not covered in detail in any of the previous editions. These include the theory of electric motors, which find application in hybrid/electric vehicles. Telematics, which is an increasingly important automotive technology, is covered in considerable detail (including the theory of GPS navigation systems). In addition, the theory of onboard diagnosis of problems with exhaust emission control systems is presented for the first time in any of the editions of this book.

It has been the intent in writing this book to emphasize the fundamental aspects of the application of electronics to automobiles. Ideally, if this objective has been achieved, the reader should understand evolving technologies as well as totally new automotive electronic technologies as they occur in future vehicles.

This page intentionally left blank

Introduction

This book covers the general topic of the application of electronics in automobiles and light trucks. Some of the technology described herein is also found in large trucks and other land vehicles, although these applications are not explicitly discussed.

The only important use of electronics in automobiles through the late 1950s was the broadcast band radio receiver, which was based upon vacuum tube technology. The development of solid-state electronics, from the first transistors through the latest high performance integrated circuits, came at a time that permitted the very sophisticated electronic systems discussed in this book to be applied to solving automotive control and instrumentation problems.

The book is organized in a way that allows the reader to select his/her own desired starting point depending upon background and/or experience. The first two chapters provide a review of linear system theory at the level found throughout the remainder of the book.

The new first and second chapters present a survey of various aspects of linear system theory. The new first chapter discuses continuous time (analog) theory and the new second chapter discuses discrete time (digital) system theory. Neither of these two chapters has sufficient breadth or depth to serve as a full ab *initio* presentation of system theory. For many readers, they will be redundant. Nevertheless, they provide a review of linear system theory and introduce notation that is used throughout this book.

Chapter 3 discusses the basics of solid-state active elements (e.g. diodes and various types of transistors) along with circuits that form the basic building blocks of automotive electronic systems. Chapter 4 discusses microprocessors/microcontrollers and certain fundamental aspects of their application in automotive electronic systems. Chapter 5 presents the fundamentals of electronic engine control. Chapter 6 surveys sensors and actuators found in automotive electronic control or instrumentation systems that are arguably the most important components in such systems. The remaining chapters discuss specific automotive systems that incorporate electronics for control or instrumentation purposes. The automotive components/systems covered include engine, drivetrain, suspension, steering, brakes, instrumentation, telematics and diagnostics as well as motion control. All of the mathematical models and performance analyses are given in terms of the theory and methods of the first and second chapters. Extensive use is made of contemporary computer simulation.

Specific examples are presented for most electronic systems/components for which component models and performance analyses are given. These examples are based upon simplified models and linear system theory such that the basic principles involved are illustrated without the cumbersome mathematical details (and nonlinearities) that are often found in the analysis of practical production vehicle systems.

Furthermore, the examples given in this book of any component/system are representative of those found in any production vehicle. Proprietary issues prohibit the detailed discussion of such production items. In addition, the technology of automotive electronics found in production vehicles is constantly evolving and any detailed discussion of a given vehicle electronic system as of the writing of this book might well be obsolete in the next model year.

There is an important notational issue concerning the analytical portions of this book. There are many variables and parameters involved in the various components/systems discussed in each chapter. The symbols used in this book include upper and lower case Roman and Greek (often subscripted) letters. However, the reader should be aware that these symbols do not have a global definition throughout the book. Rather, each symbol is defined for the section of the book in which it is used.

The units and dimensions for variables discussed in this book are English in keeping with earlier editions, even though the worldwide automotive industry uses essentially all metric units. However, conversion tables from English to metric units are readily available. Moreover, those readers in the USA, having relatively limited backgrounds in the physical sciences, are likely to be more familiar with English rather than metric units. Readers having backgrounds in the sciences will find unit conversion straightforward.

CHAPTER 1

The Systems Approach to Control and Instrumentation

Chapter Outline

Chapter 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

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-08-097097-4.00001-1

groups. This chapter 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 chapter 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 and digital or discrete time – 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 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 chapter, 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 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. 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 Figure 1.1. 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

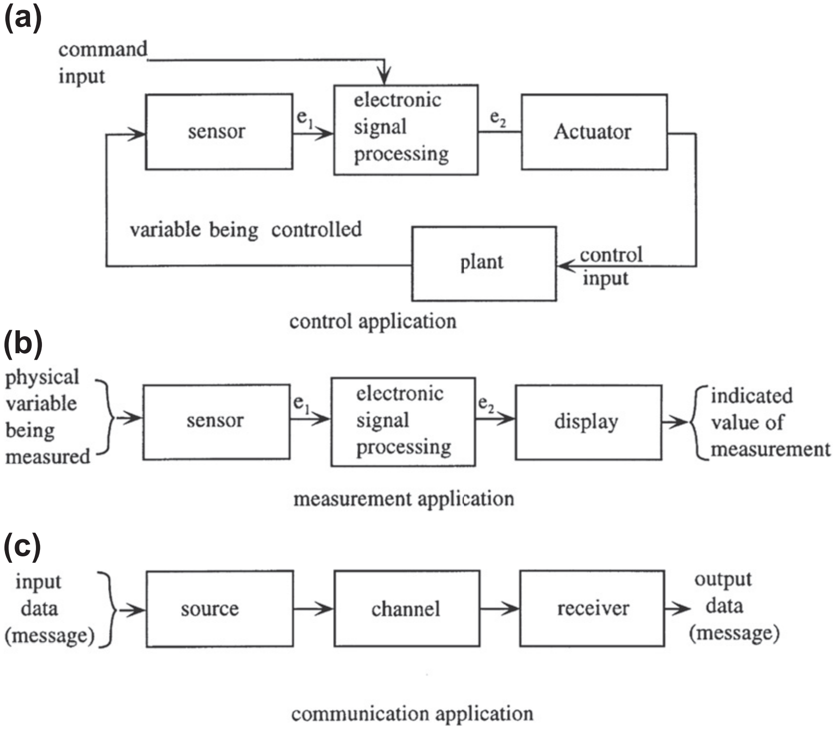


Figure 1.1:
Electronic system block diagrams variable being controlled.

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, i.e., its basic physics. Initially, this chapter will consider components, subsystems, and system blocks that can be represented by a linear mathematical model. Later in the chapter, 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 of the form

$$a_o y + a_1 \frac{\mathrm{d}y}{\mathrm{d}t} + \ldots\ a_n \frac{\mathrm{d}^n y}{\mathrm{d}t^n} = b_o x + b_1 \frac{\mathrm{d}x}{\mathrm{d}t} \ldots\ b_m \frac{\mathrm{d}^m x}{\mathrm{d}t^m} \tag{1}$$

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 chapter.

Figure 1.1a 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 required to perform the desired system function output. 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 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).

In an electronic control system, the output of the sensor is always an electrical signal (denoted e_1 in Figure 1.1). 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 Figure 1.1) 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 chapter. At this point, we are interested only in describing the control-system architecture. A redundant explanation of electronic control is presented later in this chapter.

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 Figure 1.1b. It should be noted that both control and instrumentation electronic systems use one or more sensors as well as electronic signal processing.

Figure 1.1c 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, as well as extravehicular communication. Communication systems are described in detail later in this chapter.

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 Eqn (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 the next chapter.

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 Eqn (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 $x(s) = \mathscr{L}[x(t)]$) of the input is denoted $x(s)$ and is defined as following the linear integral transform of its time domain representation:

$$x(s) = \int_{o}^{\infty} e^{-st} x(t)\mathrm{d}t + x(t)\big|_{t=0} \tag{2}$$

where

$$s = \sigma + j\omega = \text{complex frequency}$$

and where

$$j = \sqrt{-1} \tag{3}$$

Similarly, the Laplace transform of the block output is denoted $y(s)$ and is given by

$$y(s) = \int_{o}^{\infty} e^{-st} y(t)\mathrm{d}t + y(t)\big|_{t=0} \tag{4}$$

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

$$\int_{o}^{\infty} e^{-st} \frac{\mathrm{d}^m x}{\mathrm{d}t^m}\mathrm{d}t = s^m x(s) \qquad m = 1, 2 \ldots \tag{5}$$

Assuming for simplicity that the initial conditions for both input and output are zero,

$$x(t)\,\big|_{t=0} = 0, \quad y(t)\,\big|_{t=0} = 0$$

the Laplace transform of the differential equation (Eqn (1)) for the block yields

$$[a_0 + a_1 s + a_2 s^2 \cdots a_n s^n] y(s) = [b_0 + b_1 s + \cdots b_m s^m] x(s) \tag{6}$$

It is conventional for the purpose of conducting analysis for continuous time systems to define the transfer function ($H(s)$) for each block:

$$\begin{aligned} H(s) &= \frac{y(s)}{x(s)} \\ &= \frac{b_0 + b_1 s + \cdots b_m s^m}{a_0 + a_1 s + \cdots a_n s^n} \end{aligned} \tag{7}$$

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 Figure 1.2) is the product of the transfer functions of the individual blocks.

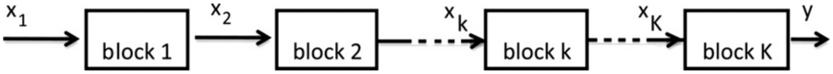


Figure 1.2:
System cascade connection block diagram.

Denoting the transfer function of the *k*th block $H_k(s)$, and for the complete system $H(s)$, the latter is given by

$$H(s) = \prod_{k=1}^{K} H_k(s) \tag{8}$$

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

$$P_N(s) = b_0 + b_1 s + \cdots b_m s^m = 0 \tag{9}$$

where

$$P_N(z_j) = 0 \qquad\qquad j = 1, 2\ldots m$$

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 equation

$$P_D(s) = a_0 + a_1 s + \cdots a_n s^n = 0 \tag{10}$$

where

$$P_D(p_i) = 0 \qquad\qquad i = 1, 2\ldots n$$

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

$$H(s) = \frac{\prod_{j=1}^{m} (s - z_j)}{\prod_{i=1}^{n} (s - p_i)} \tag{11}$$

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

$$Y(s) = H(s)X(s) \tag{12}$$

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 $H(s)$ as well as any zeros and poles of the input:

$$Y(s) = \frac{\prod_{j=1}^{m} (s - z_j)}{\prod_{i=1}^{n} (s - p_i)} \tag{13}$$

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

$$y(t) = \sum_{k=1}^{N} (s - p_k)H(s)X(s)\Big|_{s \to p_k} e^{p_k t} \tag{14}$$

In evaluating this expression, the pole at $s = p_k$ is canceled by the term $(s - p_k)$ in the remaining portion 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:

$$y(t) = \mathscr{L}^{-1}[Y(s)] \tag{15}$$

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 $H(s)$:

$$h(t) = \mathscr{L}^{-1}[H(s)] \tag{16}$$

It is the response of a linear system to a unit impulse. The output of a linear system can be found from its impulse response by the so-called *convolution theorem* which is given as follows:

$$y(t) = \int_{-\infty}^{\infty} h(\tau)\, x\,(t - \tau)\mathrm{d}\tau \tag{17}$$

For any causal stable system, the impulse response is zero for negative argument:

$$h(\tau) = 0 \qquad\qquad \tau \leq 0 \tag{18}$$

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

$$a_0 y + a_1 \frac{\mathrm{d}y}{\mathrm{d}t} = b_0 x \tag{19}$$

The transfer function for this block is given by

$$H(s) = \frac{b_0}{a_0 + a_1 s} = \frac{\frac{b_0}{a_1}}{s + \frac{a_0}{a_1}} \tag{20}$$

This transfer function has a single pole

where

$$p = -\frac{a_0}{a_1}.$$

The block output has poles at $s = 0$ and $s = -\frac{a_0}{a_1}$.

The time response of this block to a unit step is given by

$$y(t) = \frac{b_0}{a_1}\left[1 - \exp\left(-\frac{a_0 t}{a_1}\right)\right] \tag{21}$$

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 (sec) and is called the first-order time constant for the system. Continuing with the present example, Figure 1.3 depicts the unit step response of a first-order system in which $b_0 = 3$, $a_0 = 0.5$, and $a_1 = 1.0$. A second-order system has two poles which are either both negative real or a 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 step response example is presented later in this chapter. For higher than second order, no simple intuitive description of the response is possible. Such higher-order systems are encountered as examples later in this book.

The dynamic response of any linear continuous time automotive system (regardless of its physical form e.g. electronic, mechanical etc) is found in the same procedure as given above. That is, a mathematical model of the system of the form of Eqn (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 Eqn (1) by differentiating all terms with respect to time.

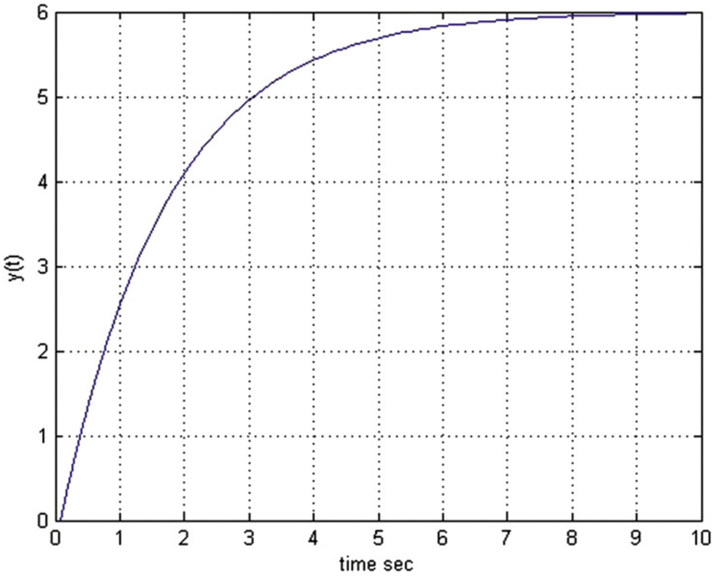


Figure 1.3:
Unit step response of a first-order system.

Alternatively, in the second step of taking the Laplace transform of the equation, the Laplace transform of the integral of a variable, e.g.

$$\int_0^t x(t')\,\mathrm{d}t'$$

(with assumed zero initial condition) is given by

$$\int_0^\infty e^{-st}\left(\int_0^t x(t')\mathrm{d}t'\right)\,\mathrm{d}t = \frac{x(s)}{s} \tag{22}$$

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 later in 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 remaining 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 Figure 1.4. This example is a highly oversimplified lowest approximation to an automotive suspension system at one wheel (i.e., quarter car). 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 applied vertically to mass M_u. The instantaneous displacement of mass M_u relative to its position for $F = 0$ is denoted $y(t)$.

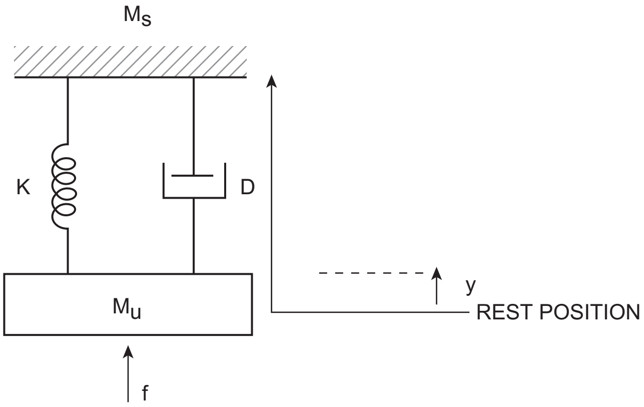


Figure 1.4:
Example second-order system configuration.

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:

$$F - M_u\ddot{y} - D\dot{y} - Ky = 0 \tag{23}$$

where

$$\dot{y} = \frac{\mathrm{d}y}{\mathrm{d}t}$$

$$\ddot{y} = \frac{\mathrm{d}^2y}{\mathrm{d}t^2}$$

The second term on the left-hand side of the equation is the inertial force associated with acceleration of mass M_u. The 3rd 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 common notation for its derivative with respect to time. This differential equation can readily by put in a form similar to our standard model of Eqn (1):

$$M_u\ddot{y} + D\dot{y} + Ky = F \tag{24}$$

It is common 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 is

$$\ddot{y} + 2\zeta\omega_0\dot{y} + \omega_0^2 y = \frac{F}{M_u} \tag{25}$$

The solution to this differential equation is most readily found using the Laplace transform method explained earlier in this chapter. Assuming for convenience that the system is initially at rest (i.e., $F(0) = 0$, $\delta y(0) = 0$), the Laplace transform of the differential equation is given by

$$\left(s^2 + 2\zeta\omega_0 s + \omega_0^2\right) y(s) = \frac{F(s)}{M_u} \tag{26}$$

The operational transfer function is given by

$$H(s) = \frac{y(s)}{F(s)} = \frac{1/M_u}{s^2 + 2\zeta\omega_0 s + \omega_0^2} \tag{27}$$

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 (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 in the form

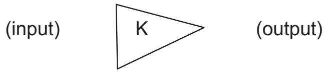
$$\ddot{y} = F(t) - \frac{D\dot{y}}{M_u} - \frac{Ky}{M_u}$$

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

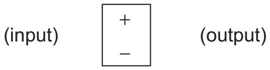
$$\boxed{\frac{1}{s}}$$

That is, the operational transfer function of a time integral is $\frac{1}{s}$ 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 ± 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:

$$\dot{y} = \int \ddot{y}\mathrm{d}t$$
$$y = \int \dot{y}\mathrm{d}t$$

These relationships are implied by the MATLAB/SIMULINK system depicted in Figure 1.5.

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

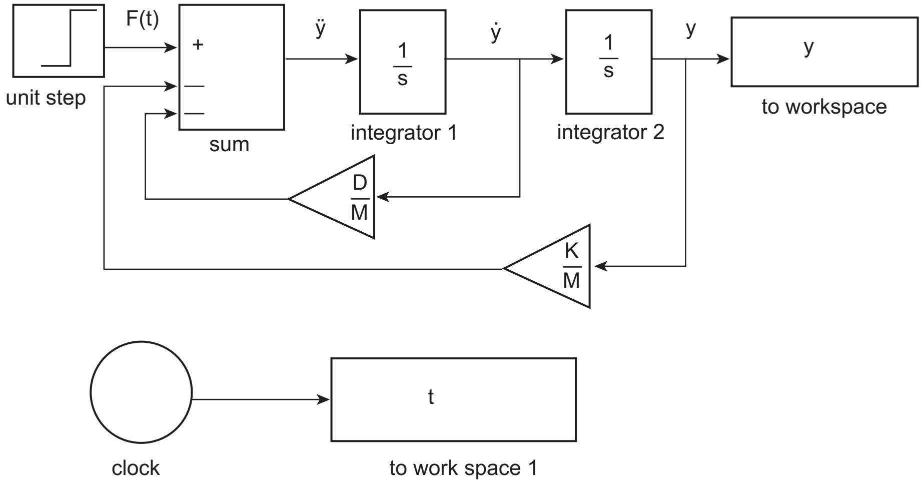


Figure 1.5:
MATLAB/SIMULINK second-order system.

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 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.

To further illustrate the use of this SIMULINK model, the following specific parameters were arbitrarily chosen:

$$M_u = 1.7$$
$$K = 50$$
$$\mathrm{D} = 5.83$$

The input is a unit step at $t = .5$ s, i.e.,

$$\begin{aligned} F(t) &= 0 \quad t \leq .5 \\ &= 1 \quad t \geq .5 \end{aligned}$$

Figure 1.6 is a graph of $y(t)$ depicting the motion of the mass M_u in response to this step.

In later chapters of this book, we will present examples of the dynamic response of automotive systems/subsystems with models which are much more accurate in representing the actual physical devices.

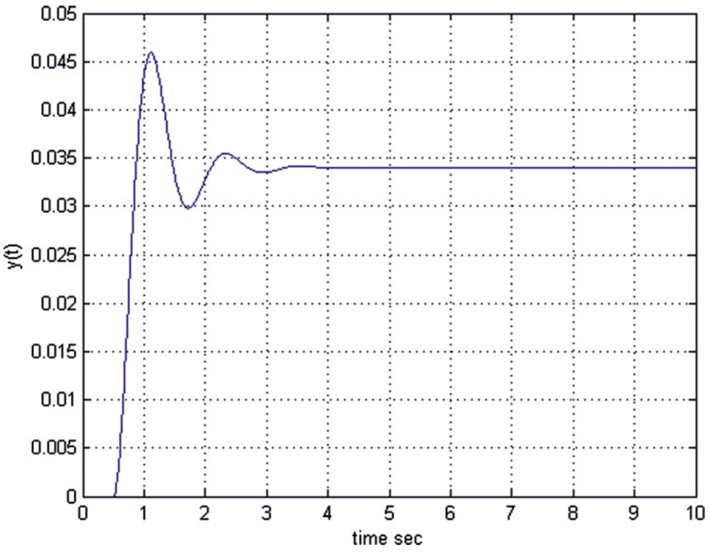


Figure 1.6:
Unit step response of example second-order system.

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 complex analysis that simplifies the calculation of the sinusoidal frequency response of any system which is given by

$$e^{j\omega t} = \cos(\omega t) + j\sin(\omega t) \tag{28}$$

This identity can also be expressed by the following relations:

$$\begin{aligned} \cos(\omega t) &= \mathrm{Re}[e^{j\omega t}] \\ \sin(\omega t) &= \mathrm{Im}[e^{j\omega t}] \end{aligned} \tag{29}$$

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 response. The transient response is the dynamic output of the system to the initial application of the sinusoid. Each component of this transient response decays exponentially to zero. Following the period in which the transient decays to zero, the remaining output is a steady-state sinusoidal (SSS) output. It is the SSS response that is the goal of the system analysis with input

$$x(t) = \mathrm{Re}[Xe^{j\omega t}]$$

The steady-state sinusoidal (SSS) frequency response system output (after all transients have decayed to zero) is of the form

$$y(t) = \mathrm{Re}[Y(j\omega)e^{j\omega t}]$$

The steady-state sinusoidal sinusoidal frequency response, which is denoted $H(j\omega)$, is defined as

$$H(j\omega) = \frac{Y(j\omega)}{X(j\omega)} \tag{30}$$

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 $(j\omega)$:

$$H(j\omega) = H(s)\Big|_{s=j\omega} \tag{31}$$

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:

$$H(j\omega) = |H(j\omega)|\; e^{j\phi(\omega)} \tag{32}$$

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

$$\begin{aligned} \text{magnitude}: &\ \ 20\log|H(j\omega)|\text{vs. }\log(\omega) \\ \text{phase}: &\ \ \phi(\omega)\text{vs. }\log(\omega) \end{aligned}$$

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

$$\dot{x}_m = \sum_{n=1}^{N} A_{mn}x_n + \sum_{k=1}^{K} B_{mk}u_k \tag{33}$$

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

$$\begin{aligned} \dot{x}_1 &= A_{11}x_1 + \cdots A_{1N}x_N + B_{11}u_1 \cdots B_{1K}u_K \\ &\vdots \\ \dot{x}_N &= A_{N1}x_1 \cdots A_{NN}x_N + B_{N1}u_1 \cdots B_{NK}u_K \end{aligned} \tag{34}$$

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:

$$\dot{x} = Ax + Bu \tag{35}$$

where, for any real physical system,

$$\begin{aligned} &x \in R^N \\ &u \in R^K \\ &A \in R^{N \times N} \\ &B \in R^{N \times K} \end{aligned}$$

The desired output variables for the system

$$y_j = \sum_{n=1}^{N} C_{jn}x_n + \sum_{k=1}^{K} D_{jk}u_k \quad j = 1, 2 \cdots J \tag{36}$$

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

$$\begin{aligned} y &= [y_1 y_2 \cdots y_J]^T \\ y &= Cx + Du \end{aligned} \tag{37}$$

where

$$\begin{aligned} &y \in R^J \\ &C \in R^{J \times N} \\ &D \in R^{J \times K} \end{aligned}$$

The complete model for the system in standard state variable form is given by

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{38}$$

This system of equations is solved 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:

$$\begin{aligned} sx(s) &= Ax(s) + Bu(s) \\ y(s) &= Cx(s) + Du(s) \end{aligned} \tag{39}$$

The first of these equations can be rewritten in the form

$$(sI - A)x(s) = Bu(s) \tag{40}$$

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 $(sI{-}A)^{-1}$:

$$x(s) = (sI - A)^{-1}Bu(s) \tag{41}$$

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 :

$$y_j(s) = \sum_{k=1}^{K} H_{jk}(s)\; u_k(s) \quad j = 1, 2, \ldots J \tag{42}$$

The response of any output variable y_j to any single input u_k is given by the operational transfer function $H_{jk}(s)$:

$$y_j = H_{jk}(s)u_k(s) \quad k = 1, 2, \ldots K$$

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 later in the book with respect to automotive electronic systems.

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 later 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 later chapters. The architecture of an open-loop system is given in the block diagram of Figure 1.7.

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 Figure 1.7, 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 Figure 1.7). This electrical signal is the input to the actuator which generates a control input (denoted u in Figure 1.7) 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 electric motor). Chapter 6 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 later.

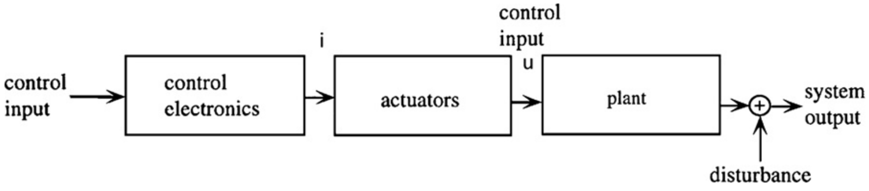


Figure 1.7:
Open-loop system configuration.

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 to 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 chapter).

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 Figure 1.8. 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 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 causes the plant input to change in such a way as to reduce the error toward zero. The configuration of Figure 1.8 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 which 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), which receives this electrical input, and cause 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

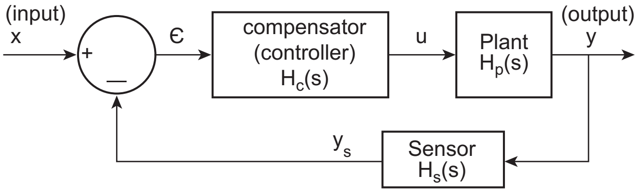


Figure 1.8:
Closed-loop feedback-control-system configuration.

is implemented by a microprocessor or microcontroller as explained in later 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. The next chapter 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:

$$\text{error } \in \ = x - y_s \tag{43}$$

$$\text{plant } \ y(s) = H_p(s)u(s) \tag{44}$$

$$\text{compensator } u(s) = H_c(s) \in (s) \tag{45}$$

$$\text{sensor } y_S(s) = H_S(s)y(s) \tag{46}$$

Combining the models for the components, the following model can be written for the closed-loop system:

$$y(s) = \left(\frac{H_p(s)H_c(s)}{1 + H_s(s)H_p(s)H_c(s)} \right) \tag{47}$$

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:

$$H_{c\ell}(s) = \frac{y(s)}{x(s)}$$

$$H_{c\ell}(s) = \frac{H_p(s)H_c(s)}{1 + H_p(s)H_c(s)} \tag{48}$$

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) $u = K_p \in + K_I \int \in \mathrm{d}t$
3. proportional—integral—differential (PID)

$$u = K_p \in + K_I \int \in \mathrm{d}t + K_D \frac{\mathrm{d}\in}{\mathrm{d}t}$$

The transfer functions for these three types are:

1. P: $H_c(s) = K_p$
2. PI: $H_c(s) = K_p + \dfrac{K_I}{s}$
3. PID: $H_c(s) = K_p + \dfrac{K_I}{s} + K_D s$

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 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 which has a transfer function

$$H_c(s) = K_p + \frac{K_I}{s} \tag{49}$$

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 a change of the output $\left(\text{i.e., } \frac{\mathrm{d}y}{\mathrm{d}t}|_{t=0}\right)$, although it may do so with an overshoot of the final intended value, depending upon the associated gain $\mathrm{K_D}$. The relative benefits of these types of controllers depend upon the particular application as well as the other system criteria.

We illustrate the influence of the compensator on transient response with the second-order system (see Figure 1.4) 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 $F(t)$ which changes its vertical position $y(t)$. It was shown above that the transfer function of this was shown to be

$$\frac{y(s)}{F(s)} = \frac{1/M_u}{s^2 + 2\zeta\omega\ _0 s + \omega_0^2} \tag{50}$$

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 which 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

$$\begin{aligned} \in &= \text{error} \\ &= x - y \end{aligned}$$

The open-loop transfer function of this plant $H_p(s) = y(s)/u(s)$ is given by

$$H_p = \frac{K_a/M_u}{s^2 + 2\zeta\omega_0 s + \omega_0^2} \tag{51}$$

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 $K_a = 50$, $K_p = 53.3$, $M_u = 1.7$, and $D = 17.6$. Using these parameters, the step response to unit force is shown in Figure 1.9.

The closed-loop transfer function is given by

$$H_{c\ell} = \frac{y(s)}{x(s)} \tag{52}$$

$$= \frac{K_pH_p(s)}{1 + K_pH_p(s)} \tag{53}$$

$$\frac{K_aK_p}{M_u\left[s^2 + 2\zeta\omega_0^2 + \omega_0^2 + \frac{K_aK_p}{M_u}\right]} \tag{54}$$

Using the parameters from the earlier example, it is possible to evaluate the step response via simulation using MATLAB/SIMULINK. Figure 1.10 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).

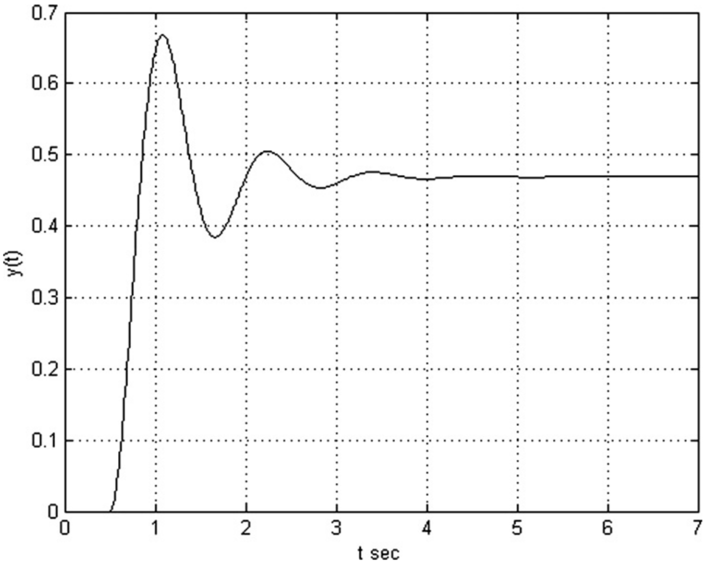


Figure 1.9:
Unit step response of example closed-loop system.

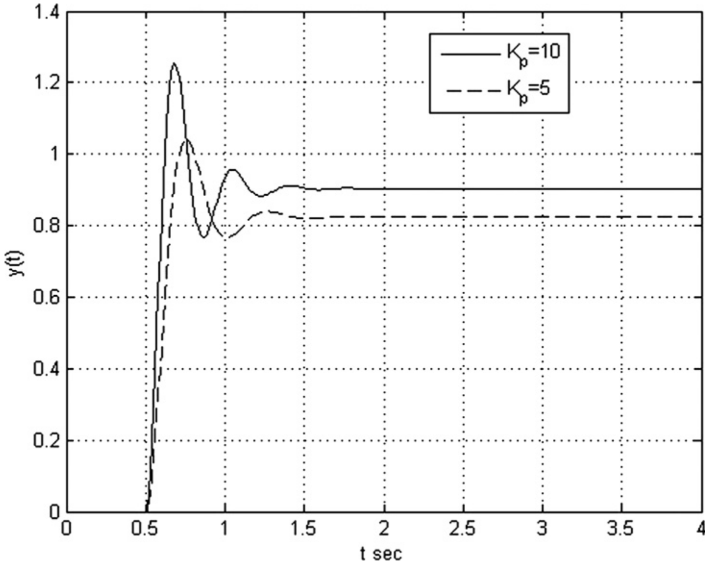


Figure 1.10:
Response of proportional feedback-control system.

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 $K_p = 5$.

Consider next the step response of a PI controller. An integral term $(K_I \int \in \mathrm{d}t)$ was added to the controller of the previous example. The closed-loop transfer function for this system is given by

$$H_{c\ell}(s) = \frac{K_a(K_p s + K_I)/M_u}{(s^2 + 2\zeta\omega_0 s + \omega_0^2)s + (K_p s + K_I)\ \dfrac{K_a}{M_u}} \tag{55}$$

Using the same second-order system parameters, 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. Figure 1.11 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

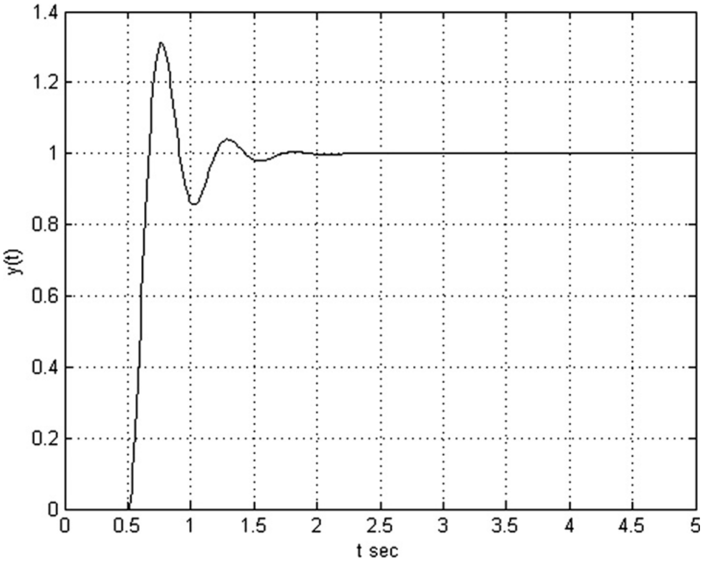


Figure 1.11:
Response of proportional—integral feedback-control system.

$$K_D \frac{d\in}{dt}$$

such that the compensator transfer function is given by

$$H_c(s) = K_p + \frac{K_I}{s} + K_D s \quad (56)$$

Figure 1.12 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. In the figure, $K_p = 5$, $K_I = 10$, and $K_D = 0.5$.

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.

Caution must be exercised by the control-system design in choosing the type of controller as well as 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 chapter.

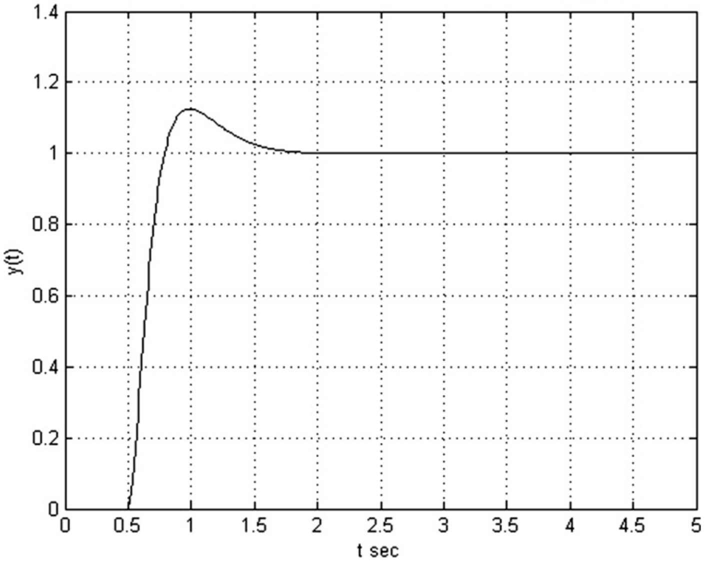


Figure 1.12:
Response of PID control system.

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.

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, 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

$$H_{cl} = \frac{H_c(s)H_p(s)}{1 + H_c(s)H_p(s)} \tag{57}$$

The poles of this transfer function are the zeros (in the complex s-plane) of the function:

$$1 + H_c(s)H_p(s) = 0 \tag{58}$$

In order to assess the influence of a parameter K on system stability, the above equation must be rewritten in the following form:

$$1 + K\,G(s) = 0 \tag{59}$$

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:

$$1 + K\,N(s)/D(s) = 0 \tag{60}$$

or

$$D(s) + K\,N(s) = 0$$

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 Figure 1.4 and having transfer function given by Eqn (51) with a PID controller. The transfer function for this controller is given by

$$H_c(s) = K_p + \frac{K_I}{s} + K_D s \tag{61}$$

which can be rewritten in the form

$$H_c(s) = \frac{K_p}{s}\left(\frac{K_D}{K_p}s^2 + s + \frac{K_I}{K_p}\right) \tag{62}$$

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_p 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

$$G(s) = \frac{\left[\left(\frac{K_D}{K_p}s^2 + s + \frac{K_I}{K_p}\right)\right]}{s(s^2 + 2\zeta\omega_0 s + \omega_0^2)} \frac{K_a}{M_u} \tag{63}$$

The root locus for the system for variations in the parameter K_p with fixed ratios

$$\frac{K_D}{K_p} = 0.5$$

$$\frac{K_I}{K_p} = 1.5$$

is given in Figure 1.13. 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 $K_p = 0$ 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 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 ∀ 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

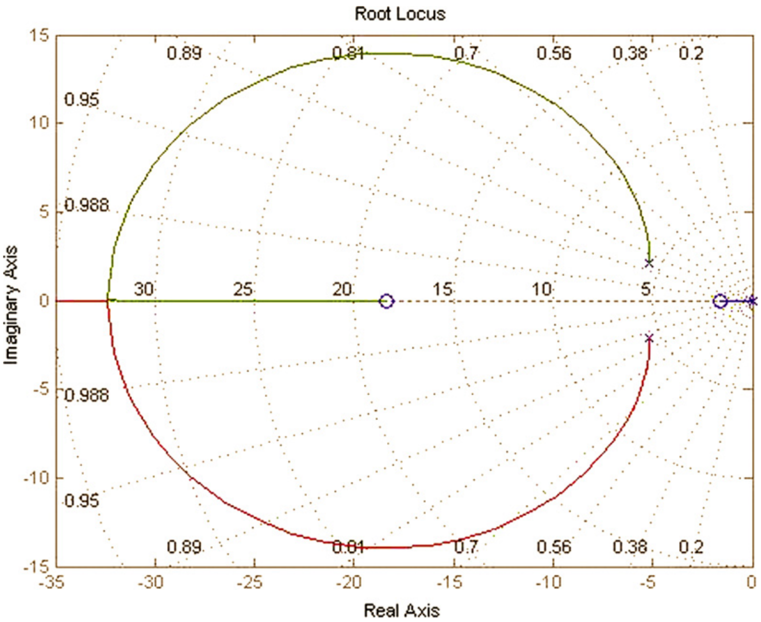


Figure 1.13:
Root-locus plot for example PID control system.

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 analyses 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 for a steady-state sinusoidal excitation. For any given set of controller gains, the characteristic equation can be written as

$$1 + L(s) = 0 \tag{64}$$

where

$$L(s) = H_c(s)\, H_\rho(s) \tag{65}$$

and where *L(s)* is called the loop gain. For steady-state sinusoidal excitation, the characteristic equation is given by

$$1 + L(j\omega) = 0 \tag{66}$$

Instability occurs whenever

$$L(j\omega) = -1 \tag{67}$$

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

$$\begin{aligned} \log|L(j\omega_G)| &= 0 \\ \angle\ L(j\omega_p) &= 180^\circ \end{aligned} \tag{68}$$

The relative stability of a closed-loop system is expressed by two quantities: 1) gain margin (GM) and 2) phase margin (PM). The gain margin is defined as

$$\mathrm{GM} = -20\log_{10}\left|L(j\omega_p)\right| \tag{69}$$

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 phase margin is defined as

$$\mathrm{PM} = \angle\ L(j\omega_G) - 180^\circ \tag{70}$$

The gain and phase margin 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 phase margin using the example second-order system with PID controller. For this illustration, the gain parameters are arbitrarily picked to be: $K_p = 10$, $K_I = 15$, and $K_D = 0.5$. The Bode plot for this example is given in Figure 1.14.

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 gain margin and a phase margin of about −80°. Any system with these GM and PM is highly stable and will remain stable with respect to relatively large system parameter variations.

The control electronics used in the example of Figure 1.10 provided what is called proportional control because the control signal is proportional to the error signal. Other combinations of control electronics are possible, and it is a challenge to the system designer to develop imaginative types of control electronics to improve the performance of a given plant.

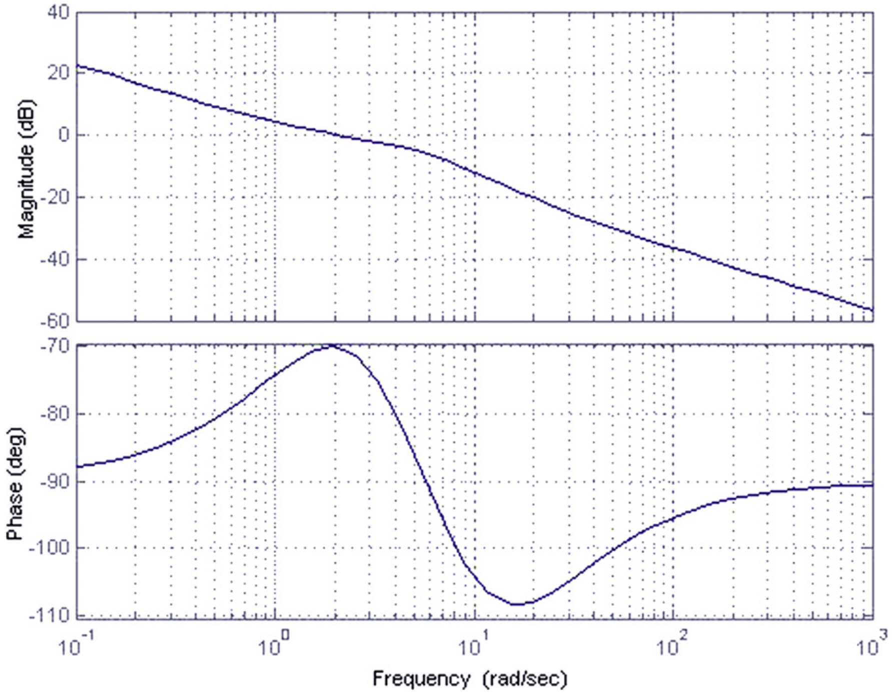


Figure 1.14:
Bode plot for example PID control system.

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.

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 Figure 1.15. 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

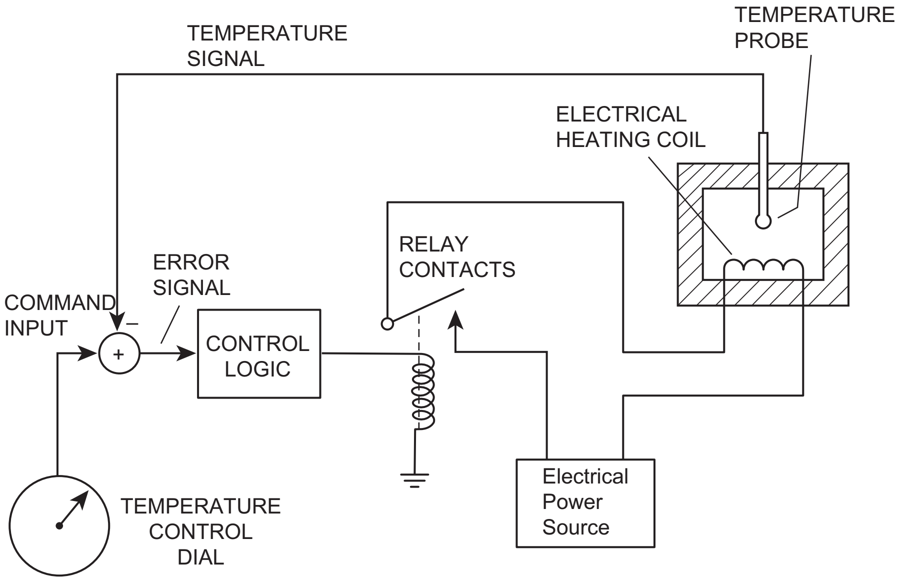


Figure 1.15:
Example limit-cycle control system.

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 in Figure 1.16.

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 Chapter 5).

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.

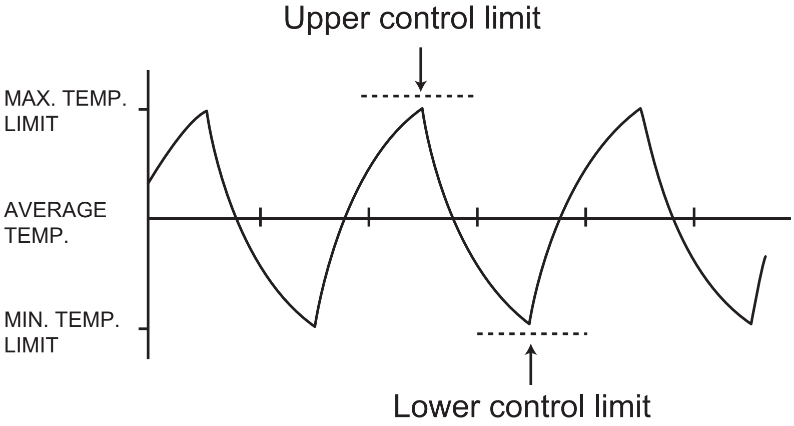


Figure 1.16:
Frequency response for limit-cycle control system.

These electronic instruments or instrumentation systems are used to measure a variety of physical quantities, including

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
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-bounds 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 10.

For an understanding of measurement instrumentation, it is helpful to review a definition of measurement. Automotive instrumentation systems, whether electrical, 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 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

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 & 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 MKS system (meter, kilogram, sec) 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 will be shown in later 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 will be seen later in this book, most automotive instruments have relatively low precision 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 chapter.

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 Figure 1.17. A voltmeter *M* having resistance R_m measures the voltage across resistance *R*. The correct voltage (v_c) is given by

$$\mathrm{v}_c = V\left(\frac{R}{R+R_1}\right) \tag{71}$$

However, the measured voltage v_m is given by

$$\mathrm{v}_m = V\left(\frac{R_p}{R_p+R_1}\right) \tag{72}$$

where R_p is the parallel combination of *R* and R_m:

$$R_p = \frac{RR_m}{R+R_m} \tag{73}$$

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 correct voltage. Loading is similarly minimized in 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 Figure 1.18. 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

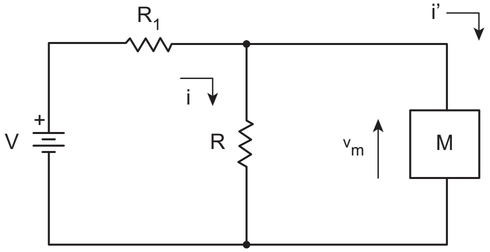


Figure 1.17:
Illustration of loading error-volt meter.

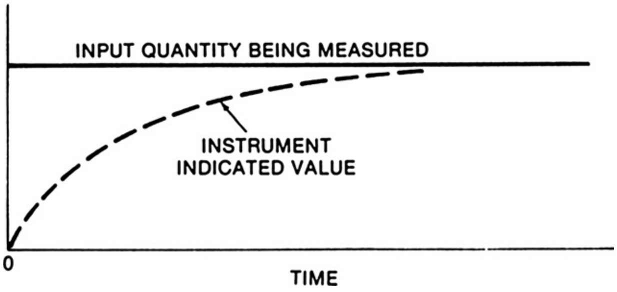


Figure 1.18:
Illustration of instrument dynamic response error.

rapidly changing input quantity varies inversely with its bandwidth as explained earlier in this chapter.

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 chapter 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.

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 Figure 1.1b. 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 Chapter 2. However, instrument systems often incorporate analog (continuous time) sensors. Consequently, for the remainder of this chapter, 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. An ideal analog sensor generates an output voltage that is proportional to the quantity q being measured:

$$\mathrm{v}_s = K_s q \tag{74}$$

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 described earlier in this chapter. Figure 1.19 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 ranges from 8 to 16, corresponding to a range of 256 to 65536 numerical values. Digital instruments belong to the class of discrete time systems, which are discussed in detail in the next chapter.

Of course, a sensor is susceptible to error just as is 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. Figure 1.20 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). Figure 1.20a depicts a square wave input to the sensor. Figure 1.20b illustrates the response that the sensor will have if its bandwidth is too small. Note that the output does

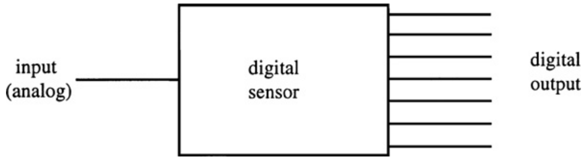


Figure 1.19:
Analog input, digital output sensor.

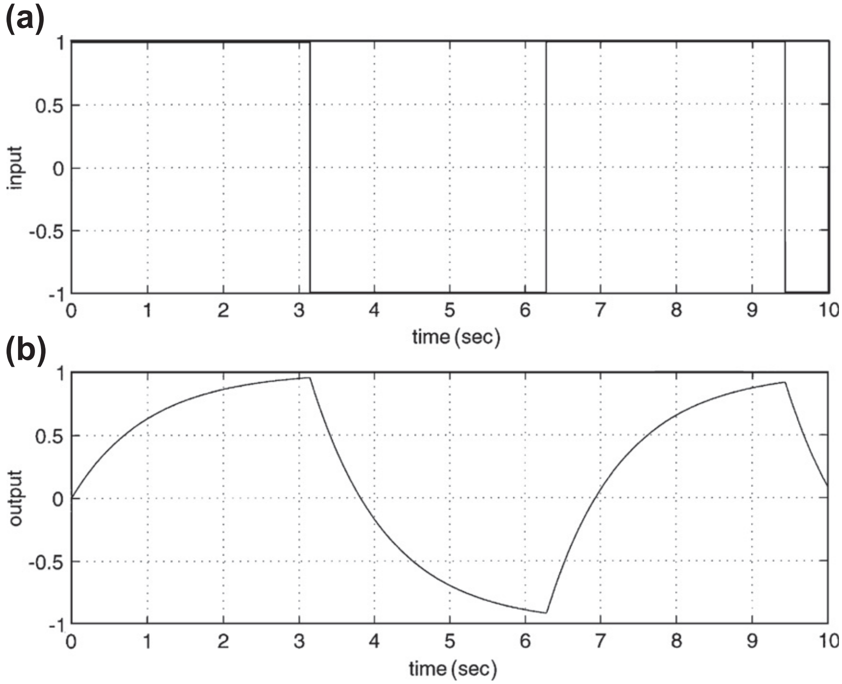


Figure 1.20:
Instrument square wave dynamic response error.

not respond to the instantaneous input changes. Rather, its output changes gradually, slowly approaching the correct value.

An ideal sensor has a *linear transfer characteristic* (or transfer function), as shown in Figure 1.21a. 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 Figure 1.21b. Sometimes, a nonlinear sensor may provide satisfactory operation without linearization if it is operated in a particular “nearly” linear region of its transfer characteristic (Figure 1.21b). Moreover, with digital signal processing, a simple

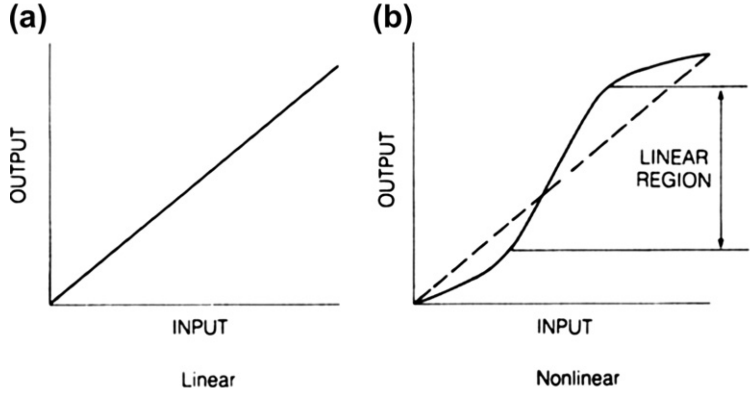


Figure 1.21:
Linearization of nonlinear sensor.

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 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 the next chapter.

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 pressure includes random error in the form of an electrical noise.

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

$$p(\mathrm{v}_n) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{\mathrm{v}_n}{\sigma}\right)^2} \tag{75}$$

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 vs.

frequency f for the noise source. The power spectral density can be determined from a sample $\mathrm{v}_T(t)$ of $\mathrm{v}_n(t)$ where

$$\begin{aligned} \mathrm{v}_T(t) &= \mathrm{v}_n(t) \quad 0 \le t \le T \\ &= 0 \qquad\quad \text{elsewhere} \end{aligned} \tag{76}$$

The spectrum of $V_T(t)$ is given formally by its Fourier transform $V_T(j\omega)$:

$$V_T(j\omega) = \int_{-\infty}^{\infty} e^{-j\omega t}\mathrm{v}_T(t)\mathrm{d}t \tag{77}$$

The power spectral density is given (with sufficient accuracy) by the following:

$$W(f) = \underset{T\to\infty}{Lim}\left(\frac{\left|V_T(j2\pi f)^2\right|}{T}\right) \tag{78}$$

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 Figure 1.22. 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 steady-state sinusoidal frequency response for the sensor which expresses its dynamic response to the noise random process. This frequency response can be found using the linear system modeling given earlier in this chapter. The "white noise source" is an ideal noise source having a constant power spectral density (W_w):

$$W_w(f) = W_o \qquad \forall f \tag{79}$$

The power spectral density of v_n (i.e., $W_n(f)$) is given by

$$W_n(f) = W_o\,|H_s(j\,2\pi f)|^2 \tag{80}$$

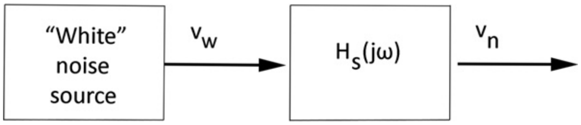


Figure 1.22:
Sensor noise model.

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

$$\begin{aligned} \tilde{v}_n &= \left[\int_o^\infty W_n(f)\mathrm{d}f \right]^{\frac{1}{2}} \\ &= \left[W_0 \int_o^\infty \left| H_s(j2\pi f)^2 \right| df \right]^{\frac{1}{2}} \end{aligned} \tag{81}$$

Thus, the noise amplitude of any sensor having a noise model depicted in Figure 1.22 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 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 chapter, 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 9. 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, typically analog or digital meters, 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

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. The trend in automotive electronic systems toward fully digital instrumentation means that the majority of automotive 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 rarely used 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 the next chapter.

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., 0 dB) characteristics and a stopband is a range of frequencies over which the attenuation is relatively large (e.g., many 10s of dB).

The filter itself is characterized by its complex frequency transfer function ($H(s)$) or its steady-state sinusoidal 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

$$\left|H(j\omega_p)\right|^2 = \frac{1}{2}|H(j0|^2 \tag{82}$$

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 bandpass filter has a passband between a pair of corner frequencies (ω_{p1} ω_{p2}) and a pair of stopbands defined:

Passband	$\omega_{p1} \leq \omega \leq \omega_{p2}$
Stopbands	$\omega < \omega_{s1}$ and $\omega > \omega_{s2}$

A bandstop filter has a single stopband between frequencies ω_1 and ω_2 and two passbands outside this range:

Passbands	$\omega < \omega_{p1}$ $\omega > \omega_{p2}$
Stopband	$\omega_{s2} < \omega < \omega_{s2}$

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 as well as classified by the function $F(\Omega)$ from which they are derived. For example, the so-called Butterworth filter is derived from the function

$$F(\Omega) = \frac{1}{1 + \Omega^{2n}} \tag{83}$$

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:

$$F(\Omega) = |H(j\,\Omega)|^2 \tag{84}$$

$$= H(S)H(-S)\Big|_{S=j\frac{\omega}{\omega_c}} \tag{85}$$

where S is a normalized complex frequency. 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 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

$$H(S) = \frac{1}{\prod_{m=1}^{n} (S - P_m)} \tag{86}$$

For example, the third-order (i.e., $n = 3$) Butterworth low-pass filter prototype is

$$H(S) = \frac{1}{(S+1)(S^2+S+1)} \tag{87}$$

The transfer function is then un-normalized by replacing S with s/ω_c.

The magnitude and phase of a third-order Butterworth filter are given in Figure 1.23 plot for $H(s)$.

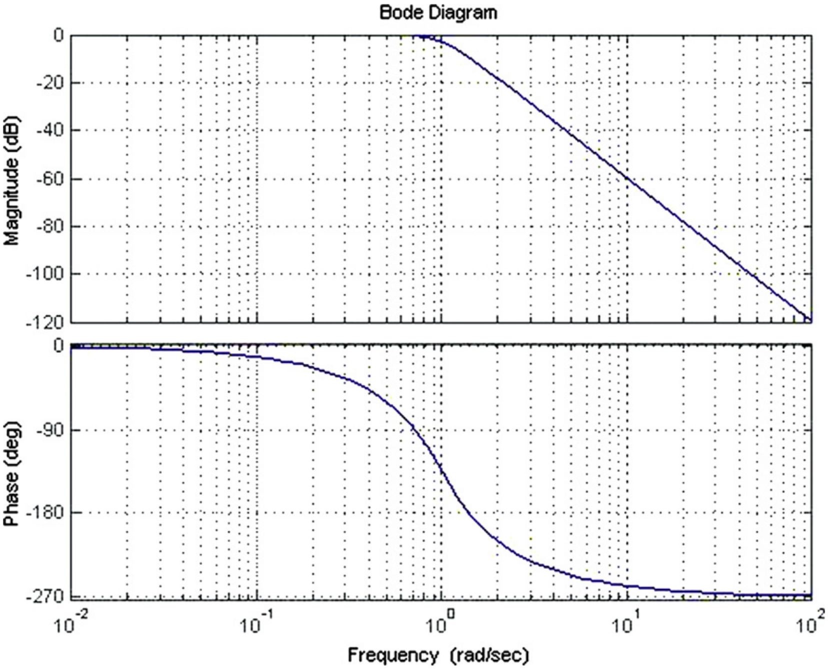


Figure 1.23:
Magnitude and phase frequency response of third-order Butterworth filter.

Note that 20 log $|H(j1)|^2$ which is equivalent to:

$$H(j\Omega)\big|_{\Omega=1} = \frac{1}{\sqrt{2}}$$

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:

$$F(\Omega^2) = \in^2 C_n^2(\Omega) \tag{88}$$

where

$$\begin{aligned} C_n(\Omega) &= \cos(n \cos^{-1}(\Omega)) \quad |\omega \leq 1| \\ &= \cosh(n \cosh^{-1}(\Omega)) \quad |\omega > 1| \end{aligned}$$

and $\in$ is a parameter determined by the passband "ripple." Cauer filter prototypes are derived from Jacobi elliptic 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.

It is clear from the above discussion of filter-design techniques that, in principle, it is possible to design a filter for 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.

This chapter has reviewed some basic principles of continuous time system theory that are applicable through the remainder 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. The next chapter reviews basic principles of such discrete time system modeling/analysis/design.

This page intentionally left blank

CHAPTER 2

Discrete Time Systems Theory

Chapter Outline

As explained in the previous chapter, 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

$$t_k = kT_s \quad k = 1, 2, 3\ldots$$

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 Figure 2.1

In this figure, the source has a continuous time electrical signal v(t) that could, for example, be a sensor output. The interface electronics-labeled A/D converter (which is modeled later in this chapter) generates a sequence of numerical values called samples at each discrete time or "sample period" t_k:

$$\mathrm{v}_k = \mathrm{v}(t_k)$$

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-08-097097-4.00002-3

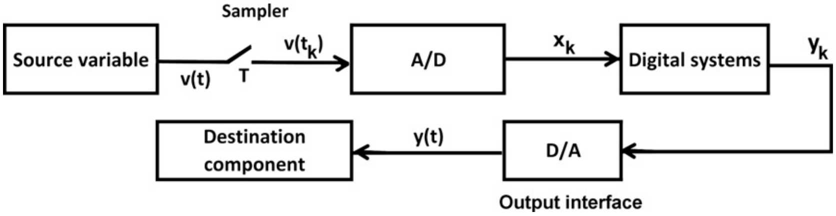


Figure 2.1:
Discrete time system configuration.

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 later chapters, which are concerned with specific automotive systems, we will deal with nonlinearities as required to explain the particular system.

As shown in the previous chapter, 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:

$$y_n = \sum_{k=0}^{K} a_k x_{n-k} - \sum_{\ell=1}^{L} b_\ell y_{n-\ell} \tag{1}$$

The dynamic response of such a system is determined by the coefficients a_k and b_ℓ.

Recall 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 **z**-transform $X(z)$ defined by

$$X(\mathrm{z}) = \mathscr{Z}(x_n) \tag{2}$$

$$X(z) = \sum_{n=-\infty}^{\infty} x_n z^{-n}$$

We illustrate with an example in which x_n is defined as

$$\begin{aligned} x_n &= c^n \quad n \geq 0 \\ &= 0 \quad\; n \leq 0 \end{aligned} \tag{3}$$

The **z**-transform is given by

$$\begin{aligned} X(z) &= \sum_{n=0}^{\infty} c^n z^{-n} \\ &= \sum_{n=0}^{\infty} (cz^{-1})^n \end{aligned} \tag{4}$$

Where $z =$ complex variable analogous to s for the Laplace transform. This latter sum is a geometric series that converges if $|cz^{-1}| \leq 1$ or $|z| \geq |c|$ to the closed form result

$$X(z) = \frac{1}{1 - cz^{-1}} \quad |z| \geq |c| \tag{5}$$

There are several important elementary properties of the **z**-transform that are important in the analysis of discrete time, digital systems that are summarized without proof below:

1. Linearity:

$$\mathscr{Z}[ax_n + by_n] = aX(z) + bY(z) \tag{6}$$

2. Time shift

$$\mathscr{Z}[x_{n+k}] = z^k X(z) \tag{7}$$

3. Convolution: let

$$W_n = \sum_{k=-\infty}^{\infty} x_k \, y_{n-k} = \sum_{n=-\infty}^{\infty} x_{n-k} \, y_k \tag{8}$$

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 **z**-transform denoted $\mathscr{Z}^{-1}[\mathrm{X}(z)]$ which is given by the following contour integral in the complex z-plane:

$$x_n = \frac{1}{2\pi j} \oint_C X(z) z^n \frac{\mathrm{d}z}{z} \tag{9}$$

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 **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 **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 **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:

$$\begin{aligned} Y(z) &= \sum_{j=1}^{n} \frac{a_j}{z - z_j} \\ &= \sum_{j=1}^{n} \frac{a_j z^{-1}}{1 - z_j z^{-1}} \end{aligned} \tag{10}$$

Each of these terms can be rewritten in the form of a Taylor series for each pole provided $|z| > |z_j|$:

$$\frac{a_j z^{-1}}{1 - z_j z^{-1}} = a_j \sum_{m=0}^{\infty} z_j^m z^{-(m+1)} \quad j = 1, 2 \cdots n \tag{11}$$

Replacing the summation index m with $k - 1$ and beginning the series sum with $k = 1$ yields an expression for each partial fraction of the same form as $Y(z)$. Combining terms of like power yields the following expression for $Y(z)$:

$$Y(z) = \sum_{k=1}^{\infty} \left[\sum_{j=1}^{n} \left(a_j z_j^{k-1} \right) \right] z^{-k} \tag{12}$$

Comparing like powers of z, the inverse z-transform of $Y(z)$ is the sequence $\{y_k\}$ where

$$y_k = \sum_{j=1}^{n} a_j z_j^{k-1} \tag{13}$$

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 Figure 2.1. Once again, assuming linear time invariance for this component, it has already been explained that its model is generally of the recursive form

$$y_n = \sum_{k=0}^{K} a_k x_{n-k} - \sum_{k=1}^{L} b_k y_{n-k} \tag{14}$$

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 **z**-transform of the above model:

$$Y(z) = \mathscr{Z}(y_n) = \sum_{n=-\infty}^{\infty} y_n z^{-n} \tag{15}$$

$$= \sum_{n=-\infty}^{\infty} \left[\sum_{k=0}^{K} a_k x_{n-k} - \sum_{k=1}^{L} b_k y_{n-k} \right] z^{-n} \tag{16}$$

Using the shift property it can be shown that

$$Y(z) = \left[\sum_{k=0}^{K} a_k z^{-k} \right] X(z) - \left[\sum_{k=1}^{L} b_k z^{-k} \right] Y(z) \tag{17}$$

which can be rewritten in the form

$$Y(z) = H(z)X(z) \tag{18}$$

The function $H(z)$ is the digital transfer function of the digital filter and is given by

$$H(z) = \frac{Y(z)}{X(z)} \tag{19}$$

$$= \frac{\sum\limits_{k=0}^{K} a_k z^{-k}}{1 + \sum\limits_{k=1}^{L} b_k z^{-k}} \tag{20}$$

The design procedures presented later in this chapter 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 **z**-transform $X(z) = 1$ and the output $Y(z) = H(z)$. The inverse **z**-transform of $H(z)$ is the sequence $\{h_n\}$, where components are given by

$$h_n = \frac{1}{2\pi j} \oint_C H(z) z^n \frac{\mathrm{d}z}{z} \tag{21}$$

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

$$\operatorname*{Lim}_{n \to \infty} h_n = 0,$$

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 (FIR) 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 (IIR).

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 **z**-transform of a sampled sinusoidal signal having frequency ω sampled at period T. The input sequence x_n is given by

$$\begin{aligned} x_n &= \sin(\Omega n) \quad t \geq 0 \\ &= 0 \qquad\qquad t < 0 \end{aligned}$$

where $\Omega = \omega T$. The sinusoid can be rewritten as

$$\sin(\Omega n) = [e^{j\Omega n} - e^{-j\Omega n}]/2j$$

for which the **z**-transform is given by

$$X(z) = \frac{1}{2j}\left[\sum_{n=0}^{\infty}(Ae^{j\Omega}\,z^{-1})^n - \sum_{n=0}^{\infty}(Ae^{-j\Omega}\,z^{-1})^n\right] \tag{22}$$

Provided $|z| \geq A$, both series converge yielding the following expression for $X(z)$:

$$\begin{aligned} X(z) &= \frac{A}{2j}\left[\frac{1}{(1 - Ae^{j\Omega}\,z^{-1})} - \frac{1}{(1 - Ae^{-j\Omega}\,z^{-1})}\right] \\ &= \frac{A\sin(\Omega)z^{-1}}{(1 - Ae^{j\Omega}\,z^{-1})(1 - Ae^{-j\Omega}\,z^{-1})} \end{aligned} \tag{23}$$

The filter output $Y(z)$ is given by

$$\begin{aligned} Y(z) &= H(z)\,X(z) \\ &= \frac{A\sin(\Omega)z^{-1}H(z)}{(1 - Ae^{j\Omega}\,z^{-1})(1 - Ae^{-j\Omega}\,z^{-1})} \end{aligned} \tag{24}$$

where $H(z) =$ transfer function for the digital filter. By partial fraction expansion, the filter output is given by

$$Y(z) = \frac{AH(e^{j\Omega})}{2j(1 - e^{j\Omega}\,z^{-1})} - \frac{AH(e^{-j\Omega})}{2j(1 - e^{-j\Omega}\,z^{-1})} + \sum_{k=1}^{K}\frac{\alpha_k z^{-1}}{(1 - \beta_k\,z^{-1})} \tag{25}$$

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 later chapters 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

$$Y_{ss}(z) = A\left\{\frac{z^{-1}\sin(\Omega)[H(e^{j\Omega}) + H(e^{-j\Omega})] - j[1 - z^{-1}\cos(\Omega)[H(e^{j\Omega}) - H(e^{-j\Omega})]]}{2[1 - 2\cos(\Omega)z^{-1} + z^{-2}]}\right\} \tag{26}$$

The inverse **z**-transform of $Y_{ss}(z)$ can be shown to be

$$\begin{aligned} y_n &= A\left\{\frac{[H(e^{j\Omega}) + H(e^{-j\Omega})]}{2}\sin(n\Omega) + \frac{[H(e^{j\Omega}) - H(e^{-j\Omega})]}{2j}\cos(n\Omega)\right\} \\ &= A\left|H(e^{j\Omega})\right|\sin[n\Omega + \phi(j\Omega)] \end{aligned} \tag{27}$$

where $\phi = \angle H(e^{j\Omega})$

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 $\angle H(e^{j\Omega})$. Thus, the behavior of $H(z)$ on the unit circle $z = e^{j\Omega}$ gives the frequency response characteristics for $-\pi \leq \Omega \leq \pi$, where Ω is 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 Figure 2.2, which depicts a subset of the components of Figure 2.1 focusing here on the components associated with digital filtering of an analog input $x(t)$ to yield an analog output.

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 (A/D) converter receives the output from the digital 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 later 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 (explained in detail later in this chapter). This device essentially clamps the

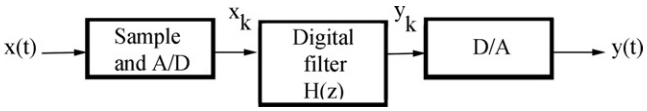


Figure 2.2:
Digital filtering of analog signal.

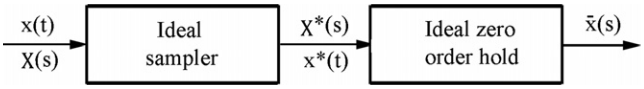


Figure 2.3:
Ideal sampler configuration.

output signal to the value of the latest sample. Although the actual sampled data system incorporates an A/D converter, for analysis purposes it is convenient to represent this system as depicted in Figure 2.3

The output of the sample and hold can be represented by the following model:

$$\begin{aligned}\overline{x}(t) = x(0)[u(t) - u(t-T)] + x(T)[u(t-T) - u(t-2T)] \\ +x(2T)[u(t-2T) - u(t-3T)]\cdots\end{aligned} \tag{28}$$

where $u(t)$ is a unit step. Taking the Laplace transform of $\overline{x}(t)$ yields $\overline{X}(s)$, which can be shown to be

$$\begin{aligned}\overline{X}(s) &= \frac{(1-e^{-Ts})}{s}\left[\sum_{n=0}^{\infty} x(nt)e^{-nTs}\right] \\ &= \frac{(1-e^{-Ts})}{s} X^*(s)\end{aligned} \tag{29}$$

The first factor is effectively the transfer function of the zero-order hold and the second $X^*(s)$, which is called the starred transform, is defined as

$$X^*(s) = \sum_{n=0}^{\infty} x(nt)e^{-nTs} \tag{30}$$

The starred transform of any variable is the **z**-transform with z replaced by e^{sT} as expressed below:

$$X^*(s) = X(z)\big|_{z=e^{sT}}$$

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

$$x^*(t) = x(0)\delta t + x(T)\delta(t-T) + \cdots x(nt)\delta(t-nT)\cdots \tag{31}$$

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 Figure 2.3.

The starred transform can also be rewritten in the form (see Reference 2.1)

$$X^*(s) = \frac{1}{T}\Big[X(s) + X(s + j\omega_s) + \cdots X(s + jn\omega_s)\cdots \\ + X(s - j\omega_s) + X(s - 2j\omega_s) + \cdots X(s - jn\omega_s) + \cdots + \frac{x(0)}{2}\Big] \tag{32}$$

where $\omega_s = 2\pi \big/ T$. 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

$$\begin{aligned} H(j\omega) &= 1 \quad \frac{\omega_s}{2} < \omega < \frac{\omega_s}{2} \\ 0 &= 1 \quad \text{elsewhere} \end{aligned}$$

provided 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 Eqn (20) based on an Butterworth analog prototype. It requires inputs n = filter order and cutoff frequency ω_n where $0 \le \omega_n \le 1$ and where $\omega_n = 1$ corresponds to $\frac{F_s}{2}$. The digital corner frequency is related to the analog corner frequency ω_c by the following relationship:

$$\Omega_c = \omega_c T \tag{33}$$

The sampling frequency must be selected such that the highest input frequency $\omega_{\max}$ satisfies

$$\omega_{\max} \le \frac{\pi}{T} \tag{34}$$

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 $F_s = 1 \big/ T$ 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 $-\pi \le \Omega \le \pi$. One such procedure utilizes a linear one-to-one mapping of normalized analog frequency band $0 \le \omega \le \infty$ into the digital frequency band $0 \le \Omega \le \pi$. This transformation is given by

$$\omega \tan\left(\frac{\Omega_c}{2}\right) = \tan\left(\frac{\Omega}{2}\right) \tag{35}$$

where the digital corner frequency is given by

$$\Omega_c = \frac{2\pi f_c}{F_s} \tag{36}$$

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 $\omega = 1$. 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 $s = j\omega$ and $z = e^{j\Omega}$ in the linear mapping transformation, which yields

$$s = j\cot\left(\frac{\Omega_c}{2}\right) \frac{\sin\left(\frac{\Omega}{2}\right)}{\cos\left(\frac{\Omega}{2}\right)} \Big|_{e^{j\Omega}=z} \tag{37}$$

$$= \cot\left(\frac{\omega_c}{2}\right) \left[\frac{e^{j\omega/2} - e^{-j\omega/2}}{e^{j\omega/2} + e^{j\omega/2}}\right] \Big|_{e^{j\Omega}=z} \tag{38}$$

$$= c\left(\frac{z-1}{z+1}\right) \tag{39}$$

where $$c = \cot\left(\frac{\Omega_c}{2}\right).$$

The digital transfer function $H(z)$ is given in terms of analog transfer function which is denoted $H^a(s)$:

$$H(z) = H^a(s)\Big|_{s=\frac{c(z-1)}{z+1}}$$

$$H(z) = H^a\left[\frac{c(z-1)}{z+1}\right] \tag{40}$$

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 Chapter 1)

$$H_a(s) = \frac{1}{(s+1)(s^2+s+1)} \tag{41}$$

It can be shown that the digital transfer function $H(z)$ is given by

$$H(z) = \frac{(z+1)^3}{K_1(z+z_1)(z^2+\alpha\, z+\beta)} \tag{42}$$

where

$$K_1 = (c+1)(c^2+c+1) \tag{43}$$

$$z_1 = \frac{(1-c)}{1+c} \tag{44}$$

$$\alpha = \frac{2(1-c^2)}{(c^2+c+1)} \tag{45}$$

$$\beta = \frac{(c^2-c+1)}{c^2+c+1} \tag{46}$$

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:

$$H(z) = \frac{1+3z^{-1}+3z^{-2}+z^{-3}}{K_1[1+(z_1+\alpha)z^{-1}+(\alpha z_1+\beta)z^{-2}+z_1\beta z^{-3}]} \tag{47}$$

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

$$H(z) = \frac{\sum\limits_{k=0}^{K} a_k z^{-k}}{1 + \sum\limits_{k=1}^{L} b_k z^{-k}} \tag{48}$$

where, for this example, $K = L = 3$. Thus, we can write

$$y_{n-k} = \{x_n + 3x_{n-1} + 3x_{n-2} + x_{n-3} - [(z_1 + \alpha)y_{n-1} + (\alpha z_1 + \beta)y_{n-2} + z_1 \beta y_{n-3}]\}/K_1 \tag{49}$$

As an example of this type of digital filter design, we present a digital version of the third-order Butterworth filter presented in Chapter 1. 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 is given by

$$\Omega_c = \frac{2\pi f_c}{F_s} = 1.2566 \tag{50}$$

The parameters of $H(z)$ are given by

$$\begin{aligned} c &= \cot\left(\frac{\Omega_c}{2}\right) = 1.3764 \\ K_1 &= (c+1)(c^2+c+1) = 10.149 \\ z_1 &= (1-c)/(c^2+c+1) = 0.1584 \\ \alpha &= 2(1-c^2)/(c^2+c+1) = -0.4189 \\ \beta &= (c^2-c+1)/(c^2+c+1) = 0.3554 \end{aligned} \tag{51}$$

The digital sinusoidal frequency response ($H\ (e^{j\Omega})$) for this example is given in Figure 2.4.

The magnitude squared of the response at the digital corner frequency is

$$\left|H(e^{j\Omega_c})\right|^2 = \frac{1}{2} \tag{52}$$

which corresponds to the response of the analog filter presented in Figure 1.21 for the normalized corner frequency of 1.

Discrete Time Control System

The previous section of this chapter, 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

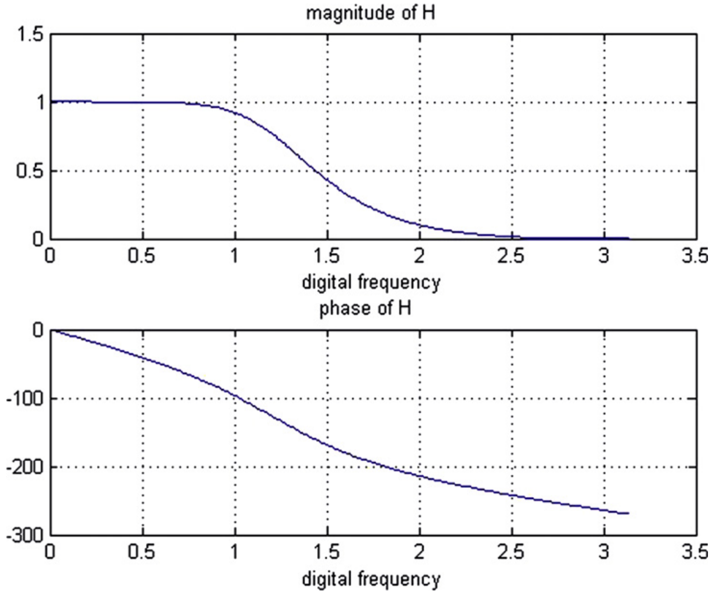


Figure 2.4:
Digital sinusoidal frequency response of third-order Butterworth filter.

instrumental 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 analytical purposes, this digital/analog conversion is taken to be a zero-order hold (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 3. The ZOH is actually an analog circuit that receives input pulses of amplitude $\overline{u}_n$. 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 $\overline{u}(t)$ is given by

$$\overline{u}(t) = \overline{u}_n \quad t_n \leq t < t_n + T \tag{53}$$

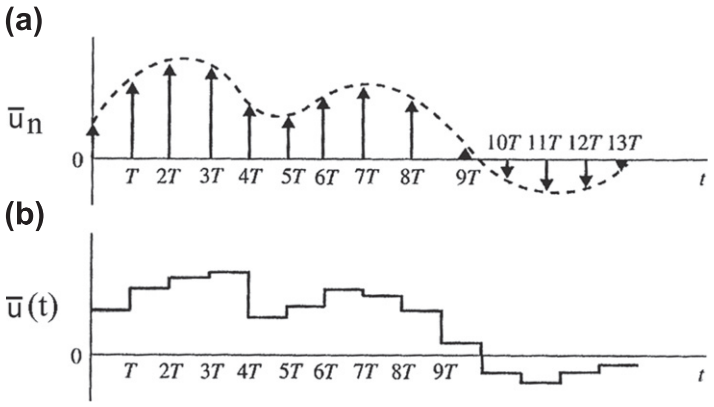
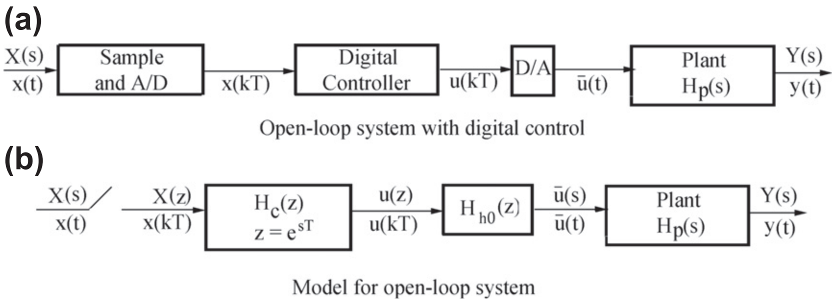


Figure 2.5:
Illustration of sample and hold signals.

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. Figure 2.5 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 as well as the precision of computation of the digital discrete time system (e.g., number of bits in the digital data) as explained in Chapter 4.

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 Figure 2.6



Open-loop system with digital control

Model for open-loop system

Figure 2.6:
Open-loop discrete time control system.

The model for the digital controller is

$$u(z) = H_c(z)X(z) \tag{54}$$

In terms of the starred transfer function this model, with the substitution $z = e^{st}$, can be written as

$$u^*(s) = H^*(s)X^*(s) \tag{55}$$

The A/D and D/A converters, controller and plant can be combined to yield the output $Y(s)$

$$Y(s) = H_p(s)\overline{u}(s) \tag{56}$$

$$= H_p(s)\left[\frac{1 - e^{-sT}}{s}\right]u^*(s) \tag{57}$$

The **z**-transform of the output $Y(z)$ is given by

$$Y(z) = \mathscr{Z}\left[\frac{(1 - e^{-sT})}{s}H_p(s)\right]u(z) \tag{58}$$

$$= \mathscr{Z}\left[\frac{(1 - e^{-sT})}{s}H_p(s)\right]H_c(z)X(z) \tag{59}$$

$$= (1 - z^{-1})\mathscr{Z}\left[\frac{(H_p(s))}{s}\right]H_c(z)X(z) \tag{60}$$

Eqn (60) is obtained from Eqn (59) from the time shift property of the **z**-transform. The **z**-transform for the combination zero-order hold and plant is known as the pulse transfer function and is normally found from tables. A sample of **z**-transform tables is given at the end of this chapter. The time domain system output at times t_k is found by computing the inverse **z**-transform of $Y(z)$:

$$y(kT) = \mathscr{Z}^{-1}[Y(z)] \tag{61}$$

As an example of the analysis of an open-loop system having a digital controller we consider a simple 1st-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:

$$u(kt) = K_p(kT) + K_D\left\{\frac{x(kT) - x[(k - 1)T]}{T}\right\} \tag{62}$$

For the purposes of illustrating this procedure, we make the numerical simplification $K_p = 1$ and $(\mathrm{K_D}/T) = 1$ yielding the following control algorithm:

$$u(kT) = 2x(kT) - x[(k-1)T] \tag{63}$$

The controller digital transfer function $H_c(z)$ can be found by taking the **z**-transform of the time domain model

$$\begin{aligned} H_c(z) &= 2 - z^{-1} \\ &= \frac{2z-1}{z} \end{aligned} \tag{64}$$

The **z**-transform of the combined zero-order hold/plant is given by

$$\mathscr{Z}\left[\frac{1-e^{-Ts}}{s(s+1)}\right] = \frac{1-e^{-T}}{z-e^{-T}} \tag{65}$$

as found from the transform tables (Table 2.1) given later in this chapter.

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 **z**-transform for this input $x(z)$ is given by

$$X(z) = \frac{z}{z-1} \tag{66}$$

The controller output $u(z)$ is given by

$$\begin{aligned} u(z) &= \left(\frac{2z-1}{z}\right)\left(\frac{z}{z-1}\right) \\ &= \frac{2z-1}{z-1} \end{aligned} \tag{67}$$

The output of this example $Y(z)$ is given by

$$\begin{aligned} Y(z) &= \left(\frac{2z-1}{z}\right)\left(\frac{1-e^{-T}}{z-e^{-T}}\right)\frac{z}{z-1} \\ &= \frac{(2z-1)(1-e^{-T})}{(z-e^{-T})(z-1)} \end{aligned} \tag{68}$$

It can be shown that the time domain system output (which is $z^{-1}[Y(\mathscr{Z})]$) is given by

$$y_k = [1 + (1 - 2e^{-T})e^{-(k-1)T}]u(k-1) \tag{69}$$

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 Figure 2.7 in which, for notational simplicity the zero-order hold, 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

$$E(s) = R(s) - Y(s) \tag{70}$$

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

$$E(s) = R(s) - H_s(s)Y(s) \tag{71}$$

Since the input to the combined plant is starred (i.e., sampled) the system output is given by

$$Y(s) = H(s)E^*(s) \tag{72}$$

The error is, then, given by

$$E^*(s) = R(s) - H(s)H_s(s)E^*(s) \tag{73}$$

The starred error $E^*(s)$ is given by

$$E^*(s) = \sum_{n=0}^{\infty} e(nT)e^{-nTs} \tag{74}$$

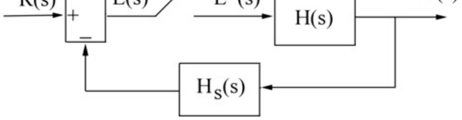


Figure 2.7:
Simplified block diagram of a closed-loop system (new figure).

Taking the starred transform of both sides of the equation yields

$$E^*(s) = R^*(s) - \overline{HH}_s^*(s)E^*(s) \tag{75}$$

where the bar over the product indicates that the product is taken before the transform. Solving the above equation for $E^*(s)$ yields

$$E^*(s) = \frac{R^*(s)}{1 + \overline{HH}_s^*(s)} \tag{76}$$

The **z**-transform for this expression is found by replacing e^{Ts} with z:

$$E(z) = \frac{R(z)}{1 + \overline{HH}_s(z)} \tag{77}$$

The system output is given by

$$\begin{aligned} Y(z) &= H(z)E(z) \\ &= \frac{H(z)R(z)}{1 + \overline{HH}_s(z)} \end{aligned} \tag{78}$$

The closed-loop transfer function $H_{c\ell}(z)$ is given by

$$\begin{aligned} H_{c\ell} &= \frac{Y(z)}{R(z)} \\ &= \frac{H(z)}{1 + \overline{HH}_s(z)} \end{aligned} \tag{79}$$

The time domain output at the sampling instants is given by the inverse **z**-transform of $Y(z)$

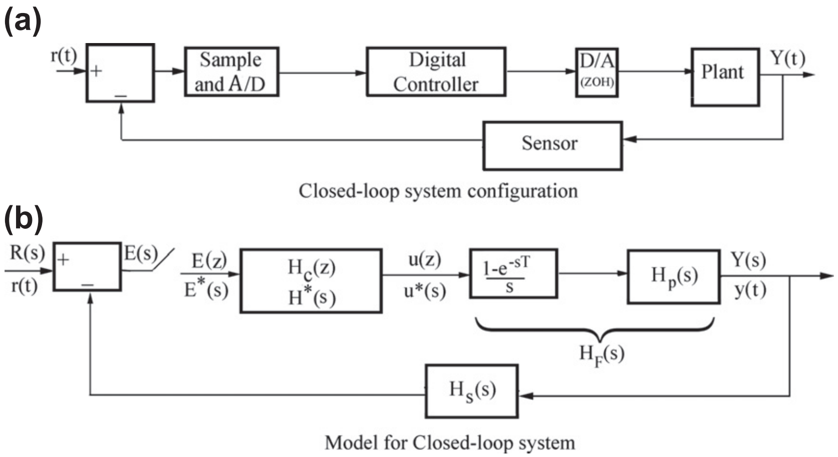
$$y(nT) = \mathscr{Z}^{-1}[Y(z)] \tag{80}$$

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

$$Y(s) = \frac{H(s)R^*(s)}{1 + \overline{HH}_s^*(s)} \tag{81}$$

However, in practice, the difficulties in analysis using this analog model generally lead to the digital transfer function approach.



Closed-loop system configuration

Model for Closed-loop system

Figure 2.8:
Discrete time closed-loop system.

We consider next the block diagram of a closed-loop system in which all of the major components are explicitly depicted and given in Figure 2.8.

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

$$\begin{aligned} E(s) &= R(s) - H(s)H_s(s)H_c^*(s)E^*(s) \\ Y(s) &= H(s)H_c^*(s)E^*(s) \end{aligned} \tag{82}$$

Finding the starred transform of both sides of the first equations yields

$$E^*(s) = R^*(s) - \overline{HH}_s^*(s)H_c(s)E^*(s) \tag{83}$$

Solving for $E^*(s)$ yields

$$E^*(s) = \frac{R^*(s)}{1 + H_c(s)\overline{HH}_s^*(s)} \tag{84}$$

From this expression, the z-transfer function can be found:

$$E(z) = \frac{R(z)}{1 + H_c(z)\overline{HH}_s(z)} \tag{85}$$

and the output is given by

$$Y(z) = \frac{H_c(z)H(z)R(z)}{1 + H_c(z)\overline{HH}_s(z)} \tag{86}$$

The closed-loop transfer function $H_{c\ell}(z)$ is given by

$$\begin{aligned} H_{c\ell}(z) &= \frac{Y(z)}{R(z)} \\ &= \frac{H_c(z)H(z)}{1 + H_c(z)\overline{HH_s}(z)} \end{aligned} \tag{87}$$

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 chapter.

It is 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 the previous chapter (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. Then, 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 ($\in_{\mathrm{n}}$) (see equation 92)

$$u_n = K_p \in_n + \frac{K_D}{T}[\in_n - \in_{n-1}] + K_I T \sum_{k=1}^{K} \frac{\in_{n-k} + \in_{n-(k+1)}}{2} \tag{88}$$

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 **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

$$u(z) = \left[K_p + \frac{K_D}{T}(1 - z^{-1}) + K_I T \sum_{k=1}^{k} \frac{z^{-k} + z^{-(k+1)}}{2} \right] E(z) \tag{89}$$

The control transfer function $H_c(z)$ is then given by

$$H_c(z) = \frac{u(z)}{E(z)} \tag{90}$$

where $E(z) = \mathscr{Z}\{\in_k\}$ and where

$$H_c(z) = K_p + \frac{K_D}{T}(1 - z^{-1}) + K_I T \sum_{k=1}^{L} \frac{z^{-k} + z^{-(k+1)}}{2} \tag{91}$$

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. Figure 2.9 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

$$\in(t) = x(t) - y(t) \tag{92}$$

where an ideal sensor (i.e., $H_s(s) = 1$) is assumed that provides the feedback signal. The sample and A/D convertor provide discrete time samples $\in_k$ of $\in$ at times $t_k = kT$:

$$\in_k = \in(t_k) \quad k = 1, 2, \ldots \tag{93}$$

where T is the sample period. The digital control generates control signal $\overline{u}_k$ in accordance with the desired control algorithm. It is assumed that the digital controller has a D/A converter such that $\overline{u}_k$ 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 $\overline{u}(t)$, 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 $H_p(s)$:

$$H_p = \frac{K_a}{s(s+p_1)} \tag{94}$$

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 discrete time system is given by

$$H_F = K_p\left(\frac{1 - e^{-Ts}}{s}\right)H_p(s) \tag{95}$$

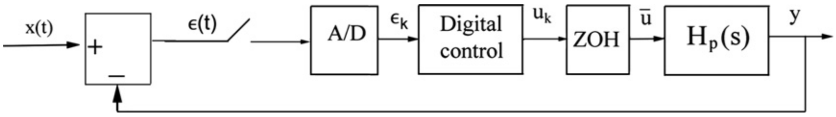


Figure 2.9:
Block diagram of example discrete time closed-loop control system (new figure).

The **z**-transform $H_F(z)$ is given by

$$H_F(z) = \mathscr{Z}\left\{K_p\left(\frac{1-e^{-Ts}}{s}\right)H_p(s)\right\} \tag{96}$$

$$= K_p(1-z^{-1})\mathscr{Z}\left[\frac{H_p(s)}{s}\right] \tag{97}$$

From the time shift property of the **z**-transform, $e^{-Ts} \rightarrow z^{-1}$.

In order to evaluate the **z**-transform of $H_p(s)/s$ (using transform tables), specific values are chosen for the following parameters:

$$K_a = 4,500$$

$$P_1 = 351.2$$

$$T = 0.001s$$

$$K_p = 14.5$$

The procedure for obtaining the **z**-transform of the function $H_p(s)/s$ is to represent it in a partial fraction expansion of the form

$$\frac{H_p(s)}{s} = \frac{\alpha_1}{s+p_1} + \frac{\alpha_2}{s+p_2} + \cdots \frac{\alpha_n}{s+p_n} \tag{98}$$

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

$$\frac{H_p(s)}{s} = \frac{A_r}{(s+s_r)^r} + \frac{A_{r-1}}{(s+s_r)^{r-1}} + \cdots \frac{A_1}{s+s_r} + \sum_{j=1}^{J}\frac{\alpha_j}{s+s_j} \tag{99}$$

where

$$A_r = \left[(s+s_r)^r\,\frac{H_p(s)}{s}\right]\Big|_{s=-s_r} \tag{100}$$

$$A_{r-k} = \left[\frac{1}{(r-k)!}\frac{\mathrm{d}^k}{\mathrm{d}s^k}\left(\frac{H_p(s)}{s}\right)\right]\Big|_{s=-s_r} \tag{101}$$

and where the poles s_j ($j=1,2\ \cdots J$) are simple poles.

The **z**-transform of each term can be found from the tables once the sample period T has been determined. For example, the **z**-transform of each simple pole is given by

$$\mathscr{Z}\left[\frac{\alpha_j}{s+s_j}\right] = \frac{\alpha_j z}{z - e^{-s_j T}} \tag{102}$$

Table 2.1 also gives the **z**-transform for terms associated with multiple poles.

The present example discrete time control system is of the form

$$\frac{H_p(s)}{s} = \frac{K_\alpha}{s^2(s+p_1)} \tag{103}$$

Table 2.1: 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	$\frac{Tz}{(z-1)^2}$
$\frac{1}{s^3}$	$\frac{t^2}{2}$	$\frac{T^2 z(z+1)}{2(z-1)^3}$
$\frac{1}{s^{n+1}}$	$\frac{t^n}{n!}$	$\lim_{\alpha \to 0} \frac{(-1)^n}{n!} \frac{\partial^n}{\partial \alpha^n}\left[\frac{z}{z-e^{-\alpha T}}\right]$
$\frac{1}{s+\alpha}$	e^{-at}	$\frac{z}{z-e^{\alpha T}}$
$\frac{1}{(s+\alpha)^2}$	te^{-at}	$\frac{Tze^{-\alpha T}}{(z-e^{-\alpha T})^2}$
$\frac{\alpha}{s(s+\alpha)}$	$1-e^{-at}$	$\frac{(1-e^{-\alpha T})z}{(z-1)(z-e^{-\alpha T})}$
$\frac{\omega}{s^2+\omega^2}$	$\sin \omega t$	$\frac{z\sin\omega T}{z^2 - 2z\cos\omega T + 1}$
$\frac{\omega}{(s+\alpha)^2+\omega^2}$	$e^{-\mathrm{at}} \sin \omega t$	$\frac{ze^{\alpha T}\sin\omega T}{z^2 - 2ze^{-\alpha T}\cos\omega T + e^{-2\alpha T}}$
$\frac{s}{s^2+\omega^2}$	$\cos \omega t$	$\frac{z(z-\cos\omega T)}{z^2 - 2z\cos\omega T + 1}$
$\frac{s+\alpha}{(s+a)^2+\omega^2}$	$e^{-\mathrm{at}} \cos \omega t$	$\frac{z^2 - ze^{-\alpha T}\cos\omega T)}{z^2 - 2ze^{-\alpha T}\cos\omega T + e^{-2\alpha T}}$

The partial fraction expansion of this function yields three terms given by

$$\frac{K_{\mathrm{a}}}{s^2(s+p_1)} = \frac{K_{\mathrm{a}}}{p_1 s^2} - \frac{K_{\mathrm{a}}}{p_1^2 s} + \frac{K_{\mathrm{a}}}{p_1^2(s+p_1)} \tag{104}$$

The **z**-transform for each of the terms above can be determined using Table 2.1 at the end of this chapter. Then the individual terms can be combined to yield the desired $H_F(z)$.

The **z**-transform of $H_F(z)$ is evaluated to be

$$H_F(s) = \frac{0.029z + 0.025z}{z^2 - 1.697z + 0.697} \tag{105}$$

The closed-loop transfer function $H_{c\ell}$ is given by

$$H_{c\ell}(z) = \frac{Y(z)}{X(z)} \tag{106}$$

$$H_{c\ell}(z) = \frac{H_F(z)}{1 + H_F(z)} \tag{107}$$

It is left as an exercise for the reader to show that $H_{c\ell}$ is given by

$$= \frac{0.029z + 0.0257}{z^2 - 1.668z + 0.7226} \tag{108}$$

The dynamic response of the closed-loop discrete time system is illustrated by its response to a unit step. The **z**-transform of the unit step $X(z)$ is given by

$$X(z) = \frac{z}{z-1} \tag{109}$$

The system output $Y(z)$ is given by

$$Y(z) = H_{c\ell}(z)X(z) \tag{110}$$

$$= \frac{z(0.029z + 0.0257)}{(z-1)(z^2 - 1.668z + 0.7226)} \tag{111}$$

The three poles of $Y(z)$ are given by

$$\begin{aligned} z_1 &= 1 \\ z_2 &= 0.8340 - 0.1645i \\ z_3 &= 0.8340 + 0.1645i \end{aligned}$$

The output sequence $\{y_k\}$ can be found using the procedures described earlier beginning with a partial fraction expansion of $Y(z)$:

$$Y(z) = \frac{\alpha_1}{z - z_1} + \frac{\alpha_2}{z - z_2} + \frac{\alpha_3}{z - z_3} \tag{112}$$

$$= \frac{\alpha_1 z^{-1}}{1 - z_1 z^{-1}} + \frac{\alpha_2 z^{-1}}{z - z_2 z^{-1}} + \frac{\alpha_3 z^{-1}}{z - z_3 z^{-1}} \tag{113}$$

where $\alpha_1 = 1.0018$, $\alpha_2 = -0.4864 + 0.2658i$, and $\alpha_3 = -0.4868 - 0.2658i$.

Each partial fraction is analytic outside the unit circle in the complex z-plane since all poles have magnitudes satisfying

$$\left|z_j\right| \leq 1 \quad j - 1, 2, 3\ldots$$

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}:

$$\frac{\alpha_j z^{-1}}{1 - z_j z^{-1}} = \alpha_j \sum_{m=0}^{\infty} z_j^m z^{-(m+1)} \tag{114}$$

By replacing $m+1 = k$ and summing over k beginning with $k = 1$, the partial fraction can be written in the form of a **z**-transform and the coefficients of z^{-k} become y_k. The output sequence terms are given by

$$y_k = \sum_{j=1}^{3} \alpha_j z_j^{k-1} j = 1, 2, 3\ldots \tag{115}$$

The output $y(t)$ is given only at the sample periods t_k:

$$y(t)\big|_{t=t_k} = y_k \tag{116}$$

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 Figure 2.10, which plots the output $y(t_k)$ for $T = 0.1$ s and for $T = 0.001$ s (i.e., 1 ms).

The output for the 1-ms 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.

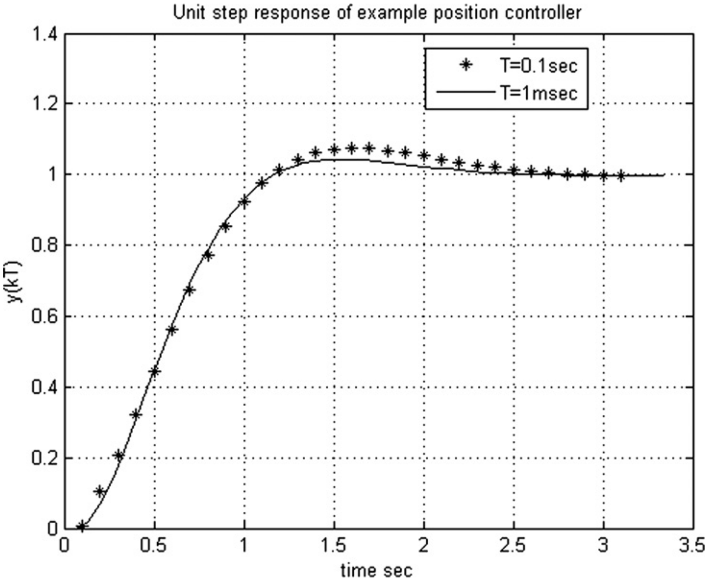


Figure 2.10:
Unit step response of example discrete time closed-loop control system.

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 **z**-transform of the discrete time forward path is normally best accomplished with respect to published tables of **z**-transforms (also available on-line). The closed-loop transfer function is found and the output $Y(z)$ for any given input $X(z)$ is given by

$$Y(z) = H_{c\ell}(z)X(z) \tag{117}$$

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 chapter 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 chapter apply to each of these exemplary systems.

CHAPTER 3

Electronics Fundamentals

Chapter Outline

This chapter is for the reader who has little 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.

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-08-097097-4.00003-5

Semiconductor Devices

All of the active circuit devices (e.g., diodes and transistors) from which electronic circuits are built are themselves 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 electrical 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 in 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 electrical 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. Although the details of electric field theory are beyond the scope of this book, roughly speaking, it 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 Figure 3.1, which depicts a relatively long, thin slab of semiconductor material across which a voltage is applied. In this

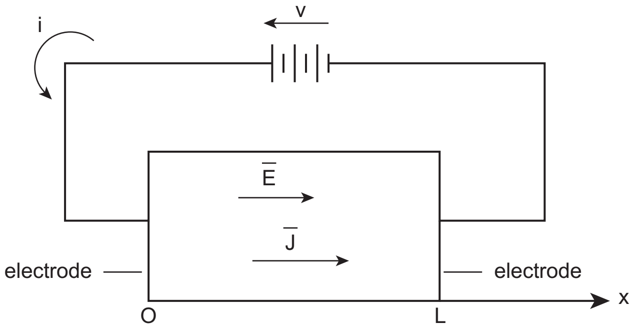


Figure 3.1:
Illustration of current conduction in semiconductor.

figure, the electric field intensity is a vector denoted $\overline{\mathbf{E}}$ which 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 $\overline{\mathbf{E}}$. 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 given approximately by

$$E = \frac{\mathrm{v}}{L}$$

Also shown in Figure 3.1 is the current density vector $\overline{\mathbf{J}}$, which is also an x-directed vector. The magnitude of the current density J is the current per unit cross-sectional area and is given by

$$J = \frac{i}{A_c} \tag{1}$$

where A_c is the cross-sectional area of the slab in the y, z plane. The current density vector is proportional to the electric field intensity

$$\overline{\mathbf{J}} = \sigma \overline{\mathbf{E}} \tag{2}$$

where σ is the conductivity of the material. The reciprocal of conductivity is known as the resistivity ρ of the material:

$$\rho = \frac{1}{\sigma} \tag{3}$$

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 outer most 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 which has an equal concentration of "free" electrons and holes which are identical, we denote this concentration $n_i = 1.5 \times 10^{10}/\mathrm{cm}^3$.

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:

$$np = n_{\mathrm{i}}^2 \tag{4}$$

There is another basic aspect of semiconductor physics which plays a role in the electrical characteristics of semiconductor electronic components. Whenever a voltage V is applied to a slab of semiconductor material it creates an electric field which is represented by the electric field intensity vector $\bar{\mathbf{E}}$ as described above (in this text the overbar 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 $\overline{\mathbf{v}}_e$ and $\overline{\mathbf{v}}_h$, respectively. These velocities are given by

$$\overline{\mathbf{v}}_e = \mu_e \overline{\mathbf{E}}$$
$$\overline{\mathbf{v}}_h = \mu_h \overline{\mathbf{E}}$$

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 $\overline{\mathbf{J}}_e$ and $\overline{\mathbf{J}}_h$, respectively:

$$\overline{\mathbf{J}}_e = nq\overline{\mathbf{v}}_e$$
$$\overline{\mathbf{J}}_h = pq\overline{\mathbf{v}}_h$$

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

A diode is a two-terminal electrical 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.

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.

The current, which flows through the diode in response to an applied voltage, depends upon the polarity of the voltage as well as its magnitude. Figure 3.2 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

$$I = I_s(\exp(V/nV_t) - 1) \tag{5}$$

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 Boltzman's constant, T is the junction absolute temperature, and q is the electron charge.

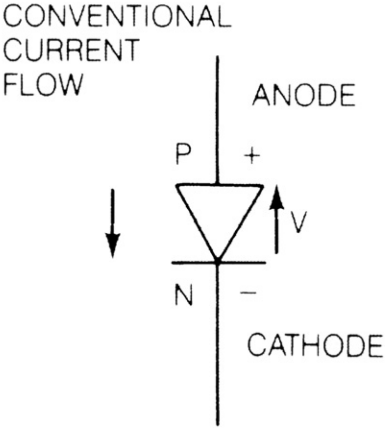


Figure 3.2:
Schematic symbol (use old 3-1a). Schematic symbol for p—n diode.

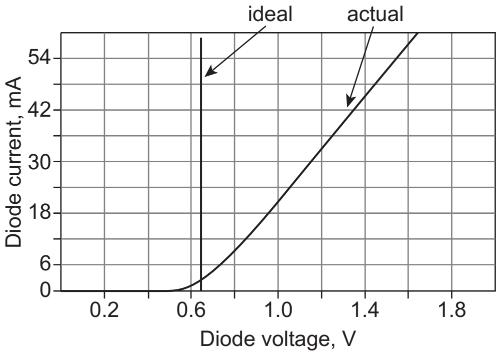


Figure 3.3:
Transfer characteristic. Diode transfer characteristics.

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. Figure 3.3 depicts this current flow vs. diode junction applied voltage V. The reverse-bias current is too small to be shown.

Although the model given above for diode voltage current characteristics is a very good representation for a practical diode (provided 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 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.

Rectifier Circuit

The circuit in Figure 3.4, 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 Figure 3.5. Notice that the output never drops below 0 V. The diode is reverse biased 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

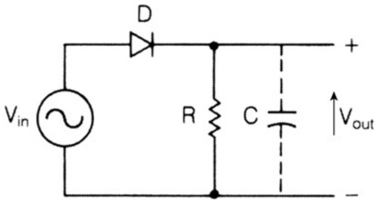


Figure 3.4:
Rectifier circuit.

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 electrical 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 Figure 3.4 is AC; the output voltage V_{out} has a DC component and a time-varying component, as shown in Figure 3.5.

The output voltage of the half-wave rectifier can be smoothed by adding a capacitor, which is represented by the dotted lines in Figure 3.4. The combination of the load resistance (R) and the capacitor (C) forms a low-pass filter, 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 dotted lines between the peaks of Figure 3.5. The amplitude of the ripples in the output voltage can be made insignificant by choosing a capacitor having sufficiently large capacitance, which lowers the low-pass filter corner frequency and attenuates the ripple components (see Chapter 1).

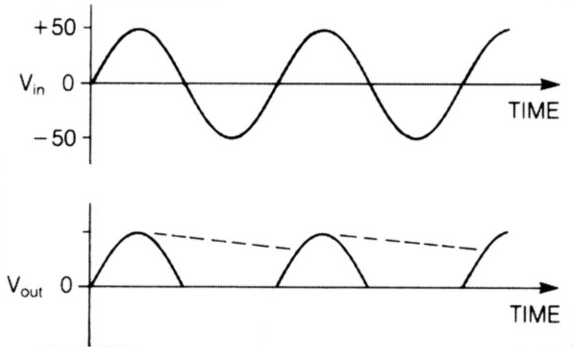


Figure 3.5:
Rectifier waveforms.

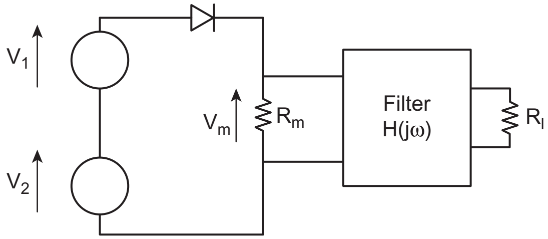


Figure 3.6:
Frequency "mixing" circuit.

There are many diode applications including some in communications 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. Figure 3.6 depicts a simplified embodiment of the frequency-mixing concept.

Let the two voltage sources have terminal voltages v_1, v_2: where $\mathrm{v}_1(t) = V_1\sin(\omega_1 t)$:

$$\mathrm{v}_2(t) = V_2\sin(\omega_2 t)$$

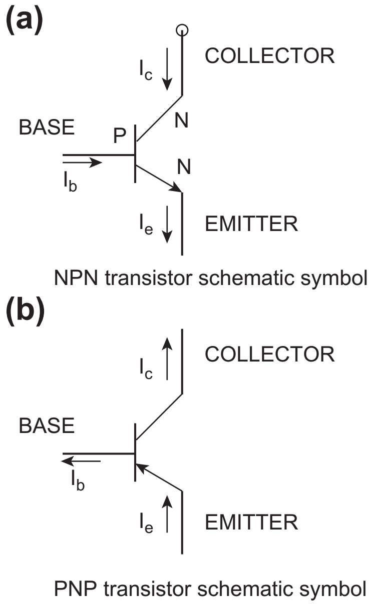
It can be shown that the voltage across R_m is given by

$$\mathrm{v}_m = \sum_{n=-\infty}^{\infty}\sum_{m=-\infty}^{\infty} C_{m,n}(V_1, V_2)\sin[(m\omega_1 + n\omega_2)t] \qquad (6)$$

where the coefficients $C_{m,n}$ are functions of V_1,V_2 as well as 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 Chapter 1).

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 1) NPN and 2) PNP.



NPN transistor schematic symbol

PNP transistor schematic symbol

Figure 3.7:
Transistor schematic symbols.

Physically, an NPN transistor structure consists of a thin p-type material (called BASE) sandwiched between 2 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.

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 Figure 3.7a 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. Figure 3.8 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 signal is taken at the collector—R_ℓ junction and is denoted 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

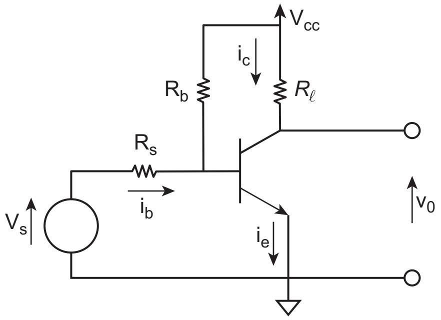


Figure 3.8:
NPN transistor amplifier circuit.

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} to 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 Figure 3.9. 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 n and the neutral portion of the base is denoted p. The reverse-bias collector–base voltage is denoted V_{cb} and is given by

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

An increase in V_{ce} (without changing V_{be}) increases the reverse bias which increases the depletion region in both the collector and base. There is no change in the depletion region of

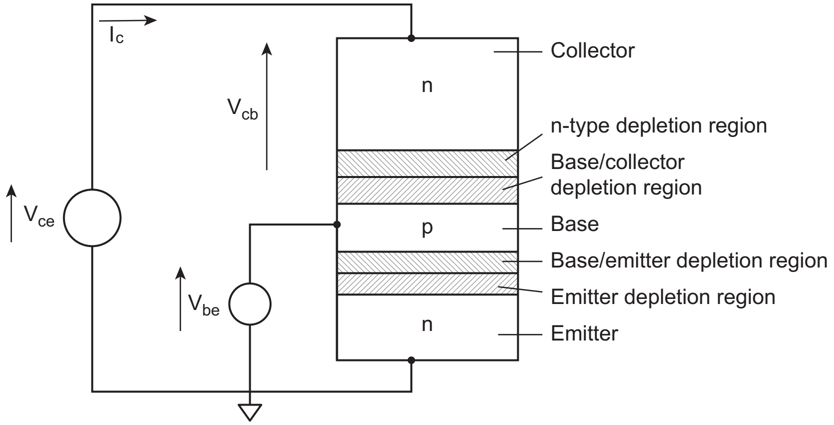


Figure 3.9:
NPN grounded emitter configuration and voltages.

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 $I_c(V_{ce},V_{be})$ known as characteristics curves. The characteristic curves for a typical small signal NPN transistor (i.e., 2N4401) in the grounded emitter configuration are given in Figure 3.10. These curves are parametrized 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:

$$I_c = I_s \exp\left(\frac{V_{be}}{V_T}\right)\left(1 + \frac{V_{ce}}{V_A}\right) \tag{7}$$

$$\beta_F = \frac{\partial I_c}{\partial I_b}\bigg|_{V_{ce}}$$

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

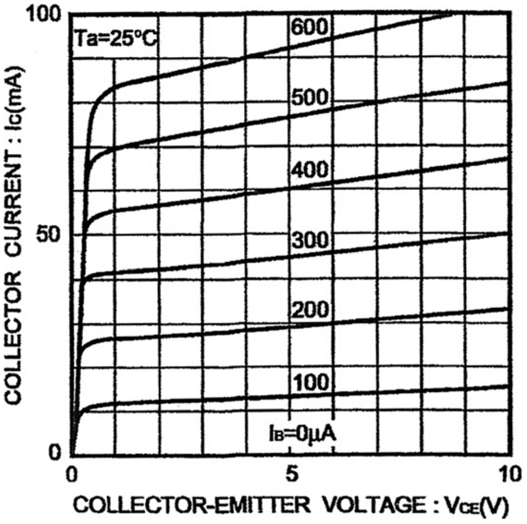


Figure 3.10:
Grounded emitter output characteristics.

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:

$$\beta_{F_0} = \frac{\partial I_c}{\partial I_b}\bigg|_{I_b} = 0$$

A small signal linear model for the transistor is given in Figure 3.11. 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 ($h_{\mathrm{fe}}I_b$) 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. Figure 3.10 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 $I_{\mathrm{o}} = 300$ μA at $V_{\mathrm{CE}} = 5\ V$ produces a variation in collector current that

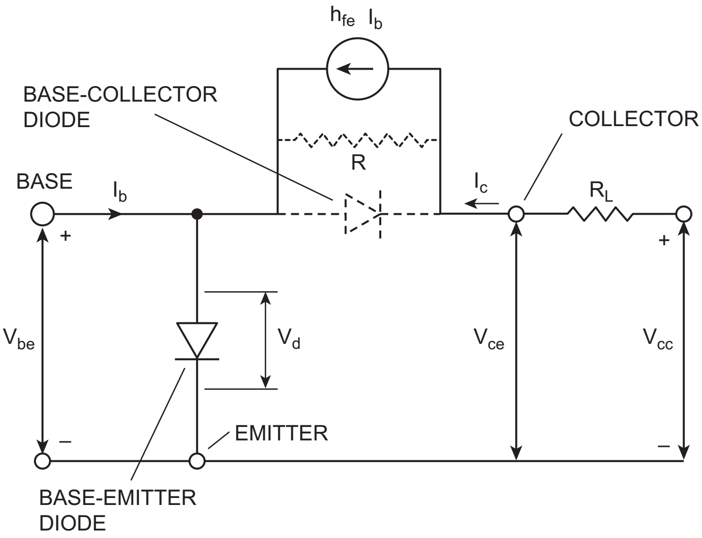


Figure 3.11:
Current and voltages for NPN transistor.

is highly linear. The so-called common emitter forward current gain, denoted h_{fe}, is given by

$$h_{\mathrm{fe}} = \frac{\partial I_c}{\partial I_b}\Big|_{V_{be}} \tag{8}$$

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

$$\begin{aligned} I_c(t) &= I_{co} + \delta I_c \\ &= I_{co} + i_c(t) \end{aligned} \tag{9}$$

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

$$\begin{aligned} I_c(t) &= I_o + \delta I_b \\ &= I_o + i_b(t) \end{aligned} \tag{10}$$

At any d-c base current (that is called the base bias current) and is denoted I_o in Figure 3.11 the collector a-c current i_c is given by

$$i_c(t) = h_{\mathrm{fe}} i_b(t) \tag{11}$$

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, I_c. The collector current is represented by a current generator in the collector circuit of the model in Figure 3.11. 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 Figure 3.8. 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 Figure 3.12a. In this figure, a signal 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 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 Figure 3.11b. The straight line (called the load line) connecting $V_{ce} = V_{cc}$ 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

$$\begin{aligned} S_{\mathrm{LL}} &= \frac{\mathrm{d}I_c}{\mathrm{d}V_{ce}} \\ &= \frac{I_{cs}}{V_{cc}} \\ &= \frac{1}{R_L} \end{aligned} \tag{12}$$

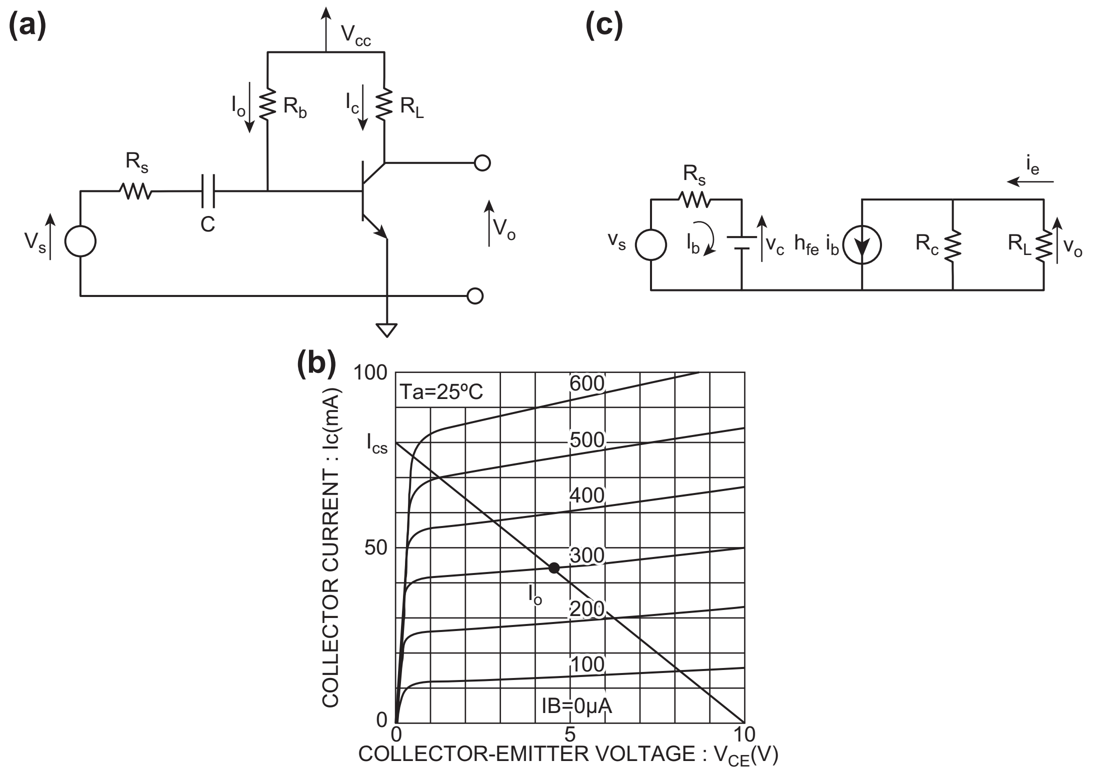


Figure 3.12:
Grounded emitter NPN transistor amplifier.

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

$$I_0 = \frac{V_{\mathrm{cc}} - V_{\mathrm{d}}}{R_{\mathrm{b}}} \cong \frac{V_{\mathrm{cc}}}{R_{\mathrm{b}}} \tag{13}$$

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 Figure 3.12c in which the load resistance at terminal V_{cc} is at a-c ground potential. The output voltage v_o of circuit in Figure 3.12a 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

$$i_c = h_{\mathrm{fe}} i_b \tag{14}$$

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

$$\mathrm{v_o}(t) = -R_{\mathrm{L}} i_{\mathrm{c}}(t) \tag{15}$$

$$\mathrm{v_o} = -h_{\mathrm{fe}} R_{\mathrm{L}} i_{\mathrm{b}}(t) \tag{16}$$

Note that the collector voltage and base current are 180° 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 Figure 3.12c. The base circuit analysis is conducted by summing the voltage components around the base circuit loop

$$\mathrm{v}_s = i_b R_s + \mathrm{v}_c \tag{17}$$

The capacitor voltage v_c is given by

$$\mathrm{v}_c(t) = \frac{1}{C}\int_o^t i(\tau)dt \tag{18}$$

Eqn (17) can be solved for base current i_b by using the Laplace transform method of Chapter 1 yielding the following Equation for $i_b(s)$:

$$\begin{aligned} i_b(s) &= \frac{\mathrm{v}_s(s)}{R_s + \frac{1}{sC}} \\ &= \frac{sC\mathrm{v}_s(s)}{1 + R_s Cs} \end{aligned} \tag{19}$$

Substituting $i_b(s)$ from Eqn (19) into Eqn (16) yields the transistor circuit voltage gain G:

$$G(s) = \frac{\mathrm{v}_o(s)}{\mathrm{v}_s(s)} \tag{20}$$

$$\begin{aligned} &= -\frac{h_{\mathrm{fe}} sCR_s R_L}{R_L(1 + sCR_s)} \\ &= -h_{\mathrm{fe}}\frac{R_L}{R_s}\left(\frac{s/\omega_o}{1 + s/\omega_o}\right) \end{aligned} \tag{21}$$

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 Chapter 1, 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

$$G(j\omega) = -h_{\text{fe}}\frac{R_L}{R_s}\left(\frac{j\omega/\omega_o}{1 + j\omega/\omega_o}\right) \tag{22}$$

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

$$G(j\omega) \underset{\omega \to \infty}{\longrightarrow} -h_{\text{fe}}\frac{R_L}{R_s} \tag{23}$$

that is, the circuit of Figure 3.12a 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 Figure 3.12a 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 Chapter 1.

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 $I_c = I_{c\text{sat}}$. 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 field-effect transistor consists of a slab of either n-type or p-type semiconductor to which electrodes are

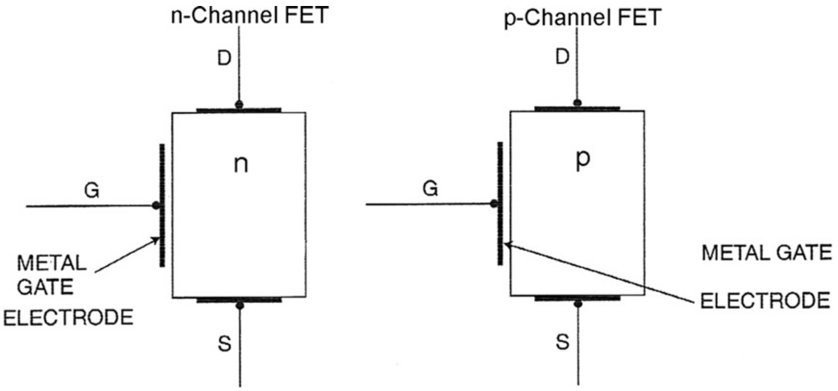


Figure 3.13:
Field-effect transistor configuration.

bonded as depicted in Figure 3.13. An FET is known as either an n-channel or a p-channel FET depending upon whether the semiconductor is n-type or p-type material, respectively.

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 S) and drain (denoted D). A third electrode, called the gate (denoted 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 circuit schematic symbols for n-type and p-type enhancement modes are shown in Figure 3.14.

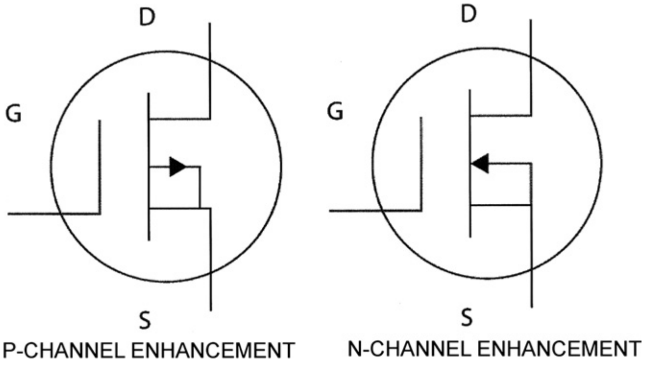


Figure 3.14:
Circuit symbol for FET transistors.

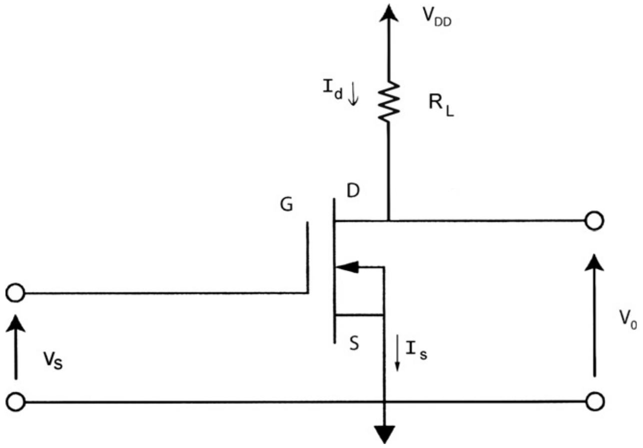


Figure 3.15:
Simple FET amplifier.

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. 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). Regardless of the fabrication technology or the type of semiconductor, FET functions as a voltage-controlled current source.

The analysis procedure for all FET-type transistors is essentially the same as for a bipolar transistor. An example-amplifying circuit, shown in Figure 3.15, depicts the current path from power supply V_{DD} 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. 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.

We illustrate such analysis with an example based upon Figure 3.12c. The small signal (a-c) equivalent circuit model for this circuit is shown in Figure 3.16.

The input impedance for an FET is sufficiently large that the gate voltage is approximately the source voltage $v_G \cong v_s$. The equivalent circuit of Figure 3.16 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. In this case, the voltage-controlled current source with current

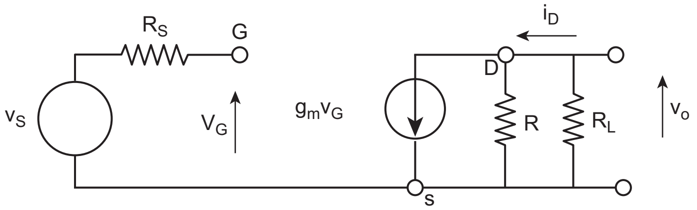


Figure 3.16:
Small signal circuit model for FET amplifier.

$$\begin{aligned} i_D &= g_m \mathrm{v}_G \\ &\cong g_m \mathrm{v}_s \end{aligned}$$

where

$$g_m = \left.\frac{\partial i_D}{\partial \mathrm{v}_G}\right|_{\mathrm{v}_{G\mathrm{bias}}}$$

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

$$\begin{aligned} \mathrm{v}_o &= -\frac{RR_L i_D}{R+R_L} \\ &= -\frac{RR_L g_m \mathrm{v}_s}{R+R_L} \end{aligned} \tag{24}$$

Note that the FET amplifier produces a 180° phase shift from v_s to v_o similar to the bipolar amplifier.

The above example illustrates the analysis procedures for any FET. 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 Chapter 1 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 Si chip). This is particularly true of digital circuits which 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 which 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 Figure 3.17a. 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.

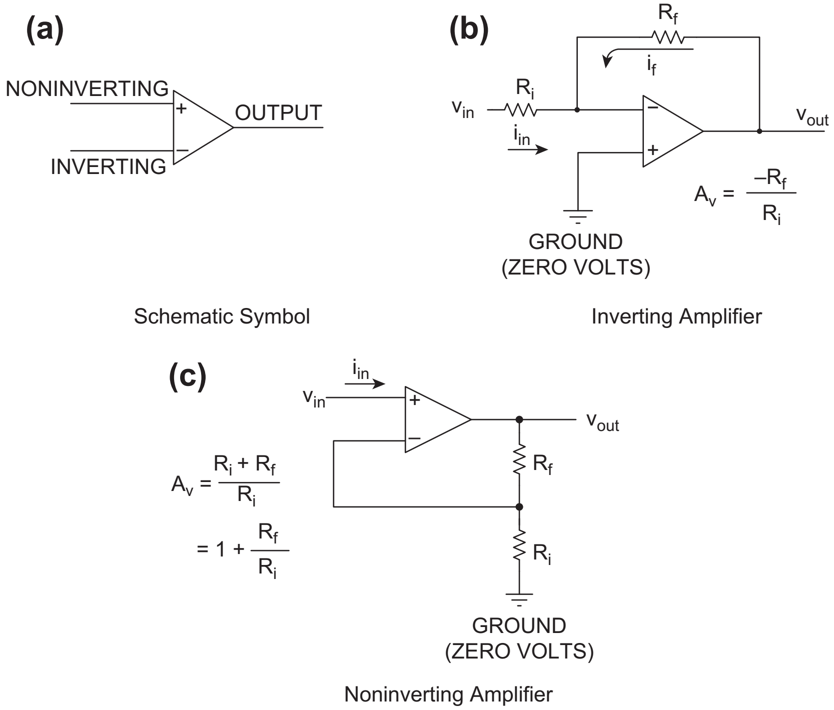


Figure 3.17:
Op amp circuits.

Use of Feedback in Op Amps

The op amp is normally not operated 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 Figure 3.17b. The output is connected to the inverting input through circuit elements (resistors, capacitors, etc.) which 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

$$v_{out} = A(v_1 - v_2) \tag{25}$$

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 $A \rightarrow \infty$):

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

Thus, the 2 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 non-inverting inputs is denoted R_{in} and is relatively large compared to external resistances used in normal up-amp applications. In the example of Figure 3.17b, 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, the noninverting for which we can write

$$v_1 = v_2 = 0$$

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

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

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

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

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

$$i_{\text{in}} + i_{\text{f}} = 0$$

$$\frac{\text{v}_{\text{out}}}{R_f} = -\frac{\text{v}_{\text{in}}}{R_i}$$

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

$$\begin{aligned} A_{c\ell} &= \frac{\text{v}_{\text{out}}}{\text{v}_{\text{in}}} \\ &= -\frac{R_f}{R_i} \end{aligned} \tag{26}$$

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

The op amp can readily be configured to implement a low-pass filter using the circuit depicted in Figure 3.18. 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, the currents into the inverting input sum to 0. Using the Laplace transform methods of Chapter 1 and the model for capacitor voltage/current relationships (i.e., $i_{\text{c}} = C\frac{\text{dv}_{\text{c}}}{\text{d}t}$), the model for the inverting mode currents is given below:

$$\frac{\text{v}_s(s)}{R_i} + \text{v}_o(s)\left[sC + \frac{1}{R_f}\right] = 0 \tag{27}$$

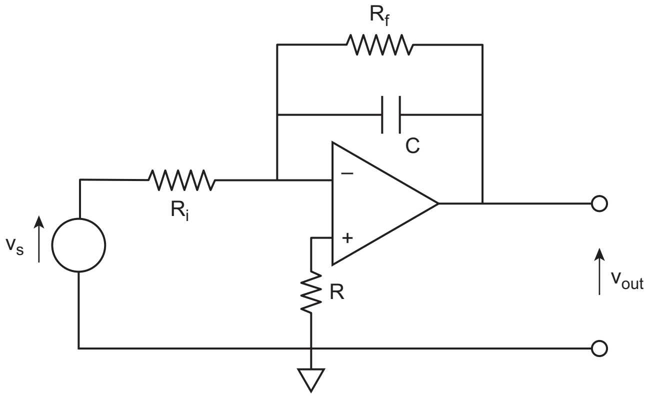


Figure 3.18:
Low-pass filter op amp circuit.

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

$$\begin{aligned} A_{c\ell}(s) &= \frac{v_o(s)}{v_s(s)} \\ &= -\frac{R_f/R_i}{1+sR_fC} \end{aligned} \tag{28}$$

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

$$A_{c\ell}(j\omega) = -\frac{R_f/R_i}{1+j\omega/\omega_c} \tag{29}$$

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 Figure 3.17b presented to input voltage v_{in} is the resistance of R_i.

A noninverting amplifier is also possible, as shown in Figure 3.17c. 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

$$A_v = \frac{v_{out}}{v_{in}} = 1 + \frac{R_f}{R_i} \tag{30}$$

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 Figure 3.17c 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. Figure 3.19 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

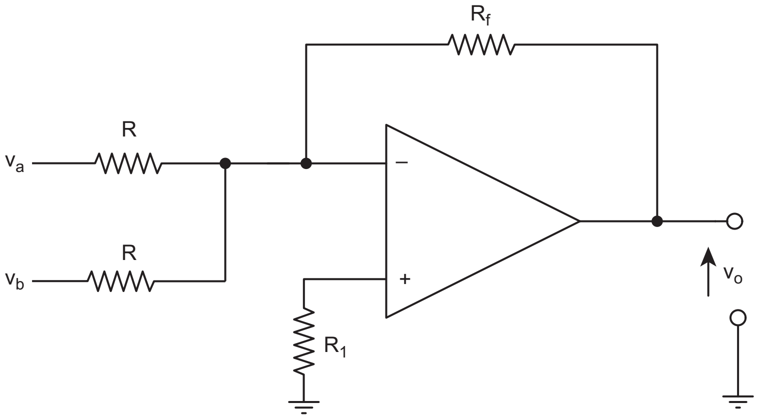


Figure 3.19:
Schematic drawing of a summing mode op amp circuit.

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:

$$v_0 = -\frac{R_f(v_a + v_b)}{R} \tag{31}$$

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 Figure 3.20.

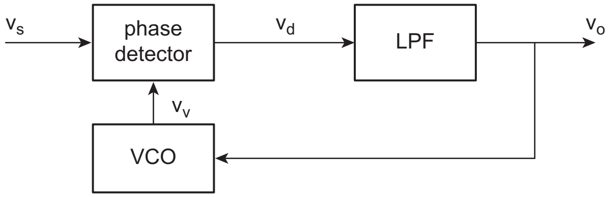


Figure 3.20:
Block diagram for PLL.

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

$$v_s(t) = V\cos(\omega_s t + \phi(t)) \tag{32}$$

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).

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

The other components in Figure 3.20 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

$$v_d = K_d v_s v_v \tag{33}$$

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

$$v_v(t) = V_v\cos(\phi_v(t)) \tag{34}$$

where

$$\phi(t) = \int_o^t \omega_v(\tau)d\tau \tag{35}$$

$$\omega_v(t) = \omega[v_o(0)] + K_v v_o(t) \tag{36}$$

where K_v is the constant for the VCO circuit.

The PLL circuit is an electronic closed-loop control system (see Chapter 1). 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

$$|\Omega_s - \omega_v(0)| \leq \Omega_c \tag{37}$$

$$\text{where } \Omega_c = \text{PLL capture range} \tag{38}$$

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

$$v_v(t) = V_v \cos[\omega_v t + \delta\phi] \tag{39}$$

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:

$$\delta\phi(t) = K_\phi \delta\omega_s \tag{40}$$

where K_ϕ is a constant for the VCO.

The phase detector output voltage is given by

$$v_d = K_d V_s V_v \cos(\omega_s t)\cos(\omega_s t + \delta\phi) \tag{41}$$

$$= \frac{K_d V_s V_v}{2}[\sin(2\omega_s t + \phi) + \sin(\delta\phi)] \tag{42}$$

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

$$v_o(t) = \frac{K_d V_s V_v K_\phi}{2}\delta\omega_s(t) \tag{43}$$

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 2 introduced the concept of ideal sample and zero-order hold circuit, which is 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.

Recall from Chapter 2 that 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:

$$x_k = \mathrm{v}(t_k, \mathrm{NB}) \tag{44}$$

where NB signifies an N-bit binary representation of $V(t_k)$ (i.e., see Chapter 4). 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).

Figure 3.21 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 4. This figure depicts the equivalent circuit of the source being sampled including its source impedance (assumed to be resistive) R_s as well as 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 Figure 3.21a, the switch S model is

$$\begin{array}{ll} \text{closed switch}\rightarrow S = 1 & t_k \le t \le t_k + \tau_s \\ \text{open switch}\rightarrow S = 0 & t_k + \tau_s < t < t_{k+1} \end{array} \tag{45}$$

The duration of the switch closure τ_s must be long enough for the A/D conversion to be complete at which time an output EOC = 1 (logical) indicating “end of conversion” to the

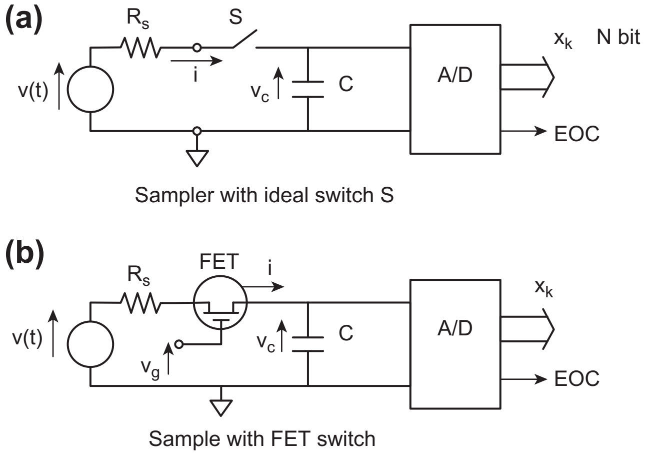


Figure 3.21:
Sample circuit.

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 $\mathrm{v}_c = \mathrm{v}(t_{k-1})$ to $\mathrm{v}_c = \mathrm{v}(t_k)$. The model for this circuit is given by

$$\begin{aligned} R_s i + \mathrm{v}_c &= \mathrm{v}(t_k) - \mathrm{v}_c(t_{k-1}) \quad t_k \le t \le t_k + \tau_s \\ i &= 0 \quad t_k + \tau_s < t \le t_{k+1} \end{aligned} \tag{46}$$

where $i = \dfrac{\mathrm{d}q}{\mathrm{d}t}$;

q is the charge on capacitor, and where $\mathrm{v}_c = \dfrac{q}{C}$; C is the capacitance of capacitor:

$$\mathrm{v}_{c(k-1)} = \mathrm{v}_c(t_{k-1} + \tau_s)$$

Eqn (24) can readily be rewritten in terms of v_c:

$$\begin{aligned} R_s C\dot{\mathrm{v}}_c + \mathrm{v}_c &= \mathrm{v}(t_k) - \mathrm{v}_{c(k-1)} && t_k \le t \le t_k + \tau_s \\ \mathrm{v}_c(t) &= \mathrm{v}_c(t_k + \tau_s) && t_k + \tau_s < t \le t_{k+1} \\ &= \mathrm{v}_{ck} \end{aligned} \tag{47}$$

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 Eqn (25) is readily obtained using the Laplace transform method given in Chapter 1:

$$\begin{aligned} \mathrm{v}_c(t) &= [\mathrm{v}(t_k) - \mathrm{v}_{c(k-1)}](1 - e{-}t/\tau) && t_k \le t \le t_k + \tau_s \\ &= \mathrm{v}_{ck} && t_k + \tau_s < t + t_{k+1} \end{aligned} \tag{48}$$

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 $\tau << \tau_s$. Furthermore, τ_s should be small compared to the time that the source changes such that

$$\mathrm{v}(t_k + \tau_s) \simeq \mathrm{v}(t_k)$$

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 was shown in Chapter 2 that the sample frequency $F_s = 1/T$ must be greater than twice the highest frequency in $\mathrm{v}(t)$ to avoid aliasing errors.

Furthermore, the sample duration must be larger than τ (i.e., $\tau_s >> \tau$) in order for the sampler to approximate the ideal sampler performance. This latter objective requires that the capacitance C satisfies the following inequality:

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

In the practical sample circuit of Figure 3.21b, the switch function is implemented via an FET whose source to drain resistance ($\mathrm{R_{SD}}$) 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

$$\begin{aligned} \mathrm{v}_g(t) &= V_g \qquad\qquad t_k \le t < t_k + \tau_s \\ &= 0 \qquad\qquad t_k \le \tau_s \le t < t_{k+1} \end{aligned} \tag{49}$$

With $\mathrm{v}_g(t)$ above applied to the FET gate, the source/drain resistance is given by

$$R_{\mathrm{SD}}(\mathrm{V_g}) = R_{\mathrm{on}} \quad \text{(transistor in saturation)}$$
$$R_{\mathrm{SD}}(0) = R_{\mathrm{off}} \quad \text{(transistor in cutoff)}$$

Ideally, these resistances should be

$$R_{\mathrm{on}} = 0$$
$$R_{\mathrm{off}} \to \infty$$

However, in practice, R_{off} is finite, but large, and R_{on} is small, but nonzero. Provided R_{off} is sufficiently large, the model for the circuit of Figure 3.21 is given by

$$\begin{aligned} (R_s + R_{\mathrm{on}})C\dot{\mathrm{v}}_c + \mathrm{v}_c &= \mathrm{v}(t_k) - \mathrm{v}_{c(k-1)} \qquad t_k \le t \le t_k + \tau_s \\ &\cong \mathrm{v}(t_k) \qquad\qquad t_k + \tau_s < t < t_{k+1} \end{aligned} \tag{50}$$

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

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

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

Zero-order hold circuit

In addition to the ideal sampler component introduced in Chapter 2, the ideal ZOH component was shown to be required whenever a digital system output must be converted to an analog electrical 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 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.

Figure 3.22 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 4. 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 $\bar{\mathbf{u}}_k$ corresponds to the numerical value of y_k. The EOC output triggers a pulse generator having output voltage $\mathrm{v}_g(t)$ given by

$$\begin{aligned} \mathrm{v}_g &= V_g \qquad && t_k \le t \le t_k + \tau_s \\ &= 0 && t_k + \tau_s < t < t_{k+1} \end{aligned} \tag{51}$$

The pulse duration τ_s must be sufficiently long that the capacitor voltage is approximately (ideally) $\bar{\mathbf{u}}_k$. 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

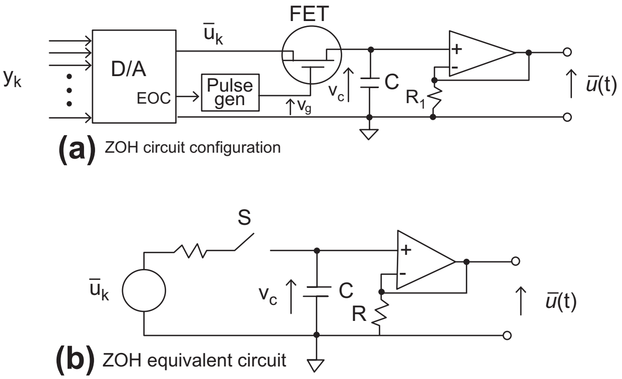


Figure 3.22:
ZOH circuit.

$$R_{\mathrm{SD}}(V_g) = R_{\mathrm{on}}$$
$$R_{\mathrm{SD}}(0) = R_{\mathrm{off}}$$

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

$$\begin{aligned}(R_{\mathrm{on}} + R_s)C\dot{\mathrm{v}}_c + \mathrm{v}_c &= \overline{u}_k - \mathrm{v}_{c(k-1)} \\ &= \overline{u}_{ck} \quad t_k + \tau_s < t < t_{k+1}\end{aligned} \tag{52}$$

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 Chapter 2. The primary differences between $\mathrm{v}_c(t)$ and $\overline{\mathbf{u}}(t)$ 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, $\overline{\mathbf{u}}(t)$ is a stepwise continuous function of time t as depicted in Chapter 2.

The circuit of 3.23 also incorporates an operational amplifier connected as a noninverting voltage follower having output voltage $\overline{\mathbf{u}}(t)$ where

$$\overline{\mathbf{u}}(t) = \mathrm{v}_c(t) \tag{53}$$

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 which would otherwise cause "loading" (with a drop in v_c from its desired value).

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 two different states. Each state is indicated by a particular voltage or current level. Binary circuits can operate in only one of 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 two modes of operation: on, conducting (at saturation), or off (in the cutoff state).

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.

Figure 3.23 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:

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

The operation of this type of transistor circuit can be illustrated assuming that it is a 2N4401 transistor having characteristic curves as depicted in Figure 3.12b.

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 $\mathrm{v}_s = V_L$, the base current and collector current are essentially zero. The output voltage v_o is given by

$$\mathrm{v}_o = V_{cc} - i_c R_L \simeq V_{cc} \tag{54}$$

It is assumed that the high voltage for this example is sufficient that the base current i_b is given by

$$i_b = \frac{V_H}{R_s} > 600\ \mu\text{A amp}$$

In this case, the output voltage is less than 0.5 V.

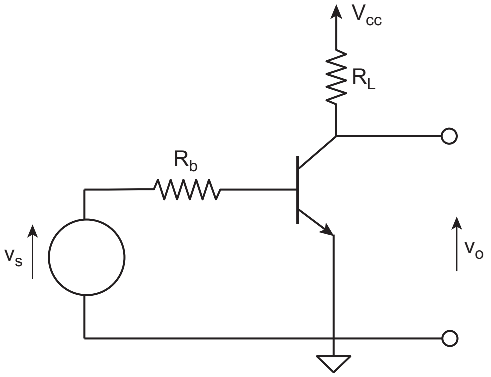


Figure 3.23:
NPN transistor digital circuit.

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 as well as the voltage conventions for representative 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. 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 as shown below (with A_M the most significant):

$$N_2 = A_M A_{M-1} \cdots A_M \cdots A_1 \tag{55}$$

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

$$N_{10} = \sum_{m=1}^{M} A_m 2^{m-1} \tag{56}$$

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 which uses octal digits such that $A_m = 0, 1, \cdots 6, 7 \cdots$ In such a system, which 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 is given for an M-digit octal number by

$$N_8 = \sum_{m=1}^{M} A_m 8^{m-1} \tag{57}$$

Logic Circuits (Combinatorial)

Digital computers can perform binary digit (bit) manipulations very easily by using three basic logic circuits, which perform arithmetic or logical operations on binary numbers or logical variables and are often euphemistically called "gates." We begin with three basic digital gates: the NOT, the AND, and the OR gate. Digital gates operate on logical variables that can have one of two possible values (e.g., true/false, saturation/cutoff, or 1/0). As was previously explained, numerical values are represented by combinations of 0 or 1 in a binary number system.

As mentioned earlier, digital circuits operate with transistors in one of 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 "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

$$V > V_H \tag{58}$$

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

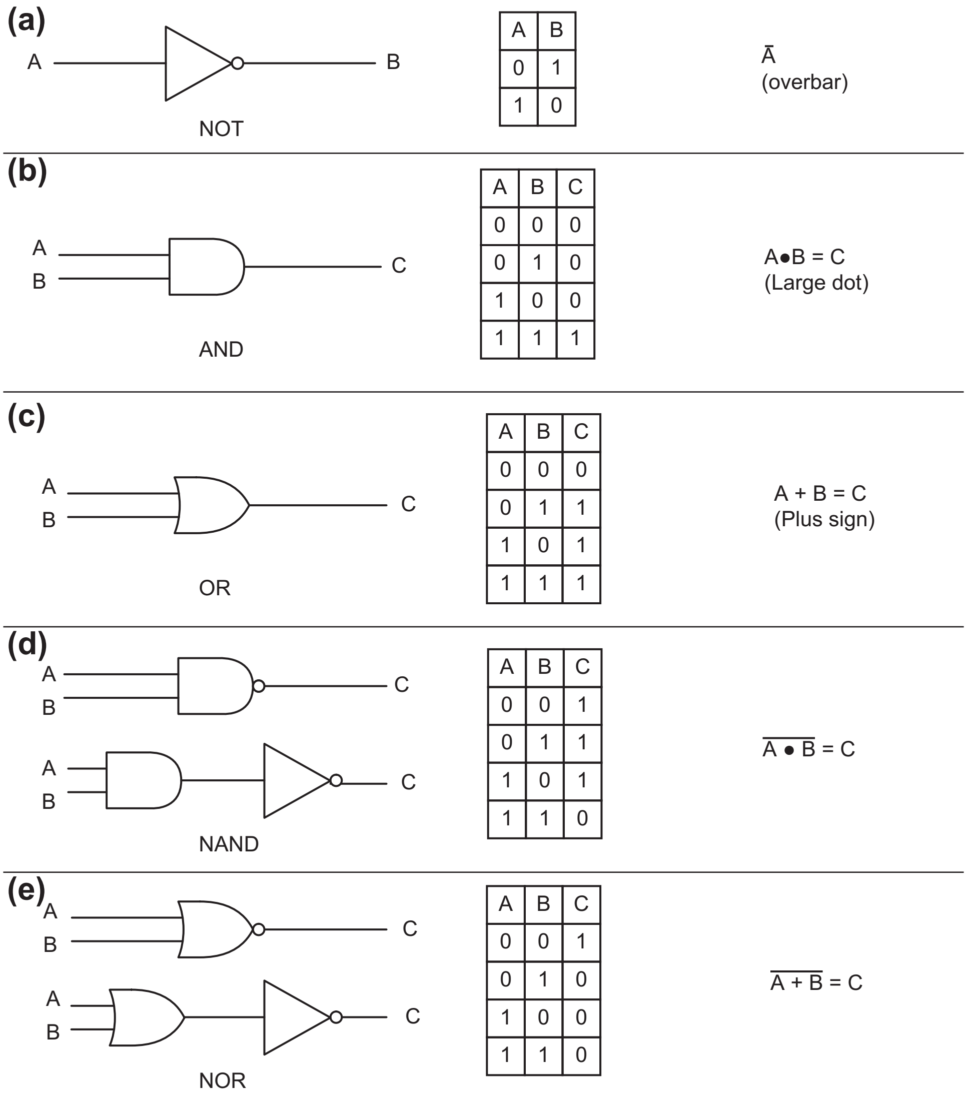
$$V < V_L \tag{59}$$

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 $V_H = 2.4$ V and $V_L = 0.8$ V for bipolar transistors.

Digital circuit operation is represented in terms of logical variables that are denoted here with uppercase letters. For example, in the next few sections A, B, and C represent logical variables that can have a value of either 0 or 1.

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 Figure 3.23 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. Figure 3.24a shows the schematic symbol for a NOT gate. Next to the schematic symbol is what is called a truth table. The truth table lists all of the possible combinations of input A and output B for the circuit. The logic symbol is shown also. The logic symbol is read as "NOT A." The bar over a logical variable indicates the logical inverse of the variable; that is, if $A = 1$, then $\overline{A} = 0$.



A	B
0	1
1	0

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

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

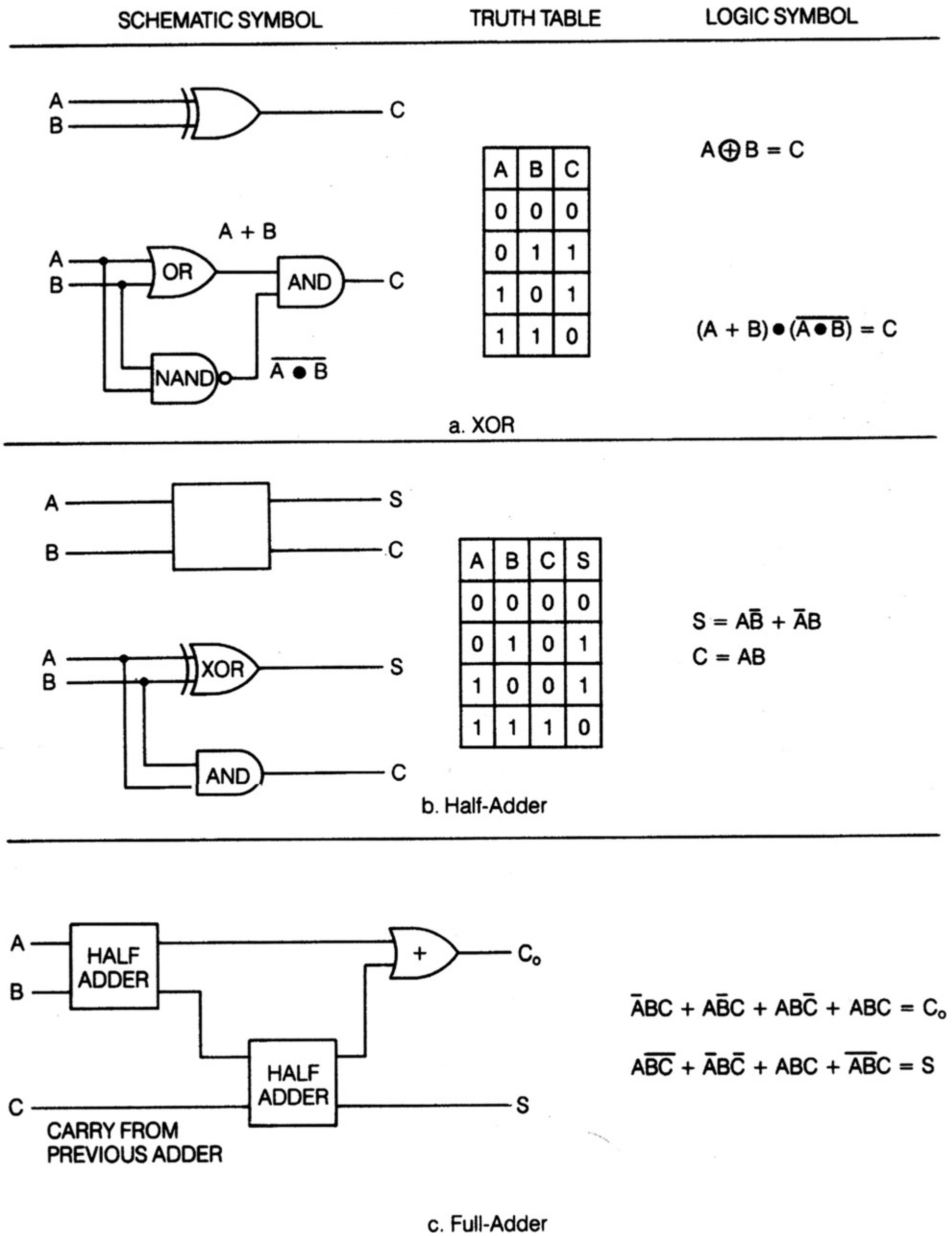
A	B	C
0	0	1
0	1	1
1	0	1
1	1	0

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

Figure 3.24:
Logic "gates."

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 Figure 3.24b 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). Figure 3.24b shows the truth



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

$A \oplus B = C$

$(A + B) \bullet (\overline{A \bullet B}) = C$

a. XOR

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

$S = A\bar{B} + \bar{A}B$

$C = AB$

b. Half-Adder

$\bar{A}BC + A\bar{B}C + AB\bar{C} + ABC = C_o$

$A\overline{BC} + \bar{A}B\bar{C} + ABC + \overline{AB}C = S$

c. Full-Adder

Figure 3.25:
Combination logic circuit.

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.

OR Gate

The OR gate, like the AND gate, has at least two inputs and one output. The one shown in Figure 3.24c 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). Figure 3.24c shows the schematic symbol, logic symbol, and truth table or the OR gate.

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. Figures 3.24d and 3.24e depict the schematic symbols for the NAND and NOR, respectively, along with truth tables and logical symbols.

Combination Logic Circuits

Still another important logical building block that can be build up with the three basic logic gates is the exclusive OR denoted XOR. This circuit has logical 1 output if one and only one of its inputs is nonzero. A two-input example is shown in Figure 3.25a. The schematic symbol for this device is depicted on the upper left of Figure 3.25a. Its implementation using the three basic gates is shown at the lower left. The XOR truth table and logic symbol are also given in Figure 3.25a.

All of these gates can be used to build digital circuits that perform all of the arithmetic functions of a calculator or computer. Table 3.1 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 Figure 3.25b. Note that it incorporates an XOR gate. It produces the sum and any necessary carry, as shown in the truth table.

Table 3.1: Addition of binary bits.

Bit *A*	0	0	1	1
Bit *B*	0	1	0	1
Sum	0	1	1	10

A half-adder circuit does not have an input to accept a carry from a previous place value. A circuit that does is called a full adder (Figure 3.25c). 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 4. Multiplication of two 1-bit numbers is characterized by elementary rules of multiplication:

$$\begin{gathered} 0 \times 0 = 0 \\ 0 \times 1 = 0 \\ 1 \times 0 = 0 \\ 1 \times 1 = 1 \end{gathered}$$

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 4.

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, 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 Figure 3.26. Here, the 4-bit numbers in place position notation are given by

$$\begin{aligned} A &= a_4 a_3 a_2 a_1 \\ B &= b_4 b_3 b_2 b_1 \end{aligned}$$

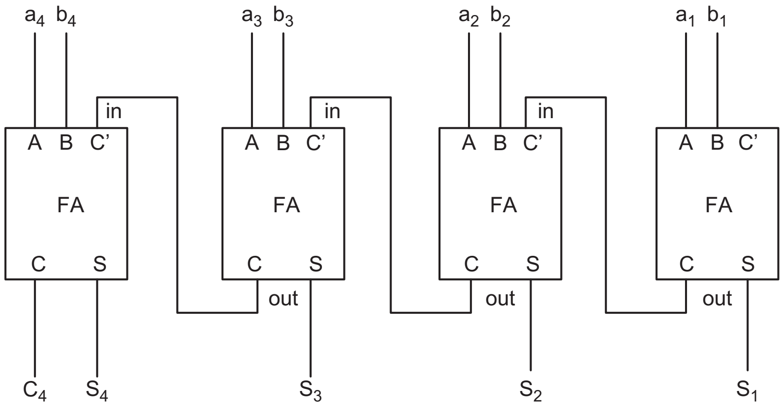


Figure 3.26:
4-bit-adder circuit.

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

$$S = C_4 S_4 S_3 S_2 S_1$$

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. 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 determine 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.

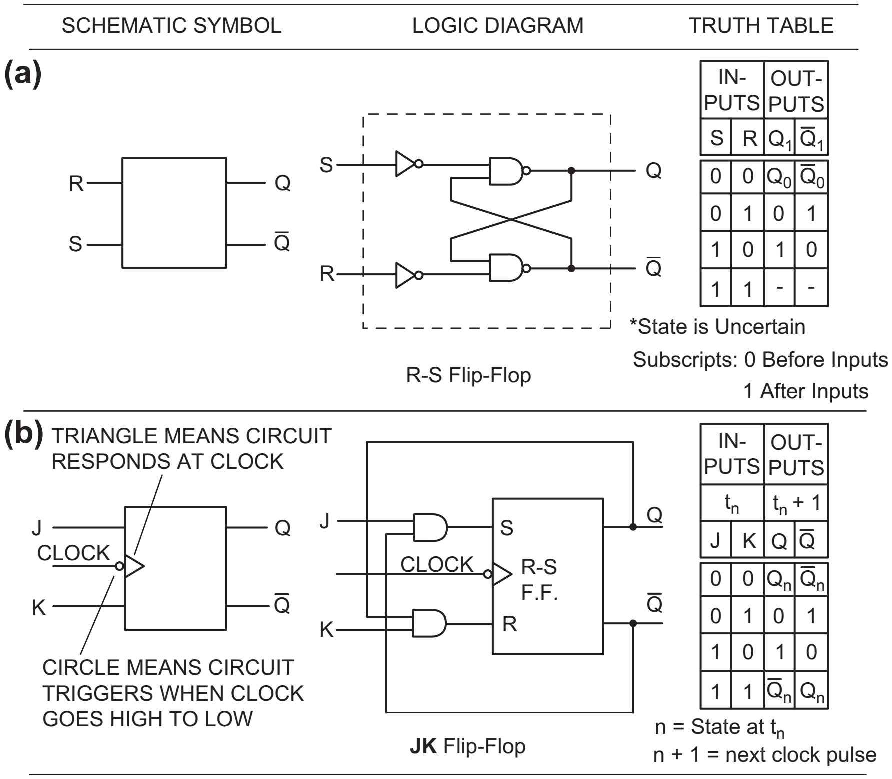
R—S Flip-Flop

A very simple memory circuit can be made by interconnecting two NAND gates, as in Figure 3.27a.

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 Figure 3.27b. It is called a J—K flip-flop and can be obtained from an R—S 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



IN-PUTS		OUT-PUTS	
S	R	Q_1	$\bar{Q}_1$
0	0	Q_0	$\bar{Q}_0$
0	1	0	1
1	0	1	0
1	1	-	-

IN-PUTS		OUT-PUTS	
t_n		$t_n + 1$	
J	K	Q	$\bar{Q}$
0	0	Q_n	$\bar{Q}_n$
0	1	0	1
1	0	1	0
1	1	$\bar{Q}_n$	Q_n

Figure 3.27:
Sequential logic circuits.

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 below, there are many uses at J–K-type circuits for their equivalent implementation in computers, which operate with a clock as explained later.

Synchronous Counter

Figure 3.28 shows a four-stage synchronous counter. 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. The waveforms of Figure 3.28b indicate how such circuitry can be used for counting, for generating other timing pulses, and for determining timed sequences.

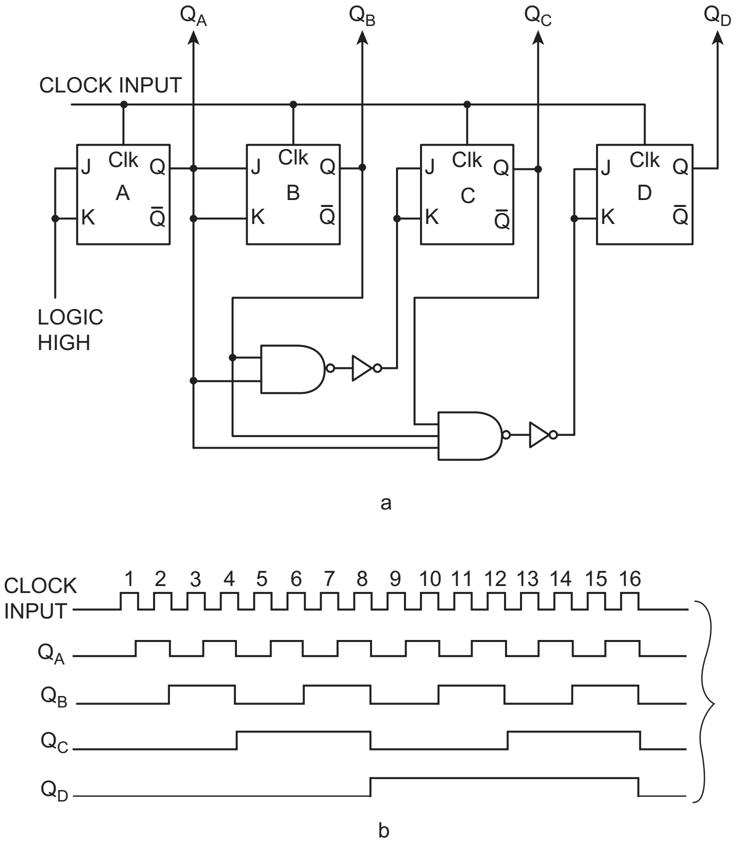


Figure 3.28:
Synchronous counter circuit.

Register Circuits

One of the most important circuits for building a computer is formed using multiple **JK**-type circuits as depicted in Figure 3.29. Such circuits are known as registers. Figure 3.29 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 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, for example, might be digital input data from a sensor and (A to D) converter), the output of a full adder, or the

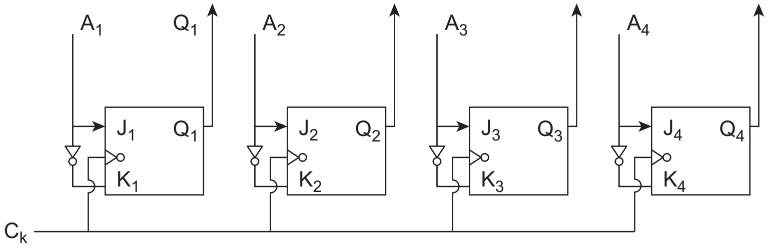


Figure 3.29:
Storage register circuit example.

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) is a so-called shift register. Figure 3.30a depicts a simple 4-bit shift register

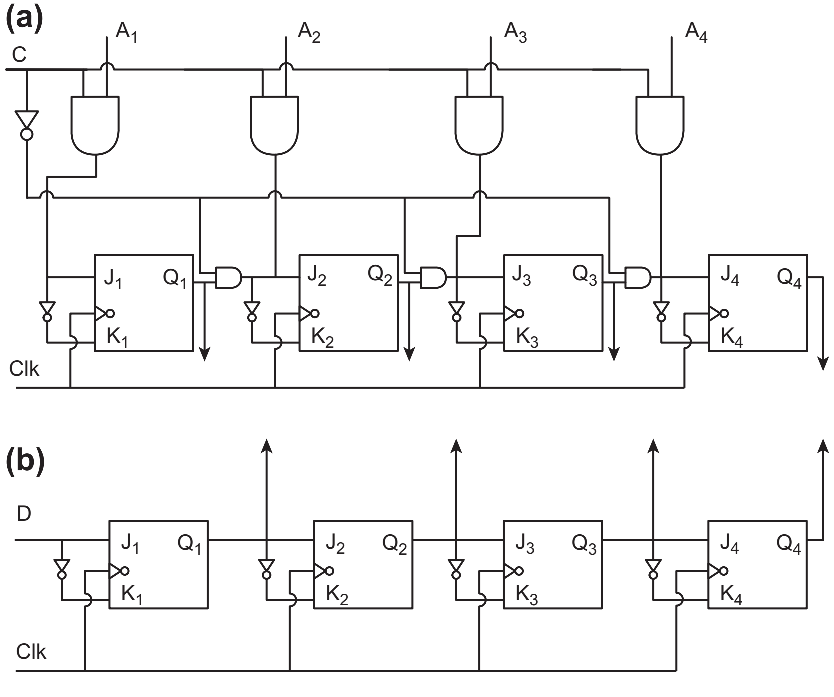


Figure 3.30:
4-bit shift register circuit example.

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,

$$N = A_M \cdots A_m \cdots A_1$$

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 Figure 3.30a, 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.

The circuit of Figure 3.30 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. 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 Figure 3.30b). Prior to clock time t_n, the previous data bit (i.e., at clock time t_{n-1}) is stored in Q_1 and that 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.

Integrated Circuits

One of the important consequences of integrated circuit (IC) technology progress has been 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 40 years. One of the important digital systems that is available as an IC is the arithmetic and logic unit (ALU).

Figure 3.31 is a sketch of a typical ALU showing the various connections. This 4-bit ALU has the capability of performing 16 possible logical or arithmetic operations on two 4-bit inputs, A and B. Table 3.2 summarizes these various operations using the logical notation explained earlier in this chapter. The ALU (implemented with more than 4-bits) is an important component of the most important single-chip IC for all digital electronic systems: the microprocessor.

The Microprocessor

Perhaps the single most important digital IC to evolve has been the microprocessor (MPU). This important device, incorporating hundreds of thousands of transistors in an area of about

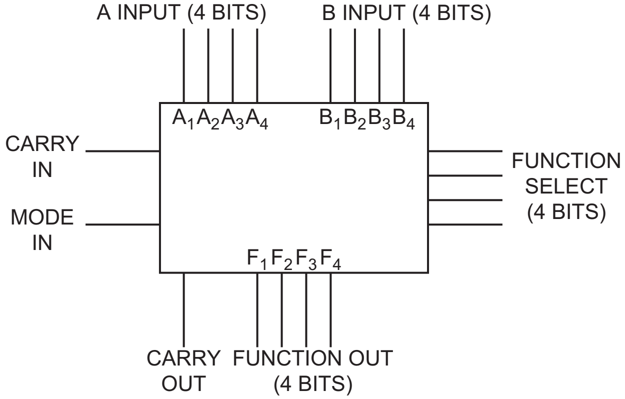


Figure 3.31:
ALU symbol.

Table 3.2: Arithmetic logic functions.

Select S Input	Logic Function, $M = 1$	Arithmetic Function, $M = 0$	
		$C_n = 1$	$C_n = 0$
0000	$F = \overline{A}$ (NOT)	$F = A$	$F = A$ Plus 1
0001	$F = \overline{A + B}$ (NOR)	$F = A + B$	$F = (A + B)$ Plus 1
0010	$F = \overline{A}B$	$F = A + \overline{B}$	$F = (A + \overline{B})$ Plus 1
0011	$F = 0$	$F =$ Minus 1 (2's Complement)	$F = 0$
0100	$F = \overline{AB}$ (NAND)	$F = A + A\overline{B}$	$F = (A + A\overline{B})$ Plus 1
0101	$F = \overline{B}$ (NOT)	$F = (A + B)$ Plus $A\overline{B}$	$F = (A = B)$ Plus $A\overline{B} + 1$
0110	$F = AB + B\overline{A}$ (Exclusive OR)	$F = (A - B)$ Minus 1	$F = A$ Minus B
0111	$F = A\overline{B}$	$F = A\,\overline{B}$ Minus 1	$F = \overline{AB}$
1000	$F = \overline{A} + B$ (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	$F = B$	$F = (A + \overline{B})$ Plus AB	$F = (A + \overline{B})$ Plus $A + 1$
1011	$F = AB$ (AND)	$F = AB$ Minus 1	$F = AB$
1100	$F = 1$	$F = A$ Plus A^*	$F = (A + A)$ Plus 1
1101	$F = A + \overline{B}$	$F = (A + B)$ Plus A	$F = (A + B)$ Plus $A + 1$
1110	$F = A + B$ (OR)	$F = (A + \overline{B})$ Plus A	$F = (A + \overline{B})$ Plus $A + 1$
1111	$F = A$	$F = A$ Minus 1	$F = A$

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

¼-inch square, has truly revolutionized digital electronic system development. A microprocessor is the operational core of a microcomputer and has broad applications in automotive electronic systems.

The MPU incorporates a relatively complicated combination of digital circuits including an ALU, registers, and decoding logic. A representative (though simplified) MPU block diagram is shown in Figure 3.32. The double lines labeled "bus" are actually sets of conductors for carrying digital data throughout the MPU. A block diagram of a microprocessor/microcontroller with more detail is given in Chapter 4. Common IC MPUs use 8, 16, or 32 (or higher multiples of 8) conductor buses.

Early twenty-first-century automobiles incorporate dozens of microprocessors that are applied for a variety of uses from advanced powertrain control to simple tasks such as automatic seat and side-view mirror positioning. A microprocessor by itself can accomplish nothing. It requires additional external digital circuitry as explained in the next chapter. One of the tasks performed by the external circuitry is to provide instructions in the form of digitally encoded electrical signals. By way of illustration, an 8-bit microprocessor 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

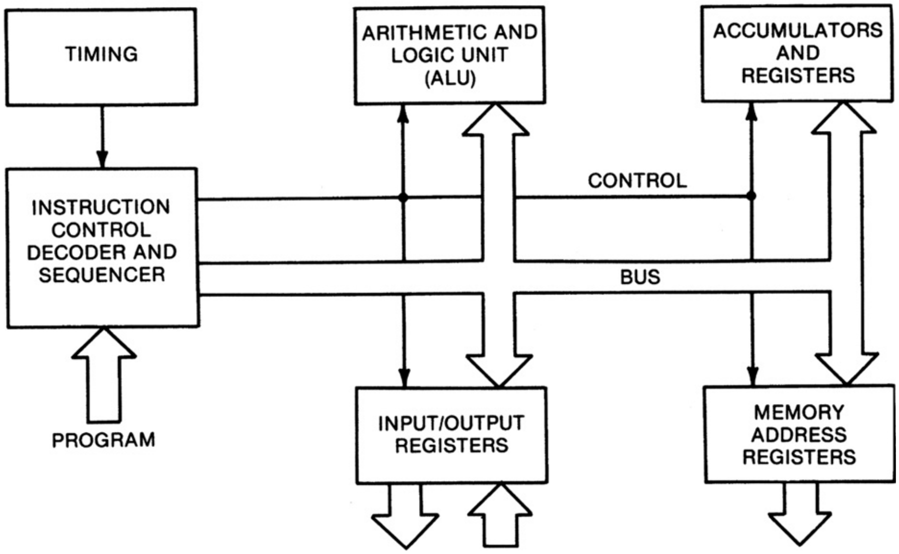

TIMING
ARITHMETIC AND LOGIC UNIT (ALU)
ACCUMULATORS AND REGISTERS
CONTROL
INSTRUCTION CONTROL DECODER AND SEQUENCER
BUS
PROGRAM
INPUT/OUTPUT REGISTERS
MEMORY ADDRESS REGISTERS

Figure 3.32:
Simplified microprocessor block diagram.

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 4

Microcomputer Instrumentation and Control

Chapter Outline

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-08-097097-4.00004-7

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.

Microcomputer Fundamentals

Digital versus Analog Computers

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 Figure 4.1. 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 7) would not include disk drive, keyboard, printer, or monitor.

The *central processing unit* (CPU) is the processor that is the heart of the system. When made in an integrated circuit, it is called a microprocessor. This is where all of the arithmetic and

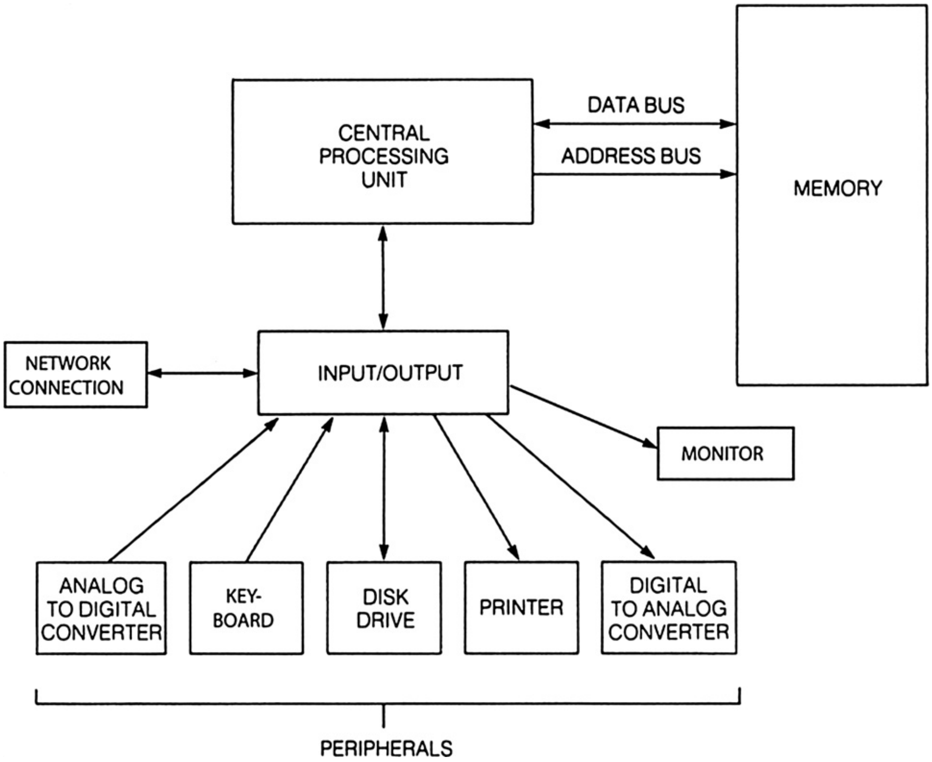


Figure 4.1:
General-purpose computer block diagram.

logic 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 Figure 4.1 indicate the flow of data.

Microcomputers versus Mainframe Computers

With this general idea of what a computer is, it is instructive to compare a general-purpose mainframe computer and 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 10^9 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 billions of arithmetic operations per second (additions, subtractions, multiplications, and divisions). A microcomputer can perform several million 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 logic 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 requested 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 minimize the use of paper tape, and 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) which 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 as well as 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 logic 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 3. 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 Figure 4.1. The central processing unit (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 3). The outputs of the register are sent at the same time over multiple wires to the computer memory and peripherals.

Buses

As shown in Figure 4.2, the group of wires that carries the address is called the address bus. (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 (Figure 4.2). The data bus is slightly different from the address bus in that the CPU uses it to obtain (*read*) data from memory or peripherals, and

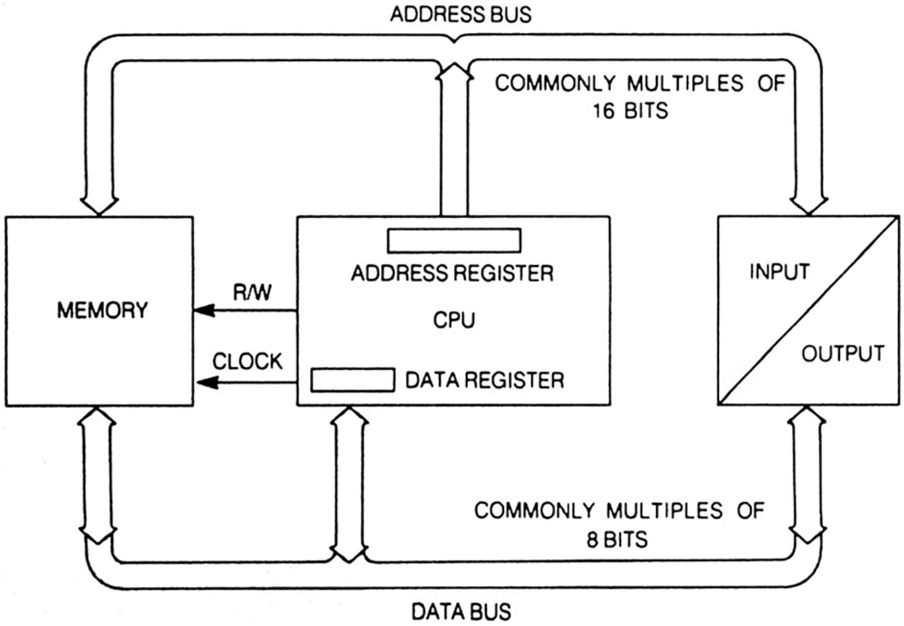


Figure 4.2:
Computer buses.

to send (*write*) data to memory or peripherals. Signals on the address bus originate only at the CPU and are sent to devices attached to the bus. Signals on the data bus 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 data bus is a two-way communication bus, while the address bus is a one-way communication bus. In addition to the address and data bus, there are sets of or wires that are called the control bus. It is this control bus 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 data bus because, although it is bidirectional, data can move in only one direction at a time. The CPU provides a special read/write control signal (Figure 4.2) 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.

The timing diagram for a memory read operation is shown in Figure 4.3.

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

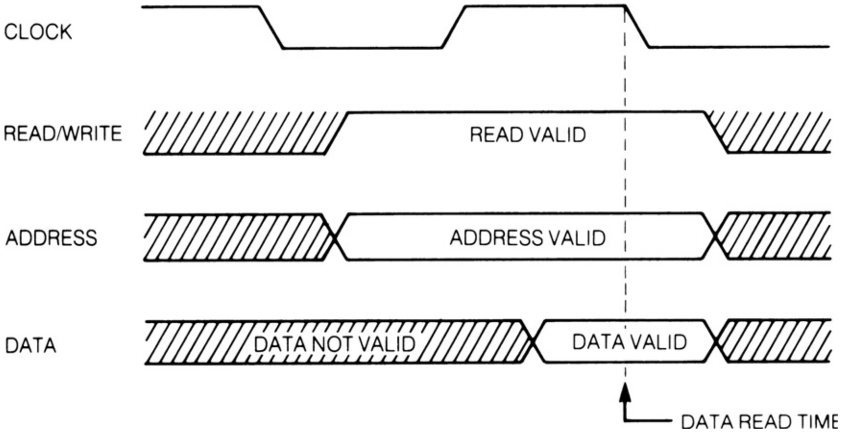


Figure 4.3:
Read/write timing.

address for location 10 is placed on the address bus ("address valid" in Figure 4.3). The number 10 in 16-bit binary (0000 0000 0000 1010) is sent to the memory in the address bus. The binary electrical signals corresponding to 10 operate the specific circuits in the memory to cause the binary data at that location to be placed on the data bus. 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 read/write control line will be set at the logic 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 data bus at the same time the destination address is placed on the address bus. 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 data bus. To allow time for this decoding, the processor institutes an appropriate time delay before receiving the information requested from the data bus. Then, at the proper time, the CPU opens the logic gating circuitry between the data bus 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 data bus. 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 data bus.

Refer again to Figure 4.3. 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 data bus 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 3).

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 Chapter 2. In such cases, the A/D converter providing the sampled input would have a specific memory location. Similarly, any D/A or ZOH 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 to the hardware designer. This model shows the programmer which

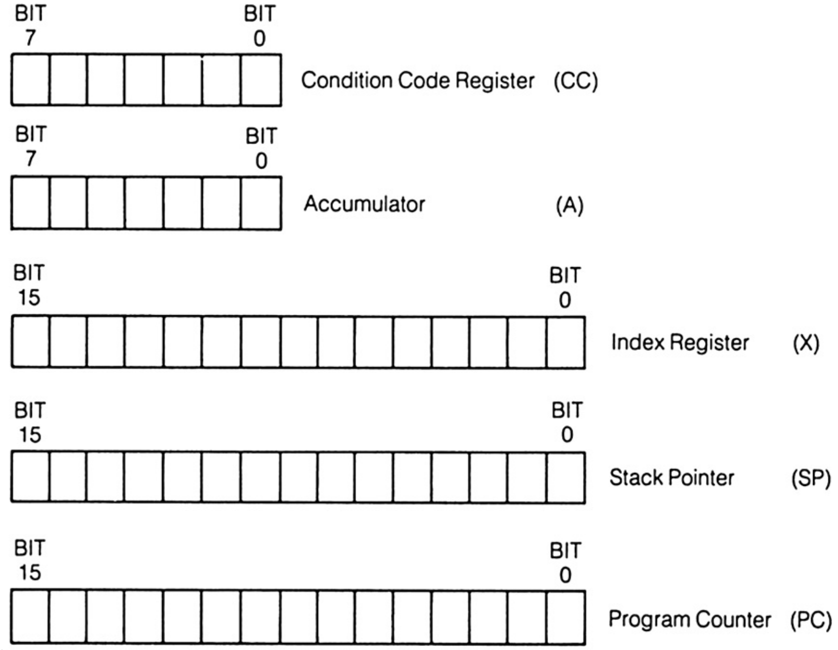


Figure 4.4:
Selected CPU registers.

registers in the CPU are available for program use, and what function the registers perform. Figure 4.4 shows a programming model microprocessor for an 8-bit microcomputer.

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.

Accumulator Register

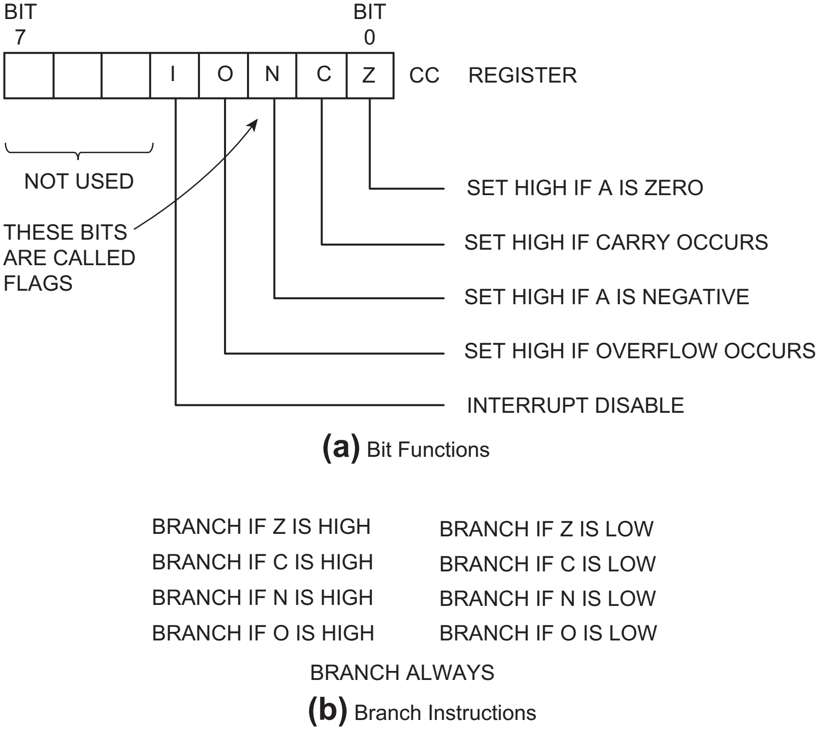
One of the 8-bit registers is an *accumulator*, a general-purpose register that is used for arithmetic and logical operations. The accumulator can be loaded with data from a memory location, or its data can be stored in a memory location. The number held in the accumulator can be added to, subtracted from, or compared with another number from memory. The accumulator is the basic work register of a computer. It is commonly 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 Figure 4.5a. One bit of the CC register indicates that the A register is all zeros. Another bit, the carry 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). Notice what happens when we add 1 to 255 in binary:

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

The eight 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



(a) Bit Functions

(b) Branch Instructions

Figure 4.5:
Condition code register bits.

the simplified 8-bit example registers. In general, the microprocessor will operate with N bits.

The condition code 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 provides programmers with status indicators (the flags) that enable them to monitor what happens to the data as the program executes 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 Figure 4.5a. Typical branch-type instructions are shown in Figure 4.5b.

Program branches are either conditional or unconditional. Eight of the nine branch instructions listed in Figure 4.5b 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 3, 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.

Figure 4.6 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

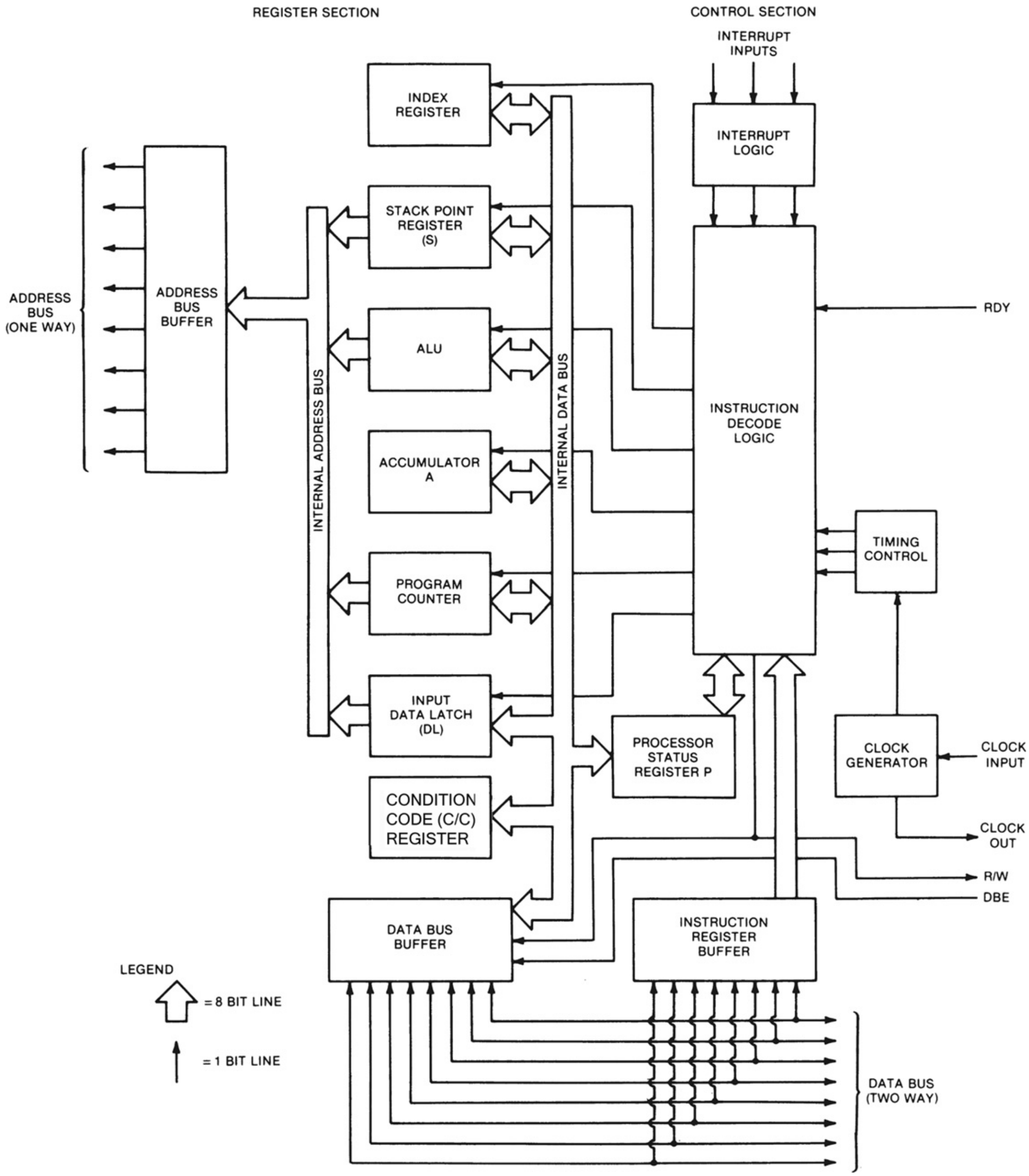


Figure 4.6:
Simplified microprocessor block diagram.

connected to the microcomputer. The connections to the external circuitry are depicted and labeled in Figure 4.6 and include address and data buses. In addition, external connections are made to an oscillator (clock) as well as inputs and outputs: interrupt, ready (rdy), read/write control (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 electrical 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 3. The result of this decoding process is a set of electrical 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 data bus buffer. From this buffer the data are then transferred to the desired component in the register section for execution of the operation.

Note that an arithmetic and logic unit (ALU) is included in the register section of a simplified, representative microprocessor as shown in Figure 4.6. This device is a complex circuit capable of performing the arithmetic and logical operations, as explained in Chapter 3. 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 register, and program counter register. The program counter register holds the contents of the program counter and is connected through the internal address bus to the address buffer register. The address bus for the example microprocessor has 16 lines, and thereby can directly address 65,536 (i.e., 64 K) 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 as well as 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 perform its various tasks at a rate which 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 (Figure 4.4) 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 data bus 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 data bus when the information on the bus is an instruction.

Operation Codes

Numeric codes called *operation codes* (or *op codes* for short) contain the instructions that represent the actual operation to be performed by the CPU. The block diagram of Figure 4.7, 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 op code instructions.

One important function of the op-code decoder is to determine how many bytes must be read to execute each instruction. Many instructions require two or three bytes. Figure 4.8 shows the arrangement of the bytes in an instruction. The first byte contains the op code. The second byte contains address information, usually the lowest or least significant byte of the address.

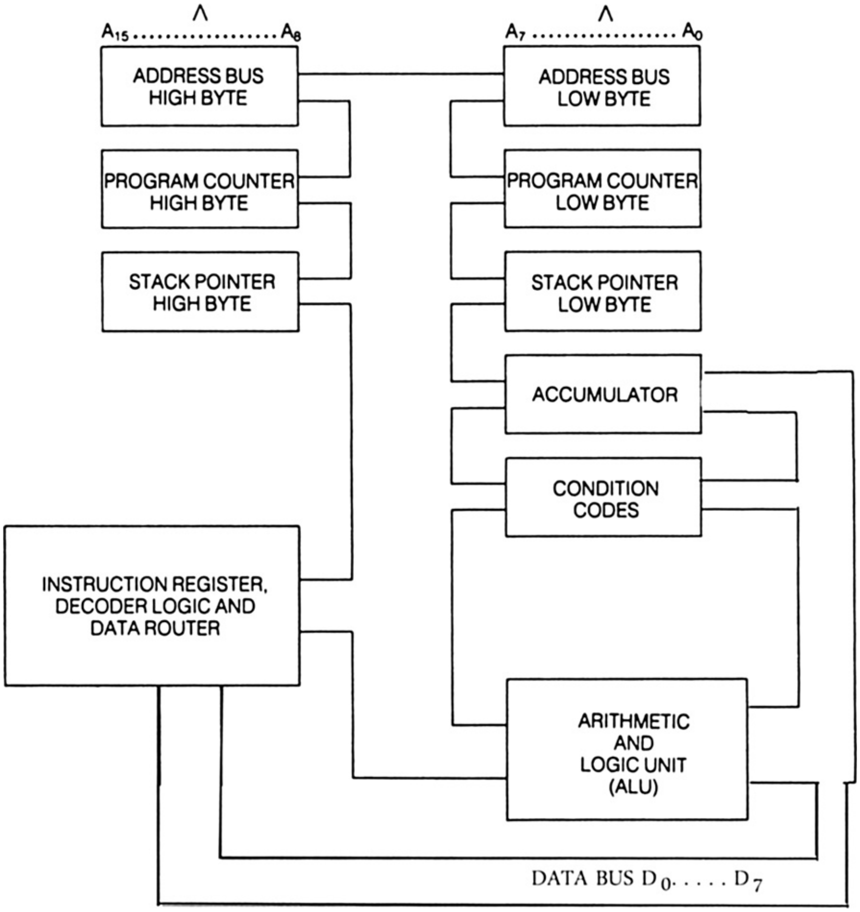


Figure 4.7:
Instruction decode subsystem.

Program Counter

The program counter is used by the CPU to address memory locations that contain instructions. Every time an op code 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 op code. 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.

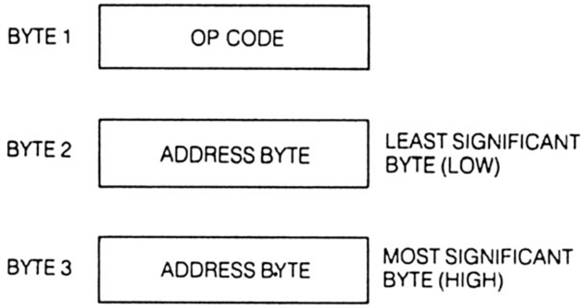


Figure 4.8:
Instruction decode bytes.

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. 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

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 address bus. The binary computation of the final address from the program counter value and second byte of the branch instruction is shown in Figure 4.9.

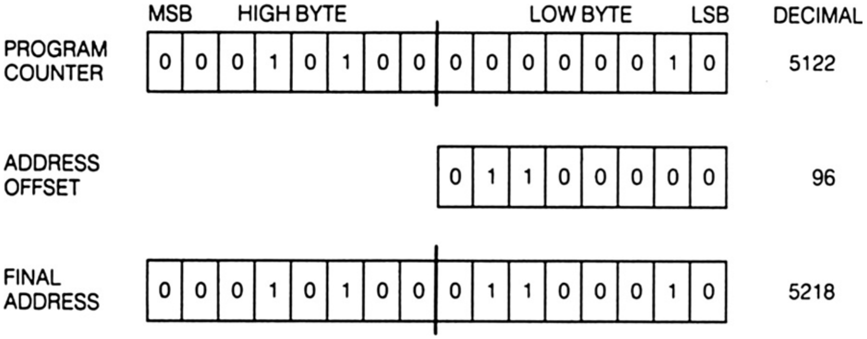


Figure 4.9:
Program counter offset.

Jump Instruction

Branch instructions have a range of +127 or −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 op code, 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 programmer can write 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 rewrite the multiplication program repeatedly. To perform the multiplication, the programmer 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 Figure 4.10 to follow the sequence. 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

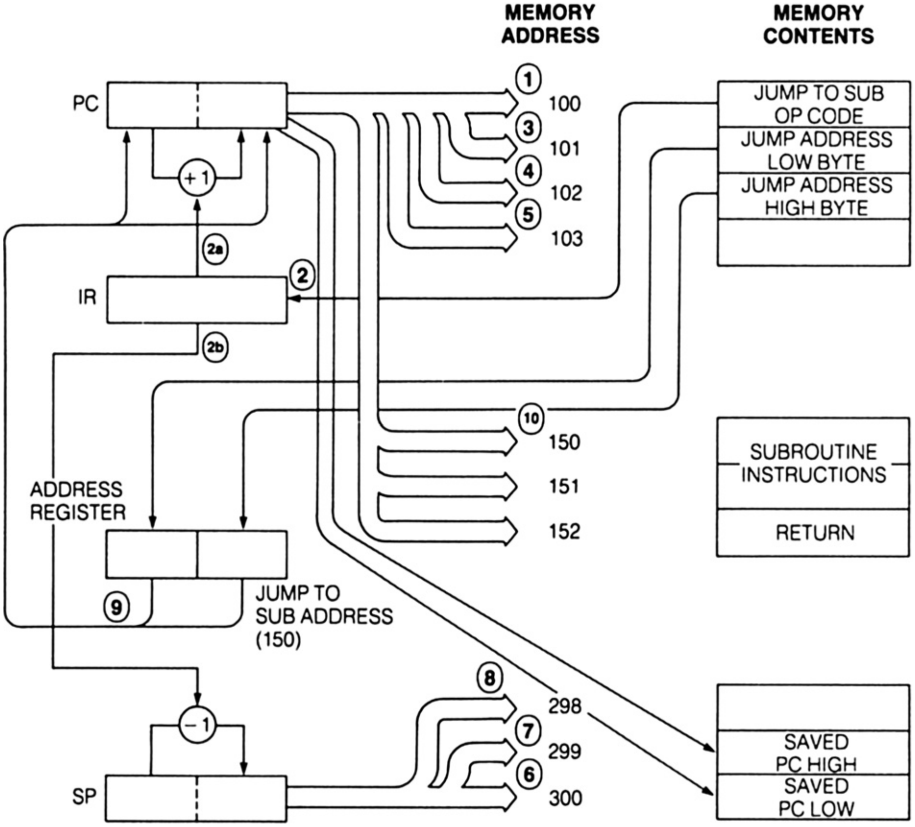


Figure 4.10:
Jump-to-subroutine sequence.

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 op-code 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 Figure 4.4. There is a register in Figure 4.4 called the stack pointer (SP). The address of the special memory location used to store the program counter content is kept in this 16-bit stack pointer register. When a jump-to-subroutine op code is encountered, the CPU uses the number code contained in the stack pointer as a memory address to store the program counter to memory (step 2b of Figure 4.10). The program counter is a 2-byte register, so it must be stored in two memory locations (in the present example). The current stack pointer is used as an address to store the lower byte of the program counter to memory (step 6). Then the stack pointer is decremented (decreased by one) and the high byte of the program counter is stored in the next lower memory location (step 7). The stack pointer 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 stack pointer 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 stack pointer as shown in Figure 4.11, step 1. It uses the stack pointer to address the stack memory to retrieve the old program counter value from the stack (steps 2 and 4). The old

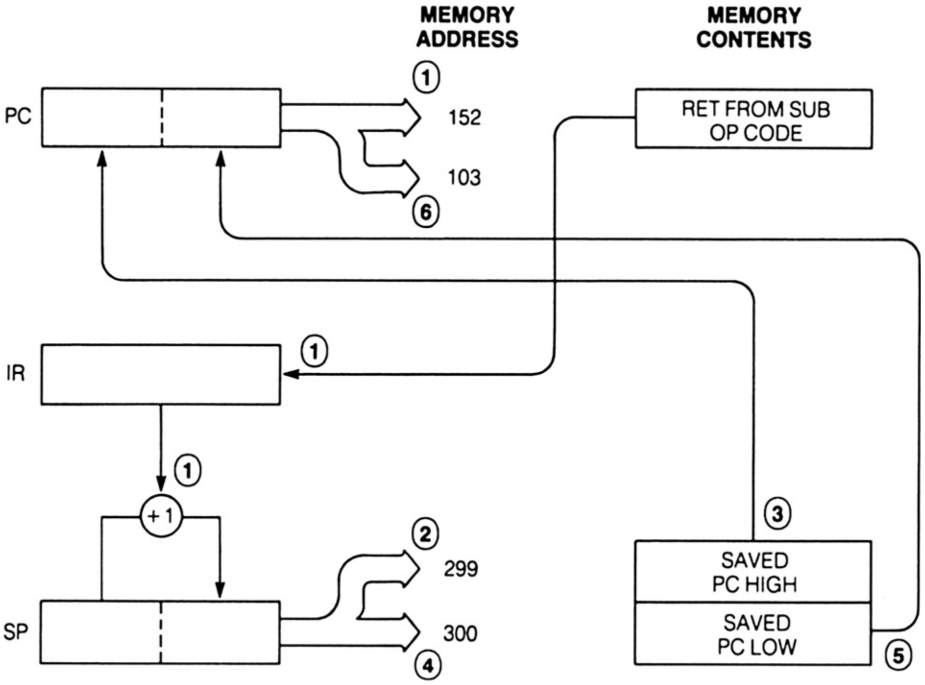


Figure 4.11:
Return-from-subroutine steps.

program counter value is loaded into the program counter register (steps 3 and 5), and execution resumes in the main program (step 6). The return-from-subroutine instruction works like the jump-to-subroutine instruction, except in reverse.

Example Use of a Microcomputer

Let us look at an 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 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* (Figure 4.12). 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 logic signals interfaced to the microcomputer, a program can be written that will perform the required logic function.

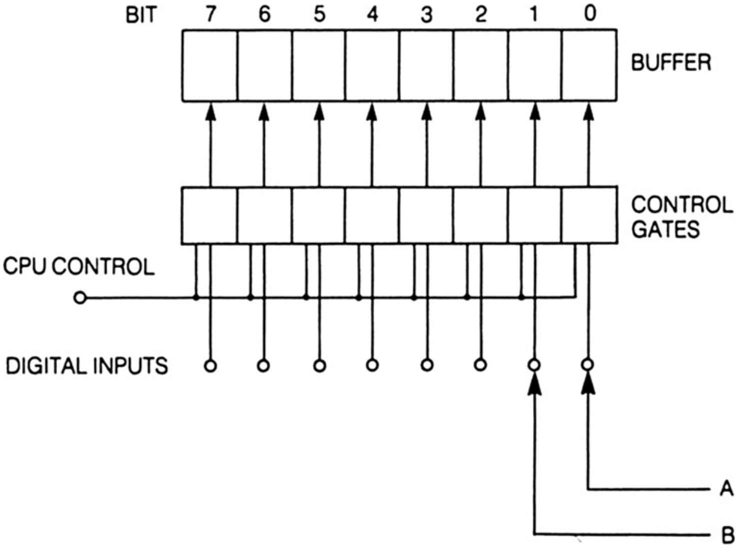


Figure 4.12:
Buffer configuration.

Programming Languages

Before writing a program, one must 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 C++ 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 through use of a program known as a compiler. The compiler converts the high-level language to a so-called machine 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

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 return from subroutine, 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.

Table 4.1 shows the assembly-language equivalents for typical microcomputer 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

Table 4.1: 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

numbers the computer uses. However, the program eventually must be converted to the 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. This enables the programmer to write the program using words that have 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 logic 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 logic functions are performed by using the contents of the accumulator and some memory location. All eight bits of the accumulator are affected, and all are changed at the same time. As shown in Figure 4.13, 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 outputs are stored back into the accumulator in the corresponding bit positions. The OR logic 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 Figure 4.13, the microcomputer uses a special type of sequential logic operation, the shift,

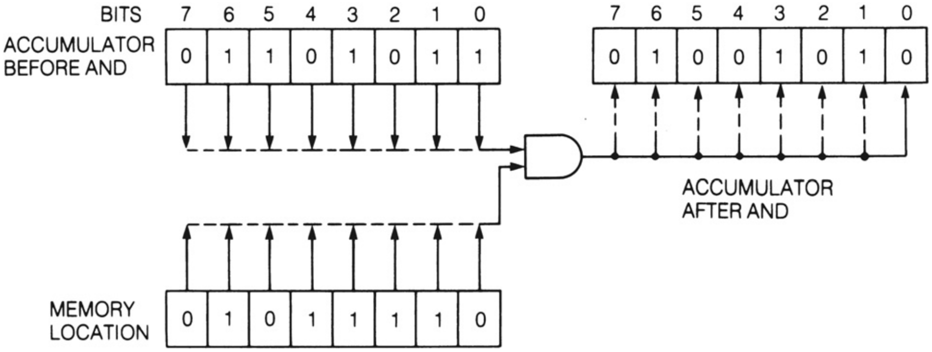
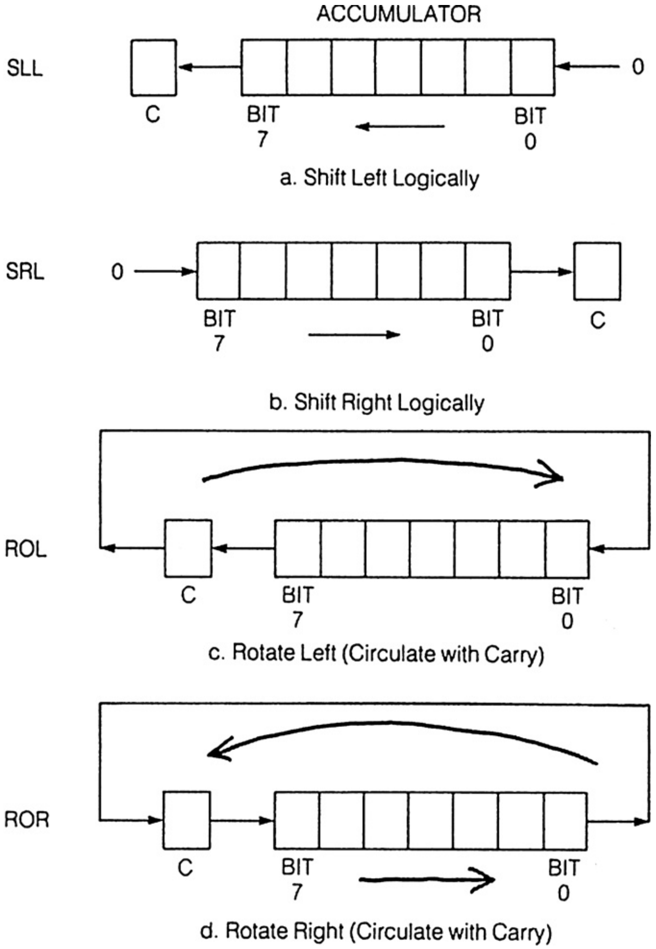


Figure 4.13:
AND logic illustration.

to move the bits to the AND gate inputs. A type of register that is capable of such shift operations was discussed in Chapter 3 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. Figure 4.14 shows the four types of shifts (logical, circulating, right, and left) and their effects on the accumulator. In a left shift, bit 7 (the leftmost bit) is shifted into the carry bit of the CC register, bit 6 is shifted into bit 7, 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 as well as the accumulator.



a. Shift Left Logically

b. Shift Right Logically

c. Rotate Left (Circulate with Carry)

d. Rotate Right (Circulate with Carry)

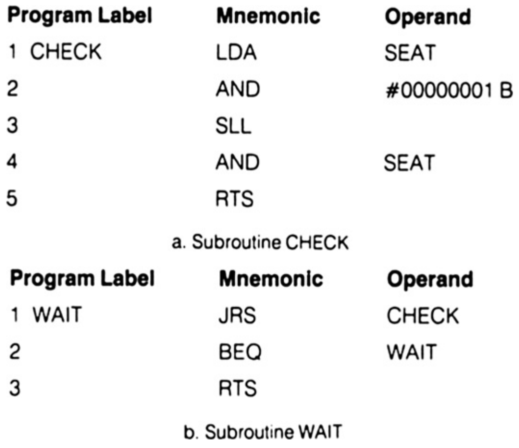
Figure 4.14:
Shift register operations.

Programming the AND Function

It is 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 4.1c. 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 Figure 4.15 performs the required AND function. The programmer must first know which memory location the digital buffer interface (Figure 4.12) 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 Figure 4.12 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.



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

a. Subroutine CHECK

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

b. Subroutine WAIT

Figure 4.15:
Assembly language AND subroutine.

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 Figure 4.15a (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.

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 (Figure 4.14a). 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).

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 Figure 4.15b 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 condition codes set as they were after the last AND instruction. The Z bit (see Figure 4.5a) is set if A and B are not both high (the accumulator is zero). The BEQ instruction (see Table 4.1) in line 2 of WAIT branches back because the accumulator is zero and causes the computer to re-execute 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 (as well as for division). The algorithm on which the subroutine is based is derived from the fundamental multiplication of a pair of bits:

$$\boxed{\begin{array}{c} 0 \times 0 = 0 \\ 1 \times 0 = 0 \\ 0 \times 1 = 0 \\ 1 \times 1 = 1 \end{array}}$$

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):

$$\begin{array}{llr l}
A = 1101 & & & (13\text{ decimal}) \\
B = 10 & & & (2\text{ decimal}) \\
A & & 1\,1\,0\,1 & \\
B & \times & \underline{\quad 1\,0} & \\
 & & 0\,0\,0\,0 & \\
 & + & \underline{1\,1\,0\,1\ \ } & \\
C & & 1\,1\,0\,1\,0 & (26\text{ decimal})
\end{array}$$

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 central processing unit is a microprocessor. It is an integrated circuit similar to the one shown in Figure 4.16a. It contains 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 arithmetic logic unit (ALU), as well as registers for data and instruction storage and a control section. The chip is housed in

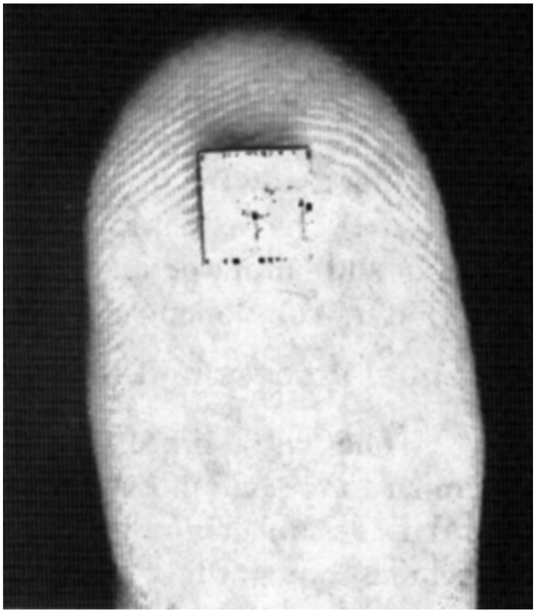


Figure 4.16:
Photograph of exemplary IC chip

a rectangular, flat package similar to the one shown in Figure 4.16b. Newer 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.

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 *read-only memory* (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 EEPROM (electrically erasable, programmable read-only-memory). 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 *random access memory* (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 read/write memory, 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 3.

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 logic 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 Chapter 2, 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 3). 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 Figure 4.17. 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 3). For the purposes of explaining the operation of this simplified

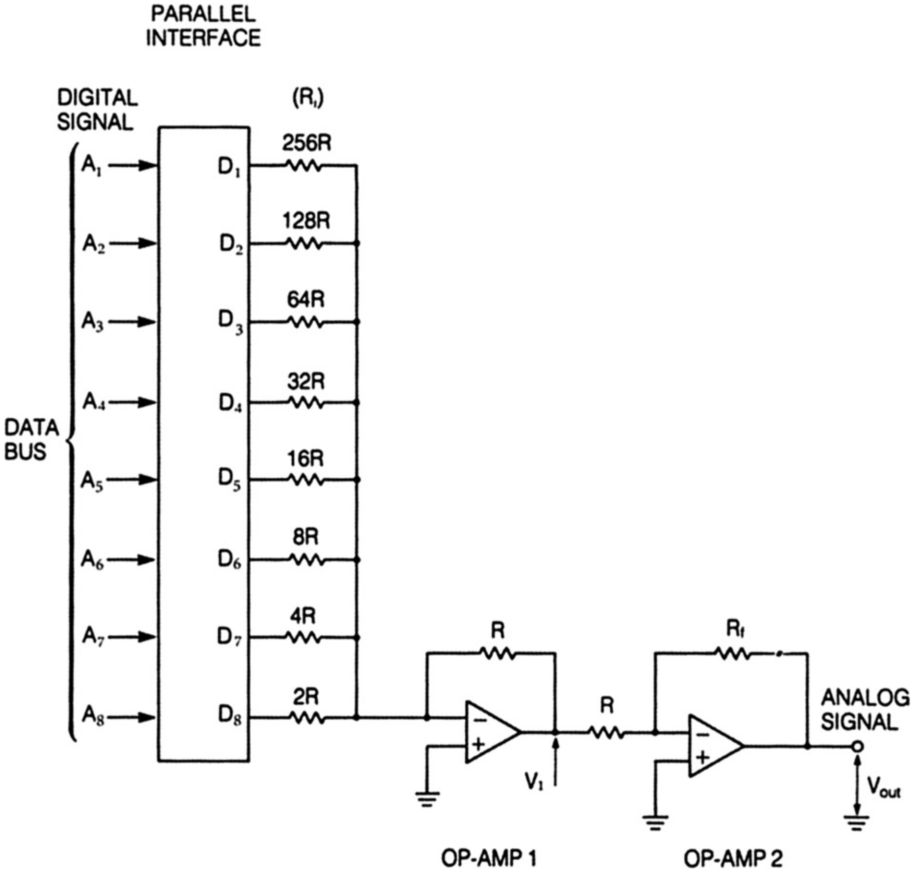


Figure 4.17:
Simplified D/A configuration.

example D/A converter, it is assumed that the parallel interface includes output circuitry associated with each data bit latch such that

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

where $A_n =$ nth bit of the 8-bit input digital data where A_8 is the MSB. In this example, the output of each latch is a digital signal that is ideally 0 V if the bit is low and 5 V if the bit is high. 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

$$R_s(n) = 2^{N-n+1}R \qquad n = 1, 2, \ldots N \tag{2}$$

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 3) is given by

$$V_\mathbf{1} = -\sum_{n=1}^{N} \frac{D_n}{2^{N-n+1}} \tag{3}$$

$$= -\frac{5}{2^N} \sum_{n=1}^{N} A_n 2^{n-1} \tag{4}$$

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

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

The second op amp has a closed-loop gain of

$$A_{c\ell} = -R_f/R \tag{6}$$

The output voltage of this 2nd op amp is given by

$$V_{\text{out}} = \frac{5R_f}{2^N R} N_{10} \tag{7}$$

$$= K_{\text{DA}} N_{10} \tag{8}$$

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 actuator (see 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 Chapter 2, 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 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. Figure 4.18 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 logic variables "reset" and convert C and input logic variable EOC as shown in Figure 4.18.

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 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 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 Figure 4.19.

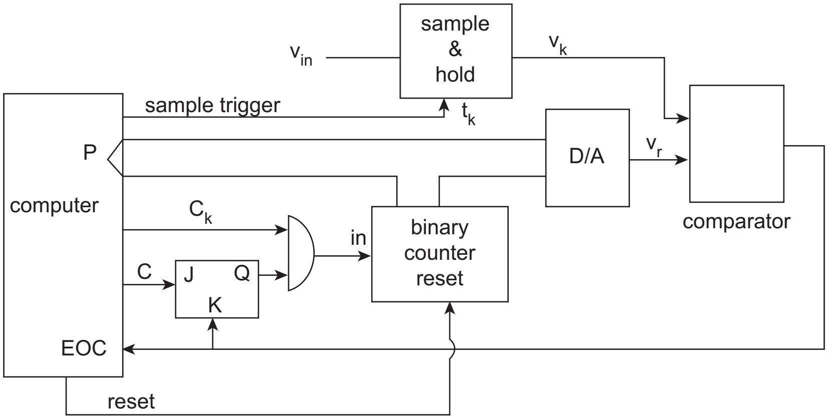


Figure 4.18:
Example A/D converter.

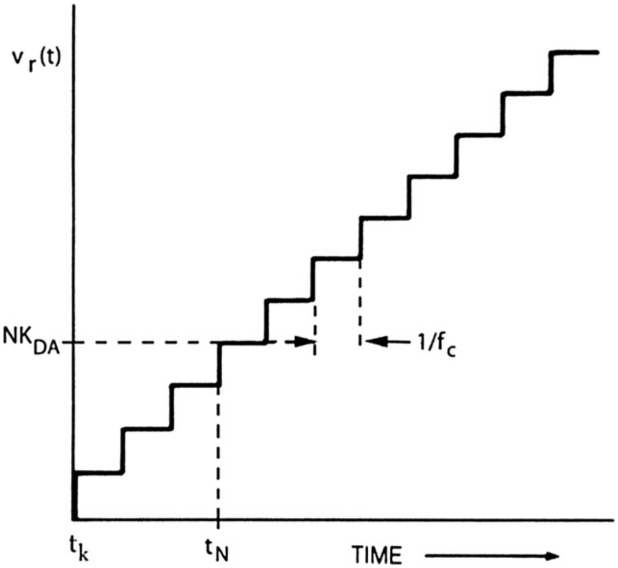


Figure 4.19:
Digital ramp waveform.

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* are the binary equivalent of *N* where *N* is the largest integer in the following:

$$N = \{\lfloor f_c(t_N - t_k)\rfloor\} \qquad t_N \le t < t_{N+1} \tag{9}$$

The ramp voltage $v_r(t)$ for the time interval specified in Eqn (9) is given by

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

as explained above for a D/A converter and shown in Figure 4.19. 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

$$v_{comp} = v_L \qquad v_r < v_k \tag{10}$$

$$= v_H \qquad v_r \geq v_k \tag{11}$$

where v_L and v_H are voltages corresponding to logic low and high, respectively. At coincidence, the ramp voltage is essentially given by

$$v_r(t_c) = v_k \tag{12}$$

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 end-of-conversion 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 $t_k + t_c$:

$$\begin{aligned} N_c &= N(t_c) \\ N_c &= \frac{v_k}{K_{DA}} \end{aligned} \tag{13}$$

The binary equivalent of N_c is denoted N_{c2}.

As shown in Figure 4.18, the counter output lines are connected to a computer parallel input (P) such that the counter contents are available to the computer data bus (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., $\max(t_c) << T$). This condition can be met with sufficiently high clock frequency (f_c).

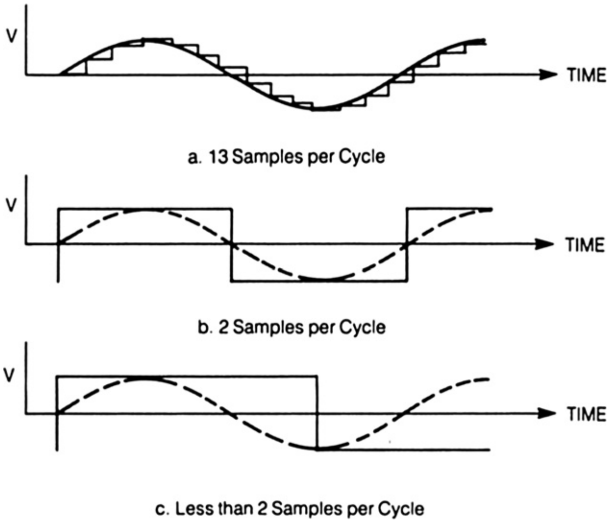
Sampling

As explained in Chapter 2, a discrete time digital system operating on continuous time variables requires sampling the input signal at the Nyquist or higher rate. Figure 4.20 shows a sine wave analog signal and some digital approximations with various sampling rates. Notice that Figure 4.20a with 13 samples per sine wave cycle follows the sinusoid much closer than Figure 4.20b, which only samples twice in a cycle. When the sampling rate is less than 2, as in Figure 4.20c, aliasing errors occur as explained in Chapter 2.

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 end of conversion, 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. This usually consists of a few lines of assembly-language code that check to see if the



a. 13 Samples per Cycle

b. 2 Samples per Cycle

c. Less than 2 Samples per Cycle

Figure 4.20:
Sampling rate illustration.

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 electrical 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 3. 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. Figure 4.21 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 MPU. It is connected to the other components by means of three buses: address bus (AB), data bus (DB), and control bus (CB). As explained above, each bus consists of a set of wires over which

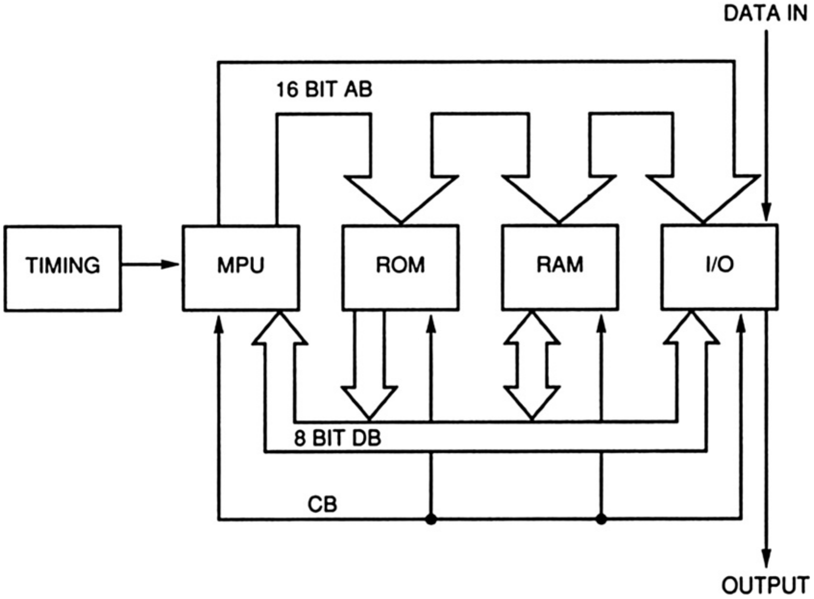


Figure 4.21:
Representative automotive computer block diagram.

binary-valued electrical 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 control bus 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. 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, as well as a port that is connected to the data bus of the computer.

Figure 4.22 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 input or output, 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 data bus.

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 electrical 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, input or output 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 block diagram of a typical computer-based instrument is depicted in Figure 4.23. 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

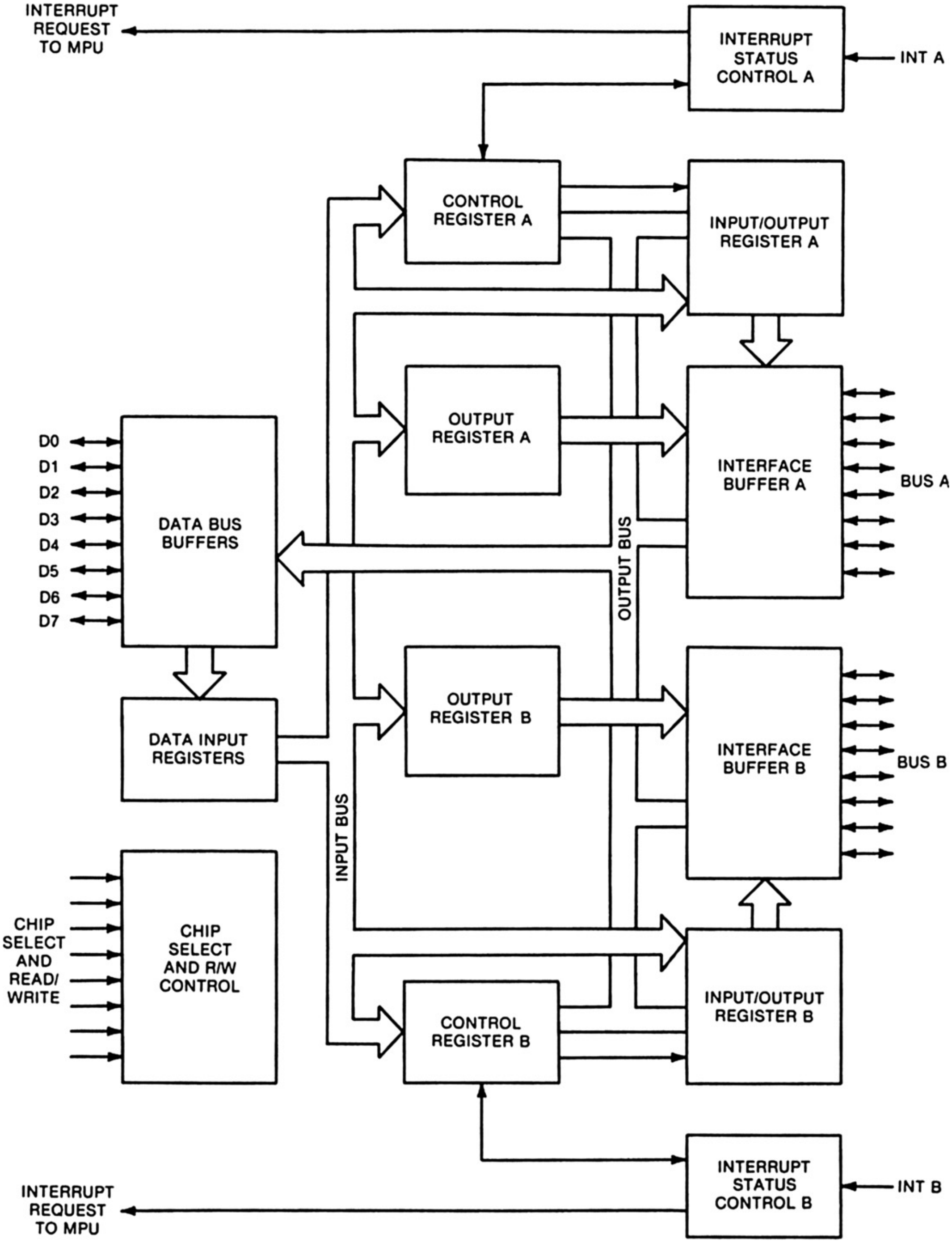


Figure 4.22:
Illustration of I/O ports.

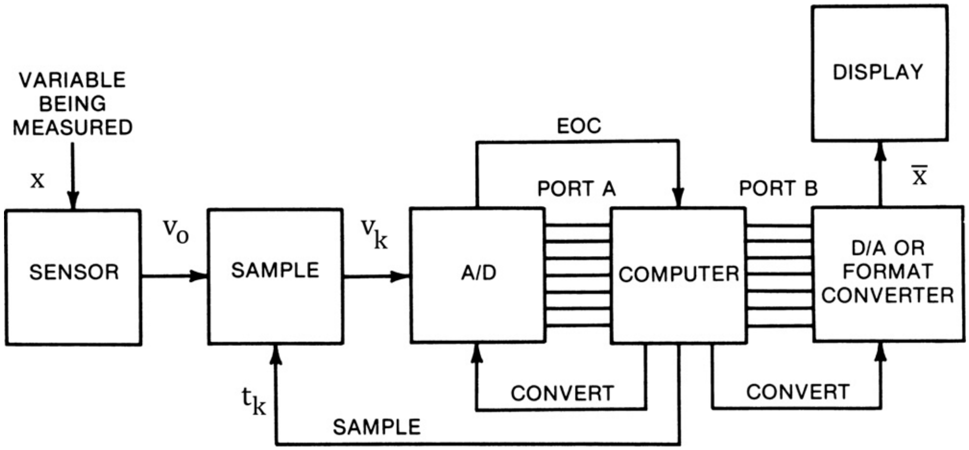


Figure 4.23:
Automotive digital instrumentation block diagram.

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.

Digital Filters

In Chapter 2, 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. Recall from the Chapter 1 discussion of filters that 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. Bandpass filters pass midrange frequencies but reject both low and high frequencies.

Chapter 2 introduced the concept of digital (discrete time) filters that perform filtering operation on samples x_k of the signal ($x(t)$) that is to be filtered. Recall from that discussion that one relatively commonly used algorithm for a digital filter is the recursive algorithm for calculating the output y_n, which is repeated here for convenience:

$$y_n = \sum_{k=1}^{K} a_k x_{n-k} - \sum_{j=1}^{J} b_j y_{n-j} \tag{14}$$

It was further shown that the z-operational transfer function $H(z)$ is given by

$$H(z) = \frac{\sum_{k=1}^{K} a_k z^{-k}}{1 + \sum_{j=1}^{J} b_j z^{-j}} \tag{15}$$

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 $x_{n-\mathrm{k}}$ 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 or in certain microprocessors by a special circuit section called "hardware multiply" in which the multiplication (e.g., $a_k x_{n-k}$) is performed by loading the variables into registers and then, by means of a control signal, the multiplication is performed by the circuitry with the result placed in a temporary storage register (e.g., in RAM). 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 electrical signal that is proportional to the height of the fuel in the center of the tank as described in Chapter 6. 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 $\overline{x}_n$ at time t_n of K previous samples of data x_k: $k = 1,2,\cdots,K$ is given by

$$\overline{x}_n = \frac{1}{K}\sum_{k=1}^{K} x_{n-k}$$

This algorithm is of the same structure as that used in a recursive filter (e.g., Eqn (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. 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

Recall the basic closed-loop control system block diagram of Chapter 1 (continuous time) and Chapter 2 (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. Recall 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., $\bar{\boldsymbol{u}}_k$ of Chapter 2) 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 Chapter 1, can be readily implemented with a microcomputer. Recall 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 against 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

Recall that in Chapters 1 and 2, the concept of a feedback control system was introduced. Those chapters dealt with the basic analytical models and control algorithms on an abstract 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 as well as the limit-cycle controller. Figure 4.24 shows the physical configuration of a control system employing a computer. 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.

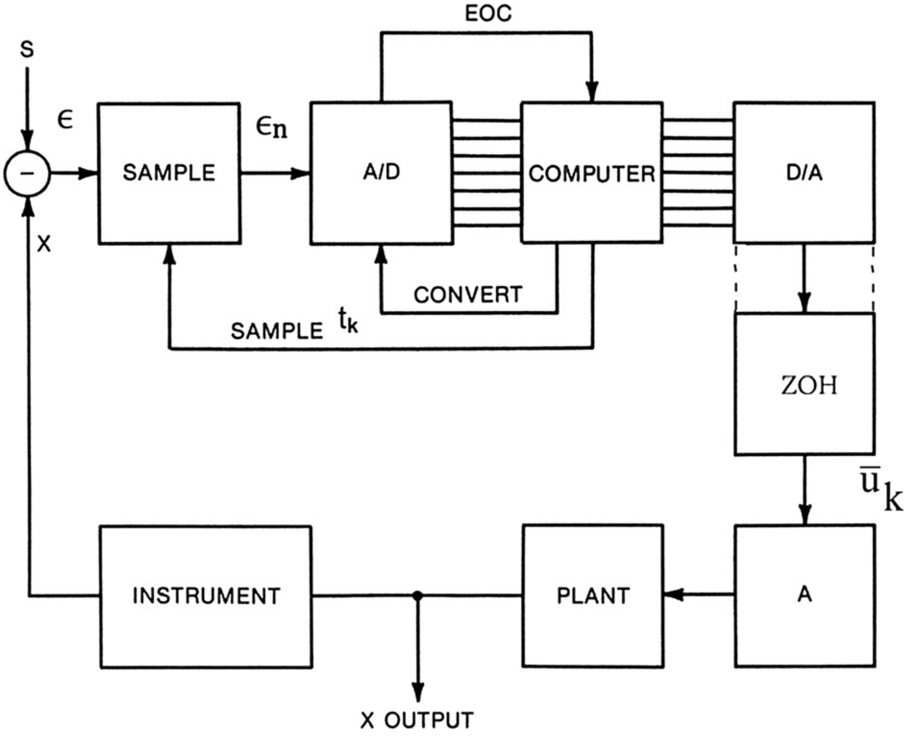


Figure 4.24:
Digital feedback control system block diagram.

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., $n = 1, 2, \cdots$). 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:

$$y_n = K_P \in_n + \frac{K_I T}{2} \sum_{k=1}^{K} (\in_{n-k} + \in_{n-k-1}) + \frac{K_D(\in_n - \in_{n-1})}{T}$$

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 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.

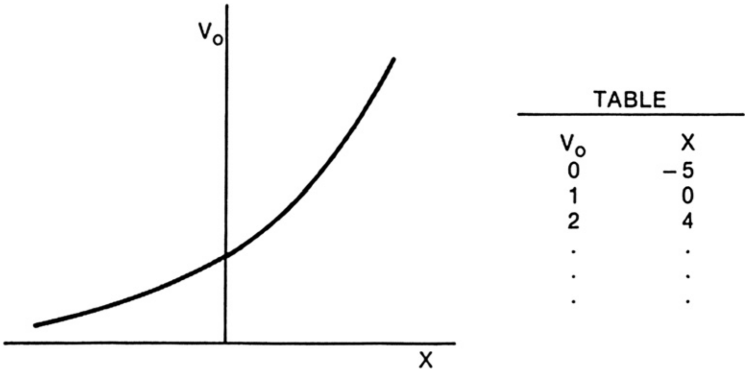
After computing y_n for each input sample, a digital version of y_n is transmitted through the I/O to the D/A converter and zero-order hold (ZOH) as explained in Chapter 2. The dashed lines between the D/A and ZOH (see Chapter 3) blocks indicate that the ZOH may be implemented as part of the D/A. There it is converted to analog format, providing a control signal $\bar{\boldsymbol{u}}_k$ 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, and
3. calibration conversion.

The concept of table lookup is illustrated in Figure 4.25, in which a pair of variables, V_o and X, are related by the graph depicted therein. Also shown in Figure 4.25 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



TABLE

V_o	X
0	−5
1	0
2	4
.	.
.	.
.	.

Figure 4.25:
Illustration of table lookup.

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 and V_2 as the nearest values for V_0 and X_1, X_2 as the corresponding tabulated values, the value for X corresponding to V_o is found by linear interpolation:

$$X = X_1 + (X_2 - X_1)(V_o - V_1)/(V_2 - V_1)$$

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 Figure 4.25. 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 Figure 4.26. 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

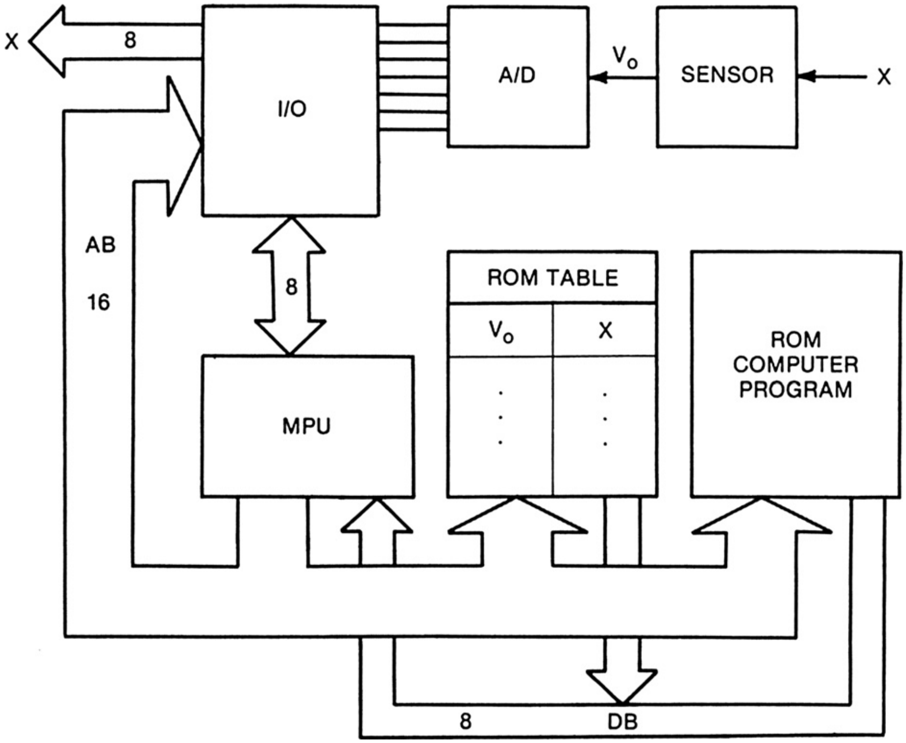


Figure 4.26:
Table lookup block diagram.

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 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 Eqn (9) (or higher-order polynomial fit algorithm).

Repeated reference will be made to the table lookup function in later chapters. In particular, Chapter 7 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. 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 9). 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 powertrain control system (see Chapter 7) involves many thousands of individual lines of code. Some assistance in the form of automatic code generation is available from certain software, although a 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.

CHAPTER 5

The Basics of Electronic Engine Control

Chapter Outline

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-08-097097-4.00005-9

Engine control in the vast majority of engines means regulating fuel and air intake as well as 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 7 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.

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 as well as 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 pick-up 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 as well as 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 NOx), 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 NOx.

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 twenty-first 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 as well as 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 miles per gallon 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 (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 are made. The U.S. federal government has published test procedures that include several steps. The first step is to place the automobile on a chassis dynamometer, like the one shown in Figure 5.1.

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

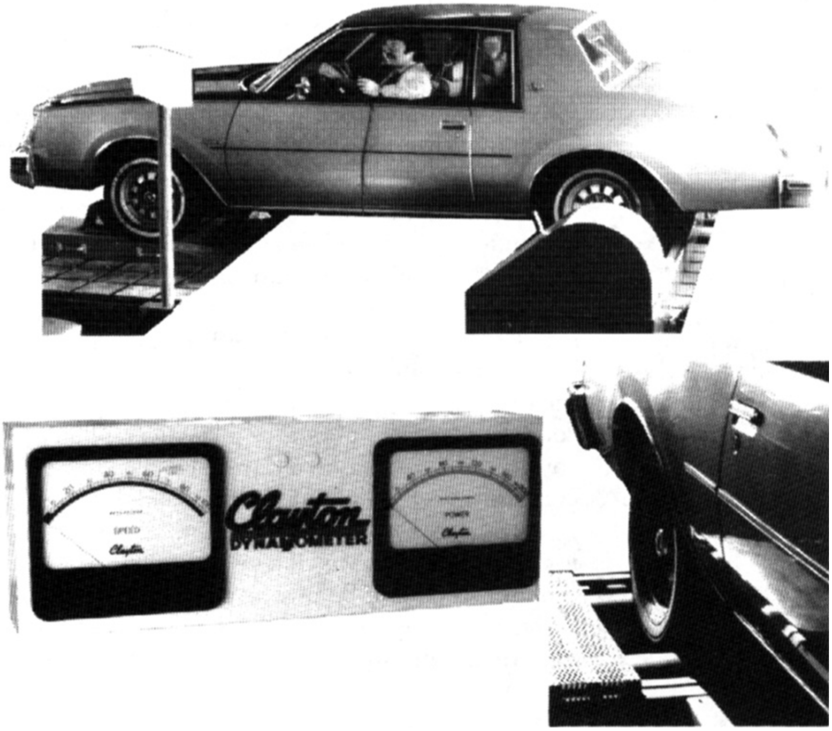


Figure 5.1:
Chassis dynamometer.

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 electrical 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 electrical 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 were 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. Figure 5.2 illustrates sample 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 miles (12 km). The highway schedule takes 765 seconds and simulates 10.24 miles (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 corporate average fuel economy (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 5.1 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 5.1 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.

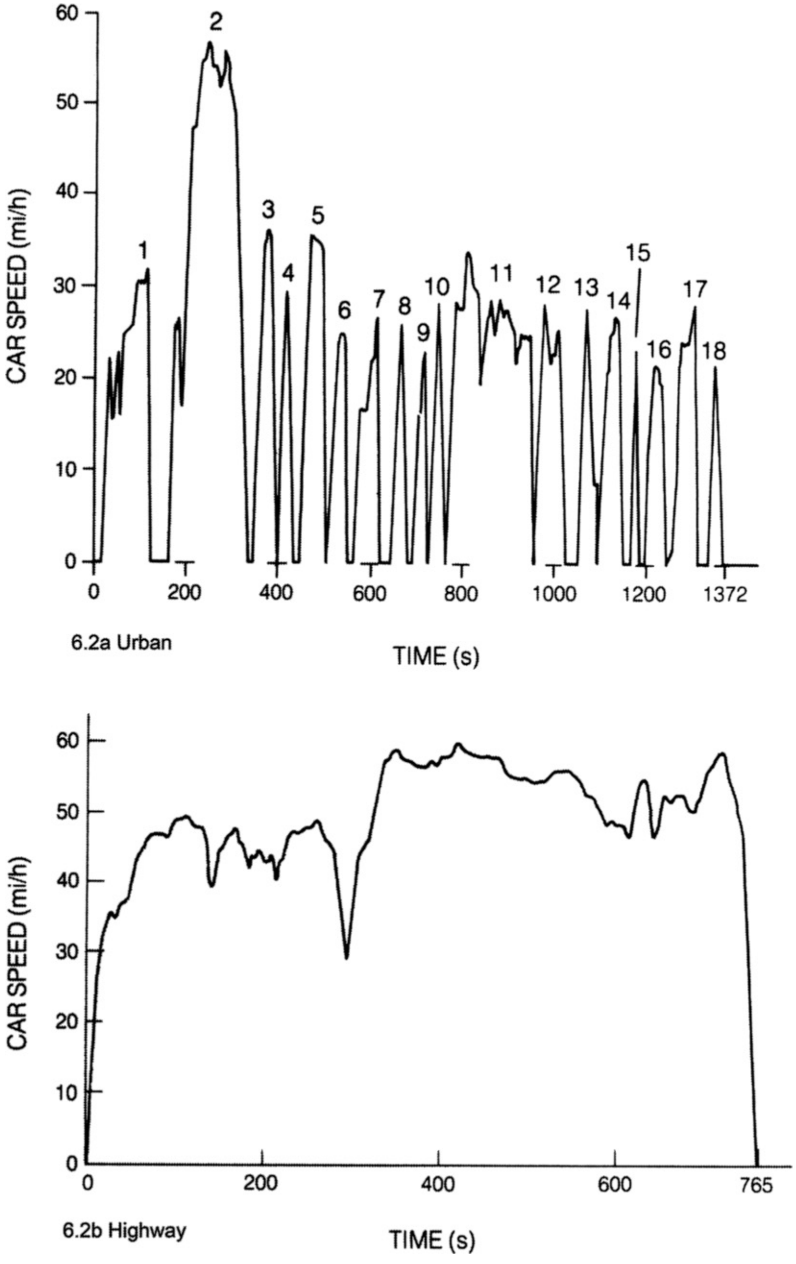


Figure 5.2:
Federal driving schedules (*Title 40 United States Code of Federal Regulations*).

Table 5.1: 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

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 miles—whichever comes first) and full life cycle (10 years or 100,000 miles) as of 1990. The standards are:

HC	0.31 g/mi
CO	4.20 g/mi
NOx	0.60 g/mi (non-diesel)
1.25 g/mi (diesel)	

Model year 1994: 40%
Model year 1995: 80%
Model year 1996: 100%

These regulations were phased in according to the following schedules:

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 5.1 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, over all operating conditions, and over the life of the vehicle to stay within the tolerance required by the EPA regulations. Each automaker must verify that each model produced will still meet emission requirements after traveling 100,000 miles. 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 make 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, as well as other on-board automobile electronic systems, commercially feasible. As pointed out in Chapter 3, 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 RPM, the mass flow rate of air into the engine varies directly with throttle plate angular position (see Figure 5.3).

As the driver depresses the accelerator pedal, the throttle angle (θ_T in Figure 5.3) increases, which increases the cross-sectional area through which the air flows, reducing the resistance to airflow and thereby allowing an increased airflow into the engine. A model for the airflow vs. 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 U.S. system of units as a rough "rule of thumb," an airflow rate of about 6 lb/h produces 1 horsepower of usable mechanical power at the output of the engine. Metric units have come to

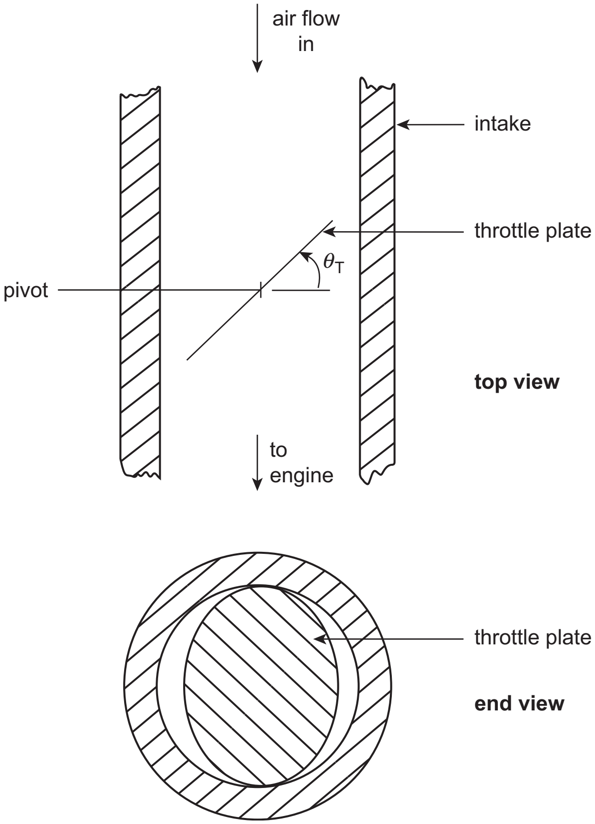


Figure 5.3:
Intake system with throttle plate.

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 (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 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. Figure 5.4 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 and closed-loop operating modes (see Chapter 1). As explained in Chapter 1, a closed-loop control system requires measurements of certain output variables such that the controller can calculate the state of the system being controlled, whereas an open-loop system does not. The electronic engine control system receives input electrical signals from the various sensors that measure the state of the engine. From these signals, the controller generates output electrical 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 6. As mentioned, the configuration and control for an automotive engine control system are determined in part by the set of sensors that is 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 variables, the desired variable is found by computation.

Figure 5.5 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

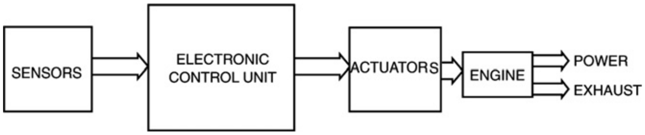


Figure 5.4:
Generic electronic engine control system.

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 re-circulated back to the input, and a catalytic converter to further control emissions. The addition of re-circulated exhaust gas to the engine intake, as well as various sensors and actuators depicted in Figure 5.5, 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 Figure 5.5.

At one stage of development, the electronic engine control consisted of separate subsystems for fuel control, spark control, and exhaust gas recirculation. The ignition system in Figure 5.5 is shown as a separate control system, although engine control is evolving toward an integrated digital system (see Chapter 7).

Inputs to Controller

Figure 5.6 identifies the major physical quantities that are sensed and provided to the electronic controller as inputs. They are as follows:

1. Throttle position sensor (TPS)
2. Mass airflow rate (MAF)
3. Engine temperature (coolant temperature) (CT)
4. Engine speed (RPM) and angular position
5. Exhaust gas recirculation (EGR) valve position
6. Exhaust gas oxygen (EGO) concentration

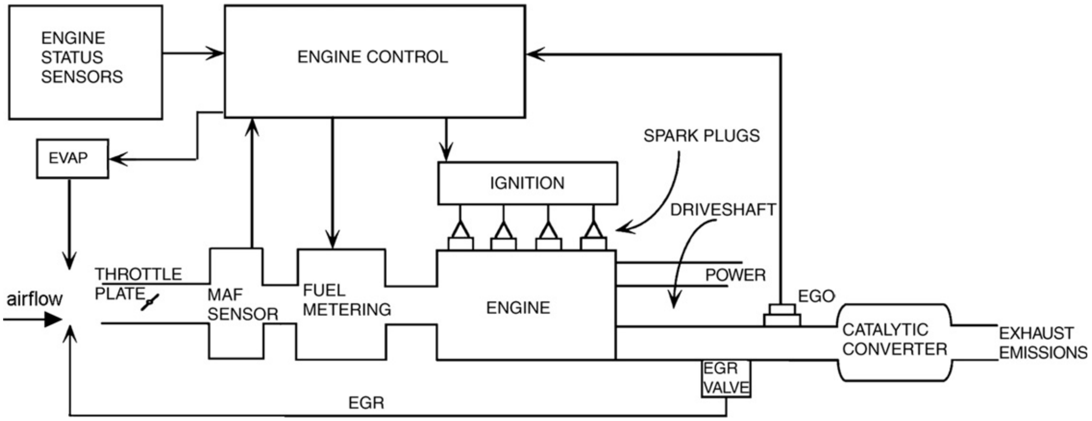


Figure 5.5:
Engine functions and control diagram.

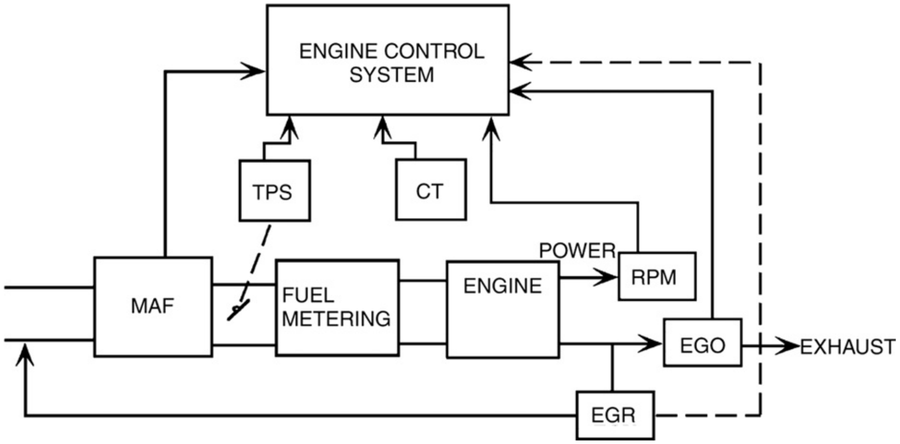


Figure 5.6:
Major controller inputs from engine.

Output from Controller

Figure 5.7 identifies the major physical quantities that are outputs from the controller. These outputs are

1. Fuel metering control
2. Ignition control (dwell and timing)
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 7 shows how these

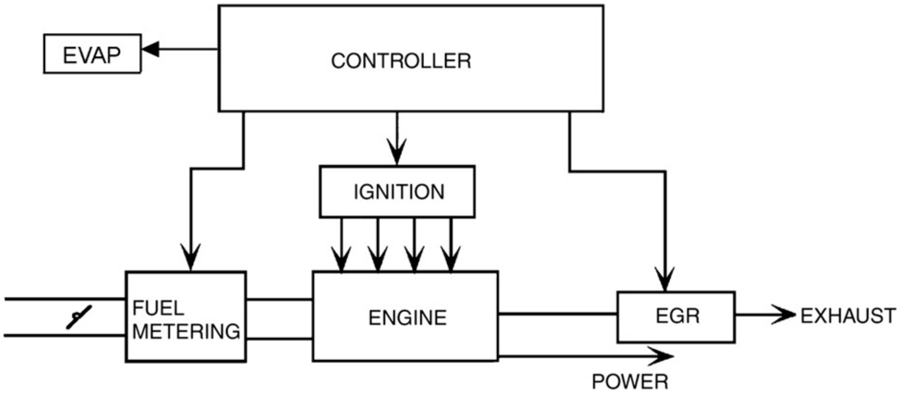


Figure 5.7:
Major controller outputs to engine.

separate control systems are being integrated into one system and are implemented with digital electronics.

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 6.

The configuration of a single cylinder of an IC engine is depicted in Figure 5.8a. Mechanical power is produced by the engine in the form of torque acting on the rotating crank shaft. 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.

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 from its highest point (called "top-dead-center" or TDC) to the lowest point (called "bottom-dead-center" or BDC). 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:

$$p_c = V_c^{\gamma} \tag{1}$$

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.

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.

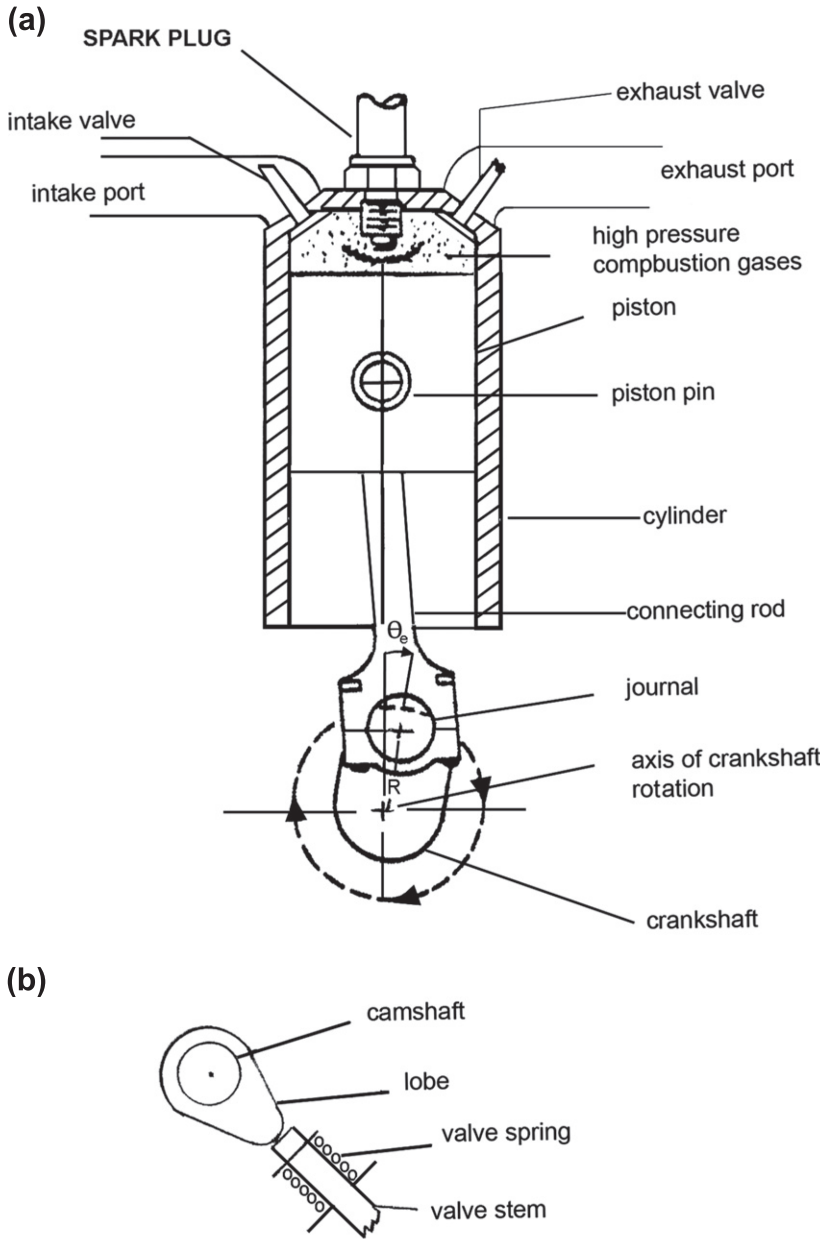


Figure 5.8:
IC engine cylinder and cam actuation mechanism

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, 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 (mostly CO_2 and H_2O) are pumped out of the cylinder into the exhaust system and released through this system to the atmosphere.

The actual point in the 720° crankshaft rotation angle at which the valves open and close (called valve timing) is determined by a mechanism that includes the camshaft and mechanical linkage connecting it to the valves. The camshaft, which is illustrated in Figure 5.8b, 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 Figure 5.8a extend to the end of the valve stem depicted in Figure 5.8b. 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 6, 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. For the present, however, the discussion is focused on basic engine processes.

Energy is produced by a four-stroke/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 as well as 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 vs. V_c for the entire cycle. Figure 5.9 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

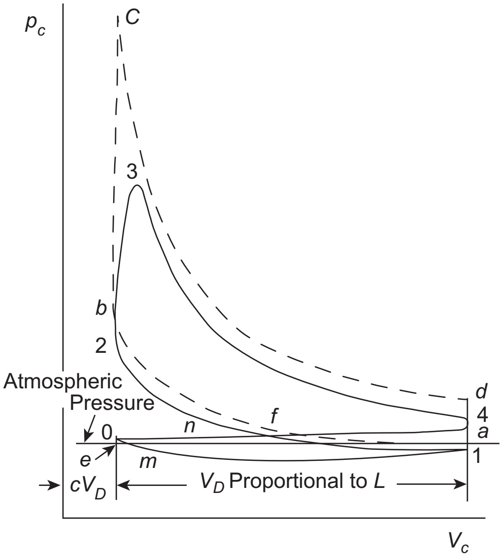


Figure 5.9:
Indicator diagram for a four-stroke engine.

negligible. In Figure 5.9, 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 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 as well as 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:

$$W_i = \oint p_c \mathrm{d}V_c \tag{2}$$

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 7. As clearly shown from Figure 5.9, in any practical engine the indicator diagram deviates from the ideal as indicated by the continuous curve of Figure 5.9.

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 powertrain 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 powertrain, it is helpful to refer to Figure 5.10, which illustrates the geometry of a single cylinder in a four-stroke IC engine. Figure 5.10 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. 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.

Figure 5.10 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)

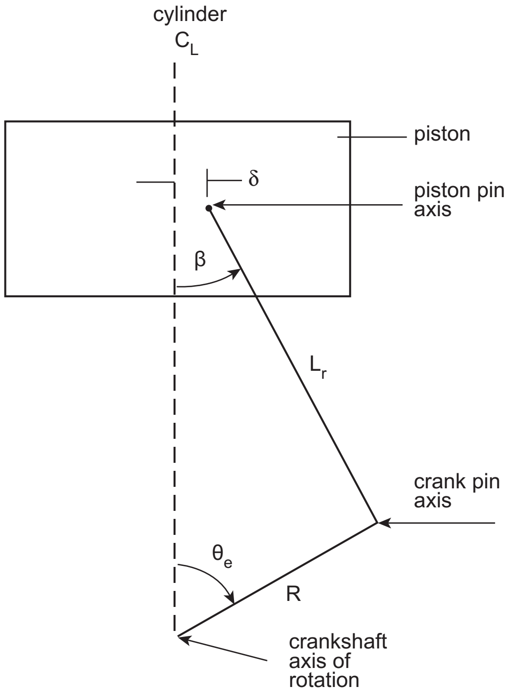


Figure 5.10:
Schematic illustration of cylinder geometry.

to be developed. In Figure 5.10 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 which 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 ($\mathrm{C_L}$) in a plane that is orthogonal to the crankshaft axes of rotation, which benefits torque production. The piston pin offset from the cylinder $\mathrm{C_L}$ is denoted δ in Figure 5.10. 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 down stroke is given by

$$T_i(\theta_e) = \frac{p_c(\theta_e) A R \sin(\theta_e + \beta)}{\cos\beta} \qquad 0 \leq \theta_e < \pi \tag{3}$$

On the up stroke T_i is given by

$$T_i(\theta_e) = \frac{p_c(\theta_e) A R \sin(\theta_e - \beta)}{\cos\beta} \qquad \pi \leq \theta_e < 2\pi \tag{4}$$

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.

The combustion chamber pressure for a representative four-stroke reciprocating IC engine is shown in Figure 5.11 for a complete engine cycle (720° of crankshaft rotation) beginning at −180° (BDC) for the start of compression and ending at 540° (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 to 180°). 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.

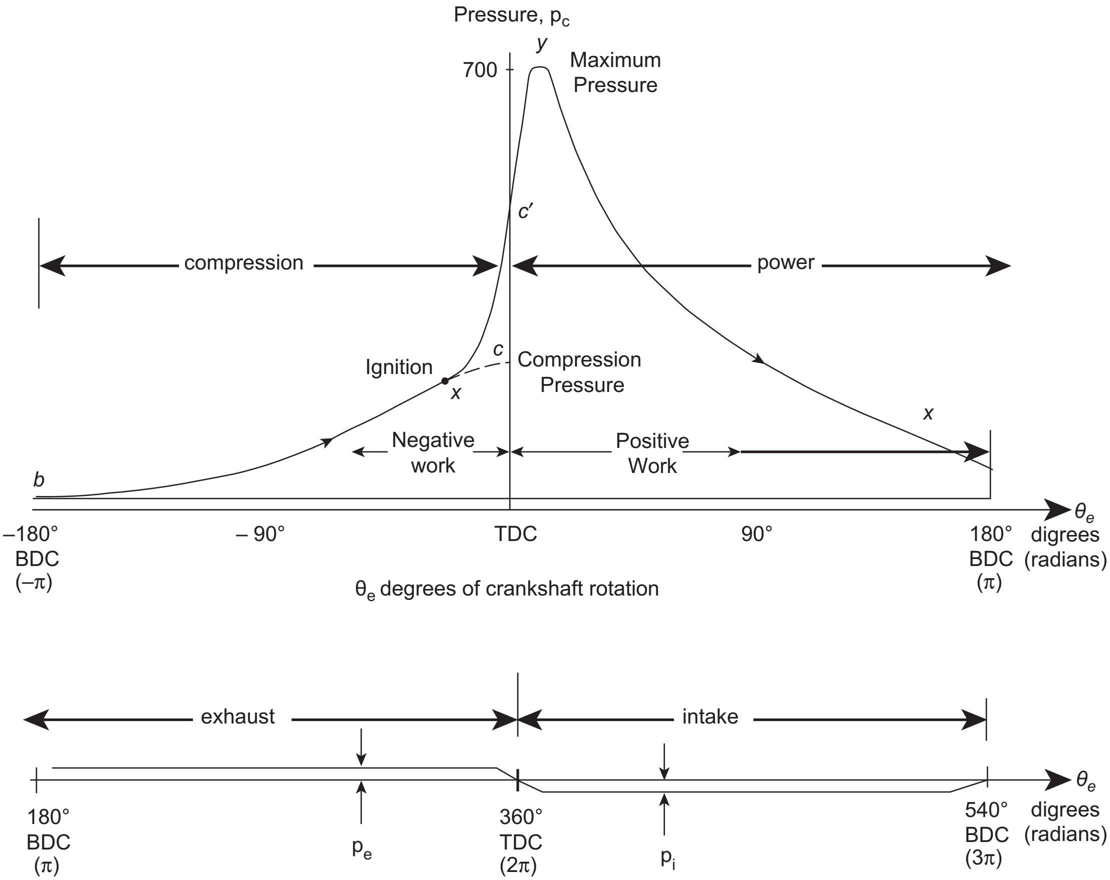


Figure 5.11:
Exemplary plot of $p_c(\theta_e)$.

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 beyond the scope of this book 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.

The indicated torque is the maximum available torque which 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 $\mathrm{p_i}$ 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 $\overline{T}_b$ and is given by

$$\overline{T}_b = \overline{T}_i - \overline{T}_{\mathrm{fp}} \tag{5}$$

where $\overline{T}_{\mathrm{fp}}$ 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 7.

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

$$\omega_e(t) = \frac{\mathrm{d}\theta_e}{\mathrm{d}t} \quad \text{in rad/sec} \tag{6}$$

The units for $P_i(t)$ are Nm/sec (metric) or ft lb/sec (English units). Normally it is the average indicated power averaged over N engine cycles $\overline{P}(N)$ that is useful as a metric for engine available power (with θ_e in radians) is given by:

$$\overline{P}_i(N) = \frac{1}{4\pi N} \int_0^{4\pi N} P_i(\theta_e)\mathrm{d}\theta_e \quad N = \text{integer} \tag{7}$$

The appropriate unit for P_i is kW, although in the USA the popular unit (with the driving public) remains horsepower (Hp), where 1Hp = 0.75 kW and 550 ft lb/s.

The engine output power at the crankshaft is known as the brake power (P_b) since historically 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, e.g., to friction and pumping of the intake mixture and exhaust gases. Generally, the cycle-averaged friction and pumping power are combined and denoted $\overline{P}_{\mathrm{fp}}$. The brake power P_b is given by

$$P_b = \overline{P}_i - \overline{P}_{\mathrm{fp}} \tag{8}$$

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

$$T_b = P_b/\omega_e \tag{9}$$

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 Eqn (2)) divided by displacement volume V_D. As in the case of torques, it is convenient to consider the indicated *mep* (*imep*) which is defined as

$$imep = \frac{W_i}{V_D} \tag{10}$$

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*):

$$fmep = \frac{W_f}{V_d} \tag{11}$$

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

$$bmep = \bar{i}mep - fmep$$

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/hr or lb/hr) 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

$$\text{BSFC} = \frac{\dot{f}}{P_b} \tag{12}$$

The unit for BSFC is kg/(kW·hr) or (lb/CHp·hr) 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:

$$V_d = A_p S_c \tag{13}$$

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.

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 as well as exhaust pressure (p_e). Assuming initially that all cylinders receive mixture at identical density the definition of e_v can be expressed by

$$e_v = \frac{2\dot{M}_i}{NV_d\rho_i} \tag{14}$$

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 which 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 which 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:

$$\begin{aligned}\dot{V}_i &= \frac{\dot{M}_i}{\rho_i}\\ &= \frac{\dot{M}_a}{\rho_a}\end{aligned} \tag{15}$$

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:

$$p_i = p_a + p_f + p_w \tag{16}$$

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

$$\rho_k = \frac{p_k}{RT_i} \tag{17}$$

where R is the perfect gas law constant. Also, it can be shown that

$$\frac{p_a}{p_i} = \frac{M_a/m_a}{\left[\dfrac{M_a}{m_a} + \dfrac{M_f}{m_f} + \dfrac{M_w}{m_w}\right]} \tag{18}$$

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

$$\rho_a = \left(\frac{p_i}{RT_i}\right) \Bigg/ \left[1 + F_i\left(\frac{m_a}{m_f}\right) + \frac{m_a}{m_w}h\right] \tag{19}$$

where $F_i = \dfrac{M_f}{M_a}$ 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 as well as 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 Figure 5.3). 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), $\dot{M}_f = \dfrac{\mathrm{d}}{\mathrm{d}t}M_f$ 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. Typically, these correct values are found from a table look-up 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 portion of the control.

The majority of present-day engines deliver fuel by means of an individual fuel injector associated with each cylinder. A fuel injector is essentially an electromechanical valve to which fuel, under pressure, is supplied. Chapter 6 explains the configuration and operation of fuel injectors. As will be explained later, each fuel injector 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 7. 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

Figure 5.12 illustrates the variation in the performance variables of indicated torque (T_i) BSFC as well as 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

BSHC = brake-specific HC concentration

$$= \frac{r_{\mathrm{HC}}}{P_b} \tag{20}$$

BSCO = brake-specific CO concentration

$$= \frac{r_{\mathrm{CO}}}{P_b} \tag{21}$$

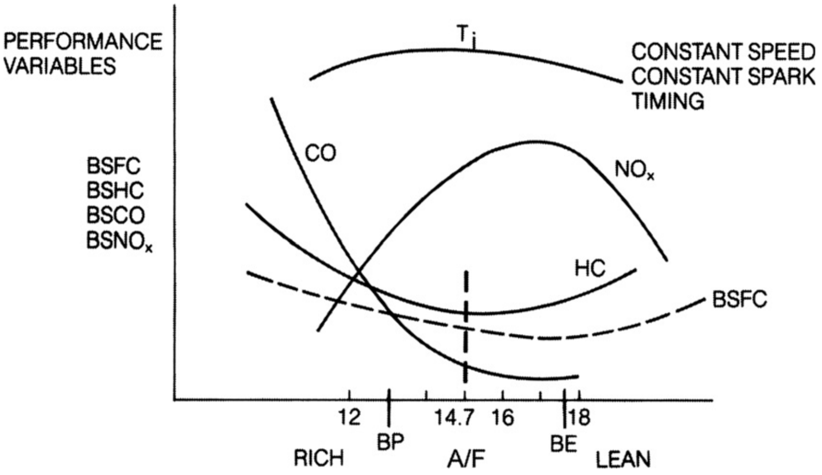


Figure 5.12:
Typical variation of performance with a variation in air/fuel ratio.

BSNOx = brake-specific NOx concentration

$$= \frac{r_{\mathrm{NOx}}}{P_b}$$

where r_{HC} is the HC rate of flow, r_{CO} the CO rate of flow, r_{NOx} the NOx rate of flow, and P_b the brake power. The indicated torque is denoted T_i in Figure 5.12.

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 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:

$$\lambda = \frac{(\mathrm{air/fuel})}{(\mathrm{air/fuel}\,@\,\mathrm{stoichiometry})}$$

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 Figure 5.12 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 NOx 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. Figure 5.13 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 NOx and HC generally increase with increased advance of spark timing. BSFC and torque are also strongly

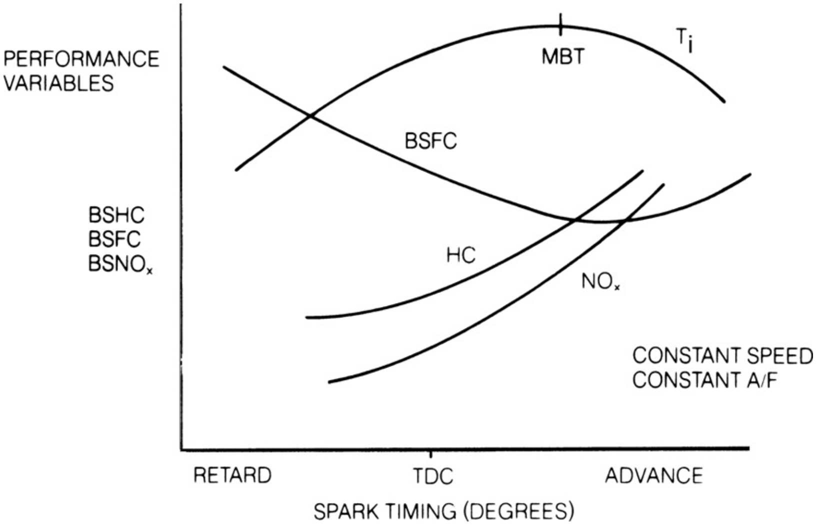


Figure 5.13:
Typical variation of performance with spark timing.

influenced by timing. Figure 5.13 shows that maximum torque occurs at a particular advanced timing (called advance 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 Exhaust Gas Recirculation 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 NOx can be significantly reduced while maintaining a relatively high level of torque. This new control variable, *exhaust gas recirculation* (EGR), consists of recirculating a precisely controlled amount of exhaust gas into the intake. The engine control configuration depicted in Figure 5.5 shows that exhaust gas recirculation is a major subsystem of the overall control system. Its influence on emissions is shown in Figures 5.14 and 5.15 as a function of the percentage of exhaust gas in the intake. Figure 5.14 shows the dramatic reduction in NOx emission when plotted against air/fuel ratio, and Figure 5.15 shows the effect on performance variables as the percentage of EGR is increased. Note that the emission rate of NOx is most strongly

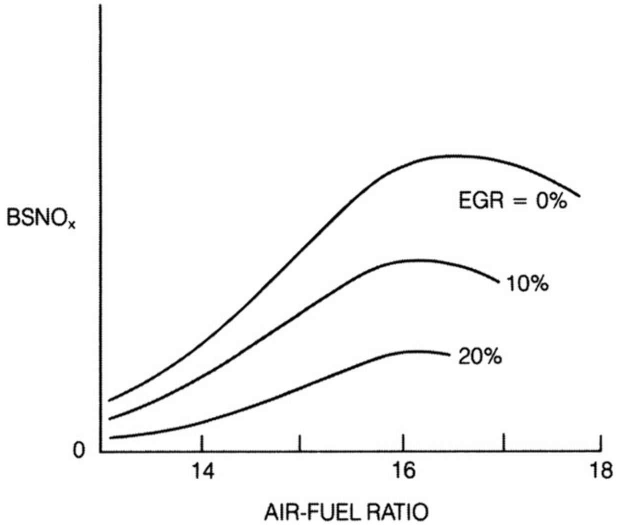


Figure 5.14:
NO$_x$ emission as a function of air/fuel ratios at various EGR%.

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 NOx reduction and HC increase is possible in which the benefits of EGR on NOx reduction far offset the adverse effect on HC emissions. This compromise amount of EGR varies with engine configuration.

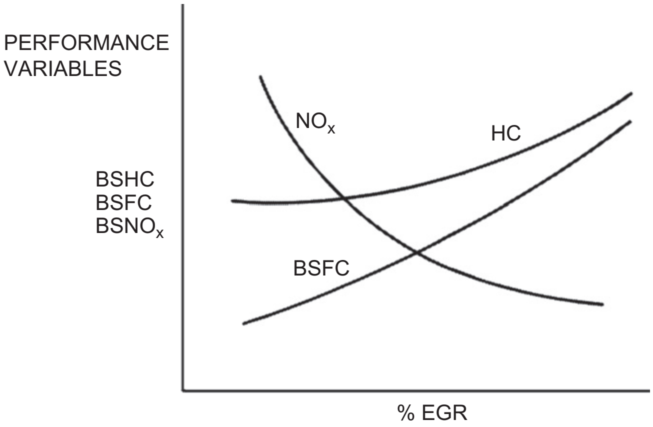


Figure 5.15:
Influence of EGR on brake-specific performance variables.

The mechanism by which EGR affects NOx production is related to the peak combustion temperature. Roughly speaking, the NOx 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 NOx 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 U.S. 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 (Figure 5.16) 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.

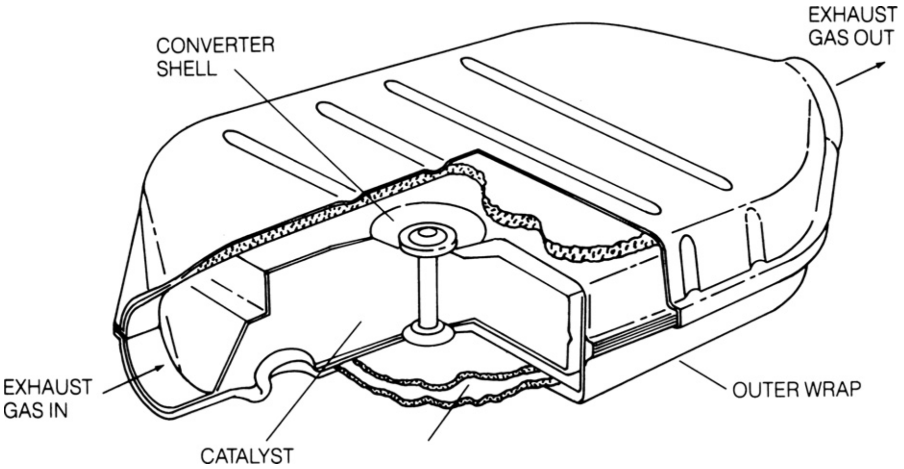


Figure 5.16:
Catalytic converter configuration.

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,

$$\eta_c = \frac{M_o}{M_{ic}} \tag{22}$$

where M_o is the mass flow rate of gas that has been oxidized leaving the converter and M_{ic} is the mass airflow rate of gas into the converter

The conversion efficiency of the OC depends on its temperature. Figure 5.17 shows the conversion efficiency (expressed as a percent) of a typical OC for both HC and CO as functions of temperature. Above about 300 degrees C, the efficiency approaches 98–99% for CO and more than 95% for HC.

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 NOx 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 three-way catalyst uses a specific chemical design to reduce all three major emissions (HC, CO, and NOx) by approximately 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 NOx conversion efficiency is high,

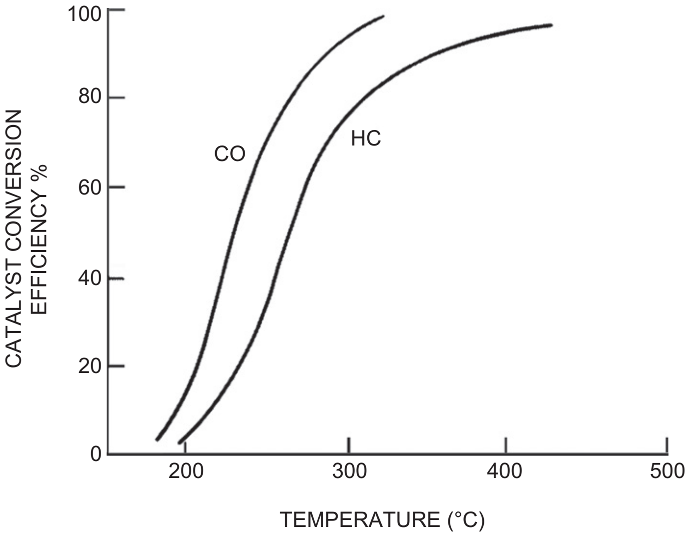


Figure 5.17:
Oxidizing catalyst conversion efficiency versus temperature.

corresponds to a very low conversion efficiency for HC and CO and vice versa. However, as shown in Figure 5.18, there is a very narrow range of air/fuel ratio (called the window and shown as the lined region in Figure 5.18) in which an acceptable compromise exists between NOx 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 is 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 mass airflow rate 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 miles.

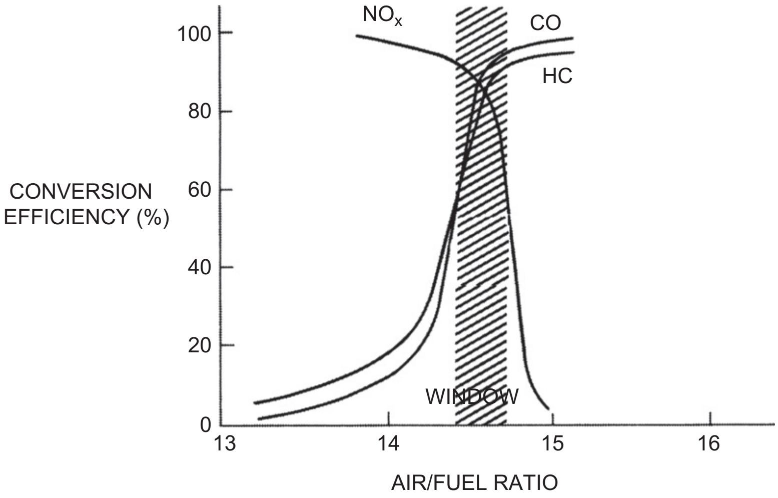


Figure 5.18:
Conversion efficiency of a TWC vs. air/fuel.

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 includes the accelerator pedal position which ultimately determines the throttle position. 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 Figure 5.19. The primary function of this fuel-control system is to determine the mass airflow rate 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). 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)

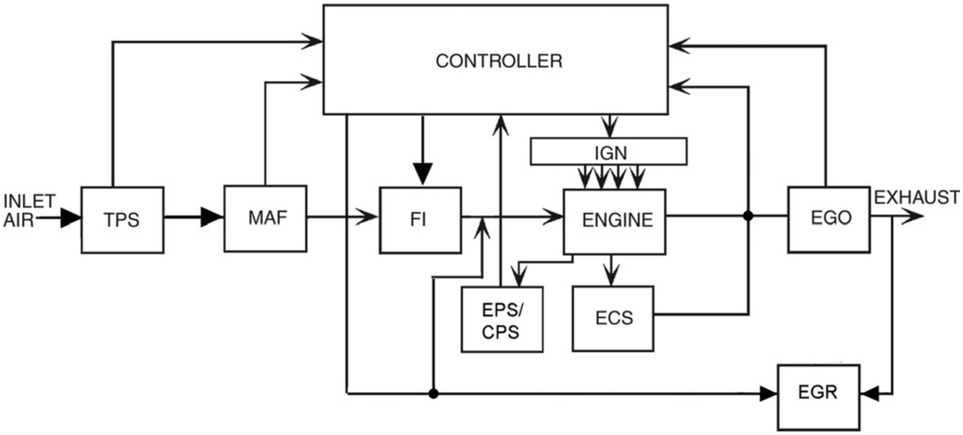


Figure 5.19:
Electronic fuel-control configuration.

5. exhaust gas oxygen sensor (EGO)
6. engine coolant sensor (ECS)
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) as well as crankshaft angular position when it is used in conjunction with a stable and precise electronic clock (in the controller) as explained in Chapter 6. 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 fuel injectors (FI). In addition, optimum ignition timing is determined and appropriate timing pulses are sent to the ignition control module (IGN).

The intake air passes through the individual pipes of the intake manifold to the various cylinders. The set of fuel injectors (one for each cylinder) are each normally located near the intake valve within the corresponding cylinder. As explained in Chapter 6, each fuel injector 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 fuel injector characteristics as well as 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 fuel injector valve is open. This time is, in turn, computed in the engine controller to achieve the desired air/fuel ratio. Typically, the fuel injector 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 fuel injector dynamic transients response. The control system generates a pulsed electrical signal of sufficient amplitude to open the fuel injector 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

$$\begin{aligned} M_f(k) &= \int_{t_{k,n}}^{t_{k,n}+\tau_k} \dot{M}_f \mathrm{d}t \\ &\cong \dot{M}\left(t_{k,n}\right)\tau_k \end{aligned} \tag{23}$$

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.

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 below.

The electronic fuel-control system operates in two modes: open-loop and closed-loop. Recall the concepts for open-loop and closed-loop control as explained in Chapter 1. In the open-loop mode (also called feedforward), the mass airflow rate $(\dot{M}_a)$ 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 closed-loop 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 6, 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 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 closed-loop 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 open-loop and partly closed-loop. The open-loop 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 closed-loop 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 as well as ambient air temperature and pressure. These conditions represent a very small fraction of the overall engine-operating times. They are discussed in Chapter 7, 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 7. 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 7. Once the engine starts and until a specific set of conditions is satisfied, the engine control operates in the open-loop 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 two minutes depending upon ambient conditions and the type of sensor used [see Chapter 6]), the EGO sensor signal is input to the controller and the system begins closed-loop operation.

Open-Loop Control

Fuel control for an electronically controlled engine operates open-loop any time the conditions are not met for closed-loop operation. Among many conditions (which are discussed in detail in Chapter 7) for closed-loop 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 open-loop 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. The mass airflow rate 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:

$$M_f = r_{fa} M_a \tag{24}$$

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 .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 coolant temperature. 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 fuel injector, an engine control system could operate open-loop at all times. In practice, owing to errors in the calculation of M_a, variations in manufactured components, as well as to factors such as wear, the open-loop 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 closed-loop mode for as much of the time as possible. Compensation for the open-loop mode variations above is possible via adaptive closed-loop control as explained in Chapter 7.

Closed-Loop Control

Referring to Figure 5.20, the control system in closed-loop 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

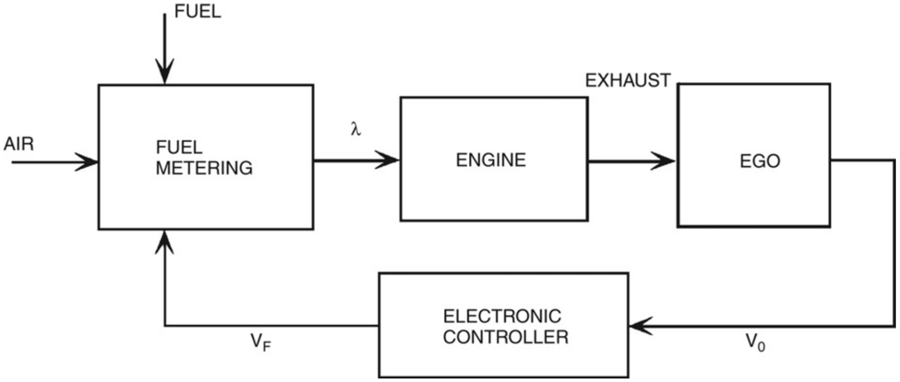


Figure 5.20:
Simplified typical closed-loop fuel-control system block diagram.

a feedback signal for the controller input that depends on the exhaust gas oxygen concentration. This concentration is a function of the intake air/fuel during the intake portion of the same cycle which is 1.5 crankshaft revolutions earlier than the time at which the EGO sensor output is measured.

One closed-loop control scheme that has been used in practice (i.e., limit-cycle control) results in the air/fuel ratio cycling around the desired set point of stoichiometry. 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 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.

Exhaust gas oxygen concentration

The EGO sensor, which provides feedback, will be explained in Chapter 6. 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 Eqn (25):

$$V_o(t) = V_1 - V_2 \mathrm{sgn}[\lambda(t - t_d) - 1] \tag{25}$$

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 which is about three-quarters of the period of an engine cycle 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 6) 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 fuel injector during the *k*th engine cycle ($M_f(k)$) is proportional to the time interval ($\tau_F(k)$) of its binary-valued control electrical signal $V_F(k)$ (see Figure 5.20). In our present example controller, the fuel injector (i.e., actuator) control voltage is given by

$$\begin{aligned} V_F(k) &= V & t_{k,n} \le t < t_{k,n} + \tau_F(k) \\ &= 0 & t_{k,n} + \tau_F(k) < t < t_{k+1,n} \end{aligned} \tag{26}$$

where $t_{k,n}$ is the injection time during *k*th engine cycle for the *n*th cylinder.

The fuel injector duration $\tau_F(k)$ consists of an open-loop component $\tau_o(k)$ plus a closed-loop component τ_{Fc}:

$$\tau_F(k) = \tau_o(k) + \tau_{Fc}(k) \tag{27}$$

where $\tau_o(k)$ is calculated based on $\dot{M}_a$ measurement.

Closed-Loop Operation

In the present example, the closed-loop 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:

$$\tau_{Fc}(k) = \tau_{Fc}(k-1) + \delta\tau \; \mathrm{sgn}[\lambda(k) - 1] \tag{28}$$

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 open-loop 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 mass airflow rate $\dot{M}_a$ measurements. We illustrate the operation of this example fuel-control system in Figure 5.20.

Lambda is used in the block diagram of Figure 5.20 to represent the equivalence ratio at the intake manifold. The exhaust gas oxygen 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 Figure 5.21. 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 fuel injector. Increasing this duration causes more fuel to be delivered, thereby

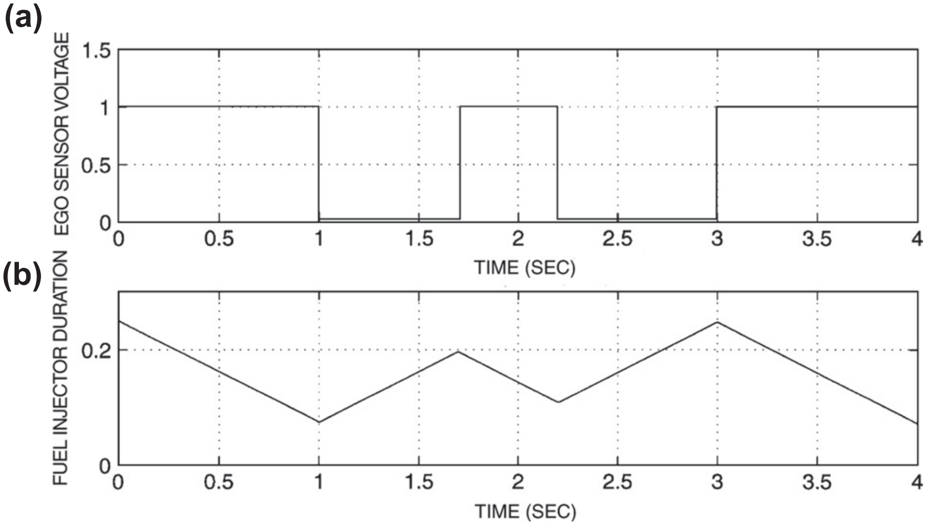


Figure 5.21:
Simplified waveforms in a closed-loop fuel-control system.

causing the mixture to become richer. Correspondingly, decreasing this duration causes the mixture to become leaner. Figure 5.21b shows the fuel injector signal duration.

In Figure 5.21a 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 (Figure 5.21b) to decrease during this interval. At time 1 second, 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 EGO sensor switches high, causing the controller to decrease the fuel injector actuating interval. The process continues this way, cycling back and forth between rich and lean around stoichiometry.

The engine controller continuously computes the desired fuel injector actuation duration and maintains the current value in memory. At the appropriate time in the intake cycle, the controller reads the value of the fuel injector duration and generates a pulse of the correct duration to activate the proper fuel injector 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 $\overline{\mathbf{V}}_{\text{EGO}}$ should be halfway between the rich and the lean values:

$$\overline{\mathbf{V}}_{\text{EGO}} = \frac{V_{\text{rich}} + V_{\text{lean}}}{2} \tag{29}$$

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 7 explains this adaptive control in more detail than is given here.

Frequency and deviation of the fuel controller

Recall from Chapter 1 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 Figure 5.21, 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 Figure 5.22 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 ±1.0. Although the air/fuel ratio is constantly swinging 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 7.

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.

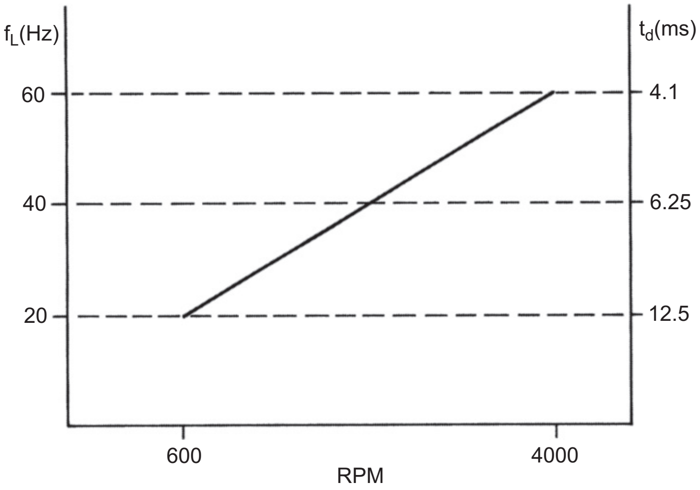


Figure 5.22:
Typical limit-cycle frequency versus RPM.

The preceding discussion applies only to a simplified idealized fuel-control system. Chapter 7 explains the operation of practical electronic fuel-control systems in which the calculation of fuel injector duration is done numerically in a microprocessor-based engine control system.

Analysis of Intake Manifold Pressure

As explained earlier, fuel control is based on a measurement of mass airflow rate 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.

Figure 5.23 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 (MAP) is at atmospheric pressure. This is the highest intake MAP for a non-supercharged 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.

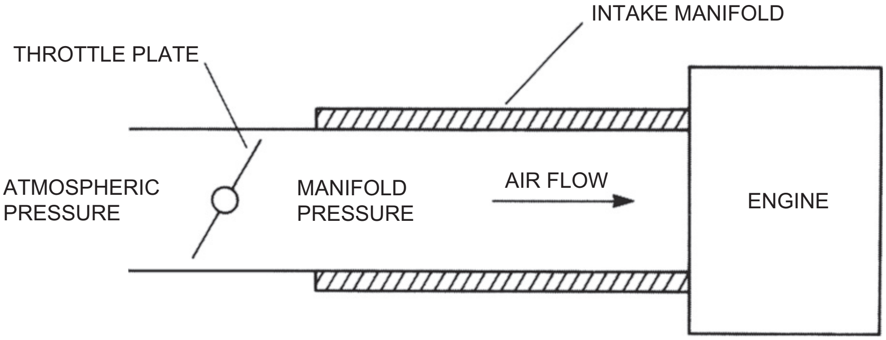


Figure 5.23:
Simplified intake system configuration.

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

$$f_p = \frac{N \times \mathrm{RPM}}{120}$$

Figure 5.24 shows manifold pressure fluctuations qualitatively as well as average MAP.

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 Chapter 1

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

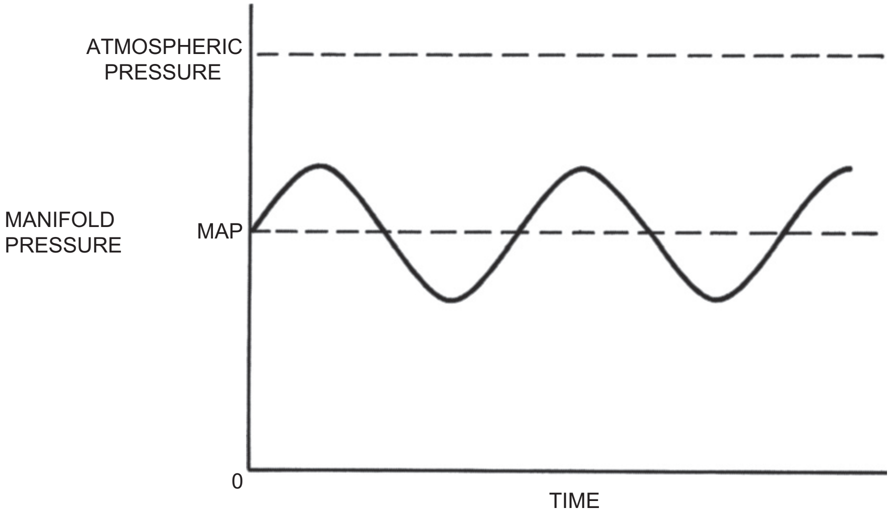


Figure 5.24:
Intake manifold pressure fluctuations.

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 uniformly).

There are two methods of determining the mass flow rate of air into the engine. One method uses a single sensor that directly measures mass airflow rate. The operation of this sensor is explained in Chapter 6. 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

The concept for this method is based on the mass density of air as illustrated in Figure 5.25a.

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

$$\rho_a = \frac{M_a}{\mathrm{V}} \tag{30}$$

This concept can be extended to moving air, as depicted in Figure 5.25b. 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.

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. Earlier, it was shown that the mass airflow rate $\dot{M}_a$ in the engine is given by

$$\dot{M}_a = \dot{\mathrm{V}}_i \rho_a \tag{31}$$

where ρ_a is the density of the mixture of fuel air and water vapor:

$$\rho_a = \frac{p_a}{RT_i} \tag{32}$$

with p_a being manifold absolute pressure, T_i being inlet air absolute temperature, and R being constant for and 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 6, 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

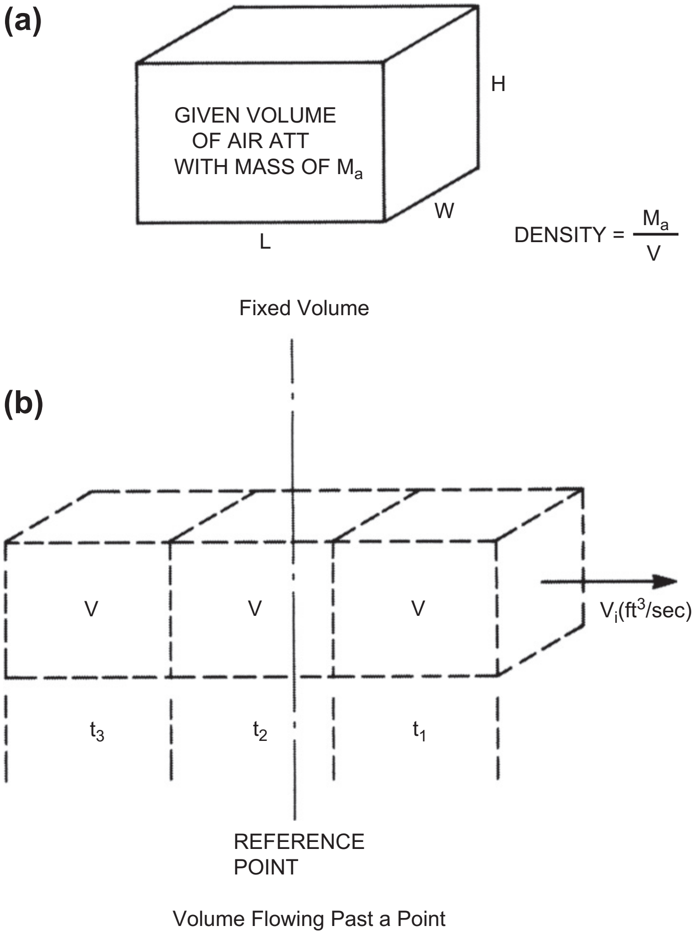

(a)
GIVEN VOLUME OF AIR ATT WITH MASS OF M_a
H
W
L
DENSITY = $\frac{M_a}{V}$
Fixed Volume
(b)
V
V
V
V_i(ft^3/sec)
t_3
t_2
t_1
REFERENCE POINT
Volume Flowing Past a Point

Figure 5.25:
Volume flow rate calculation.

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

$$\dot{V}_i = e_v \frac{N}{2} V_D \tag{33}$$

where

$$N = \frac{\text{RPM}}{60}$$

V_D is the displacement, and e_v 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 vs. 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 look-up, 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 mid-range RPMs as well as 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. Variable valve timing is also called variable valve phasing (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 6. There it is shown that either (or both) intake and exhaust valve timing is varied relative to the engine cycle. The control strategy for regulating VVP is explained in Chapter 7.

In essence, the exhaust valve is open primarily during the exhaust stroke and that 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 6 and 7.

Including EGR

Calculating $\dot{V}_i$ is relatively straightforward in a computer-based control system. Another factor must be taken into account in determining mass airflow rate. Exhaust gas recirculation 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 $(\dot{V}_{\mathrm{EGR}})$ from $\dot{V}_i$. The total cylinder air charge rate $(\dot{V}_a)$ is thus given as follows:

$$\dot{V}_a = \dot{V}_i - \dot{V}_{\mathrm{EGR}} \tag{34}$$

The volume flow rate of EGR is known from the position of the EGR valve and from engine-operating conditions, as explained in Chapter 7.

Substituting the equation for $\dot{V}_a$, the volume flow rate of air is

$$\dot{V}_a = \frac{NV_D e_{\mathrm{V}}}{2} - \dot{V}_{\mathrm{EGR}} \tag{35}$$

Knowing $\dot{V}_a$ and the density ρ_a gives the mass flow rate of air $\dot{M}_a$ as follows:

$$\dot{M}_a = \dot{V}_a \rho_a \tag{36}$$

Knowing $\dot{M}_a$ the stoichiometric mass flow rate for the fuel, $\dot{M}_f$, can be calculated as follows:

$$\dot{M}_f = \frac{\dot{M}_a}{14.7} \tag{37}$$

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 electrical signal to operate the fuel injectors 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 as well as 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 as well as switching electrical 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 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 mass airflow rate. Chapter 6 discusses various actuators having application for idle airflow control. For the present discussion, we assume a model for the idle mass airflow rate that is representative of the practical actuator configurations discussed in Chapter 6. (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

$$T_{iI} = K_I \dot{M}_{aI} \tag{38}$$

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:

$$\dot{M}_{aI} = K_m x_I \tag{39}$$

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 as well as 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:

$$m_I \ddot{x}_I + d_I \dot{x}_I + k_I x_I = K_a u \tag{40}$$

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

$$\Omega_I = \pi \frac{\mathrm{RPM}_I}{30} \tag{41}$$

where

$$\mathrm{RPM}_I = \mathrm{RPM\ at\ idle}$$

In general for relatively small changes in Ω_I, the load torques (including friction 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:

$$T_i \cong J_e \dot{\Omega}_I + T_L(\Omega) \tag{42}$$

where J_e is the moment of inertia of engine and load rotating components.

Using the Laplace transform methods of Chapter 1, it is possible to obtain the engine transfer function at idle $H_{eI}(s)$:

$$H_{eI}(s) = \frac{\Omega_I(s)}{T_i(s)} \tag{43}$$

$$= \frac{1}{J_e s + R_e} \tag{44}$$

Similarly, the transfer function for the idle speed actuator dynamics $H_{aI}(s)$ is given by

$$\begin{aligned} H_{aI}(s) &= \frac{x_I(s)}{u(s)} \\ &= \frac{K_a}{m_I(s^2 + 2\zeta_I \omega_I s + \omega_I{}^2)} \end{aligned} \tag{45}$$

where

$$\omega_I = \sqrt{k_I/m_I}$$

$$\zeta_I = \frac{d_I}{2m_I\omega_I}$$

These transfer functions can be combined to yield the transfer function (in standard form) of the idle speed control "plant" $H_{pI}(s)$:

$$H_{\mathrm{p}I}(s) = \frac{\Omega_I(s)}{u(s)} \tag{46}$$

$$= \frac{K_a K_m K_I}{J_e m_I\left[\left(s^2 + 2\zeta\omega_I + \omega_I\right)\ \left(s + \frac{R_e}{J_e}\right)\right]} \tag{47}$$

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 as well as parameter changes due to variations in operating environmental conditions. On the other hand, closed-loop control is well suited to regulating idle speed to a desired value. Figure 5.26 is a block diagram of such an idle speed control system.

Using the analysis procedures of Chapter 1 and denoting the idle speed set point Ω_s, it can be shown that the idle speed control closed-loop transfer function H_{CLI} is given by

$$\begin{aligned} H_{CLI}(s) &= \frac{\Omega_I(s)}{\Omega_s(s)} \\ &= \frac{H_{cI}(s)H_{pI}(s)}{1 + H_s(s)H_{cI}(s)H_{pI}(s)} \end{aligned} \tag{48}$$

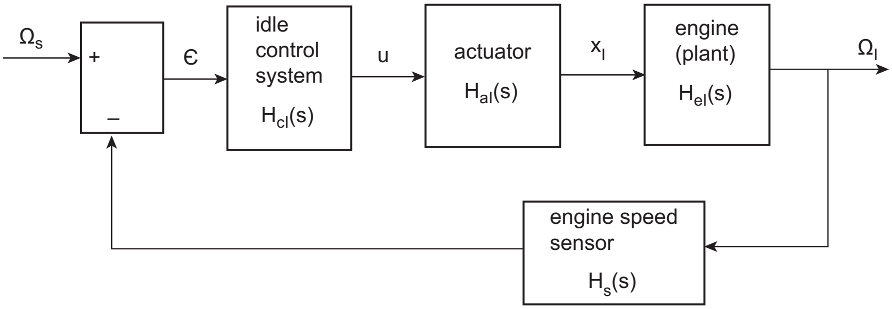


Figure 5.26: Idle speed control system block diagram.

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 Chapter 1, 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 Chapter 1 that a proportional—integral (PI) control had zero steady-state error but could potentially yield an unstable closed-loop 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

$$H_{cI}(s) = K_p + \frac{K_I}{s} = K_p\left(\frac{s+s_0}{s}\right) \tag{49}$$

For the purpose of illustrating exemplary idle speed control performance, we assume the following set of parameters:

$$\begin{aligned}
\zeta_I &= 0.5 \\
\omega_I &= 25 \text{ rad/sec} \\
\omega_e &= R_e/J_e = 10\text{rad/sec} \\
K_{\text{num}} &= K_a K_m K_I = 250 \\
K_{\text{den}} &= J_e m_I = 0.05 \\
s_0 &= K_I/K_p = 10
\end{aligned}$$

The forward transfer function $H_F(s)$ is defined by the following expression:

$$\begin{aligned}
H_F(s) &= H_{cI}(s)H_{pI}(s) \\
&= \frac{K_{\text{num}}(s+s_0)}{K_{den}[(s^3+2\zeta\omega_I s^2+\omega_I^2 s)(s+\omega_e)]}
\end{aligned} \tag{50}$$

The present analysis is simplified by assuming a perfect angular speed sensor such that $H_s(s) = 1$. In this case, the closed-loop idle speed control transfer function ($H_{CLI}(s)$) is given by

$$H_{CLI}(s) = \frac{K_p H_F(s)}{1+K_p H_F(s)} \tag{51}$$

The influence of proportional gain on stability of this closed-loop idle speed control can be evaluated via root locus techniques as explained in Chapter 1. Figure 5.27 is a plot of the root locus for this idle speed control with the assumed parameters.

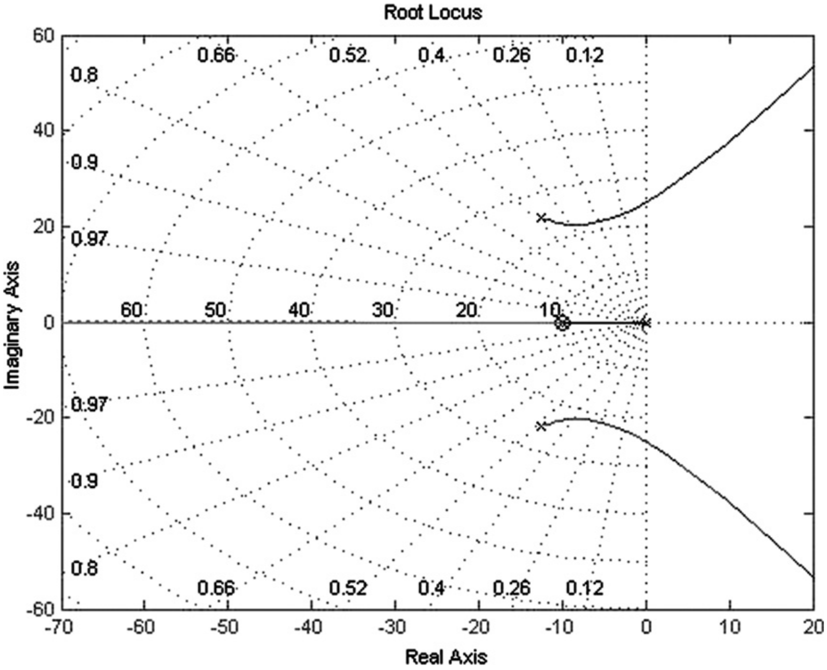


Figure 5.27:
Root locus for idle speed control.

It can be seen from this figure that the closed-loop 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%.

Using this value for K_p (i.e., $K_p = 1.2$), the closed-loop dynamic response for the system was examined by commanding a step change in RPM from an initial 550 RPM to 600 RPM at $t = 0.5$ s. Figure 5.28 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 7 where digital engine (powertrain) control is discussed, a discrete time control is modeled.

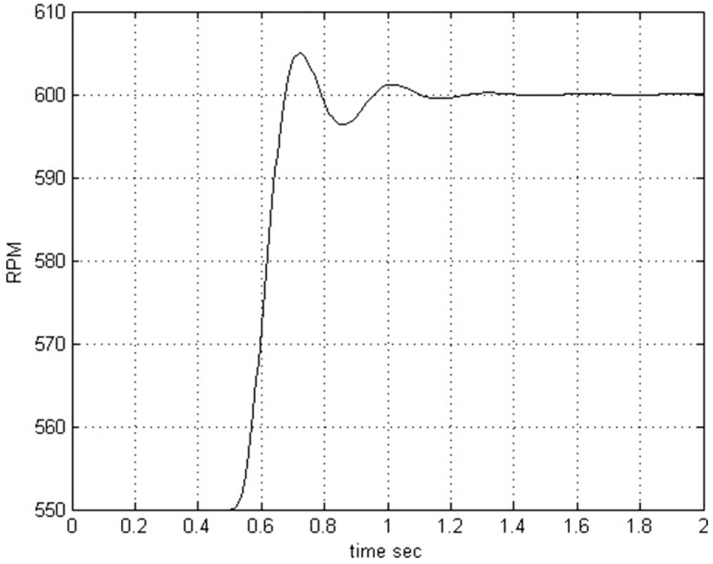


Figure 5.28:
Step response of idle speed control.

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 Chapter 1). The spark advance (relative to TDC) is determined in the electronic engine control based on a number of measurements made by sensors. As will be explained in Chapter 7, 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.

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 Figure 5.29.

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

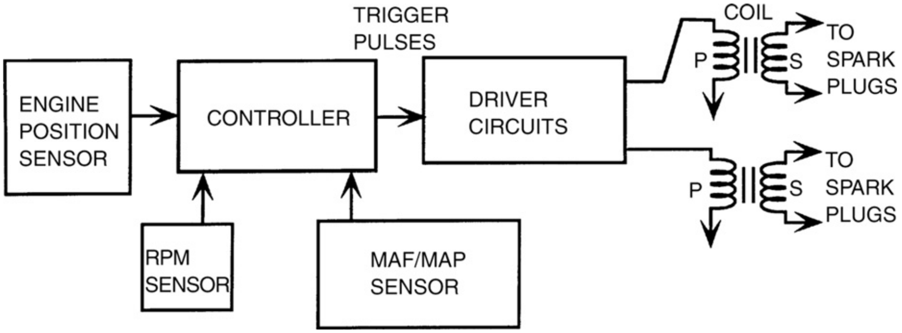


Figure 5.29:
Electronic ignition system configuration.

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 6. 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 powertrain, however, it is necessary to explain and develop models for the critically important components of a control system: sensors and actuators. Chapter 6 is devoted to these important components.

This page intentionally left blank

CHAPTER 6

Sensors and Actuators

Chapter Outline

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-08-097097-4.00006-0

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 powertrain (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 microcontroller. 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.

Recall from Chapter 1 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 Chapter 1) 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 can also 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 automotive 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 powertrain control. To motivate the discussion of engine control sensors and actuators, it is helpful to review the variables measured (sensors) and the controlled variables (actuators). Figure 6.1 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 5, 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 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).

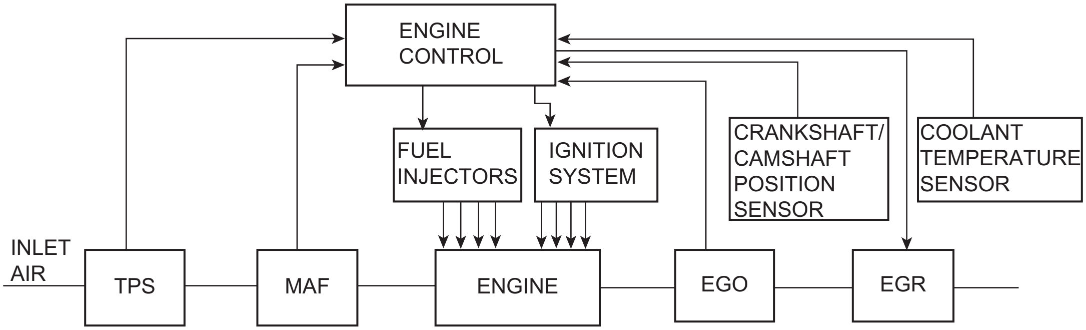


Figure 6.1:
Representative electronic engine control system.

Variables to be Measured

The set of variables sensed for any given powertrain is specific to the associated engine control configuration. Space limitations for this book preclude a complete survey of all powertrain control systems and relevant sensor and actuator selections for all car models. Nevertheless, it is possible to review a superset 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 superset 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, and
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, and
5. transmission gear selection.

Airflow Rate Sensor

In Chapter 5, 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. (Recall from Chapter 1 that the 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 that a 4-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) superposed on a quasi-steady 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 quasi-steady component of airflow changes with load and speed. It is this quasi-steady component of $\dot{M}_a(t)$ that is measured by the MAF for engine control purposes. One way of characterizing this quasi-steady state component is as a short-term time average over a time interval τ (which we denote $\dot{M}_{a\tau}(t)$) where

$$\dot{M}_{a\tau}(t) = \frac{1}{\tau} \int\limits_{t-\tau}^{t} \dot{M}_a(t')\mathrm{d}t' \tag{1}$$

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 quasi-steady component of mass airflow can be represented by a low-pass-filtered version of the instantaneous flow rate. Recall from Chapter 1 that a low-pass filter (LPF) can be characterized (in continuous time) by an operational transfer function ($H_{\mathbf{LPF}}(s)$) of the form

$$H_{\mathrm{LPF}}(s) = \frac{b_o + b_1 s + \cdots b_m s^m}{a_o + a_1 s + \cdots a_n s^n} \tag{2}$$

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 powertrain 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 Chapter 2).

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 as well as 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 is mounted a honeycomb flow straightener 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 air stream (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 as well as 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 air stream, thereby producing a change in its resistance. A convenient model for the sensing element resistance (R_{SE}) at temperature (T) is given by

$$R_{\mathrm{SE}}(T) = R_o + K_T \Delta T \tag{3}$$

where R_o is the resistance at some reference temperature T_{ref} (e.g., 0 °C), $\Delta T = T - T_{\mathrm{ref}}$, and K_T is the resistance/temperature coefficient. For a conducting sensing element, $K_T > 0$, and for a semiconducting sensing element, $K_T < 0$.

The mass flow rate of the moving air stream 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 Figure 6.2.

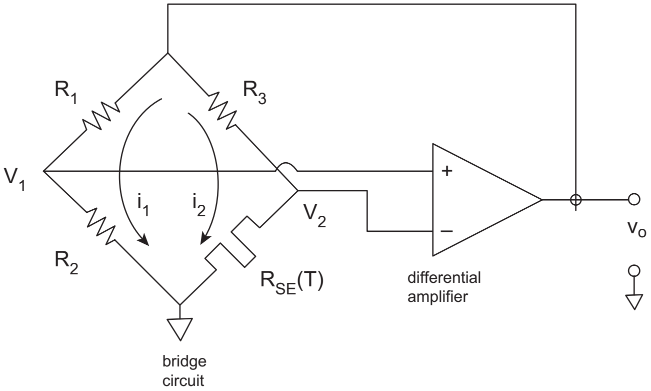


Figure 6.2:
Mass airflow sensor.

In the bridge circuit, three resistors (R_1, R_2, and R_3) are connected as depicted in Figure 6.2 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 Figure 6.2) are connected to the inputs of a relatively high-gain differential amplifier. The output voltage of this amplifier $\mathrm{v_o}$ is connected to the bridge (as shown in Figure 6.2) and provides the electrical excitation for the bridge. This voltage is given by

$$\mathrm{v}_o = G(V_1 - V_2) \tag{4}$$

where G is the amplifier voltage gain

In this bridge circuit, only that sensing element is placed in the moving air stream 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 regards the moving air.

The combination bridge circuit and differential amplifier form 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 °C). 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

$$\Delta V = V_1 - V_2 \tag{5}$$

$$= \mathrm{v}_0 \left[\frac{R_2}{R_1 + R_2} - \frac{R_{\mathrm{SE}}}{R_{\mathrm{SE}} + R_3}\right] \tag{6}$$

However, it has been shown that $\mathrm{v_o} = G\Delta V$, so the following equation can be shown to be valid:

$$\frac{1}{G} = \left[\frac{R_2}{R_1 + R_2} - \frac{R_{\mathrm{SE}}}{R_{\mathrm{SE}} + R_3}\right] \tag{7}$$

In the present MAF sensor configuration, it is assumed (as is often found in practice) that $G >> 1$. For sufficiently large G, from Eqn (6.7), we can see that R_{SE} is given approximately by

$$R_{\mathrm{SE}}(T) = \frac{R_2 R_3}{R_1} \tag{8}$$

In this case, it can be shown using Eqn (3) that the temperature difference between the sensing element and the ambient air is given approximately by

$$k_T \Delta T = \frac{R_2 R_3}{R_1} - [R_0 + k_T(T_a - T_{\text{ref}})] \tag{9}$$

where T_{ref} is an arbitrary reference temerature.

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

$$R_3(T_a) = R_{3o} + k_{T3}(T_a - T_{\text{ref}}) \tag{10}$$

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

$$k_T \Delta T = \left(\frac{R_2 R_{3o}}{R_1} - R_0\right) + \left(\frac{R_2}{R_1} k_{T3} - k_T\right)\left(T_a - T_{\text{ref}}\right) \tag{11}$$

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

$$\Delta T = \frac{1}{k_T}\left[\frac{R_2 R_{3o}}{R_1} - R_o\right] \tag{12}$$

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., $\dot{M}_a$ = constant). The mass airflow into an automotive engine is rarely constant, so it is useful to consider the MAF sensor dynamic response to time-varying $\dot{M}_a$. The combination bridge circuit and differential amplifier has essentially instantaneous dynamic response to changes in $\dot{M}_a$. The dynamic response of the MAF of Figure 6.2 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

$$\Delta \dot{T} + \frac{\Delta T}{\tau_{\text{SE}}} = \alpha_1 P_{\text{SE}} - \alpha_2 \dot{M}_a \tag{13}$$

where $P_{\text{SE}} = i_2^2 R_{\text{SE}}$:

$$= \left(\frac{\text{v}_o}{R_{\text{SE}} + R_3}\right)^2 R_{\text{SE}}$$

In equation 13, i_2 = current shown in Figure 6.2
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 Chapter 1 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. In this case, the change in power dissipation from the zero airflow condition is given by

$$\alpha_1[P_{SE}(\dot{M}_a) - P_{SE}(0)] = \alpha_2 \dot{M}_a \tag{14}$$

It can be shown from Eqn (14) that MAF sensor output voltage varies as given below:

$$\mathrm{v}_o(\dot{M}_a) = [\mathrm{v}_o^2(0) + K_{MAF}\dot{M}_a]^{1/2} \tag{15}$$

where K_{MAF} is the constant for the MAF configuration.

As an example of this variation, Figure 6.3 is a plot of the sensor voltage vs. 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 Figure 6.3 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 7, in which digital engine control is discussed, it is necessary to

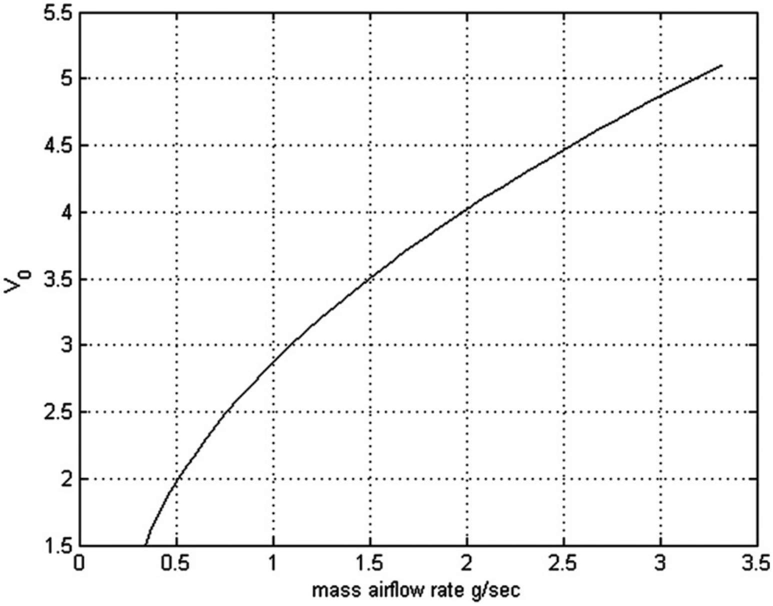


Figure 6.3:
Output voltage for example MAF vs. mass flow rate g/s.

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 4). The engine control system can calculate $\dot{M}_a$ from v_o using the known functional relationship $v_o(\dot{M}_a)$.

Pressure Measurements

There are numerous potential applications for 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. In essentially 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 pressure (MAP). Although it is obsolete in contemporary vehicles, the speed−density method (discussed in Chapter 5) of calculating mass airflow in early emission regulation vehicles used such an MAP sensor.

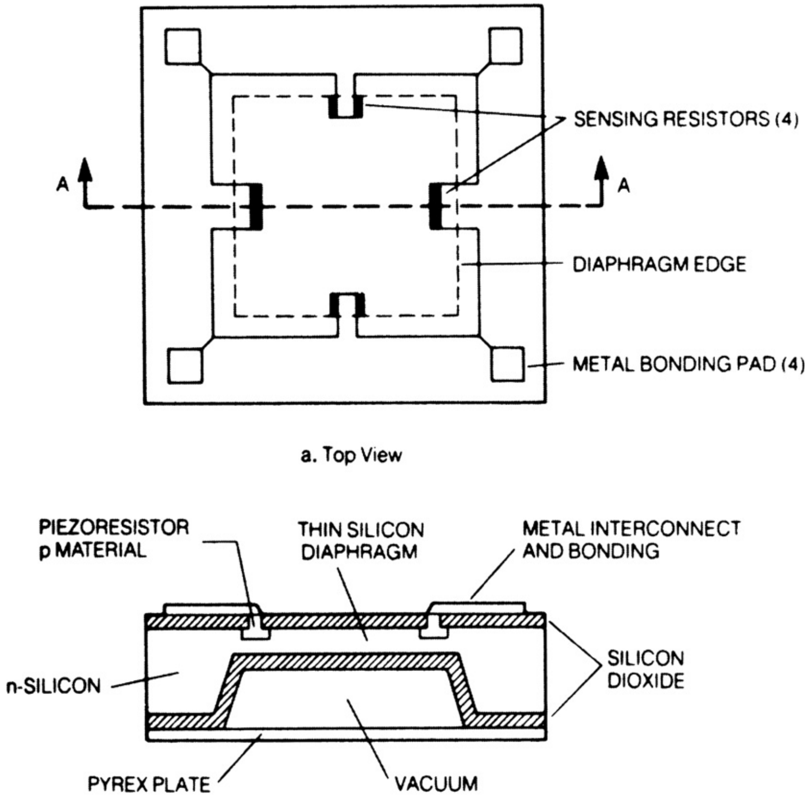
Strain gauge MAP sensor

One relatively inexpensive MAP sensor configuration is the silicon-diaphragm diffused strain gauge sensor shown in Figure 6.4. This sensor uses a silicon chip that is approximately 3 millimeters square. Along the outer edges, the chip is approximately 250 μm ($1\ \mu\text{m} = 10^{-6}$ 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 Figure 6.4. 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, $\delta L << L$. The elongation is quantitatively represented by its strain $\in$, which is given by

$$\in = \frac{\delta L}{L} \tag{16}$$



a. Top View

b. Section A-A

Figure 6.4:
Exemplary manifold pressure sensor configuration.

In any diaphragm made from a linear material the strain is proportional to the applied pressure (p):

$$\in = K_D p \tag{17}$$

where K_D is a constant which is determined by the diaphragm configuration (e.g., its shape and area exposed to p as well as 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) 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, as shown in the schematic of Figure 6.5a. 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 Figure 6.5a. 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 Figure 6.5b.

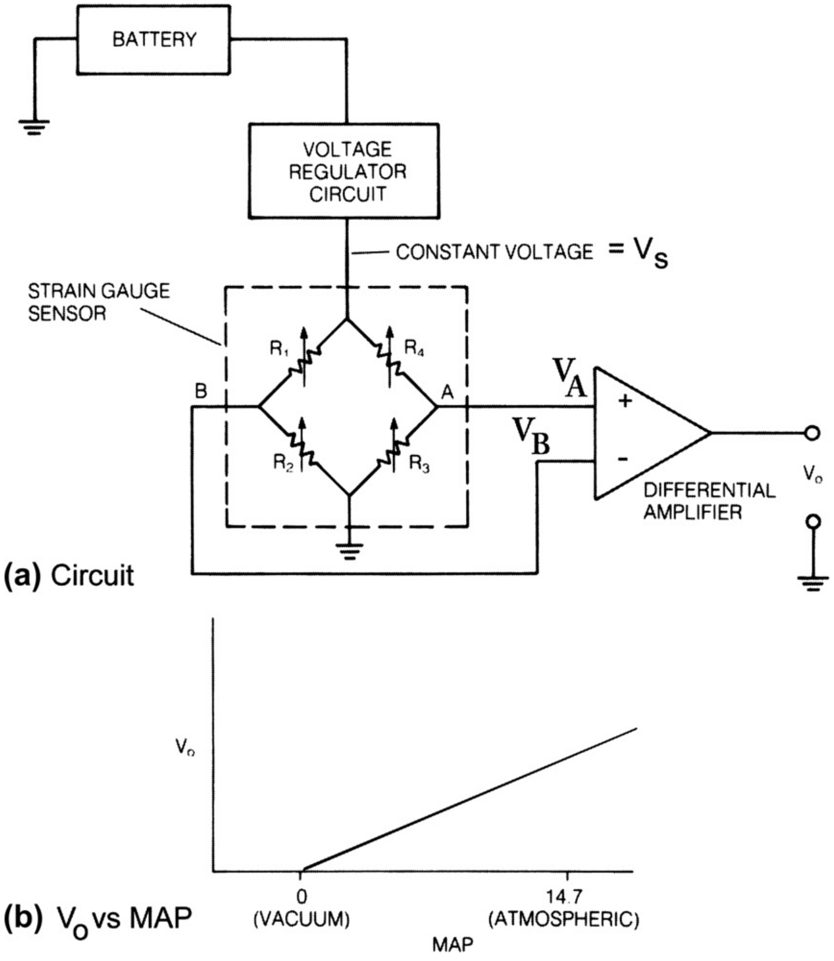


Figure 6.5:
Example MAP sensorr circuit.

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

$$R_n(\in) = R_o + R_\in \in \quad n = 1, 3 \tag{18}$$

where

$$R_\in = \left.\frac{\mathrm{d}R}{\mathrm{d}\in}\right|_{\in = o} > 0 \tag{19}$$

For resistances R_2 and R_4, the model for resistance is given by

$$R_m(\in) = R_o - R_\in \in \quad m = 2, 4 \tag{20}$$

The voltages V_A and V_B are given respectively by

$$V_A = V_S\left(\frac{R_3}{R_3 + R_4}\right) = V_S\frac{(R_o + R_\in \in)}{2R_o} \tag{21}$$

$$V_B = V_S\left(\frac{R_2}{R_2 + R_4}\right) = V_S\frac{(R_o - R_\in \in)}{2R_o} \tag{22}$$

The voltage difference $V_A - V_B$ is given by

$$V_A - V_B = V_S\frac{R_\in \in}{R_o} \tag{23}$$

The differential amplifier output voltage (V_o) is given by

$$V_o = G_A(V_A - V_B) \tag{24}$$

$$= G_A\frac{V_S R_\in \in}{R_o} \tag{25}$$

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 Chapter 2).

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.

The principles involved in measuring rotating shafts can be illustrated by 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 Figure 6.6. 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 Figure 6.6. 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 zero degrees. For our purposes, assume that this angular position corresponds to the No. 1 cylinder at TDC (top dead center) on either intake or power strokes. As the crankshaft rotates, this angle increases from zero to 360° 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 zero to 720°. During each cycle, it is important to measure the crankshaft position

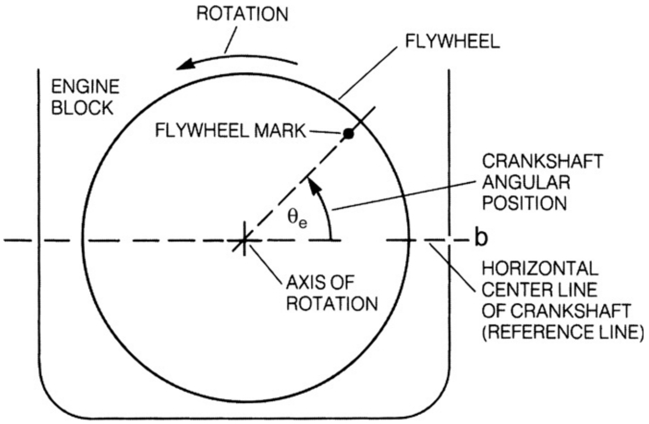

ROTATION
FLYWHEEL
ENGINE BLOCK
FLYWHEEL MARK
CRANKSHAFT ANGULAR POSITION
θ_e
b
AXIS OF ROTATION
HORIZONTAL CENTER LINE OF CRANKSHAFT (REFERENCE LINE)

Figure 6.6:
Illustration of crankshaft angular position representation.

relative to the reference for each cycle in each cylinder. This information is used by the electronic engine controller to set ignition timing and, in most cases, to set the fuel injector pulse timing.

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 zero to 360° for one complete engine cycle. Either 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 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 as well as crankshaft position. Typically, it is sufficient to sense camshaft position at one point in a complete revolution. At the present time, there appears to be a trend toward measuring crankshaft position directly rather than indirectly via 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 Figure 6.7.

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 Figure 6.7 for illustrative purposes, the steel disk has four protruding tabs, which is the minimum number of tabs for an 8-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 on its power stroke, although other reference positions are also possible. The crankshaft position θ at all other times in the engine cycle are given by

$$\theta - \theta_n = \int_{t_n}^{t} \omega(t)\mathrm{d}t \qquad t_n < t < t_{n+1} \quad (26)$$

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

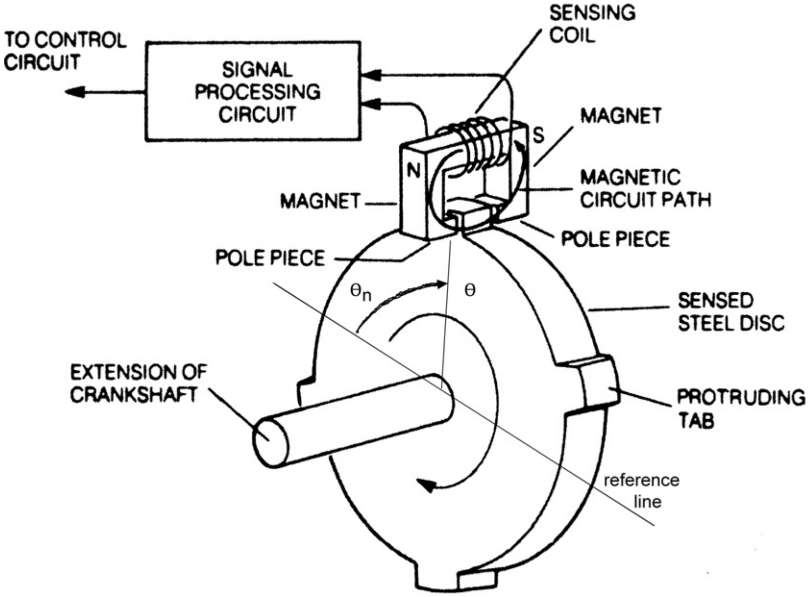


Figure 6.7:
Magnetic crankshaft angular position sensor configuration.

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 Figure 6.7 (as well as any magnetic sensor) incorporates one or more components of its structure which 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 to the voltage and current of an ordinary electric circuit. One of these quantities is called the *magnetic field intensity vector* $\overline{H}$. 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* $\overline{B}$, which is analogous to current. In these two quantities, the over bar 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.

Figure 6.8 illustrates the functional relationship of the scalar magnitudes $B(H)$ for a typical ferromagnetic material having a configuration such as is depicted in Figure 6.7.

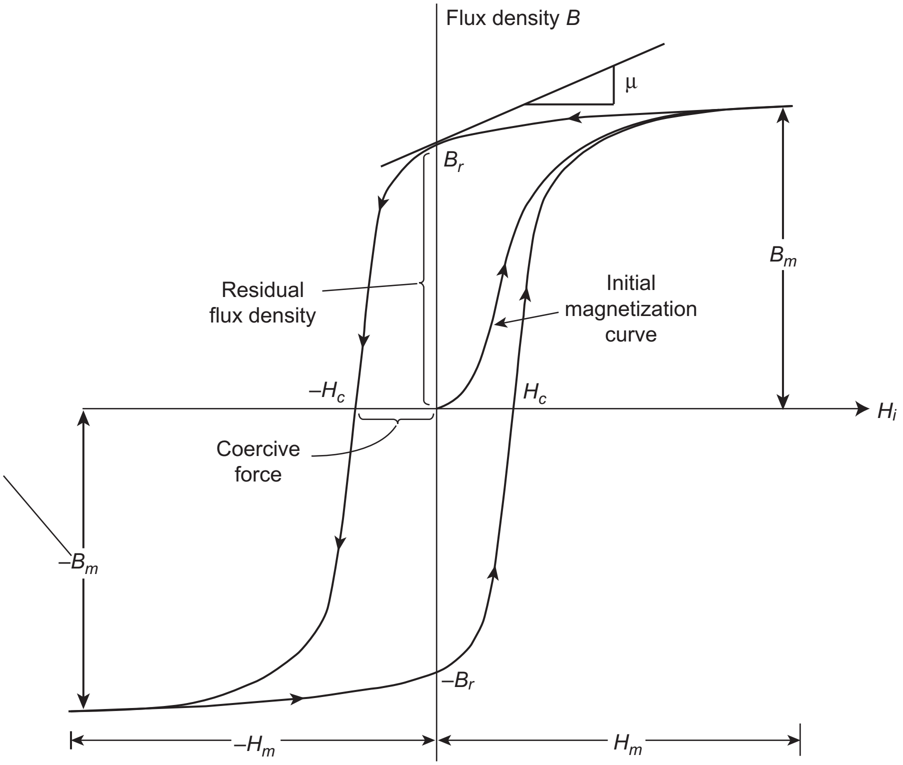


Figure 6.8:
Magnetization curve for exemplary ferromagnetic material.

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 Figure 6.8 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

$$M = \frac{B}{\mu_o} - H_i \tag{27}$$

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 Figure B-1. If the applied field is reduced from saturation to zero, the ferromagnetic material has a nonzero flux density denoted B_r in Figure B-1 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 Figure 6.8. 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 Figure 6.7 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 Figure 6.7 operates with small B and H incremental changes about the remanent magnetization such that B is given approximately by

$$B = B_r + \mu H_i \tag{28}$$

where

$$\mu = \left.\frac{\mathrm{d}B}{\mathrm{d}H_i}\right|_{H_i=0} \tag{29}$$

$$= \text{incremental permeability of the ferromagnetic materials}$$

The straight line of Figure 6.8 passing through $B = B_r$, $H_i = 0$ 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 $\mu_r >> 1$.

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 $\overline{H}$ along a closed contour C and is given by

$$\oint_C \overline{H} \cdot \mathrm{d}\overline{\ell} = I_T \tag{30}$$

where I_T is the total current passing normal to and through the surface enclosed by C. This integral equation will be shown to be useful for analyzing magnetic automotive sensors of the type depicted in Figure 6.7.

Another relationship that is useful for developing the model for a magnetic sensor is continuity of the normal component of $\overline{B}$ at the interface of any two materials. This continuity is expressed by the relationship

$$\overline{B}_1 \cdot \hat{n} = \overline{B}_2 \cdot \hat{n} \tag{31}$$

where $\overline{B}_1$ and $\overline{B}_2$ 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 Figure 6.7 and other similar magnetic sensors.

The path for the magnetic flux of the sensor of Figure 6.7 is illustrated in Figure 6.9.

In Figure 6.9, 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 Figure 6.9, the lines of constant magnetic flux follow paths as indicated in the figure. The following notation is used:

$\overline{B}_m$ is the flux density within the ferromagnetic material,
$\overline{H}_m$ is the magnetic field intensity within the ferromagnetic material,
$\overline{B}_g$ is the flux density within air gaps, and
$\overline{H}_g$ is the magnetic field intensity within air gaps.

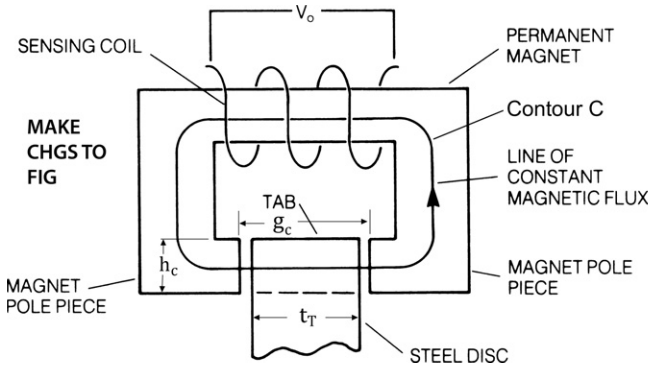


Figure 6.9:
Magnetic circuit of the sensor of Figure 6.7.

From Eqn (30) above, the following equation can be written for the contour shown in Figure 6.9:

$$\int_C \overline{H}\cdot \mathrm{d}\overline{\ell} \cong H_g g_a + H_m L_m \tag{32}$$

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

$$H_g \cong -H_m L_m / g_a \tag{33}$$

From Eqn (31), the following equation can be written for the interface between the ferromagnetic material and the air gap:

$$\overline{B}_g \cdot \hat{n}_g = \overline{B}_m \cdot \hat{n}_m$$

However, since the lines of magnetic flux are normal to this interface

$$\overline{B}_g \cdot \hat{n}_g = B_g$$

and

$$\overline{B}_m \cdot \hat{n}_m = B_m$$

or

$$B_g = B_m \quad \text{(at the interface)}$$

That is, the magnetic flux density for the configuration of Figure 6.9 is constant along the path denoted therein. Within the material, the following relationship is valid:

$$\begin{aligned} B_m &= \mu_o(H_m + M_r) \\ &= B_g \\ &= \mu_o H_g \end{aligned} \tag{34}$$

where M_r is the remanent magnetization of the pole piece. Thus, we can write

$$H_m = H_g - M_r \tag{35}$$

For a magnetized ferromagnetic material $M_r >> H_g$ such that

$$H_g \cong -M_r \tag{36}$$

Combining Eqns (29) and (30), the flux density is given by

$$\begin{aligned} B_g &= \mu_o H_g \\ &= \mu_o \frac{M_r L_m}{g_a} \end{aligned} \tag{37}$$

Eqn (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 Figure 6.8, 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 Figure 6.7, 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 Figure 6.7 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:

$$V_o = N \frac{\mathrm{d}\Phi}{\mathrm{d}t}$$

where

$$\begin{aligned} \Phi &= \int_{A_c} B \mathrm{d}s \\ &= \frac{\mu_o M_r L_m A_c}{g_a} \end{aligned}$$

where $A_c = h_c w_c$

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

$$\Phi \approx \frac{\mu_o M_r L_m h_c \mathrm{w}_c}{g_c} \tag{38}$$

where w_c is the width of the magnet normal to the page.

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

$$\Phi = \frac{\mu_o M_r L_m h_c w_c}{(g_c - t_T)} \tag{39}$$

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 Figure 6.10. 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 Figure 6.8 is known as "reluctance" and is denoted $\Re$, which is given by

$$\Re = \frac{\mu_o L_m h_c w_c}{g_a}$$

Since the air gap g_a varies with the position of the steel disk in the sensor depicted in Figure 6.7, 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.

One of the disadvantages of the inductive type of variable reluctance sensor as depicted in Figure 6.7 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.

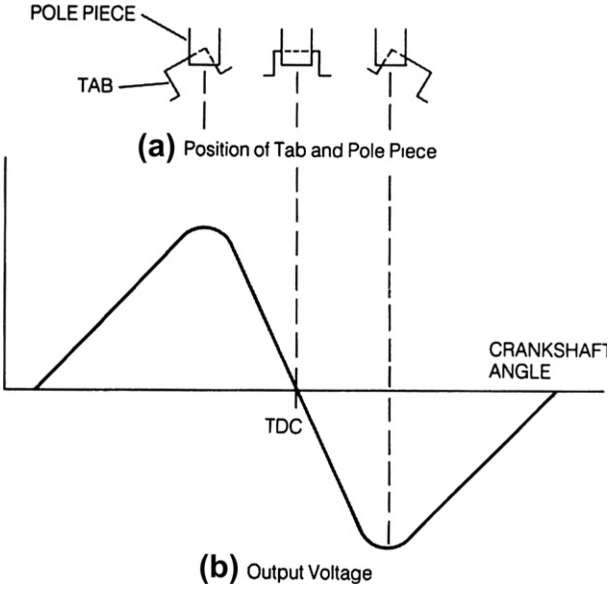


Figure 6.10:
Variable reluctance sensor voltage.

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. Figure 6.11 gives an equivalent circuit for this sensor in which the open circuit voltage source is represented by the voltage waveform of Figure 6.10b. 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

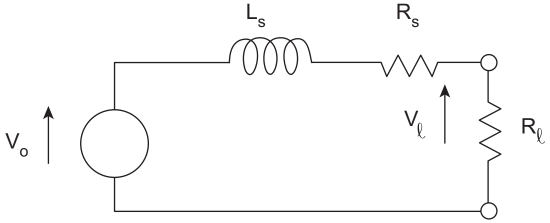


Figure 6.11:
Equivalent circuit for variable reluctance sensor.

sensor of Figure 6.7 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 Figure 6.11. At any sinusoidal frequency ω the approximate phase shift $\varphi(\omega)$ between $\mathrm{v_o}$ and v_ℓ is given by

$$\phi = \tan^{-1}\left(\frac{\omega L_s}{R_s + R_\ell}\right)$$

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 7.

Figure 6.7 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 ½ 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 $\phi(\omega_s)$ 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 Figure 6.7 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 Figure 6.10b. For a running engine, the sensor output consists of a continuous stream of such voltage pulses. We denote the time of the *n*th 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:

$$V_o(t_n) = 0$$

$$\left.\frac{\mathrm{d}V_o}{\mathrm{d}t}\right|_{t=t_n} < 0 \tag{40}$$

The crankshaft angular speed ($\omega_e(t)$ in rad/sec) is given by

$$\omega_e(t) = \frac{2\pi}{M(t_{n+1} - t_n)} \tag{41}$$

where $M =$ number of tabs (four in the example illustrated in Figure 6.6). Thus, a measurement of the time between any pair of successive zero crossings of $\mathrm{v_o}$ 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 4. 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

$$B_c = f_c(t_{n+1} - t_n) \tag{42}$$

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

$$\begin{aligned} V_{\mathrm{ZCD}} &= V_1 \quad t_n \le t \le t_n + \tau_{\mathrm{ZCD}} \\ &= V_2 \quad t_n + \tau_{\mathrm{ZCD}} < t < t_{n+1} \end{aligned} \tag{43}$$

where the time interval $\tau_{\mathrm{ZCD}} << (t_{n+1} - t_n)$ 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 or within the controller/microprocessor (see Chapter 4).

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 (as well as its use in timing fuel delivery and ignition) is described in Chapter 7. 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.

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 Figure 6.12 (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 Figure 6.12, the sensor output voltage resembles that of Figure 6.9b with a polarity reversal; that is, the output voltage satisfies the conditions

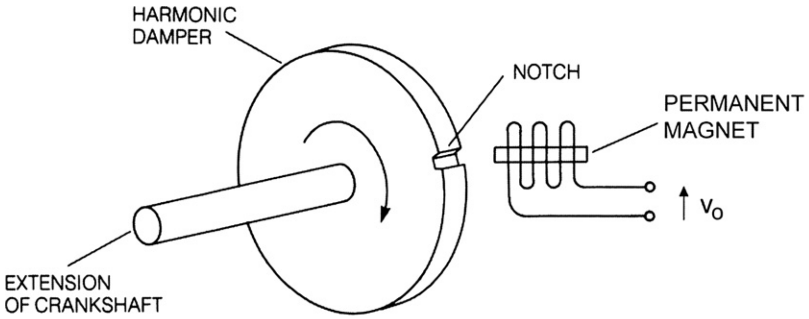


Figure 6.12:
Exemplary camshaft angular sensor configuration.

$$
\begin{aligned}
V_o &< 0 \quad t < T_{\text{notch}} \\
V_o &> 0 \quad t > T_{\text{notch}}
\end{aligned}
$$

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 Figure 6.13. 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 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 Figure 6.13, θ_n is the angle between the reference line and the center of the nth tab as shown.

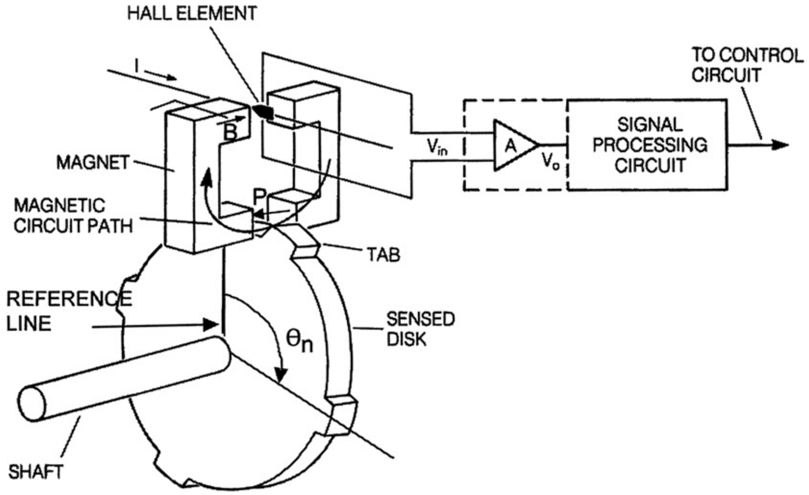


Figure 6.13:
Representative Hall-effect sensor configuration.

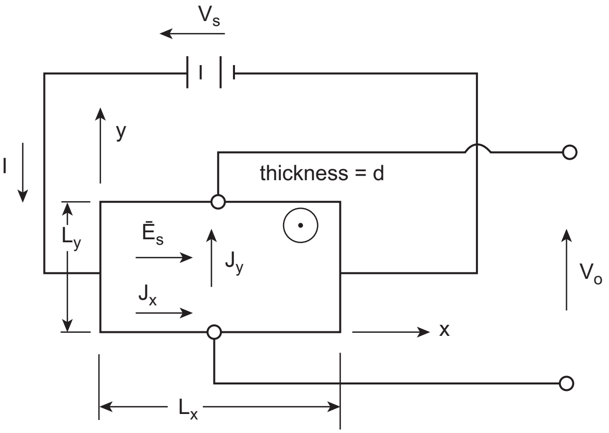


Figure 6.14:
Schematic illustration of Hall-effect sensor.

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. Figure 6.14 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 $\overline{E}_s$ 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., $L_x >> L_y >> d$) it is nearly uniform over much of the sample and given approximately by

$$\overline{E}_s \cong \frac{V_s}{L_x}\hat{x} = E_x\hat{x} \tag{44}$$

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

$$I = \int_o^{L_y}\int_o^{d} J_x \mathrm{d}y\mathrm{d}z \tag{45}$$

where J_x is the current density (i.e., the current/unit area across any y, z plane):

$$J_x = q[nv_{ex} + pv_{hx}] \tag{46}$$

where

$v_{ex} = \mu_e E_x =$ electron drift velocity,
$v_{hx} = \mu_h E_x =$ hole drift velocity

and where n and p are the electron and hole concentrations and 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 $\overline{F}_{\text{Le}}$ (electrons) and $\overline{F}_{\text{Lh}}$ (holes), which are proportional to the vector product of $\overline{B}$ and velocities ($\overline{v}_e$ and $\overline{v}_h$):

$$\overline{F}_{\text{Le}} = q\overline{v}_e \times \overline{B} \tag{47}$$

$$\overline{F}_{\text{Lh}} = q\overline{v}_h \times \overline{B}$$

where $\overline{B} = B_z\hat{z}$ 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 creating a current flow in this direction represented by current density J_y:

$$J_y = q[pv_{hy} + nv_{ey}]$$

If (as is the usual case) the input impedance of the differential amplifier A in Figure 6.13 is extremely large, $J_y \simeq 0$ which means that $pv_{hy} = -nv_{ey}$. 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

$$V_o = \int_{-L_y/2}^{L_y/2} E_y \mathrm{d}x$$

$$\cong E_y L_y = R_H J_x B_z L_y$$

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 Figure 6.13 is based upon the variation of magnetic flux density normal to the Hall element and its relationship to

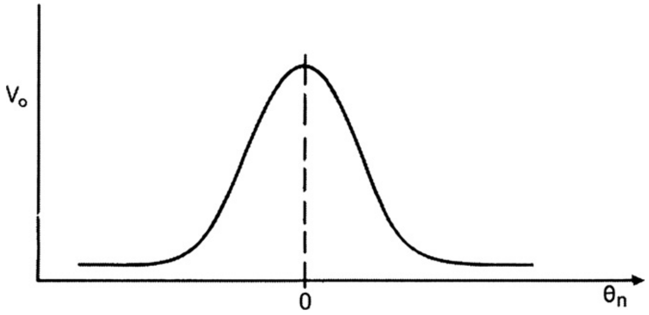


Figure 6.15:
Hall sensor output voltage waveform.

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 Figure 6.13. 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 *n*th tab is denoted θ_n (as shown in Figure 6.13), then the terminal voltage V_o of the sensor has a waveform as depicted in Figure 6.15; 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 Figure 6.15 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, as explained above.

Shielded-field sensor

Figure 6.16a 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 Figure 6.16b that the waveform is just the opposite of the one in Figure 6.15.

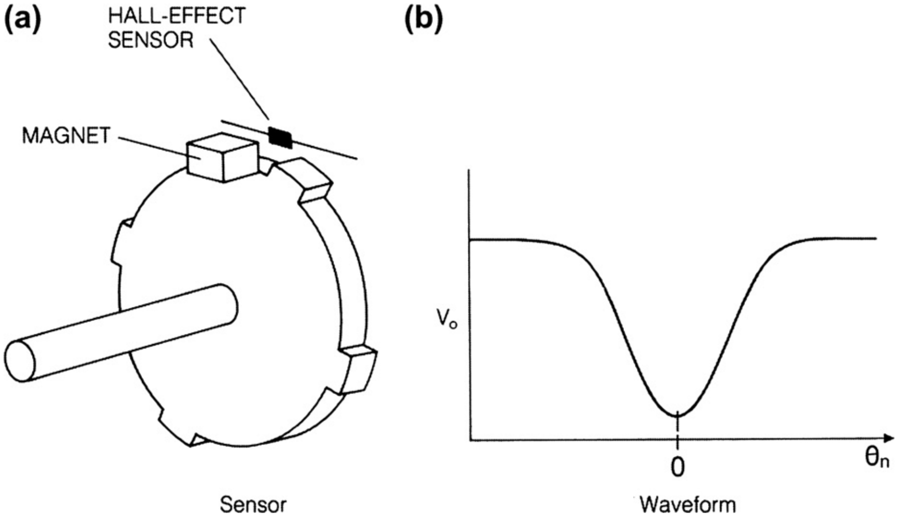


Figure 6.16:
Shielded-field Hall-effect sensor.

Optical Crankshaft Position Sensor

In a sufficiently clean environment, a shaft position can also be sensed using optical techniques. Figure 6.17 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 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 material, fabrication and excitation voltage. There is even now a white light LED.

The other important component of the optical position sensor of Figure 6.17a 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

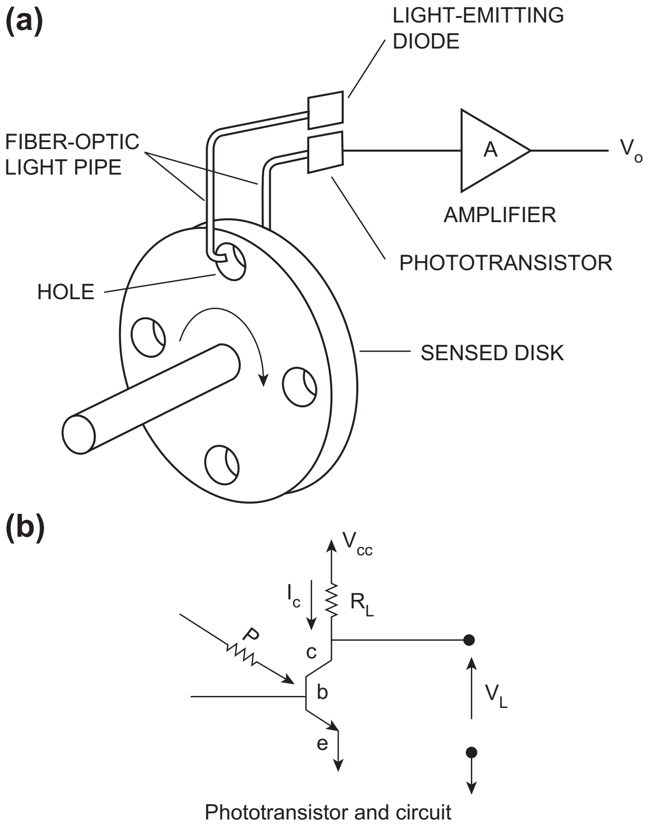


Figure 6.17:
Optical angular position sensor.

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 3) determines the wavelength of light to which the phototransistor responds.

Figure 6.17b 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 3, 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

$$I_c = I_o + \beta\gamma P \tag{48}$$

where β = grounded emitter current gain
γ = conversion constant from light intensity to equivalent base current.

The load voltage V_L is given by

$$\begin{aligned} V_L &= V_{cc} - I_c R_L \\ &= V_{cc} - R_L(I_o + \beta\gamma P) \end{aligned} \tag{49}$$

Each time a hole in the disk passes the fiber-optic light path depicted in Figure 6.17a, 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 7. Most throttle angle sensors are essentially potentiometers. A *potentiometer* consists of a resistor with a movable contact, as illustrated in Figure 6.18.

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., $\sqrt{A} << L$), the current density is essentially uniform across the cross-sectional area. The current density of a current flowing through this area J is related to the electric field intensity E along the conductor long axis by

$$J = \sigma E \tag{50}$$

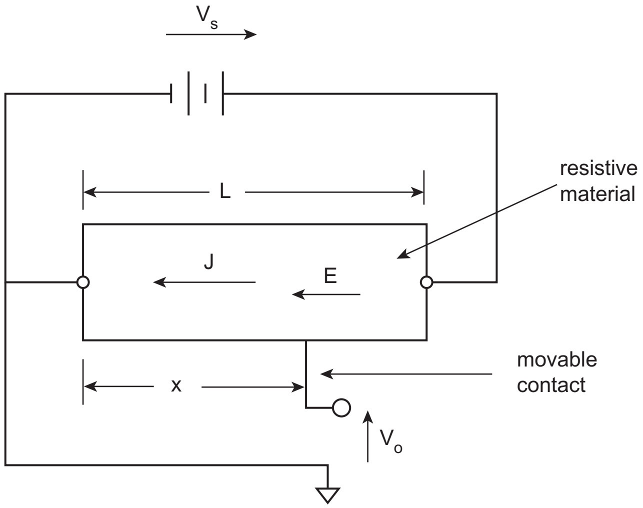


Figure 6.18:
Potentiometer schematic circuit

where σ is the conductivity of the material. The total current through the conductor I for uniform J is given by

$$\begin{aligned} I &= \int_A J \mathrm{d}s \\ &\cong JA \end{aligned} \tag{51}$$

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

$$V = -\int_o^L E\mathrm{d}x \cong EL \tag{52}$$

where the x coordinate is along the long axis. The voltage relative to ground (at $x = 0$) varies linearly with position x:

$$V(x) = \frac{V_S x}{L} \quad 0 \leq x \leq L \tag{53}$$

The resistance R of this conductor is defined as

$$\begin{aligned} R &= \frac{V}{I} \\ &= \frac{EL}{\sigma EA} \end{aligned} \tag{54}$$

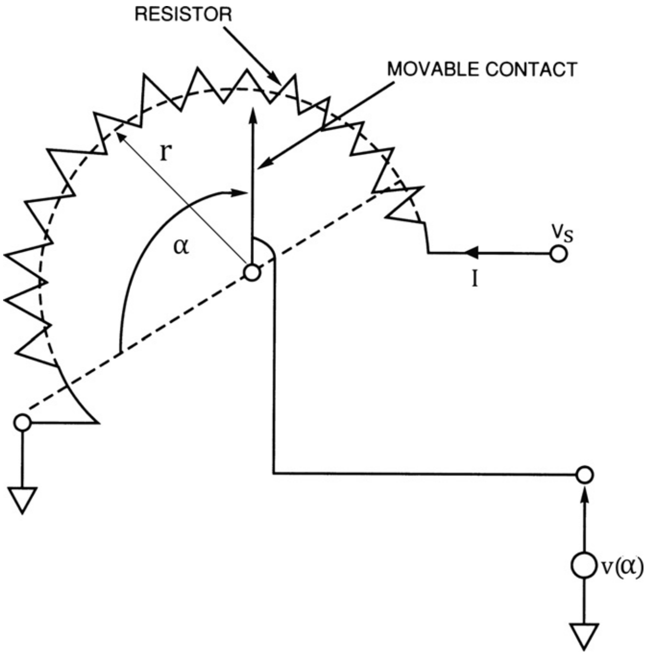


Figure 6.19:
Throttle angle sensor: a potentiometer.

$$R = \frac{L\rho}{A} \tag{55}$$

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 Figure 6.19. Let the radial dimension and the thickness of the material be uniform and small compared to 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 Figure 6.19 is known as a rotary potentiometer (or just as a potentiometer). Let the total resistance from the end of the material which 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

$$R(\alpha) = \frac{R_p \alpha}{\alpha_{\max}} \tag{56}$$

The current I flowing into this potentiometer is given by

$$I = \frac{V_s}{R_p} \tag{57}$$

The open circuit voltage at the movable contact $V(\alpha)$ is given by

$$\begin{aligned} V(\alpha) &= IR(\alpha) \\ &= \frac{V_s}{R_p} R(\alpha) \\ &= V_s \frac{\alpha}{\alpha_{\max}} \end{aligned} \tag{58}$$

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 Figure 6.19 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.

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 Figure 6.20, 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 which, in turn, depend upon the difference in energy between the

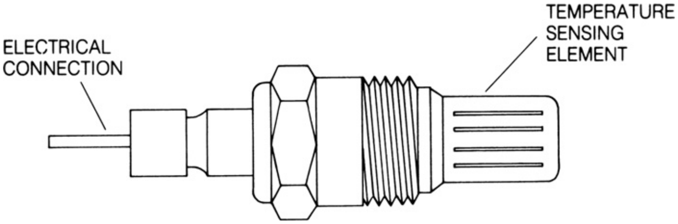


Figure 6.20:
Coolant temperature sensor.

valence and conduction band and which 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

$$\ell n(R_T) = \frac{A}{T} - B \tag{59}$$

where, for an exemplary sensor, the coefficients are approximately

$A \cong 5000$, $B \cong 3.96$, and *T* is the absolute temperature (K).

The sensor is typically connected in an electrical circuit like that shown in Figure 6.21, 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:

$$V_T = V\frac{R_T}{R + R_T} \tag{60}$$

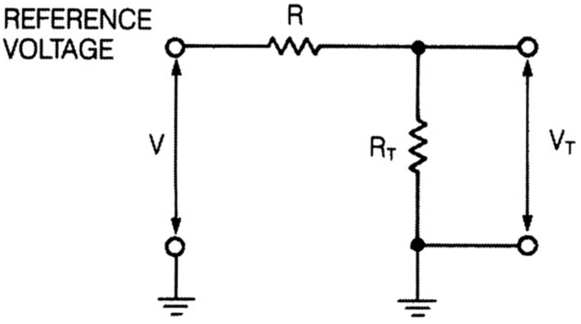


Figure 6.21:
Temperature sensor circuit.

Combining equations 59 and 60 yields the following equation for temperature T:

$$T = A/\{B + \ell n\,[V_T R/(V - V_T)]\}$$

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).

Sensors for Feedback Control

The sensors that we have discussed until now have been part of the open-loop (i.e., feedforward) control. Next, we consider sensors that are appropriate for feedback engine control. Recall from Chapter 5 that feedback control for fuel delivery is based on maintaining the air/fuel ratio at stoichiometry (i.e., 14.7:1). The primary sensor for fuel control is the exhaust gas oxygen sensor.

Exhaust Gas Oxygen Sensor

Recollect from Chapter 5 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 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 5):

$$\lambda = \frac{(\mathrm{air/fuel})}{(\mathrm{air/fuel}/@\ \ \mathrm{stoichiometry}} \tag{61}$$

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 the most commonly used type today. Figure 6.22a is a photograph of a typical ZrO_2 EGO sensor. Figure 6.22b schematically depicts the mounting of the sensor on the exhaust system. Figure 6.22c 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.

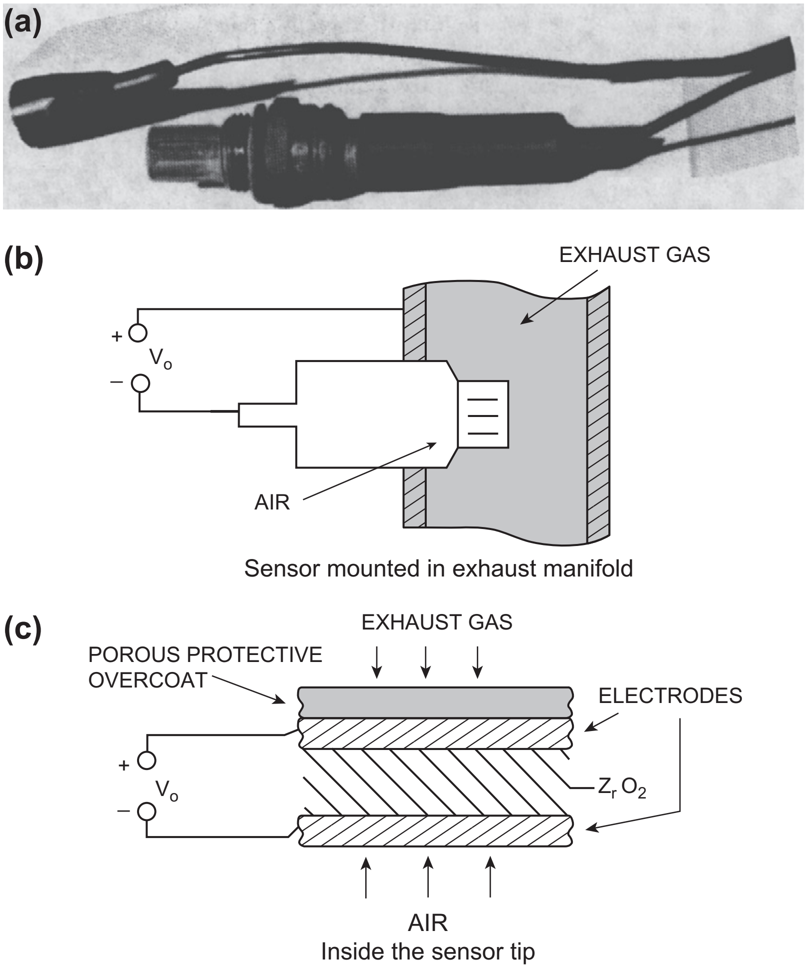


Figure 6.22:
Illustration of EGO sensor.

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} atmosphere. 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 volt 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 5 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, and
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 Figure 6.23. 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 5 for explaining closed-loop fuel control in which the hysteresis was taken to be negligible.

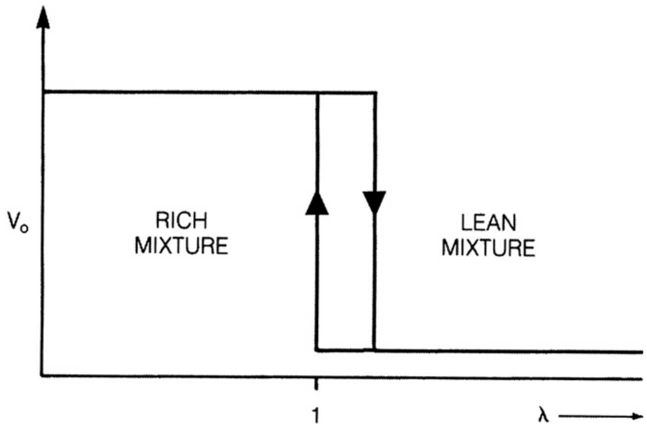


Figure 6.23:
Switching characteristics of ideal EGO sensor.

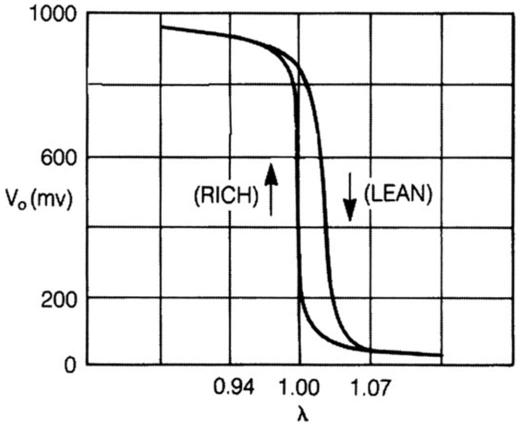


Figure 6.24:
Commercial EGO sensor voltage vs. λ.

Figure 6.24 depicts the actual sensor voltage/equivalence ratio characteristics for a common commercially available (fully warned) 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 Figure 6.25. Note that the time per division is twice as much for the display at 350 °C as at 800 °C. This means that the switching times are roughly 0.1 second 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 Figure 6.26 shows the temperature dependence of an EGO sensor output voltage for lean and rich mixtures and for two different load resistances — 5 MΩ 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 7). 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. 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

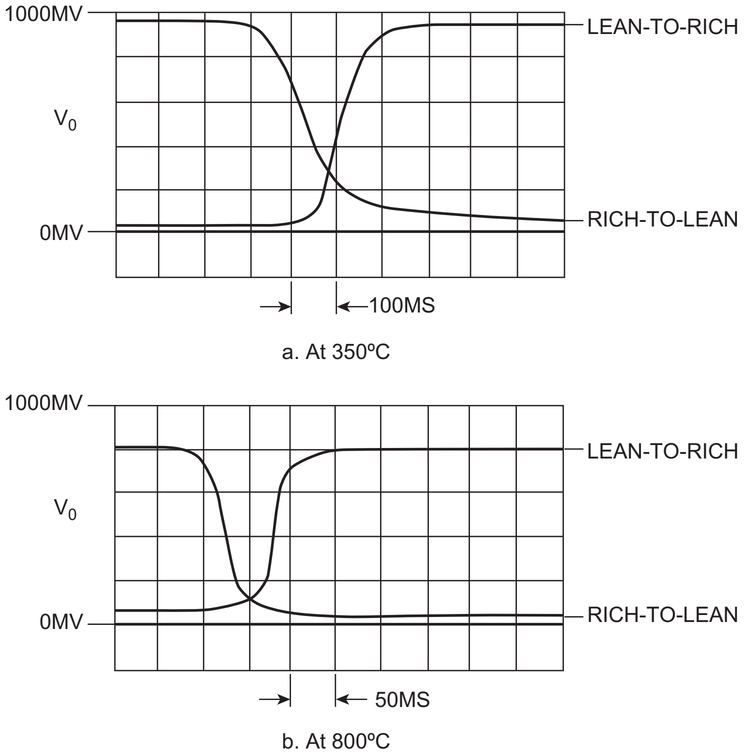


Figure 6.25:
EGO sensor switching transients.

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

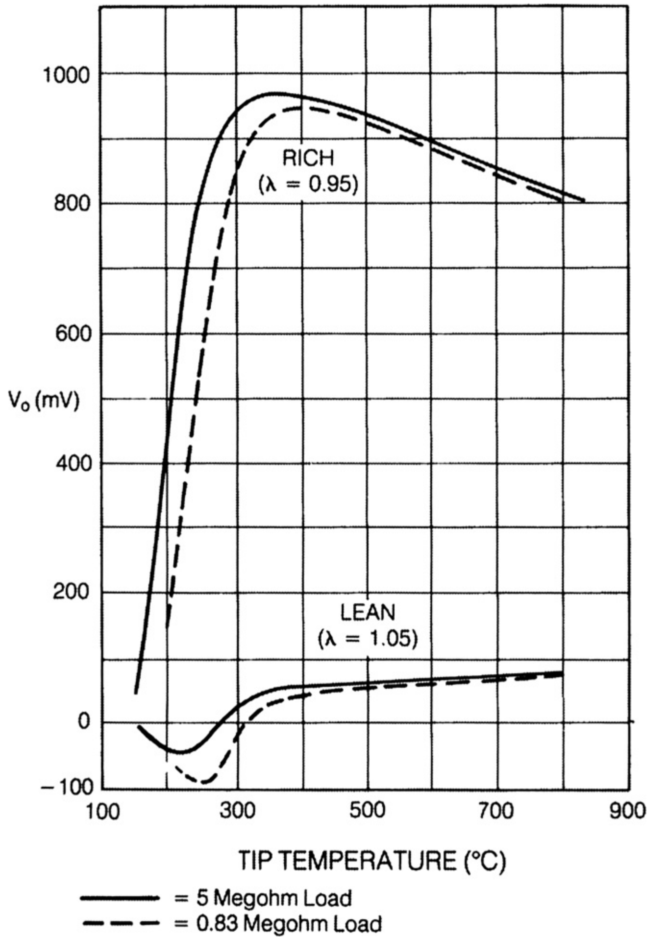


Figure 6.26:
EGO sensor temperature characteristics.

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 Figure 6.27. 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

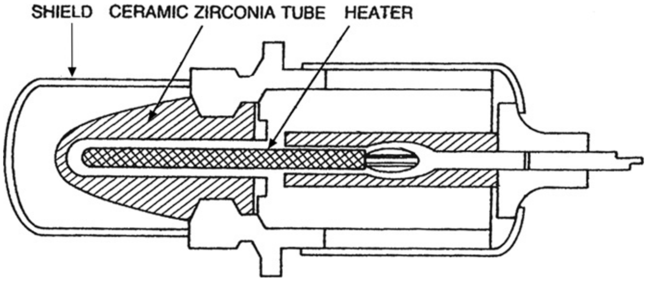


Figure 6.27:
Heated EGO sensor configuration.

heat is largely maintained within the sensor housing, thereby leading to a relatively rapid temperature rise.

Normally, the heating element need 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.

Knock Sensors

Another sensor having applications in closed-loop engine control is the so-called knock sensor. As explained in Chapter 7, this sensor is employed in closed-loop ignition timing to prevent undesirable knock. Although a more detailed explanation of knock is given in Chapter 7, 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 7, 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 acoustical signals associated with the rapid pressure rise of cylinder pressure. The phenomenon is called knock because the acoustical 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 KHz range depending on engine configuration. The associated cylinder pressure waveform is depicted in Chapter 7 in Figure 7.11.

The configuration of a representative knock sensor using magnetostrictive techniques is shown in Figure 6.28a. *Magnetostriction* is a phenomenon whereby the magnetic properties

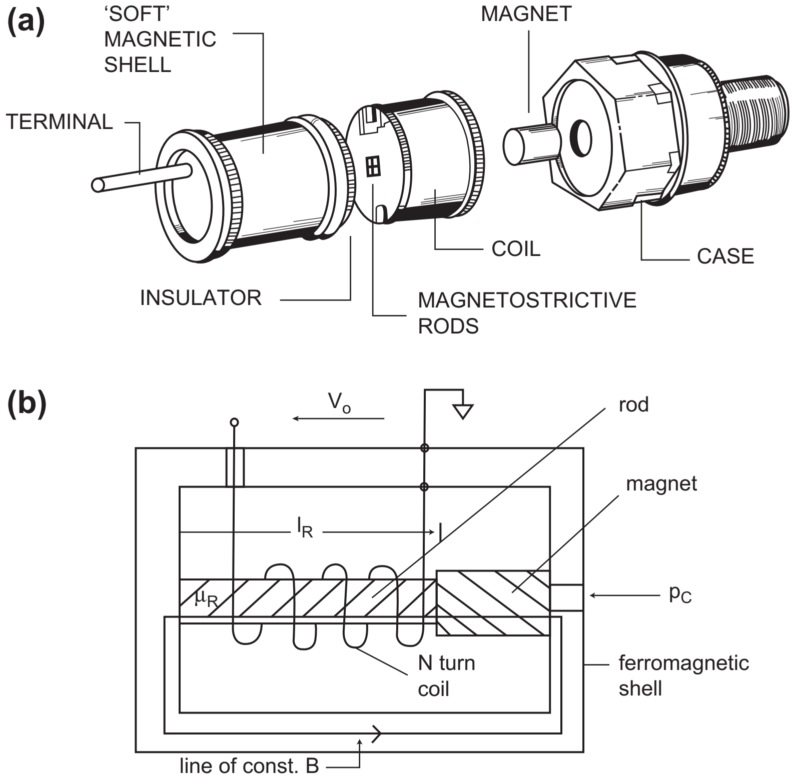


Figure 6.28:
Knock sensor configuration.

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 Figure 6.28a, 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 Figure 6.7. 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 Figure 6.28a, the small magnet creates a magnetic field having a magnetic flux density $\overline{B}$ in the form of closed-loops passing through the magnet, magnetostrictive rods, the coil of N turns, and return through the magnetically "soft" (i.e., relatively high magnetic permeability) magnetic shell (i.e., see Figure 6.28b). These loops are basically lines of constant flux density magnitude. The strength of this flux density is determined by the magnet as well as the permeability of the magnetostrictive rods μ_R. The magnetic permeability of the shell μ_s is assumed to satisfy the inequality: $\mu_s >> \mu_R$.

A simplified model for the amplitude of the flux density is given by

$$B \cong \frac{M\mu_R}{\ell_R} \tag{62}$$

where ℓ_R is the length of the magnetostrictive rods and M is a constant for the magnet. The total magnetic flux Φ through the rods is approximately given by

$$\begin{aligned}\Phi &= \int_{A_R} B\mathrm{d}s \\ &\cong BA_R\end{aligned} \tag{63}$$

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

$$\Phi = \frac{MA_R\mu_R}{\ell_R} \tag{64}$$

The sensor terminal voltage V_o is given by

$$\begin{aligned}V_o &= N\frac{\mathrm{d}\Phi}{\mathrm{d}t} \\ &= \frac{NMA_R}{\ell_R}\frac{\mathrm{d}\mu_R}{\mathrm{d}t}\end{aligned} \tag{65}$$

where $N =$ number of turns of the coil.

The time derivative of μ_R is due to magnetostrion in the rods. An approximate model for μ_R is given by:

$$\mu_R = \mu_1 + \mu_2\sigma_R$$

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 7). 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 7.

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 7).

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.

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 Figure 6.29. 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 Figure 6.29. 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

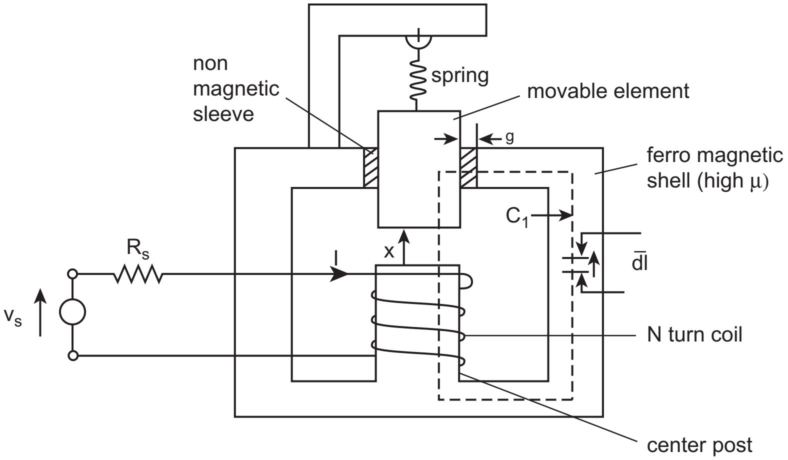


Figure 6.29:
Solenoid configuration.

ferromagnetic material as well as the nonmagnetic sleeve of thickness g and nonmagnetic air gap of length *x*.

Figure 6.29 also shows a movable element which 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 can be related to the total current *I* that passes through the surface enclosed by C_1 using one of the fundamental equations (Eqn (30)) given above

$$I_T = \oint_{C_1} \overline{H} \cdot \mathrm{d}\overline{\ell} \tag{66}$$

where $\mathrm{d}\overline{\ell}$ 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 $\overline{H}$ along any contour C_1 is given by

$$H = \frac{B}{\mu}$$

where

$$\begin{aligned} \mu &= \mu_o && \text{in the sleeve and gap} \\ &= \mu_r\mu_o && \text{in the ferromagnetic material} \end{aligned}$$

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

$$I_T \simeq H(x+g) \tag{67}$$

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

$$\begin{aligned} I_T &= NI \\ &= \frac{B}{\mu_o}(x+g) \end{aligned}$$

The total flux linking the N turn coil λ is given by

$$\lambda = N \iint_{A_c} B\mathrm{d}s$$

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

$$\begin{aligned} \lambda &\cong NBA_C \\ &= \mu_o \frac{N^2 I A_C}{x+g} \end{aligned}$$

The important circuit parameter that characterizes the electrical model for the coil, which is called its inductance L, is defined as

$$\begin{aligned} L &= \frac{\lambda}{I} \\ &= \frac{\mu_o N^2 A_C}{(x+g)} \end{aligned}$$

The terminal voltage v_o of a two-terminal device having inductance L is given by

$$\mathrm{v}_o = \frac{\mathrm{d}(LI)}{\mathrm{d}t} \tag{68}$$

For a fixed inductor, this model becomes familiar:

$$\mathrm{v}_o = L\frac{\mathrm{d}I}{\mathrm{d}t} \quad \text{(for constant L)} \tag{69}$$

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 electrical 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

$$W_m = \int_0^I \lambda(i,x)\mathrm{d}i \tag{70}$$

$$= \frac{1}{2}L(x)I^2 \tag{71}$$

If the total energy stored in the magnetic field is denoted W_m, conservation of energy requires that

$$\frac{\mathrm{d}W_m}{\mathrm{d}t} = IL\frac{\mathrm{d}I}{\mathrm{d}t} - f_e\frac{\mathrm{d}x}{\mathrm{d}t} \tag{72}$$

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 Figure 6.25 is given by

$$f_e = \frac{\partial W_m}{\partial x} \tag{73}$$

$$= -\frac{\mu_o N^2 A_C I^2}{2(x+g)^2} \tag{74}$$

Note that the minus sign indicates a force that is reducing x and that it varies inversely with x.

The solenoid of Figure 6.29 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 Figure 6.25, 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:

$$\lambda(I, x) = \frac{L_o I}{(1 + x/g)} \tag{75}$$

where

$$L_o = \frac{\mu_o A_C N^2}{g}$$

and where L_o is the inductance of the solenoid at $x = o$. Summing the voltages around the loop formed by the source $\mathrm{v_s}$, R_s and the solenoid terminals yields:

$$\begin{aligned} \mathrm{v}_s &= IR_s + \frac{\mathrm{d}\lambda}{\mathrm{d}t} \\ &= IR_s + \frac{L_o}{(1 + x/g)}\frac{\mathrm{d}I}{\mathrm{d}t} - \frac{L_o I}{g^2(1 + x/g)^2}\frac{\mathrm{d}x}{\mathrm{d}t} \end{aligned} \tag{76}$$

The first term on the right-hand side of equation 76 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:

$$f_e = M\frac{\mathrm{d}^2 x}{\mathrm{d}t^2} + D\frac{\mathrm{d}x}{\mathrm{d}t} + K_s(x - \ell)$$

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:

$$-\frac{1}{2}\frac{L_o I^2}{g(1+x/g)^2} = M\frac{d^2x}{dt^2} + D\frac{dx}{dt} + K_s(x-\ell) \tag{77}$$

Since these equations are nonlinear in I and x, they cannot be solved by the Laplace operator method of Chapter 1. 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 as well as 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 Figure 6.30, 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. 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

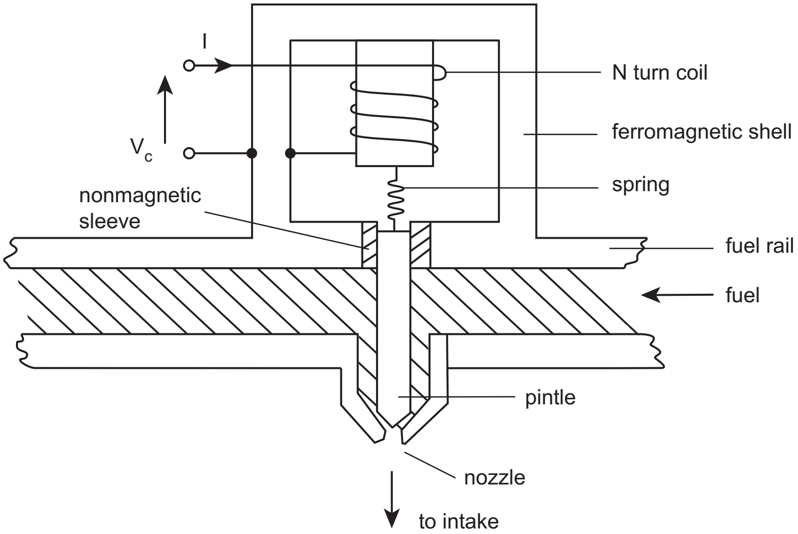


Figure 6.30:
Simplified fuel injector configuration

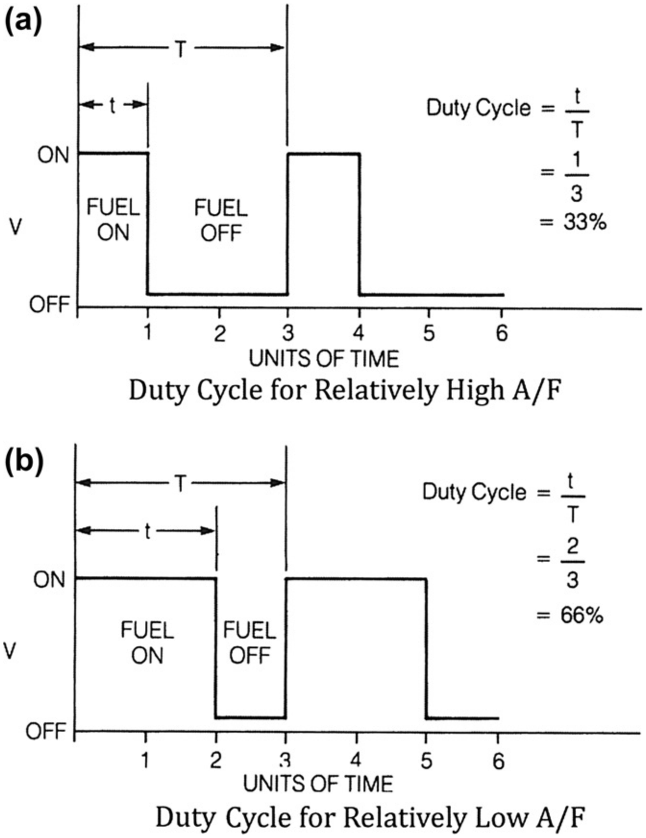
for brief transient periods, the quantity of fuel injected into the air stream is proportional to the time the valve is open. The control current that operates the fuel injector is pulsed on and off to deliver precise quantities of fuel.

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 Figure 6.31 is the terminal voltage applied to the fuel injector by the electronic engine control system. Figure 6.31a and b depict 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 Figure 6.31, the fuel is assumed to flow at an essentially constant rate (i.e., $\dot{M}_f = \text{constant}$) 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



Duty Cycle for Relatively High A/F

Duty Cycle for Relatively Low A/F

Figure 6.31:
Fuel injector terminal voltage.

the *duty cycle* δ_{FI}. This is shown in Figure 6.31. The fuel injector is energized for time t_k to allow fuel to spray from the nozzle into the air stream 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 Figure 6.31a, is used for a relatively high air/fuel ratio (lean mixture), and a high duty cycle (Figure 6.31b) is used for a relatively low air/fuel ratio (rich mixture).

Exhaust Gas Recirculation Actuator

In Chapter 5 it was explained that exhaust gas recirculation (EGR) is utilized to reduce NOx emissions. The amount of EGR is regulated by the engine controller, as explained in Chapter 7. When the correct amount of EGR has been determined by the controller based on

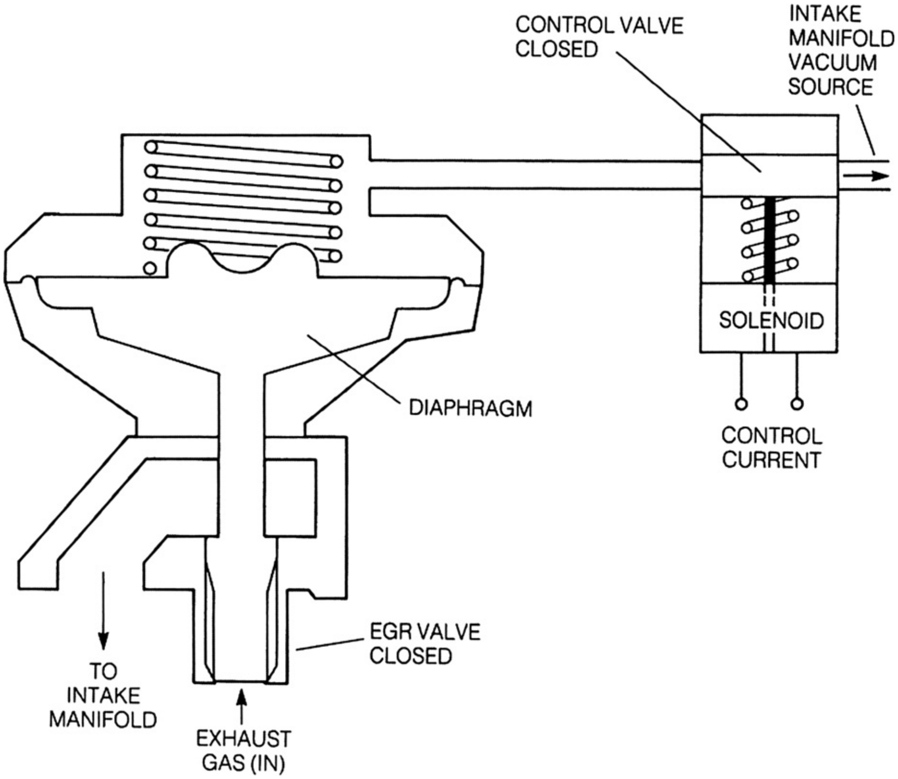


Figure 6.32:
EGR actuator.

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 Figure 6.32. 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) as well as 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 cyclical 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 Figure 6.33. Figure 6.33a 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.

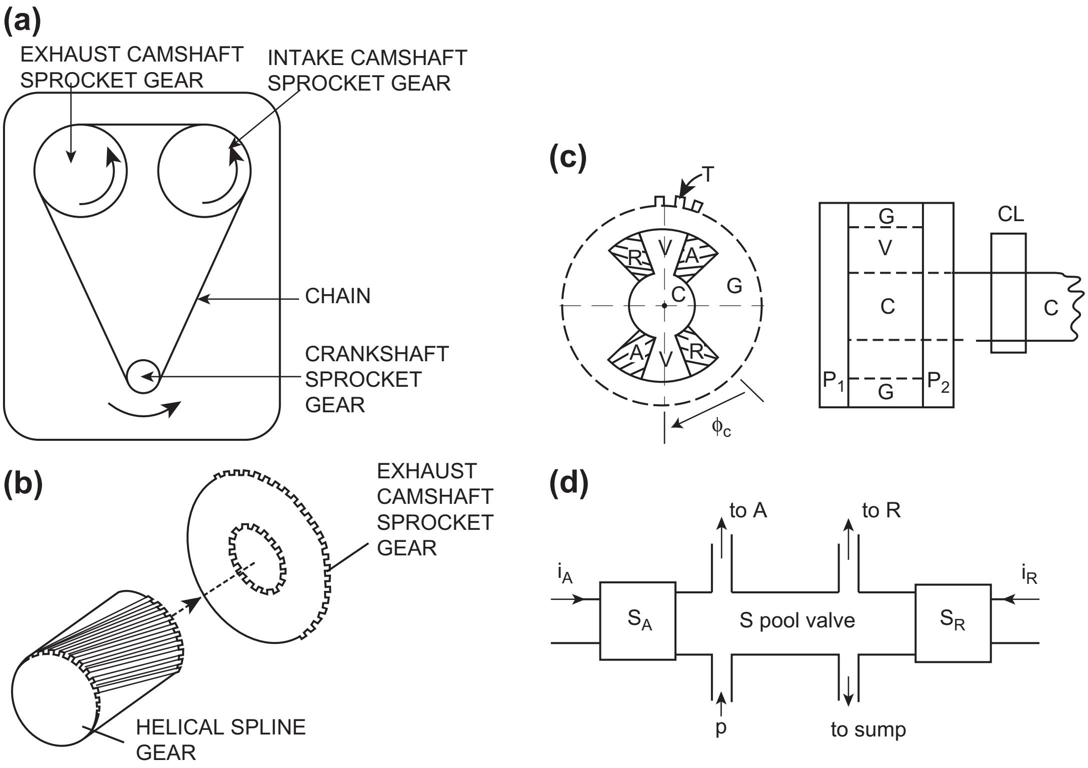


Figure 6.33:
Representative VVP mechanisms.

In this hypothetical example, each cam sprocket includes a housing within which is a helical spline gear that engages an inner gear connected rigidly to this sprocket gear. Figure 6.33b 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 Figure 6.33c. 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.

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 Figure 6.33c, there are passageways that supply the oil under pressure.

Figure 6.33d 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 Figure 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) as well as 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. Figure 6.33d 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 7 to explain the operation of the VVP under powertrain control. In the implementation shown in Figure 6.33c and d, the spool valve is centered via a spring when $i_{\mathrm{A}} = 0$ and $i_{\mathrm{R}} = 0$. 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

$$p_{\mathrm{A}} = k_{p\mathrm{A}}\delta_{\mathrm{A}}$$

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

$$T_c = k_c\delta_{\mathrm{A}} \quad \text{for } i_{\mathrm{A}} \text{ nonzero} \tag{78}$$

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

$$T_c = -k_c\delta_{\mathrm{R}} \quad i_{\mathrm{R}} \text{ nonzero} \tag{79}$$

With proper design, such a system can be modeled as a linear actuator in the following form:

$$T_c = k_c u \tag{80}$$

where u is the control signal from electronic engine control system:

$$\begin{aligned} &= \delta_{\mathrm{A}} && i_{\mathrm{A}} \text{ nonzero} \\ &= -\delta_{\mathrm{R}} && i_{\mathrm{R}} \text{ nonzero} \end{aligned} \tag{81}$$

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 Figure 6.30c). It is convenient to represent this dynamic motion with the following approximate linear model:

$$J_c\ddot{\phi}_c + B_c\dot{\phi}_c = T_c \tag{82}$$

$$= k_c u \tag{83}$$

where J_c = moment of inertia of the components that rotate relative to the gear
B_c = viscous damping coefficient for VVP mechanism.

The VVP actuator mechanism can now be modeled as a transfer function $H_p(s)$, which is given by

$$\begin{aligned} H_p &= \frac{\phi_c(s)}{u(s)} \\ &= \frac{k_a}{s(s+s_o)} \end{aligned} \tag{84}$$

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/powertrain control system for an engine with VVP. The electronic control for this VVP is described and analyzed in Chapter 7.

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, position seats as well as for actuators on airflow control at idle (see Chapter 7). 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 by 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.

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 analytical 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 Figure 6.34, 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

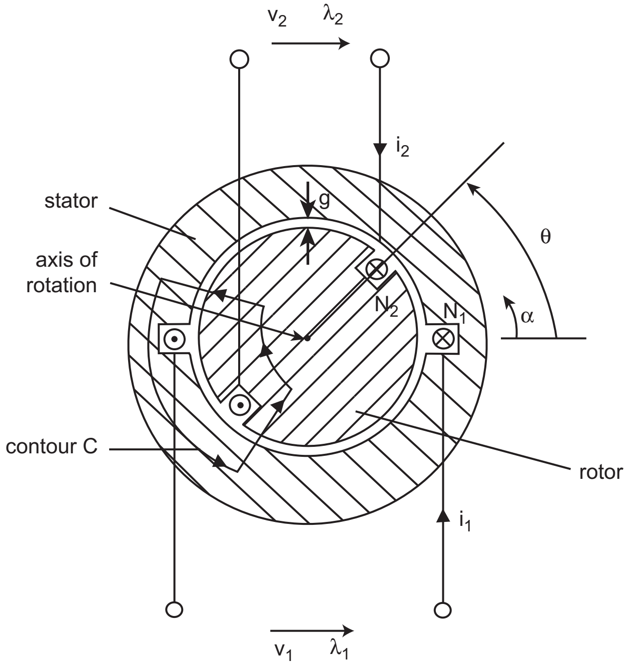


Figure 6.34:
Schematic representation of electric motor.

relatively complex structure of a practical motor. In Figure 6.34, 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

$$B_r = \mu_o H_r \tag{85}$$

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 Figure 6.34) that encloses the two coils is given by

$$I_T = \oint_C \overline{H} \cdot \overline{\mathrm{d}} \overline{\ell} = 2gH_{\mathrm{r}}(\alpha) \tag{86}$$

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:

$$\begin{aligned} 2gH_r(\alpha) &= N_1 i_1 - N_2 i_2 & 0 \le \alpha < \theta \\ &= N_1 i_1 + N_2 i_2 & \theta < \alpha < \pi \\ &= -N_1 i_1 + N_2 i_2 & \pi < \alpha < \pi + \theta \\ &= -N_1 i_1 - N_2 i_2 & \pi + \theta < \alpha < 2\pi \end{aligned}$$

The magnetic flux linkage for the two coils λ_1 and λ_2 are given by

$$\lambda_1 = N_1 \int\limits_{o}^{\pi} B_r(\alpha)\ell R_r \mathrm{d}\alpha$$

$$\lambda_2 = N_2 \int\limits_{\theta}^{\pi+\theta} B_r(\alpha)\ell R_r \mathrm{d}\alpha \qquad (87)$$

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:

$$\lambda_1 = L_1 i_1 + L_m i_2 \qquad (88)$$

$$\lambda_2 = L_m i_1 + L_2 i_2 \qquad (89)$$

where

$$L_1 = N_1^2 L_o = \text{self inductance of coil 1} \qquad (90)$$

$$L_2 = N_2^2 L_o = \text{self inductance of coil 2} \qquad (91)$$

$$L_o = \frac{\mu_o \ell R_r \pi}{2g} \qquad (92)$$

The parameter L_m is the mutual inductance for the two coils which 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 as well as 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 symmetric function of θ. It can be formally expanded in a Fourier series in θ having only cosine terms in odd harmonics as given below:

$$L_m(\theta) = M_1 \cos(\theta) + M_3 \cos(3\theta) + M_5 \cos(5\theta) + \ldots \qquad (93)$$

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

$$L_m \simeq M\cos(\theta) \tag{94}$$

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

$$\lambda_s = L_s i_s + M i_r \cos\theta$$

$$\lambda_r = L_r i_r + M i_s \cos\theta$$

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

$$T_e = J_r \frac{\mathrm{d}^2\theta}{\mathrm{d}t^2} + B_\mathrm{v}\frac{\mathrm{d}\theta}{\mathrm{d}t} + C_c\,\mathrm{sgn}\left(\frac{\mathrm{d}\theta}{\mathrm{d}t}\right)$$

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

$$\begin{aligned} i_s &= I_s \sin(\omega_s t) \\ i_r &= I_r \sin(\omega_r t) \\ \theta(t) &= \omega_m t + \gamma \end{aligned} \tag{95}$$

where ω_m is the rotor rotational frequency (rad/sec) and γ expresses an arbitrary time phase parameter. The motor power is given by

$$P_m = T_e \omega_m \tag{96}$$

$$= -\omega_m I_s I_r M \sin(\omega_s t)\sin(\omega_r t)\sin(\omega_m t + \gamma) \tag{97}$$

This equation can be rewritten using well-known trigonometric identities in the form

$$P_m = -\frac{\omega_m I_s I_r M}{4}\{\sin[(\omega_m+\omega_s-\omega_r)t+\gamma]+\sin[(\omega_m-\omega_s+\omega_r)t+\gamma] \\ -\sin[(\omega_m+\omega_s+\omega_r)t+\gamma]-\sin[(\omega_m-\omega_s-\omega_r)t+\gamma]\} \tag{98}$$

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:

$$\omega_m = \pm\omega_s \pm \omega_r \tag{99}$$

For example, whenever $\omega_m = \omega_s + \omega_r$, the motor time average power $P_{m_{av}}$ is given by

$$P_{m_{av}} = \frac{\omega_m I_s I_r M}{4}\sin\gamma \tag{100}$$

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

$$\sin\gamma = \frac{4P_L}{\omega_m I_s I_r M} \tag{101}$$

provided

$$P_L \le \frac{\omega_m I_s I_r M}{4} \tag{102}$$

The above frequency conditions (Eqn (99)) 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.

A two-phase motor has two sets of windings displaced at 90° in the θ direction and excited by currents with a 90° phase for both stator and rotor. A so-called balanced two-phase motor will have its coil excited by currents i_{as}, i_{bs} for phases a and b, respectively, where

$$i_{as} = I_s \cos(\omega_s t) \qquad (103)$$

$$i_{bs} = I_s \sin(\omega_s t)$$

The rotor is also constructed with two sets of windings displaced physically by 90° and excited with currents i_{ar} and i_{br} having 90° phase shift:

$$i_{ar} = I_r \cos(\omega_r t) \qquad (104)$$

$$i_{br} = I_r \sin(\omega_r t)$$

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 $\mathrm{v}_{ar} = \mathrm{v}_{br} = 0$, 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

$$\begin{aligned}
\lambda_{as} &= L_s i_{as} + M i_{ar} \cos\theta - M i_{br} \sin\theta \\
\lambda_{bs} &= L_s i_{bs} + M i_{ar} \sin\theta + M i_{br} \cos\theta \\
\lambda_{ar} &= L_r i_{ar} + M i_{as} \cos\theta + M i_{bs} \sin\theta \\
\lambda_{br} &= L_r i_{br} - M i_{as} \sin\theta - M i_{bs} \cos\theta
\end{aligned} \qquad (105)$$

The torque T_e and instantaneous power P_m for the two-phase induction motor are given by

$$T_e = M[(i_{ar} i_{bs} - i_{br} i_{as})\cos\theta - (i_{ar} i_{as} + i_{br} i_{bs})\sin\theta] \qquad (106)$$

$$P_m = \omega_m M I_s I_r \sin[(\omega_m - \omega_s + \omega_r)t + \gamma]$$

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

$$\frac{\mathrm{d}\lambda_{ar}}{\mathrm{d}t} = \frac{\mathrm{d}\lambda_{br}}{\mathrm{d}t} = 0 \qquad (107)$$

The two rotor currents, thus, satisfy the following equations:

$$0 = R_r i_{ar} + L_r \frac{\mathrm{d}i_{ar}}{\mathrm{d}t} + M I_s \frac{\mathrm{d}}{\mathrm{d}t}[\cos(\omega_s t)\cos(\omega_m t + \gamma) + \sin(\omega_s t)\sin(\omega_m t + \gamma)] \qquad (108)$$

$$0 = R_r i_{br} + L_r \frac{\mathrm{d}i_{br}}{\mathrm{d}t} + M I_s \frac{\mathrm{d}}{\mathrm{d}t}[-\cos(\omega_s t)\sin(\omega_m t + \gamma) + \sin(\omega_s t)\cos(\omega_m t + \gamma)] \qquad (109)$$

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

$$L_r \frac{\mathrm{d}i_{ar}}{\mathrm{d}t} + R_r i_{ar} = MI_s(\omega_s - \omega_m)\sin[(\omega_s - \omega_m)t - \gamma] \tag{110}$$

$$L_r \frac{\mathrm{d}i_{br}}{\mathrm{d}t} + R_r i_{br} = -MI_s(\omega_s - \omega_m)\cos[(\omega_s - \omega_m)t - \gamma] \tag{111}$$

The current i_{ab} is identical to i_{ar} except for a 90° phase shift as can be seen from Eqn (111). 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

$$i_{ar} = \frac{(\omega_s - \omega_m)MI_s}{\sqrt{R_r^2 + (\omega_s - \omega_m)^2 L_r^2}} \cos[(\omega_s - \omega_m)t - \alpha] \tag{112}$$

where

$$\alpha = -\left(\frac{\pi}{2} + \gamma + \beta\right)$$

and

$$\beta = \tan^{-1}\left[\frac{(\omega_s - \omega_m)}{R_r} L_r\right] \tag{113}$$

The current in phase b is identical except for a 90° 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

$$T_e = \frac{(\omega_s - \omega_m)M^2 R_r I_s^2}{R_r^2 + (\omega_s - \omega_m)^2 L_r^2} \tag{114}$$

The mechanical output power P_m is given by

$$\begin{aligned} P_m &= \omega_m T_e \\ &= \left[\frac{\omega_s^2 M^2 I_s^2}{(R_r/s)^2 + \omega_s^2 L_r^2}\right]\left(\frac{1-s}{s}\right) R_r \end{aligned}$$

where s is called slip and is given by

$$s = \frac{\omega_s - \omega_m}{\omega_s} \tag{115}$$

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 $i_r^2 R_r$ 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 Eqn (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

$$0 \le \omega_m^* \le \omega_s$$

The torque has a negative slope given by

$$\frac{\mathrm{d}T_e}{\mathrm{d}\omega_m} < 0 \quad \omega_m > \omega_m^*$$

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, i.e. $T_e(\omega_m) = T_L(\omega_m)$.

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

$$T_L = K_L \omega_m \tag{116}$$

Figure 6.35 illustrates the motor and load torques for a load that varies linearly with ω_m.

For convenience of presentation, Figure 6.35 presents normalized motor torque and load torque normalized to the maximum torque $T_{\max}$ where

$$T_{\max} = \max_{\omega_m}(T_e(\omega_m)) \tag{117}$$

This maximum occurs at $\omega_m = \omega_m^*$, which, for the present hypothetical normalized example, is given by

$$\frac{\omega_m^*}{\omega_s} \cong .68$$

Figure 6.33 also presents two load torques normalized to $T_{\max}$:

$$T_{L1} = K_{L1}\omega_m / T_{\max}$$
$$T_{L2} = K_{L2}\omega_m / T_{\max}$$

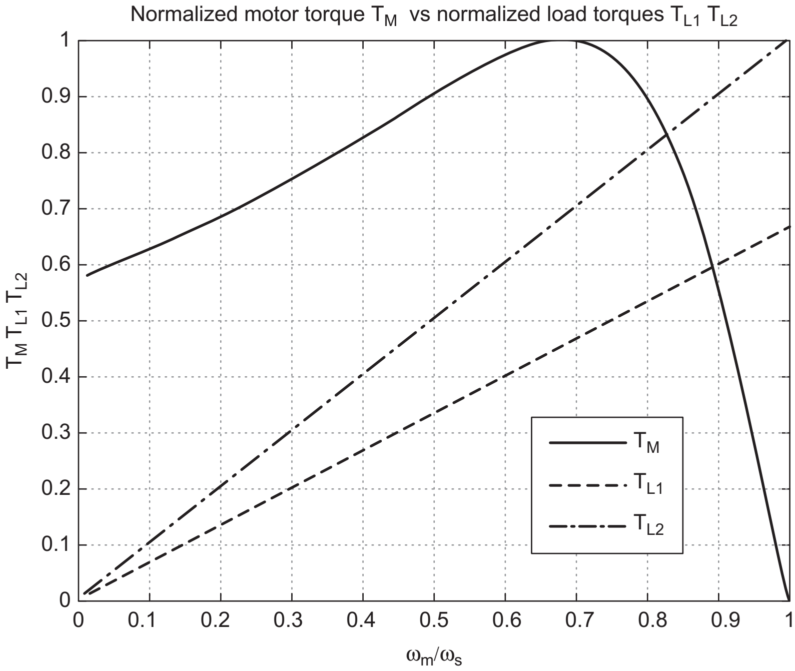


Figure 6.35:
Normalized torque T_m vs. normalized load torques T_{L1} T_{L2}.

where

$$K_{L2} > K_{L1}$$

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 7 discusses the control of an induction motor that is used in a hybrid electric vehicle. There the model for load torque vs. 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. This type of motor incorporates a permanent magnet in the rotor and electromagnet poles in the stator as depicted in Figure 6.36. 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 Figure 6.36 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 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.

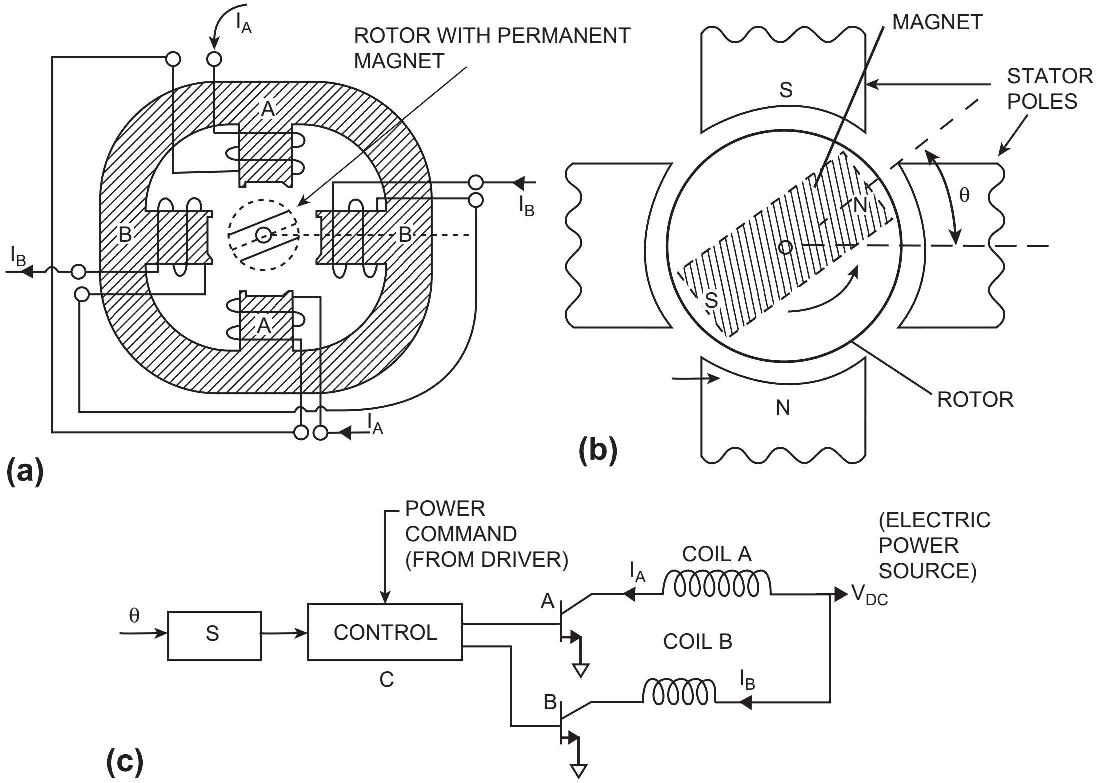


Figure 6.36:
Brushless DC motor.

The torque $\overline{T}_m$ applied to the rotor by the magnetic field intensity vector $\overline{H}$ created by the stator windings is given by the following vector product

$$\overline{T}_m = \gamma(\overline{M} \times \overline{H}) \tag{118}$$

where $\overline{M}$ 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 $\overline{H}$ (which is proportional to the excitation current). The magnitude of the torque T_m is given by

$$T_m = \gamma MH\sin(\theta)$$

where $M =$ magnitude of $\overline{M}$, $H =$ magnitude of $\overline{H}$ and $\theta =$ angle between $\overline{M}$ and $\overline{H}$.

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 Figure 6.36 (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 which rotates at rotor frequency (ω_r).

A simplified block diagram of the two-pole motor control system for the motor of Figure 6.36a and b is shown in Figure 6.36c. 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 as well as 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 on-board 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°, 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 d-c 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 Figure 6.34 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 7 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 Figure 6.36 has excitation currents i_A and i_B that are sequences of nonoverlapping pulses. The relative phasing of the pulses determine 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 during the end of the compression stroke.

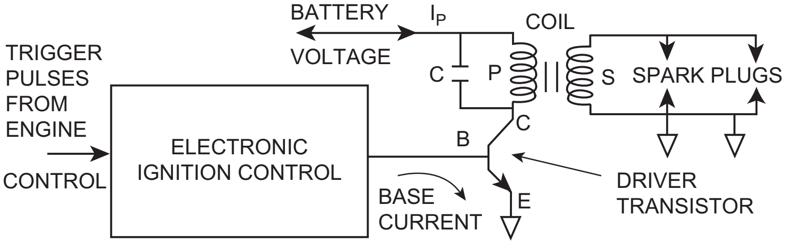


Figure 6.37:
Electronic ignition block diagram.

Figure 6.37 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 Figure 6.37) 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 3). The coil secondary S is connected to one or more spark plugs, as explained in Chapter 7. A model for the operation of the ignition system is developed next.

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.

Figure 6.37 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 $N_s >> N_p$. Although this figure depicts only a single spark plug and coil, there must be, of course, a separate circuit for each cylinder.

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 3). 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:

$$\mathrm{V}_b = R_{on} I_p + L_p \frac{\mathrm{d}I_P}{\mathrm{d}t} \tag{119}$$

where V_b is the vehicle power bus voltage and L_p is the primary coil inductance. Using the Laplace transform methods of Chapter 1, 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.

The physical process of creating the spark is the very large coil secondary voltage v_s which is given by

$$\mathrm{V}_s = N_s\frac{\mathrm{d}\Phi}{\mathrm{d}t}$$

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 Figure 6.38. 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

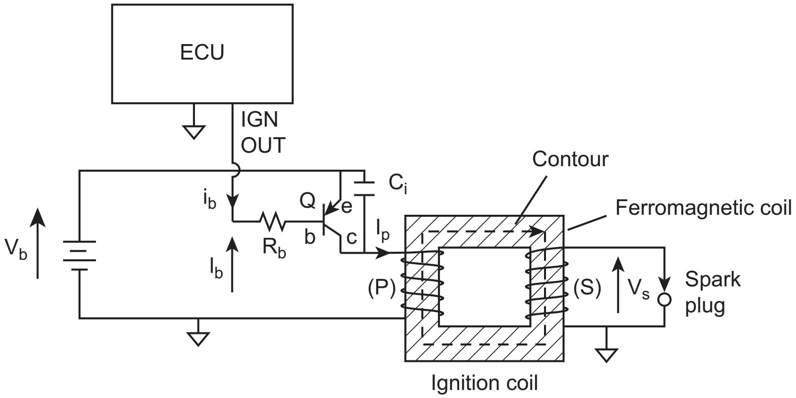


Figure 6.38:
Simplified electronic ignition circuit.

$$\Phi_s \cong \frac{\mu_c A_c \mathrm{V}_b}{\ell_c R_{on}}$$

where μ_c is the core permeability, ℓ_c is the distance around contour C, 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. The capacitor C_i, which is shown in Figure 6.38, 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.

This page intentionally left blank

CHAPTER 7

Digital Powertrain Control Systems

Chapter Outline

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-08-097097-4.00007-2

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 emissions 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 an 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. 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 5 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 5, 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 proven 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 (for example, 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

$$\begin{aligned} t_k &= kT_s \quad k = 1, 2, 3\ldots \\ T_s &= \text{sample time} \end{aligned} \tag{1}$$

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 Chapter 2. However, as explained in Chapter 5, the majority of automotive control or instrumentation systems employ some analog sensors and actuators (or in the instrumentation case, displays).

In Chapter 6, it was shown that the majority of sensors and actuators are analog devices that are modeled as functions of continuous time t. As described in Chapter 2, 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

$$V_k = V_0(t_k) \tag{2}$$

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 Chapter 2, one form for the relationship between the input and output of a linear SISO digital system is given by the recursive algorithm below:

$$y_k = \sum_{m=0}^{M} a_m x_{k-m} - \sum_{n=1}^{N} b_n y_{k-n} \tag{3}$$

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 Chapter 2.

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.

Digital Engine Control Features

Recall from Chapter 5 that one primary purpose of the electronic engine control system is to regulate the mixture (i.e., air—fuel), the ignition timing, and 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, 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.

Figure 7.1 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. Spark plugs for this four-cylinder example are denoted S.P.

Typically, the controller incorporates hardware for the multiply/divide operation as well as ROM (see Chapter 4). 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 as well as calibration

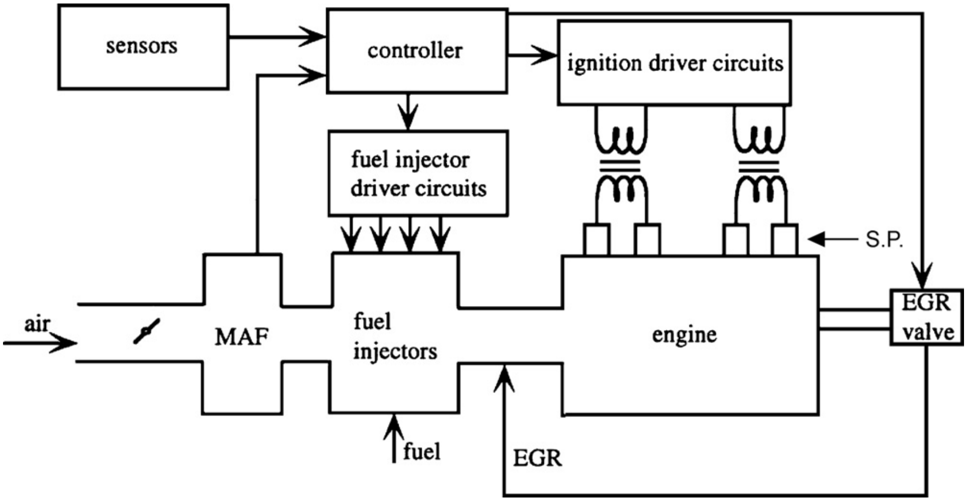


Figure 7.1:
Components of an electronically controlled engine.

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 5.

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 6, the MAF sensor generates an output terminal voltage v_o given by

$$\mathrm{v}_o(t) = f_m(\dot{M}_a) \tag{4}$$

where $\dot{M}_a$ is the instantaneous mass airflow rate into the engine intake system (kg/s).

As explained in Chapter 6, the function F_m for a representative production MAF sensor is given by

$$\mathrm{v}_o(\dot{M}_a) = \sqrt{\mathrm{v}_o^2(0) + K_{\mathrm{MAF}}\dot{M}_a}$$

However, a digital fuel control system can invert a nonlinear function to obtain the value $\dot{M}_a$ of mass airflow:

$$\dot{M}_a = f_m^{-1}(\mathrm{v}_o) \tag{5}$$

As explained in Chapter 5, the intake to the engine includes EGR as well as air. As will be shown below, the digital engine control system is able to determine the EGR mass flow rate $\dot{M}_{\mathrm{EGR}}$ 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 $\dot{M}_{\mathrm{EGR}}$ in the MAF sensor output is a straightforward computation.

An ideal engine control would determine the mass of air drawn into the *m*th cylinder during the *n*th 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 process would assure that all cylinders achieved the desired stoichiometric mixture for each cycle as desired for optimum exhaust emissions in conjunction with the catalytic converter. However, this ideal fuel control is not practically achievable.

On the other hand, suboptimal fuel control that is very close to optimal can be achieved in practice. 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 rotations) is given by

$$M_{aT}(t) = \int_{\theta_e(t)-4\pi}^{\theta_e(t)} \dot{M}_a(\theta_e)\mathrm{d}\theta_e \qquad (6)$$

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

$$= \frac{120}{\mathrm{RPM}}$$

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

$$M_a(n,m) = \frac{M_{aT}}{M_c} \quad m = 1, 2\ldots M_c \tag{7}$$

where M_c is the number of cylinders.

Note: 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

$$f(n,m) = \frac{M_a(n,m)}{R_{a/f}} \tag{8}$$

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 $R_{a/f}$ 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 6, 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 6 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 as well as 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

$$F(n,m) = \int_{t_{nm}}^{t_{nm}+\tau_f(n,m)} \dot{M}_f \mathrm{d}t \tag{9}$$

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 a 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

$$F(n,m) \cong \dot{M}_f \tau_F(n,m) \tag{10}$$

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, as well as distribution to the individual cylinders, took place in the intake manifold system.

The more stringent exhaust emissions 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.

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 an 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. We consider next the individual engine control modes.

Engine Start

When the ignition key is switched on initially, 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, the low air/fuel ratio may be 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. This condition 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 6 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 until the EGO sensor cools and fails to read a rich mixture for a certain length of time or a hard acceleration or deceleration occurs. If the sensor cools, the control mode logic selects the open-loop mode again.

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 NOx production.

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 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 Figure 7.2.

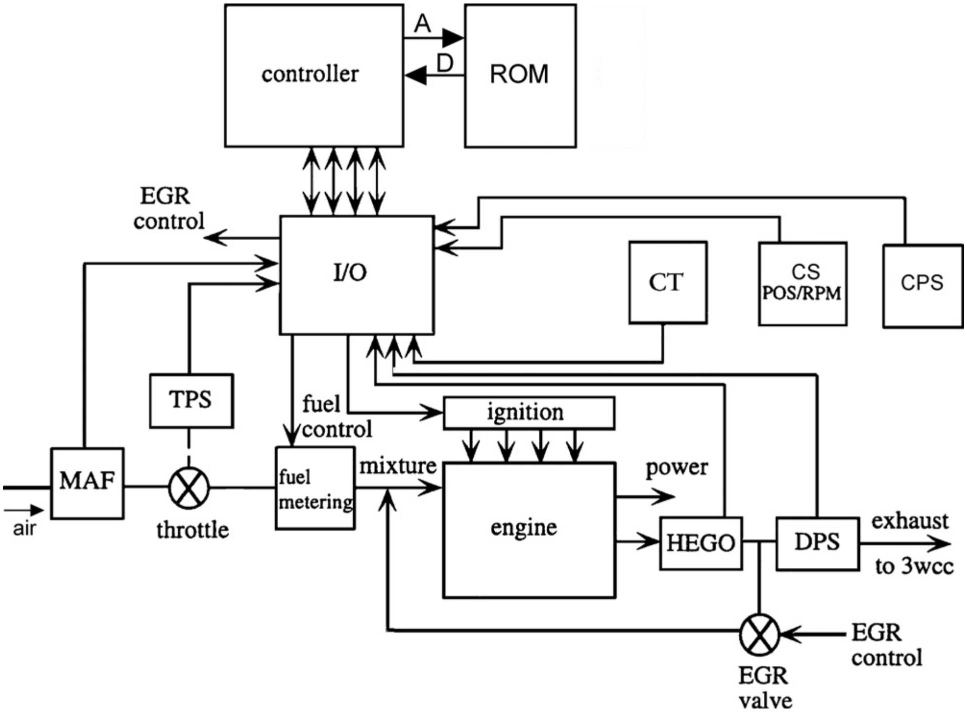


Figure 7.2:
Digital engine control system diagram.

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 4). There is also a section of ROM continuing parameter values for specific control modes and tables of data for various control functions as explained later in this chapter. Of course, any microprocessor-based system must have RAM for temporary storage of data during computation (see Chapter 4). 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 Figure 7.2) as well as 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 4 for a detailed discussion of these components.)

With reference to Figure 7.2, the sensors that measure various engine variables for control are as follows:

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),
camshaft position sensor for determining start of each engine cycle (CS POS/RPM),
throttle position sensor (TPS), and
differential pressure sensor (exhaust to intake) for EGR control (DPS).

Other sensors that might be used on older model cars that are not given in Figure 7.2 include the following:

manifold pressure sensor (MAP),
inlet air temperature (IAT),
ambient air pressure (AAP), and
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 as well as 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, 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. Therefore, 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. The controller selects a warm-up time from a lookup table based on the temperature of the coolant. In certain cases, a fully warmed engine is switched off by the driver for a brief period such that temperature remains sufficiently high that warm-up mode is either very short or not used at all.

A diagram illustrating the lookup table selection of desired air/fuel ratios is shown in Figure 7.3. 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 (A/B). The data stored at this address in the ROM are desired

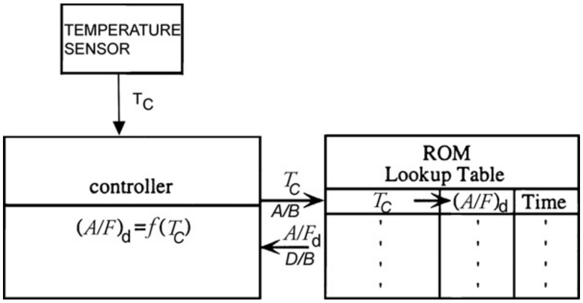


Figure 7.3:
Illustration of table lookup.

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

$$\begin{aligned} S &= \frac{\mathrm{d}y}{\mathrm{d}x} \\ &= \frac{y_{m+1} - y_m}{x_{m+1} - x_m} \end{aligned} \tag{11}$$

The linearly interpolated value for y is given by

$$\begin{aligned} y &= y_m + S(x - x_m) \\ &= y_m + \left(\frac{y_{m+1} - y_m}{x_{m+1} - x_m}\right)(x - x_m) \end{aligned} \tag{12}$$

Alternatively, it is possible to obtain a polynomial model which gives the best fit to measured data in a least squared error sense. Let an empirical data set be given by $\{x_n, y_n: n = 1,2\ldots N\}$. The polynomial which best represents this data is of the form

$$y = a_0 + a_1x + a_2x^2 + \cdots a_Mx^M \quad M < N \tag{13}$$

The mean squared error between this polynomial and the data is

$$\mathrm{MSE} = \sum_{n=1}^{N} [y(x_n) - y_n]^2 / N \tag{14}$$

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 Eqn (13) and the stored values for $\{a_m\}$ for the polynomial model method and by Eqn (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.

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

$$\tau_F(n,m) = \tau_F(n) \quad \forall m \tag{15}$$

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 $\tau_b(n)$:

$$\tau_F(n)(\text{open loop}) = \tau_b(n) \tag{16}$$

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. Corrections to the base pulse width are then made using the actual battery voltage.

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 6 that for a fully warmed EGO sensor, the output voltage of the sensor is high (approximately 1 V) when the exhaust oxygen concentration is low (i.e., for a rich mixture relative to stoichiometry). The EGO sensor voltage is low (approximately 0.1 V) whenever the exhaust oxygen concentration is high (i.e., for a mixture that is lean relative to stoichiometry).

Chapter 1 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 5. A large body of theory (both linear and nonlinear) exists which 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 5 and 6, no cost-effective sensor for measuring these regulated exhaust gases is available for production vehicles. On the other hand, as explained in Chapter 5, 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 6, 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 Chapter 1 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 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.

The physical configuration for the closed-loop fuel control system is depicted in Figure 7.4a. 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 ($\dot{M}_a$) 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. This pulse duration consists of a base pulse duration $\tau_b(n)$ and a closed-loop correction factor ($C_L(n)$) in the representative form

$$\tau_F(n) = \tau_b(n)[1 + C_L(n)] \tag{17}$$

One commonly employed 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

$$C_L(n) = \alpha I(n) + \beta P(n) \tag{18}$$

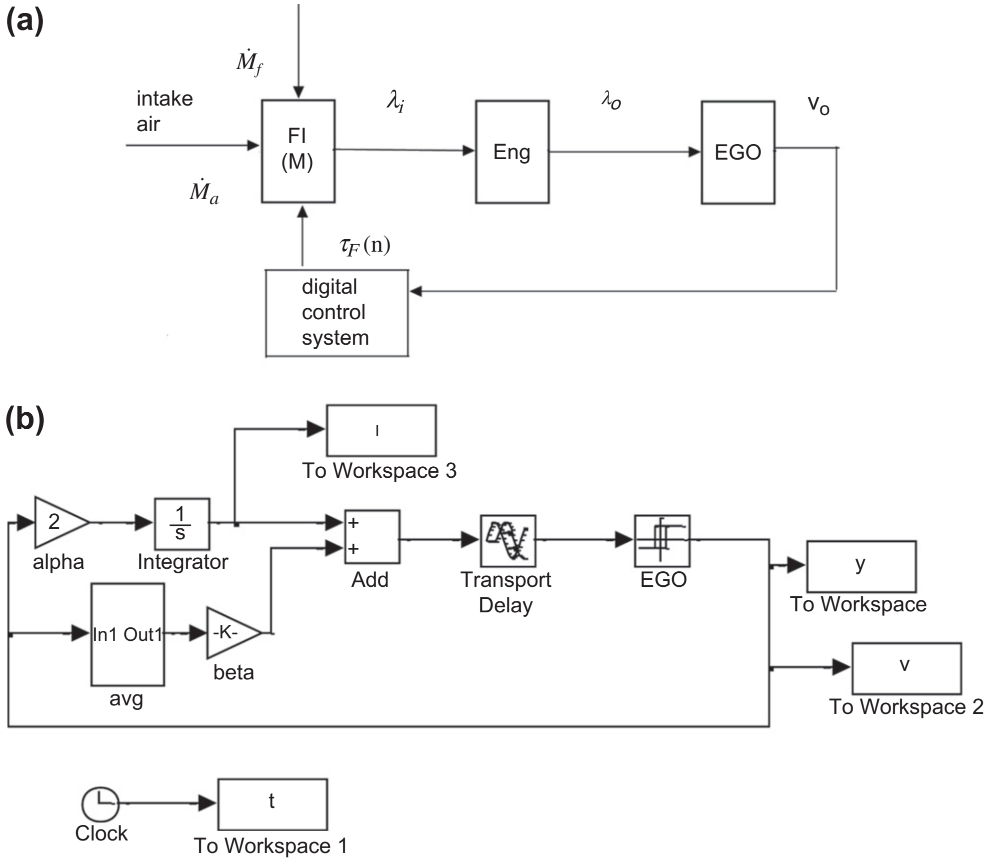


Figure 7.4:
Closed-loop fuel control system.

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

$$\lambda_o(t) \cong \lambda_i(t - T_e) \tag{19}$$

where T_e is the engine cycle time

$$= \frac{120}{\text{RPM}}$$

With this notation, the EGO sensor voltage is given by

$$\begin{aligned} \mathrm{v}_o(\lambda_o) &= V_H \quad \lambda_o < 1(\text{rich mixture}) \\ &= V_L \quad \lambda_o > 1(\text{lean mixture}) \end{aligned} \tag{20}$$

where V_H is the EGO sensor "high" level ≈1 volt 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

$$\begin{aligned} I(k+1) &= I(k) - 1 \quad \lambda_0(k) < 1 \\ &= I(k) + 1 \quad \lambda_0(k) > 1 \end{aligned} \tag{21}$$

In this algorithm, the computation time t_k is given by

$$\begin{gathered} t_k = kT_s \quad k = 1, 2 \ldots \\ T_s = \text{sample time} \end{gathered}$$

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 linearly 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:

$$P(n) = \frac{1}{K}\left[\sum_{k=1}^{K} \mathrm{v}_o(t_n - t_k)\right] - \mathrm{v}_{\mathrm{om}} \tag{22}$$

where v_{om} is the EGO sensor mid-range 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 Figure 7.4b. 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 $\alpha = 2T_s$, $\beta = 0.025$.

The simulation block diagram uses an ideal model for the EGO sensor (see Figure 6.23), combined with the integral control logic of Eqn (21). 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 Eqn (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 0.1 air/fuel ratio for this ideal sensor. The time delay is $T_e = 0.067$ s and is implemented in a transport delay SIMULINK block.

Figure 7.5 is a sample of the waveform where the solid curve is the EGO sensor output voltage and the dashed curve is the integral portion of the C_L and the deviation of the air/fuel ratio is the dash-dot curve. Note that this deviation is ±0.1 air/fuel ratios.

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 Figure 7.5, it can be seen that the air/fuel oscillates within ±0.1 air/fuel ratio of stoichiometry (14.7). This performance should be sufficient that the tail pipe gases after passing through the three-way converter should meet government-mandated limits.

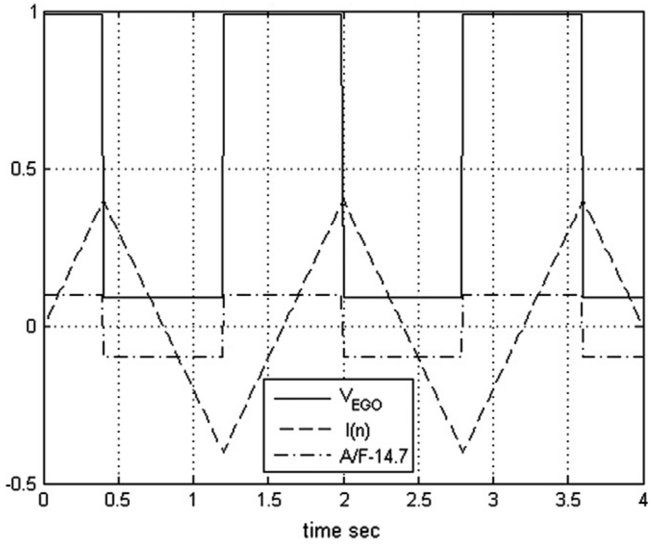


Figure 7.5:
Example limit cycle operation.

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, 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

$$r_T(k) = \frac{\theta_T(k) - \theta_T(k-1)}{T_s} \tag{23}$$

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

$$F(r_T) = F_d(r_T) \quad r_T < r_{Td} \tag{24}$$

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 Figure 7.6a, 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 are powered by separate driver circuits. The configuration of a stepper motor is similar to that of a brushless DC motor as explained in Chapter 6 (see Figure 6.36). Such a motor can be operated in either direction by supplying pulses in the proper phase to the windings as explained in Chapter 6. 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 Figure 7.6a. The variables have the same notation as given in Chapter 5.

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 5, an idle speed control system (ISC) was introduced based upon the continuous time control theory of Chapter 1. As explained in Chapter 5, 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 many

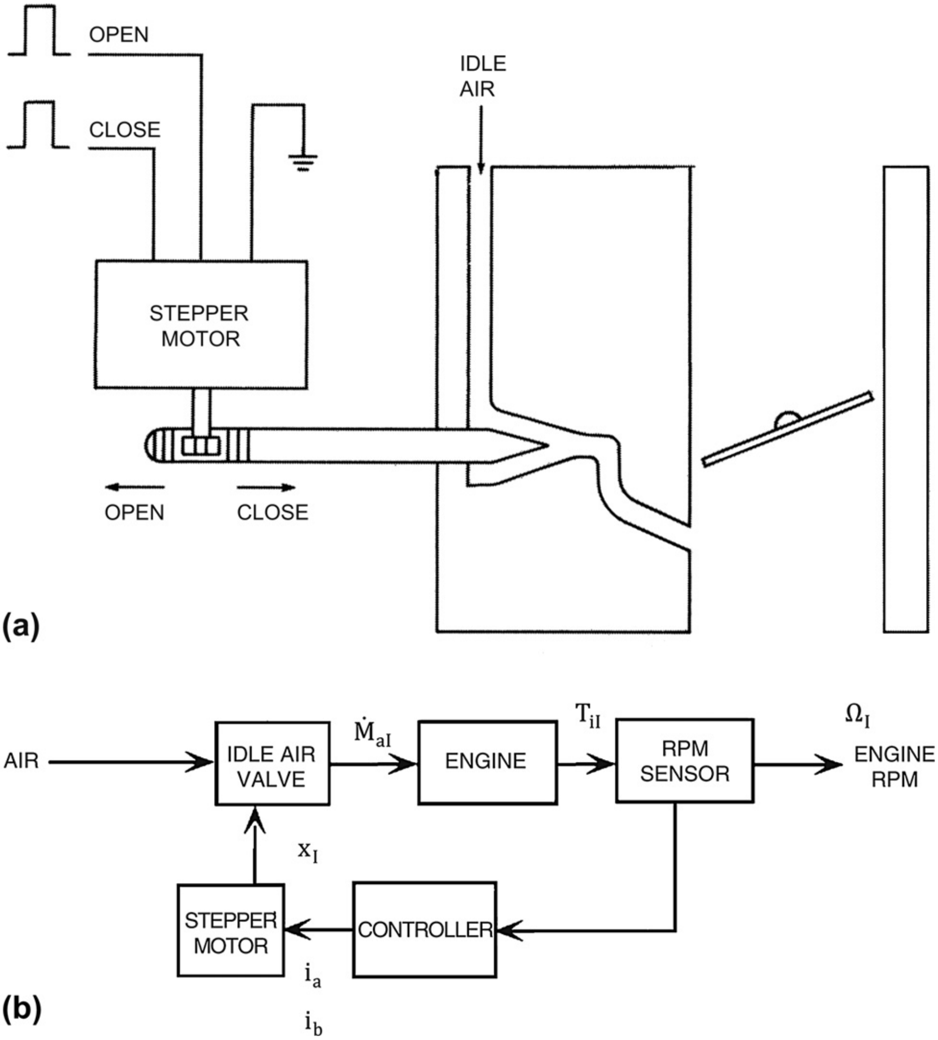


Figure 7.6:
Idle speed control system.

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 5. Figure 7.7 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.

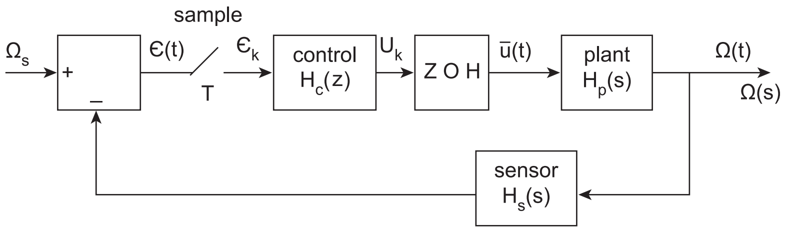


Figure 7.7:
Discrete time idle speed control block diagram.

The present discussion is an example of discrete time control introduced in Chapter 2. 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 Chapter 1, it was shown in Chapter 5, that, for the example ISC, the plant transfer function $H_p(s)$ is given by

$$H_p(s) = \frac{5000}{s^3 + 35s^2 + 875s + 6250} \tag{25}$$

The desired idle angular speed (or set point for the controller) is denoted Ω_s in Figure 7.7. 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 $\in$:

$$\in(s) = \Omega_s - H_s(s)\Omega(s) \tag{26}$$

In the example of Chapter 5 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 Chapter 2 we assume an ideal sampler/quantizer (i.e. A/D converter) such that the input to the discrete time control system is $\in$:

$$\in_k = \in(kT) \quad k = 1, 2, 3\ldots \tag{27}$$

We further assume that in keeping with the continuous time system, the control is PI. The continuous time model for the control system is given by its operational transfer function:

$$\begin{aligned} H_c(s) &= \frac{u(s)}{\in(s)} \\ H_c(s) &= K_p + \frac{K_I}{s} \end{aligned} \tag{28}$$

In the time domain, the control variable $u(t)$ can be written as

$$u(t) = K_p \in (t) + K_I \int_o^t \in (t')\mathrm{d}t'$$

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 is the 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:

$$\int_o^{t_k} \in (t)\mathrm{d}t \cong \int_o^{t_{k-1}} \in (t)\mathrm{d}t + \frac{T}{2}(\in (t_k) + \in (t_{k-1})) \tag{29}$$

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:

$$u_k = u_{k-1} + \frac{K_I T}{2}(\in_k + \in_{k-1}) \tag{30}$$

Taking the z-transform of this equation yields the following expression:

$$u_I(z) = z^{-1}u_I(z) + \frac{K_1 T(1 - z^{-1})}{2} \in (z) \tag{31}$$

This equation can be rewritten as

$$u_{\mathrm{I}}(z) = \frac{K_{\mathrm{I}} T}{2} \frac{(z+1)}{(z-1)} \in (z) \tag{32}$$

It can be shown that the z-transform operational transfer function $H_c(z)$ is given by

$$\begin{aligned} H_c(z) &= \frac{u(z)}{\in (z)} \\ H_c(z) &= K_p + \frac{K_I T(z+1)}{2(z-1)} = \frac{(K_p + K_I T/2)z + (K_I T/2 - K_p)}{(z-1)} \end{aligned} \tag{33}$$

The controller outputs a sequence $\{u_k\}$ control signal that is converted to a piecewise continuous time control signal $\overline{u}(t)$ via the ZOH (see Chapter 2) which operates the plant actuator.

It was also shown in Chapter 2 that the z-transform operational transfer function of the combination ZOH and plant $G(z)$ is given by

$$G(z) = \left(1 - z^{-1}\right)\mathscr{Z}\left(\frac{H_p(s)}{s}\right) \tag{34}$$

As shown in Chapter 2 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 Chapter 2 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. It can be shown using this procedure that $G(z)$ for the example system with sample time $T = 0.01$ s is given by

$$G(z) = \frac{10^{-3}(0.762z^2 + 2.788z + 0.6391)}{(z^3 - 2.629z^2 + 2.3381z - 0.7040)}$$

The z-transform operational transfer function for the forward path $H_F(z)$ is given by

$$\begin{aligned} H_F(z) &= H_c(z)G(z) \\ &= \frac{10^{-3}[0.9563z^3 + 2.6247z^2 - 2.3879z - 0.7323]}{z^4 - 3.6286z^3 + 4.9672z^2 - 3.0432z + 0.704} \end{aligned} \tag{35}$$

The closed-loop transfer function $H_{\mathrm{CL}}(z)$ (as explained in Chapter 2) is given by

$$H_{\mathrm{CL}}(z) = \frac{H_c(z)G(z)}{1 + H_c(z)G(z)} \tag{36}$$

It can be shown that, using the parameters of the example of Chapter 5, $H_{\mathrm{CL}}(z)$ is given by

$$H_{\mathrm{CL}}(z) = \frac{10^{-3}(0.9563z^3 + 2.624z^2 - 2.3879z - 0.7323)}{z^4 - 3.629z^3 + 4.9698z^2 - 3.0456z + 0.7040} \tag{37}$$

The four poles of $H_{\mathrm{CL}}(z)$ are given by

$$\begin{aligned} z_1 &= 0.9236 + 0.1912\mathrm{i} \\ z_2 &= 0.9236 - 0.1912\mathrm{i} \\ z_3 &= 0.9246 \\ z_4 &= 0.8559 \end{aligned}$$

All four poles are inside the unit circle ($|z| = 1$) in the complex z-plane so the system is stable.

In Chapter 5, 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

$$\Omega_S(t) = 550 + 50u(t) \tag{38}$$

where $u(t) =$ unit step at $t = 0$. The z-transform of the ISC dynamic response to this input is given by

$$\begin{aligned}\Omega(z) &= 550 + 50\frac{z}{z-1}H_{\mathrm{CL}}(z)\\ &= 550 + \frac{0.05(0.9563z^4 + 2.624z^3 - 2.3879z^2 - 0.7323z)}{z^5 - 4.622z^4 + 8.5975z^3 - 8.0154z^2 + 3.7496z - 0.7040}\end{aligned} \tag{39}$$

The system output at times t_k can be found by writing the partial fraction expansion for the product $y(z)$:

$$y(z) = \frac{zH_{\mathrm{CL}}(z)}{z-1} \tag{40}$$

As shown in Chapter 2 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 which is denoted y_k was shown in Chapter 2 (by equating coefficients of z^{-k} on both sides of the above equation) to be given by

$$y_k = \sum_{m=1}^{5} \alpha_m {z_m}^{k-1}$$

Figure 7.8 is a plot of $\Omega(t_k)$ where

$$\Omega(t_k) = 550 + 50y_k$$

A comparison of Figure 5.29 in Chapter 5 with Figure 7.8 shows that the dynamic performance of the discrete time digital version of ISC is nearly identical with the

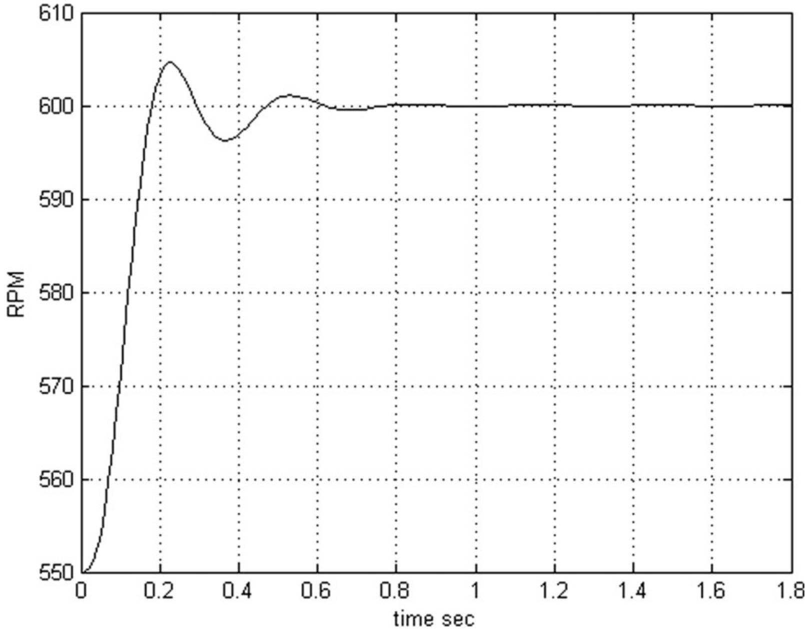


Figure 7.8:
Step response of discrete time idle speed control.

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. 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 5, 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 NOx 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 NOx 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 6. 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).

A simplified block diagram for an exemplary EGR control system is depicted in Figure 7.9. 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 6. 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

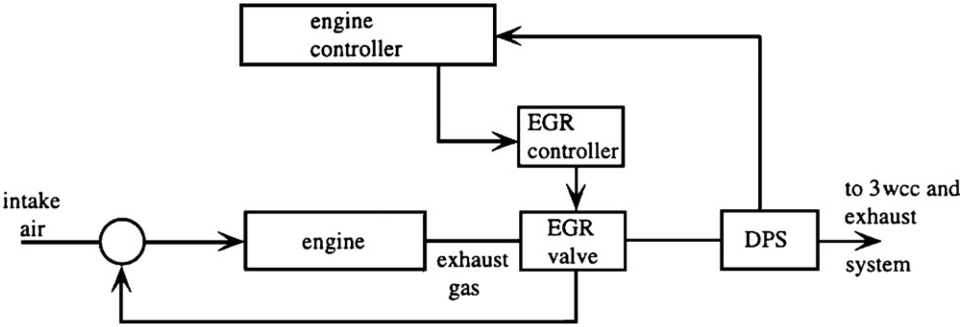


Figure 7.9:
EGR control block diagram.

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 5 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 6, a representative mechanism was discussed for varying camshaft phasing that can be used for varying either/both intake and exhaust camshaft phasing. This system improves volumetric efficiency by varying valve overlap from exhaust closing to intake opening as well as the absolute phase of valve opening and closing. In addition to improving volumetric efficiency, this variable valve phasing 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 6). Load is measured either using MAF as well as RPM or via an MAP sensor (see Chapter 6).

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 6, 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 variable valve phasing 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 6. For the following discussion of VVP, it is assumed that such a sensor is part of the system. Figure 7.10a depicts a physical configuration of a representative camshaft phasing control system.

Control of a variable valve phasing (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 6 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 Figure 7.10a. The dynamic response of the VVP control system should be robust with respect

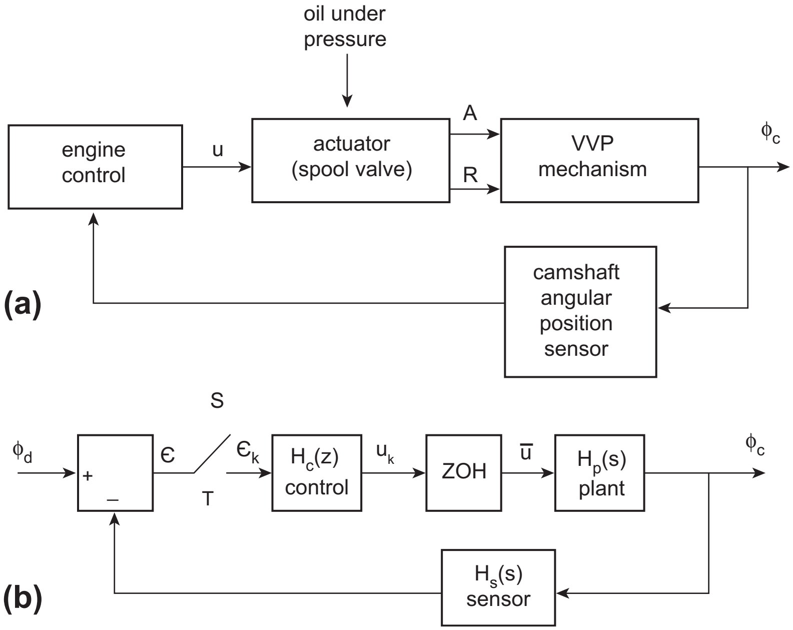


Figure 7.10:
Physical configuration and block diagram of VVP system.

to oil viscosity, which changes with changes in engine temperature; that is, the closed-loop gain for the control system should have large gain and phase margins (see Chapter 1).

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 Figure 7.10b. As explained in Chapter 2, a discrete time control system that regulates a continuous time plant requires a sample and A/D converter as well as a zero order hold (ZOH), both of which are incorporated in the block diagram of Figure 7.10b. 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 6, it was shown that the plant transfer function for this VVP configuration is given by

$$H_p(s) = \frac{K_a}{s(s+s_o)} \tag{41}$$

For the present example, the following parameters are chosen:

$$\begin{aligned} K_a &= 2600 \\ s_o &= 17 \end{aligned}$$

A PID control law is selected to provide sufficient flexibility to meet design objectives. The continuous time PID control law is given by

$$u = K_p \in + K_D \frac{\mathrm{d}\in}{\mathrm{d}t} + K_I \int \in \mathrm{d}t \tag{42}$$

Using the root locus techniques of Chapter 1, the following gain parameters are given which satisfy the overshoot and response time criteria:

$$\begin{aligned} K_p &= 0.080 \\ K_D/K_p &= 0.020 \\ K_I/K_p &= 0.100 \end{aligned}$$

One of the requirements for the VVP control system is robust stability; in Chapter 1, it was shown that robustness is expressed meaningfully by gain and phase margins as determined by the bode plot for the product $H_c(s)H_p(s)$. Figure 7.11 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 .02 rad/sec where the gain margin is more than 100 dB. This system has very robust stability.

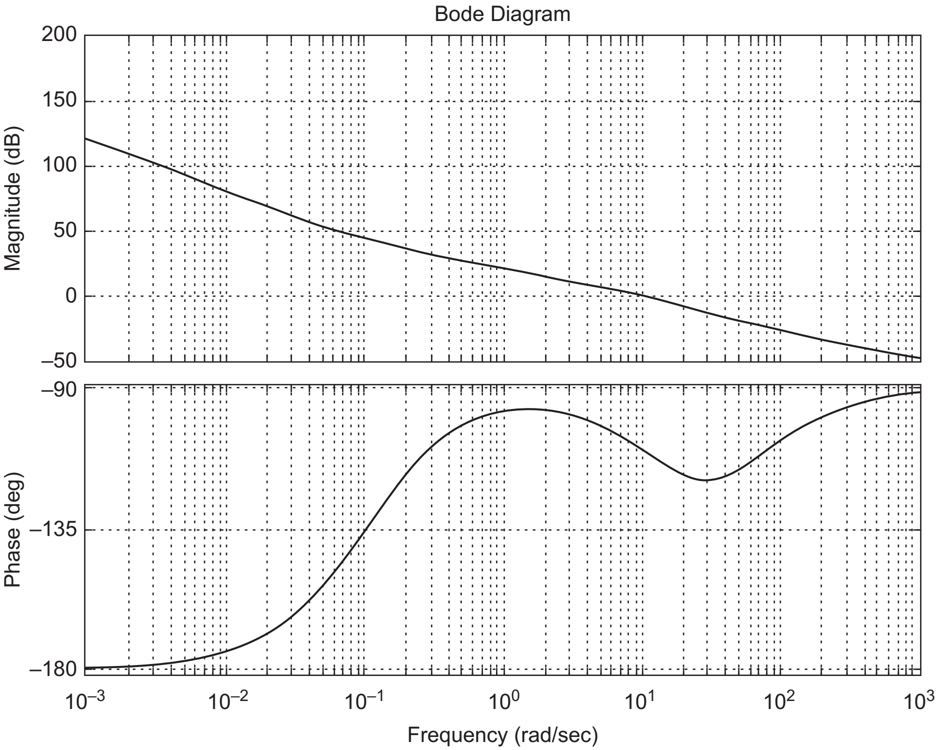


Figure 7.11:
Bode plot of $H_F(s)$ for VVP system.

The discrete time model for the control system is given by

$$u_k = K_p \in_k + \frac{K_D}{T}(\in_k - \in_{k-1}) + K_I u_{kI} \tag{43}$$

where $\in_k = \in(t_k),\ t_k = kT \quad k = 1, 2 \ldots,$ and T is the 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

$$\begin{aligned} u_{kI}(z) &= K_I \mathscr{Z}\left[\int \in \mathrm{d}t\right] \\ &= \frac{K_I T(z+1)}{2(z-1)} \end{aligned}$$

Combining all three terms in the control variable u_k of Eqn (37), the control system transfer function $H_c(z)$ is given by

$$H_c(z) = \frac{u(z)}{\in(z)}$$

$$\begin{aligned} H_c(z) &= K_p + \frac{K_D(z-1)}{Tz} + \frac{K_I T}{2}\frac{(z+1)}{(z-1)} \\ &= \frac{\left[\left(K_p + \frac{K_D}{T} + \frac{K_I T}{2}\right)z^2 - \left(K_p + \frac{2K_D}{T} - \frac{K_I T}{2}\right)z + \frac{K_D}{T}\right]}{z(z-1)} \end{aligned} \tag{44}$$

The plant and ZOH z-transform operational transfer function $G(z)$ is found using the method given in Chapter 2:

$$G(z) = (1 - z^{-1})\mathscr{Z}\left(\frac{H_p(s)}{s}\right)$$

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

$$\frac{H_p(s)}{s} = \frac{K_a}{s^2(s+s_o)} \tag{45}$$

This function has a double pole at $s = 0$. Using the parameters for the plant given above, the partial fraction expansion is given by

$$\frac{H_p(s)}{s} = \frac{8.9965}{s+17} - \frac{8.9965}{s} + \frac{152.94}{s^2} \tag{46}$$

Using the tables of z-transforms from Chapter 2 and assuming a sample period $T = 0.01$ sec, the operational transfer function $G(z)$ is given by

$$G(z) = (1 - z^{-1})\left[\frac{8.9965z}{z - z_1} - \frac{8.9965z}{z-1} + \frac{152.94zT}{(z-1)^2}\right] \tag{47}$$

where $\begin{aligned} z_1 &= e^{-s_0 T} \\ &= 0.6538. \end{aligned}$

The poles of $G(z)$ are all on or in the unit circle which assures a stable system with a combined transfer function:

$$G(z) = \frac{0.7087z + 0.6152}{z^2 - 1.6538z + 0.6538} \tag{48}$$

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

$$\begin{aligned} H_F(z) &= \frac{\phi_c(z)}{\in(z)} \\ H_F(z) &= H_c(z)G(z) \qquad (49) \\ &= \frac{0.1021z^3 - 0.0587z^2 - 0.0825z + 0.0394}{z^4 - 2.6538z^3 + 2.3075z^2 - 0.6538z} \end{aligned}$$

The closed-loop z-transfer function $H_{cl}(z)$ is given by

$$H_{cl}(z) = \frac{\phi_c(z)}{\phi_d(z)} \qquad (50)$$

$$= \frac{H_F(z)}{1 + H_F(z)}$$

$$= \frac{0.1021z^3 - 0.0587z^2 - 0.0825z + 0.0394}{z^4 - 2.5517z^3 + 2.2489z^2 - 0.7363z + 0.0394} \qquad (51)$$

The poles of the closed-loop transfer function are given by

$$\begin{aligned} z_1 &= 0.9975 \\ z_2 &= 0.7442 + 0.2166\mathrm{i} \\ z_3 &= 0.7442 - 0.2166\mathrm{i} \\ z_4 &= 0.0657 \end{aligned}$$

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

$$\begin{aligned} \phi_d &= 0 \qquad t < 0 \\ &= 10^\circ \quad t \geq 0 \end{aligned}$$

The z-transform of ϕ_d is given by

$$\Phi_d(z) = \frac{10z}{z-1}$$

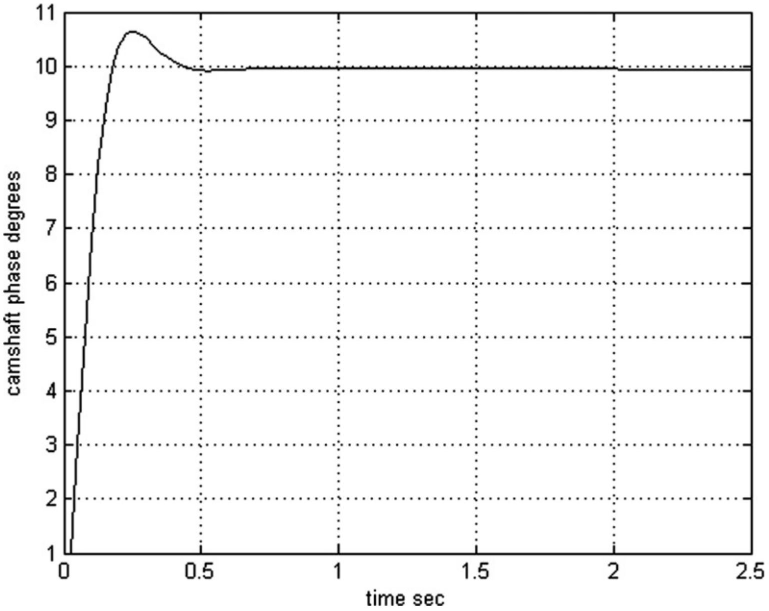


Figure 7.12:
VVP response to 10-degree step command.

The camshaft phase (i.e., system output) is given by

$$\Phi_c(z) = H_{\text{CL}}(z)\Phi_d(z)$$

The output sequence $\phi_c(k)$ at time t_k is found using the method of finding the inverse z-transform explained in Chapter 2. 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.

Figure 7.12 is a plot of this sequence vs. time t_k: The transient response error has essentially decayed to zero in less than 0.5 sec and the overshoot is 6.5%. Thus, this digital variable camshaft phase control system meets the original objectives.

Electronic Ignition Control

As explained in Chapter 5, 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 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 which 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:* In some cases, the coil is on the spark plug as an integral unit.)

Figure 7.13 illustrates such a system for an example 4-cylinder engine.

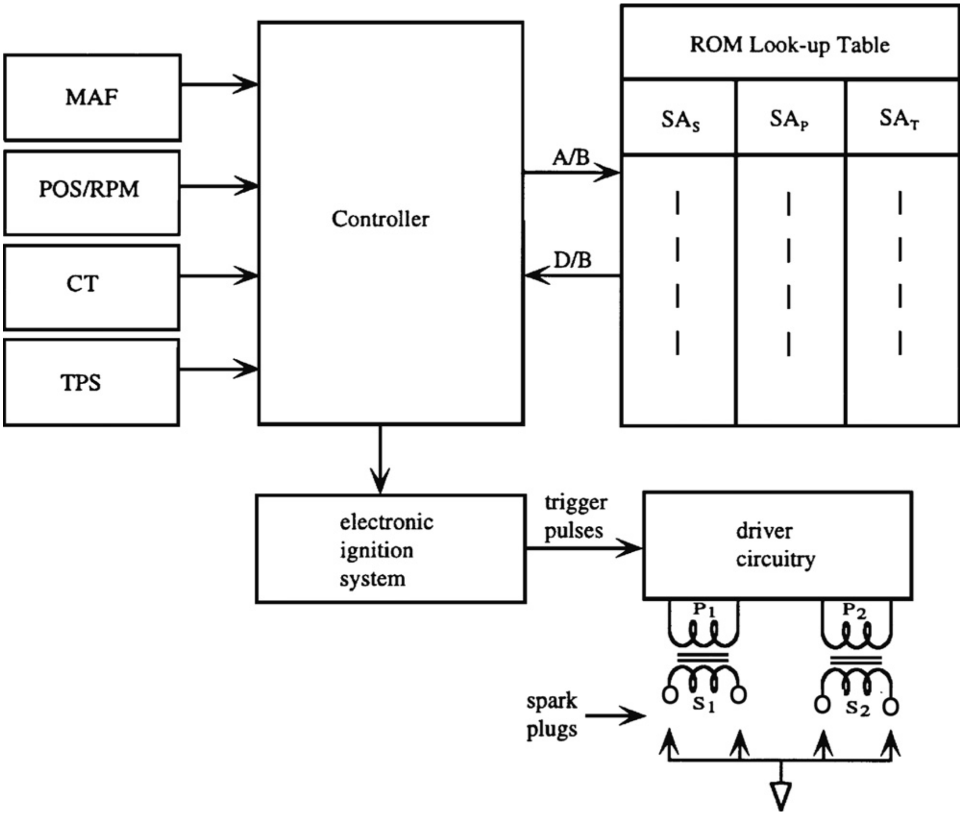


Figure 7.13:
Example integrator circuit diagram.

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.

Although the mixture for contemporary vehicle engines is constrained by emissions 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 5, this optimum ignition timing is known as "spark advance for mean best torque" which 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 Figure 7.13, 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 Figure 7.13, 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 Figure 7.13. These primary currents build up during the so-called *dwell period* before the spark is to occur. The process of spark generating for ignition purposes was explained in Chapter 6. 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 Figure 7.13, a pair of coil packs, each firing two spark plugs, is shown. Such a configuration would be appropriate for a 4-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:

$$SA = SA_S + SA_P + SA_T \tag{52}$$

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. Figure 7.14 depicts a representative variation in SA_S vs. 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.

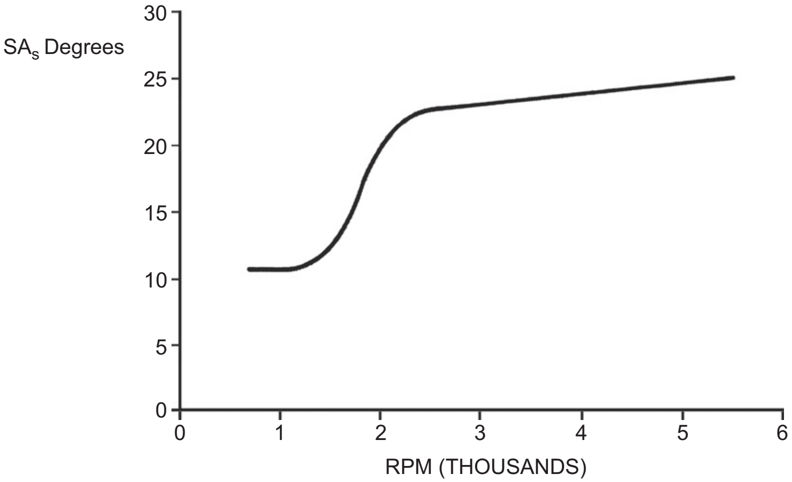


Figure 7.14:
Representative SA curve versus RPM.

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 beyond the scope of this book.

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 Figure 7.15.

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 5, this spark advance is known as the SA for *mean best torque*, or MBT.

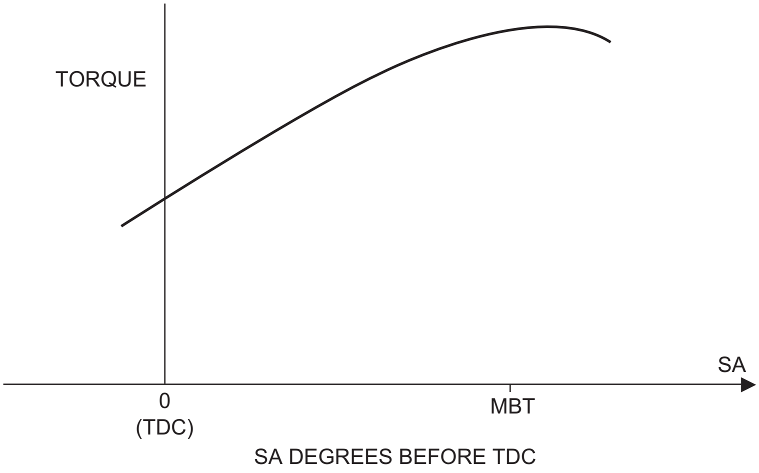


Figure 7.15:
Engine brake torque vs. SA.

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. Figure 7.16 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 6. This sensor responds to the acoustical energy in the spectrum of the rapid cylinder pressure oscillations, as shown in Figure 7.16.

Figure 7.17 is a diagram of an exemplary instrumentation system for measuring knock intensity. Output voltage V_E of the knock sensor is proportional to the acoustical energy in the engine block at the sensor mounting point. This voltage is sent to a narrow bandpass 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

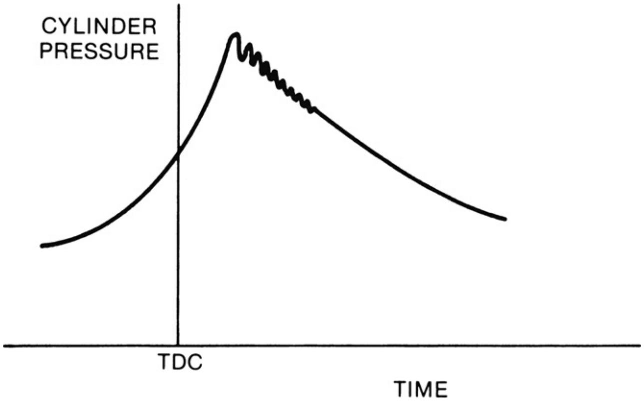


Figure 7.16:
Cylinder pressure under knock conditions.

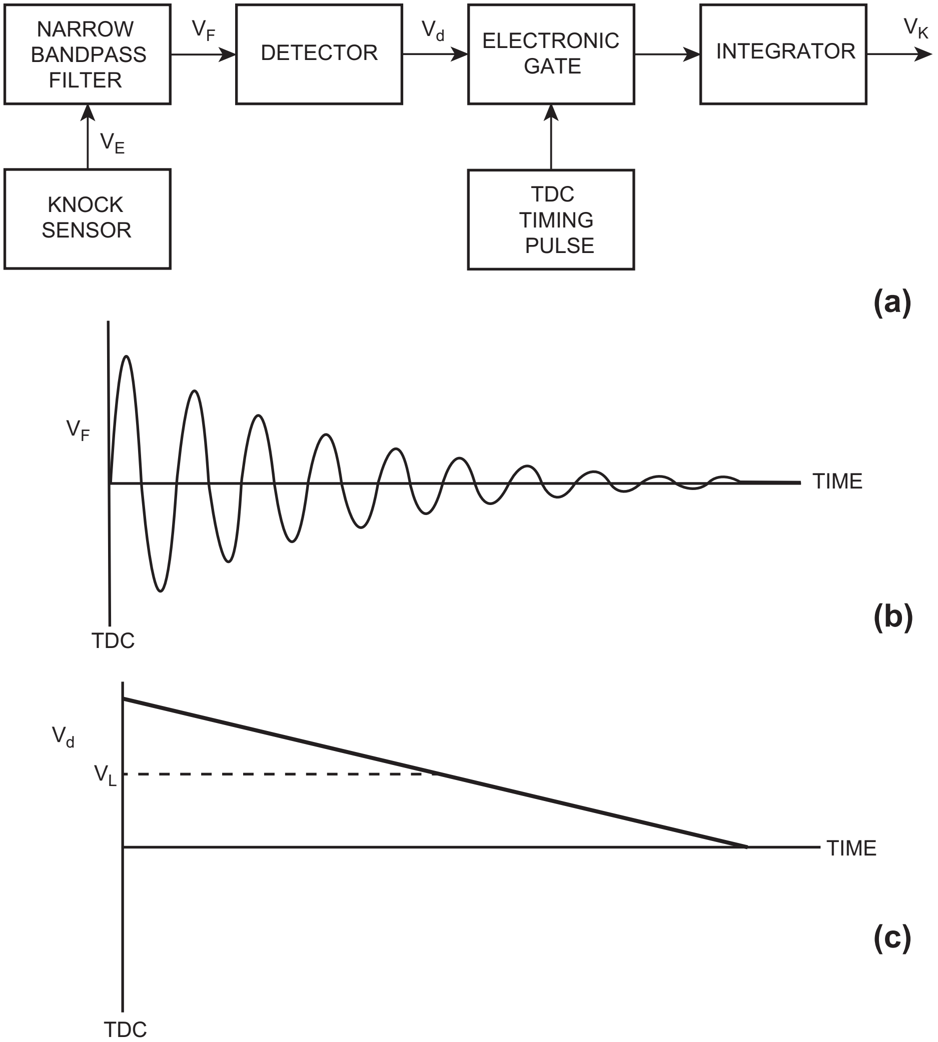


Figure 7.17:
Instrumentation for measuring knock intensity.

which can, for example, be implemented with a rectifier-type circuit which includes a diode and a capacitor (see Chapter 3).

Following the detector in the circuit of Figure 7.17 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 Figure 7.18a 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

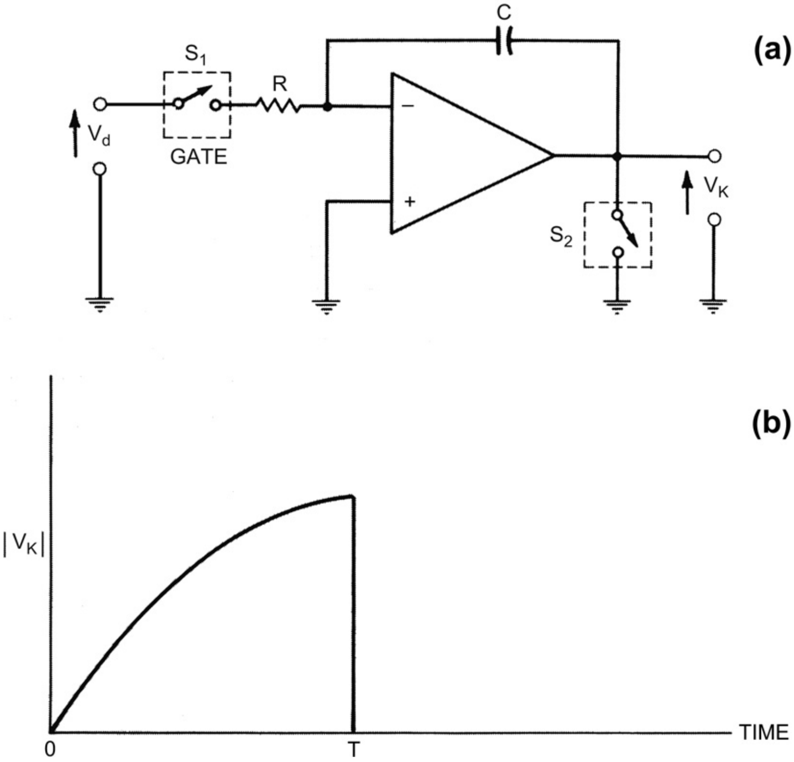


Figure 7.18:
Analog integrator for knock detection system.

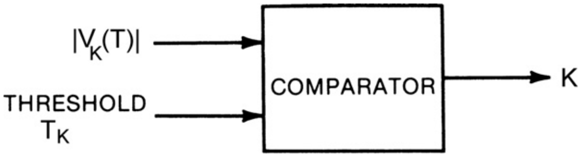


Figure 7.19:
Comparator for knock detector.

$$V_K = -(1/RC) \int_0^T V_d(t)\mathrm{d}t \tag{53}$$

This voltage increases sharply in magnitude but is negative for V_d as depicted in Figure 7.18b because the input is connected to the op amp inverting input. Figure 7.18b is a plot of the absolute magnitude of V_k (i.e., $|V_k|$). This voltage reaches a maximum amplitude at the end of the gate interval, as shown in Figure 7.18b, 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 Figure 7.19).

In such an implementation, the controller generates a binary-valued variable (denoted K in Figure 7.19) having the following algorithm:

$$\begin{aligned} K &= 0 \quad |V_K(T)| < T_K \\ &= 1 \quad |V_K(T)| > T_K \end{aligned} \tag{54}$$

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.

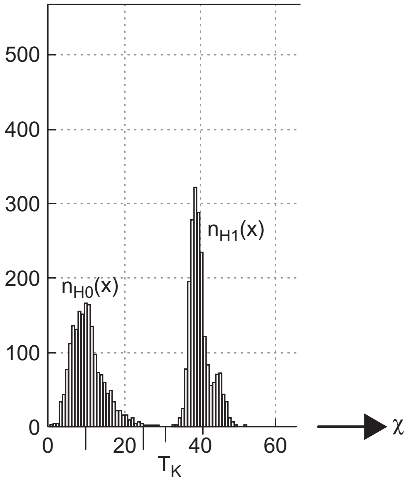


Figure 7.20:
Histogram for hypotheses H_0 and H_1.

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 Figure 7.20 for voltage $V_K(T)$ for a large sample of engine cycles under the two hypotheses:

H_0: normal combustion and
H_1: knocking conditions.

For notational convenience, we let $x = |V_K(T)|$ in Figure 7.20. 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.

The detection threshold T_K is depicted in Figure 7.20. The total probability of error P_e for our example knock detection method is given by

$$P_e = \int_{T_K}^{\infty} p_{H_0}(x)\mathrm{d}x + \int_{O}^{T_K} p_{H_1}(x)\mathrm{d}x \tag{55}$$

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 acoustical noises in the engine increase with RPM.

Spark Advance Correction Scheme

Although the details of spark advance control vary from manufacturer to manufacturer, 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°) whenever knock is detected. Then the spark advance is incremented in one-degree 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 degree of spark advance to cause knocking, more than one degree must be removed to eliminate knocking). The fast correction scheme is depicted qualitatively by the waveform depicted in Figure 7.21.

In the slow correction scheme (Figure 7.22), 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-degree 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).

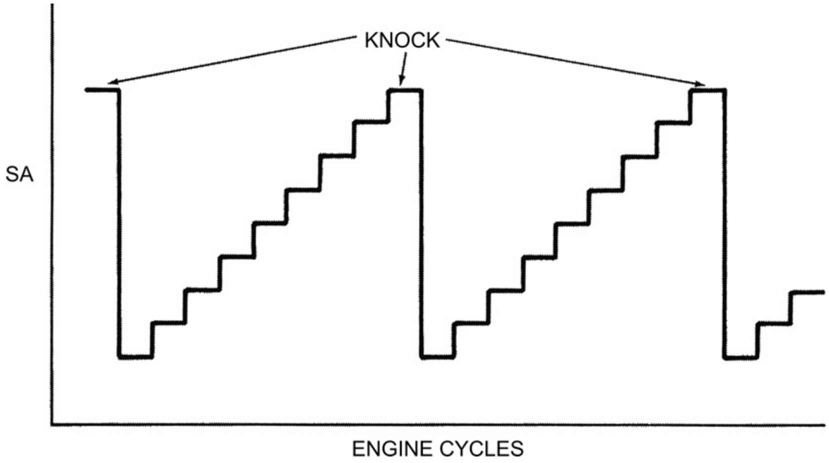


Figure 7.21:
Fast correction of SA.

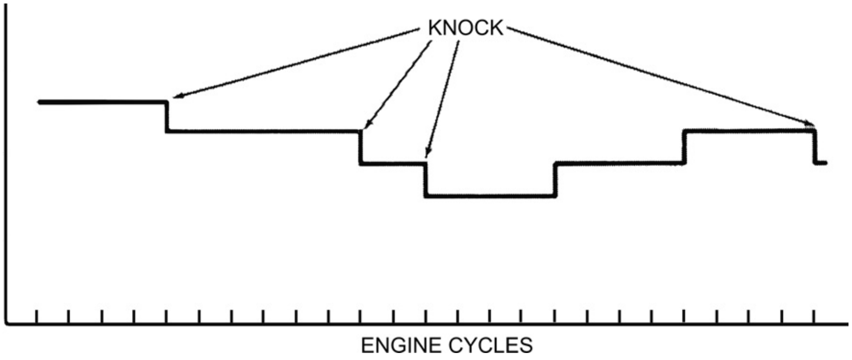


Figure 7.22:
Slow correction of SA.

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 NOx. 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 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 NOx 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 Figure 7.23.

Evaporative Emissions Canister Purge

In pre-emission 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.

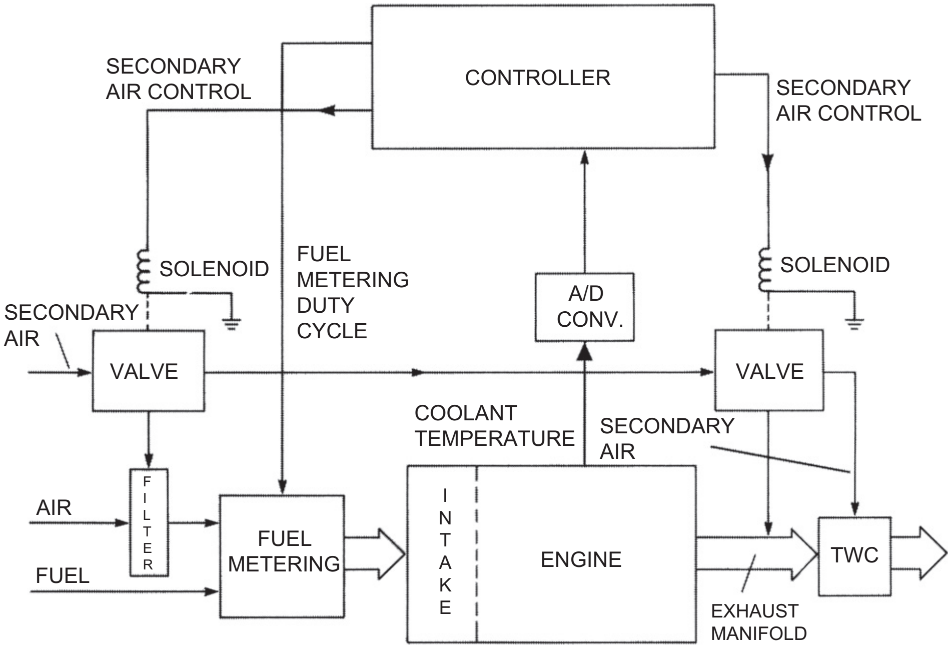


Figure 7.23:
Secondary air system.

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 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 in 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. Keeping the system totally integrated gives the microcomputer controller access to more sensor inputs so they can be checked. Chapter 10 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, and
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,
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, and
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. emissions 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, and
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), 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), and
8. relatively poor emissions 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, and
9. possibly relatively poor fuel economy during idle, but fuel consumption kept to minimum possible (except for hybrid electric vehicle (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 structures of a semitoroidal shape (i.e., a donut-shaped object split along the plane of symmetry). Figure 7.24 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 Figure 7.24. 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

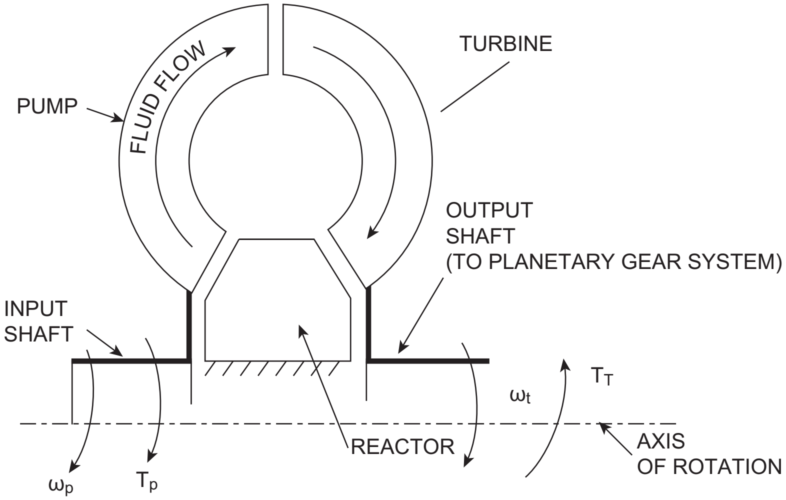


Figure 7.24:
Torque converter configuration.

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 analytical model for a torque converter is given in a paper by Allen Kotwicki.* In this paper, it is explained that a torque converter is a form of fluid coupling device in which a reactor is added which 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 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 Figure 7.24. 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

$$\begin{aligned} T_p &= A\omega_p Q + BQ^2 \\ T_t &= A\omega_p Q - C\omega_t Q + DQ^2 \end{aligned} \tag{56}$$

where ω_p = the pump angular speed (rad/s),

ω_t = the turbine angular speed (rad/s),
Q = the fluid volume flow rate,
$\mathrm{A} = \rho\, R_{px}^2$,
$\mathrm{B} = \rho\left[\frac{R_{px}\tan\alpha_{px}}{A_{px}} - \frac{R_{rx}\tan\alpha_{rx}}{A_{rx}}\right]$,
$\mathrm{C} = \rho\, R_{tx}^2$, and
$\mathrm{D} = \rho\left[\frac{R_{px}}{A_{px}}\tan\alpha_{px} - \frac{R_{rx}}{A_{tx}}\tan\alpha_{tx}\right]$

where ρ is the transmission fluid density.

In these equations a double subscript on a variable means: first subscript p → pump, r → reactor, t → turbine and the second subscript e → entrance, x → 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, and
α is the element blade angle relative to axis.

[*] Dynamic Models for Torque Converter Equipped Vehicles, Allen Kotwicki, SAE paper # 820393, 1982.

It is further shown that the volume flow rate is given by

$$Q = -\frac{(H\omega_t - G\omega_p)}{2I} + \frac{\left[\left(H\omega_t - G\omega_p\right)^2 + 4I\left(E\omega_p^2 + F\omega_t^2\right)\right]^{\frac{1}{2}}}{2I} \tag{57}$$

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

$$\omega_t \cong S\omega_p \text{ is assumed}$$

where S is the speed ratio.

Using this approximation, it is shown in the reference that the torque ratio T_R is given by

$$T_R = \frac{T_t}{T_p} = \frac{(A + D\alpha_1)\omega_p + (D\beta - C)\omega_t}{(A + B\alpha_1)\omega_p + B\beta\omega_t} \tag{58}$$

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 development of transmission controls.

The planetary gear system consists of a set of three types of gears connected together as depicted in Figure 7.25a. 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 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.

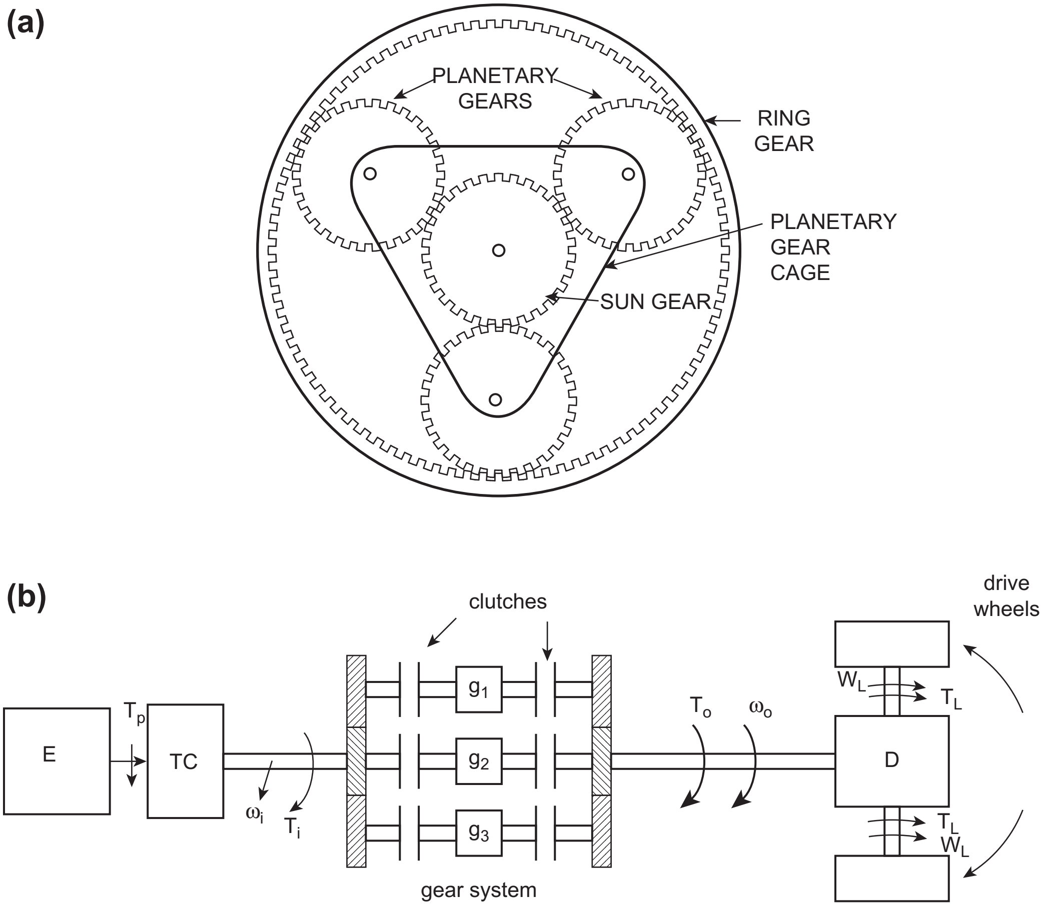


Figure 7.25:
Schematic automatic transmission configuration.

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

$$\begin{aligned} T_o &= gT_i \\ \omega_o &= \omega_i/g \end{aligned} \tag{59}$$

where g is the gear ratio

$$= N_p/N_s$$

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, g_3 in Figure 7.25b), each with its own set of hydraulically actuated clutches as depicted schematically in Figure 7.25b. In an electronically controlled automatic transmission, the clutches are electrically or electrohydraulically actuated via solenoid type actuators such as are described in Chapter 6.

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.

Figure 7.25b 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

$$g_T = g_D \prod_{n=1}^{N} g_n \tag{60}$$

Otherwise, for a parallel connected system of gears as shown in Figure 7.25b, the gear ratio is given by

$$g_T = g_D g_n \tag{61}$$

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 ($P_b = T_b \omega_e$). For example, under certain powertrain operating conditions, the gear ratio, which maximizes this transfer of engine power to load power ($P_L = T_L \omega_L$) $g,^*$ is given approximately by

$$g^* = \sqrt{\frac{T_b/\omega_e}{T_L \omega_L}}$$

The controller selects the nearest available gear ratio from the set of possible choices.

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 piston-like 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 6.

During normal driving, the electronic transmission controller determines the desired gear ratio from measurements of engine load and RPM as well as transmission output shaft RPM. These RPM measurements are made using noncontacting angular speed sensors (usually magnetic in nature) as explained in Chapter 6. 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 lock-up 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 lock-up."

Differential and Traction Control

The transmission output shaft is coupled to the drive axles via the differential. 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. Figure 7.26 depicts the configuration for a differential. Unfortunately, wherever there is a large difference between the tire/road friction from left to right, the differential will tend

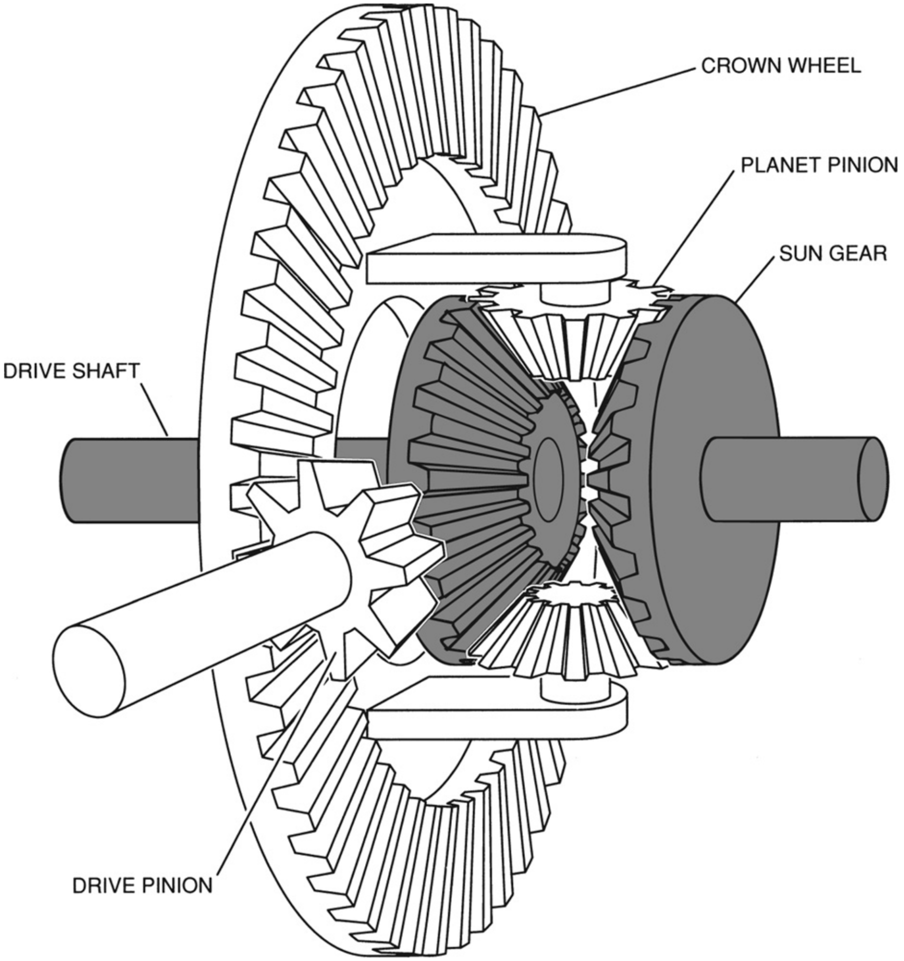


Figure 7.26:
Differential configuration.

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.

Certain cars are equipped with so-called traction control devices that can overcome this disadvantage of the differential. Such cars have 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.

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 wheel spin 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 hybrid electric vehicle (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 5, 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 differential 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 which 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 electrical power to the battery pack. The discussion of induction motors in Chapter 6 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 were to begin. It is highly impractical to do such preprogramming. 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). Figure 7.27 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 electrical 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 electrical 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.

Figure 7.28 is a schematic of a hybrid vehicle type known as a parallel hybrid. The parallel hybrid of Figure 7.28 can operate with ICE alone by engaging both solenoid-operated clutches on either side of the EM but with no electrical 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 electrical power to the EM from the high-voltage bus

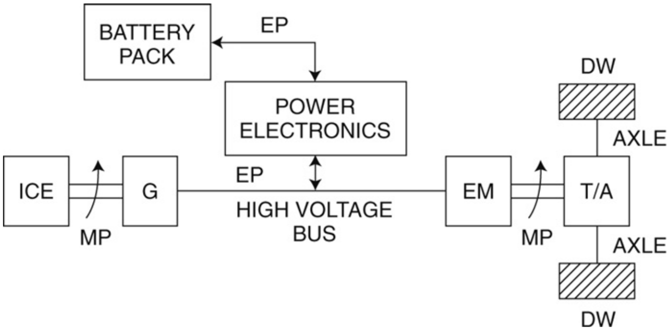


Figure 7.27:
Series HEV schematic.

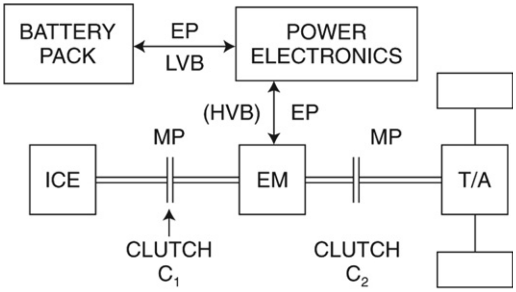


Figure 7.28:
Parallel hybrid schematic.

(HVB). Of course, if both ICE and EM are to produce propulsive power, then both clutches are engaged. Not shown in Figure 7.28 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 Figure 7.29 operates similarly to that of Figure 7.28 except that mechanical power from ICE and EM are combined in a mechanism denoted coupler. For the system of Figure 7.29 pure ICE propulsion involves engaging clutch $C_{1,}$ disengaging clutch C_2, and providing no electrical power to the EM. Alternatively, pure EM propulsion involves disengaging clutch C_1, switching off the ICE, engaging clutch C_2, and providing electrical power to the EM via the high-voltage bus (HVB). Simultaneous ICE and EM propulsion involves running the ICE, providing electrical power to the EM, and engaging both clutches.

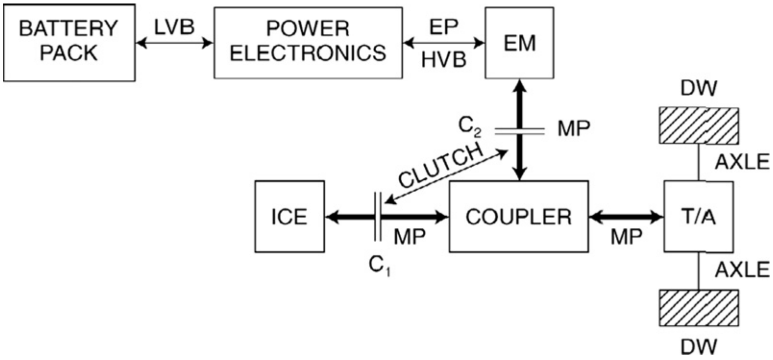


Figure 7.29:
HEV with mechanical coupler.

In principle, any type of electric motor could be used to provide the electric propulsion in a hybrid vehicle. However, in practice, there are two main types in common use today: the brushless DC motor and the induction motor. Both are explained and modeled in Chapter 6. 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 a hybrid vehicle during electric propulsion requires that an electric system convert the stored DC electric power to 3-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 high-power switching transistors). Ideally, the stator excitation should be three sinusoidal voltages of equal amplitude which in phasor notation are given by

$$\begin{aligned} V_A &= Ve^{j\omega_s t} \\ V_B &= Ve^{j(\omega_s t + 2\pi/3)} \\ V_C &= Ve^{j(\omega_s t + 4\pi/3)} \end{aligned} \tag{62}$$

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 6).

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, neodynium.

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 $V_A V_B V_C$ in each winding:

$$V_A = \frac{d\Phi_A}{dt}$$

$$V_B = V_A e^{j(2\pi/3)}$$

$$V_C = V_A e^{j(4\pi/3)}$$

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 3). 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 6 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 6 where it was shown that the frequency condition for average torque generation is automatically satisfied.

Induction motors for hybrid vehicle use are normally three-phase, meaning that three separate windings (one for each phase) are required for both stator and rotor. In Chapter 6 it was shown that the torque produced by the induction machine (with current excitation amplitude I_s) is given by

$$T_e = \frac{(\omega_s - \omega_m)M^2R_r^2I_s^2}{R_r^2 + (\omega_s - \omega_m)^2L_r^2} \tag{63}$$

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 6. It is also shown in Chapter 6 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:

$$T_e(\omega_o) = T_L(\omega_o) \tag{64}$$

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 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

$$F_{\mathrm{rr}} = \mu_{\mathrm{rr}}W_V$$

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 $0.02 \leq .\mu_{\mathrm{rr}} \leq 0.04$.

The aerodynamic drag D is given by

$$D = \frac{\rho}{2} C_D S_{\mathrm{ref}} V^2$$

where ρ is the local air density (kg/m^3 or slug/ft^3), C_D is the drag coefficient, S_{ref} is the a 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

$$F_V = \mu_{\mathrm{rr}} W_V + (1/2)\rho C_D S_{\mathrm{ref}} V^2$$

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

$$F_V = \mu_{\mathrm{rr}} W_V + (1/2)\rho C_D S_{\mathrm{ref}} V^2 + W_V \sin\theta \tag{66}$$

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 θ, 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:

$$T_V = r_T F_V / g_{\mathrm{v}} \tag{67}$$

where g_V is the gear ratio from motor to drive wheels and r_T is the tire effective radius.

Figure 7.30 is a plot of normalized motor torque T_m and load torque T_L (normalized to the maximum motor torque $T_{\max}$) vs. the ratio ω_m/ω_s where

$$T_m = \frac{T_e}{T_{\max}}$$

$$T_L = \frac{T_V}{T_{\max}}$$

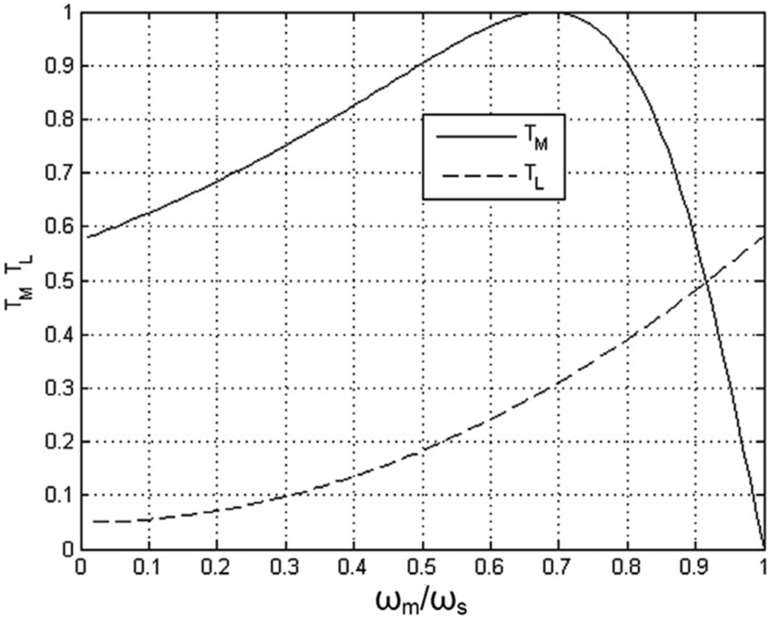


Figure 7.30:
Normalized motor torque T_M vs. normalized load torque T_L.

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.

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, vehicle-operating conditions and driving patterns (e.g., urban or highway), etc.

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 6, the relationship between V_s and I_s is given by

$$\mathrm{V}_s = j\omega_s L_s I_s + \frac{\omega_s^2 M^2 I_s (R_s/s)}{(R_s/s)^2 + \omega_s^2 L_r^2} - j\frac{\omega_s^3 M^2 L_r I_s}{(R_r/s)^2 + \omega_s^2 L_s^2} \tag{68}$$

This expression gives the voltage/current relationships for each phase. See Chapter 6 for the definitions of all parameters. Solving the above equation for I_s and substituting it into the equation for motor torque yields

$$T_e = \frac{(M^2/\omega_s L_s L_r)\,(L_v/L_s)\,(R_r/s)\,V_s^2}{\left[\omega_s\left(1-\dfrac{M^2}{L_r L_s}\right)L_r\right]^2 + (R_r/s)^2} \tag{69}$$

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 (see Figure 7.30). Normal operation of an induction motor is in a region in which

$$\frac{\mathrm{d}T_e}{\mathrm{d}\omega_s} < 0$$

and s is relatively small. In this region, the motor torque is given approximately by

$$T_e \cong \left(\frac{M^2}{\omega_s L_s^2}\right)\,\frac{s}{R_r}\mathrm{V}_s^2 \tag{70}$$

On the other hand, when slip is relatively large the torque can be shown to be given approximately by

$$T_e \cong \frac{M^2 R_r \mathrm{V}_s^2}{(L_r L_s)^2 \omega_s^3 \left(1 - \frac{M^2}{L_r L_s}\right) s} \tag{71}$$

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 as well as 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 electronics 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) and even special capacitors called ultra-caps. 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 which 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 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 at 12 V (nom), a common battery pack might consist of a connection of multiple 12-volt batteries.

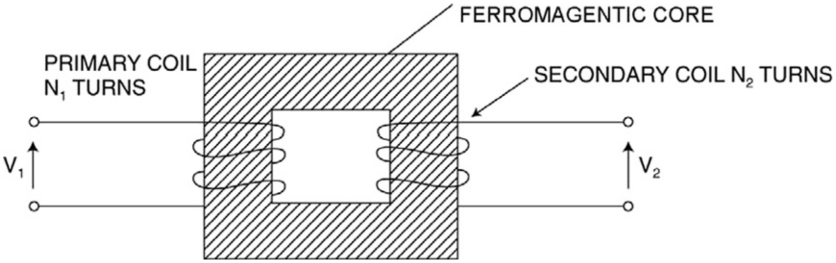


Figure 7.31:
Transformer configuration.

Conversion of electrical 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. Figure 7.31 schematically illustrates transformer structure and the conversion of voltages from one level to another.

A transformer consists of a core of magnetically permeable material (usually 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.

Assuming (arbitrarily) that AC electrical 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 electrical 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. Figure 7.32a 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 supplies 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 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 3). 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.

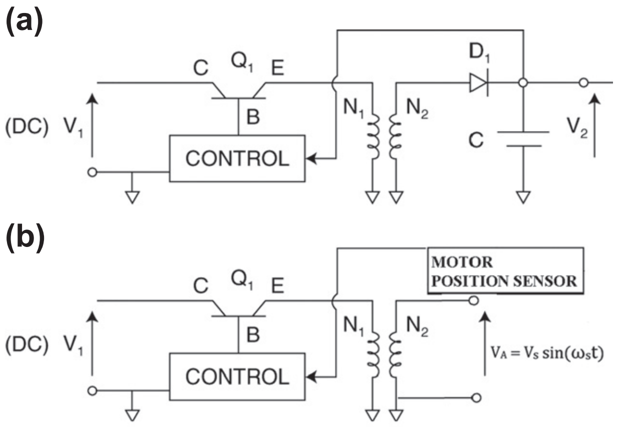


Figure 7.32:
Voltage conversion circuit.

A variation of the circuit of Figure 7.32a 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 V_C for a three-phase motor) can be accomplished using circuits of the configuration shown in Figure 7.32b. 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 computes the signals that control the phases (i.e., 0, 2π/3 and 4π/3) of the remaining three phases. 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 as well as 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 (for brushless DC motors), and
- 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 V_C),
- clutch engage/disengage, and
- 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. 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 as well as 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 8

Vehicle-Motion Controls

Chapter Outline

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. The vertical axis is taken as the *z*-axis with the positive sense up.

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-08-097097-4.00008-4

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 powertrain (including, in selected models, traction control), the brakes, the aerodynamic drag, and 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 powertrain is balanced against all resisting forces to maintain a constant speed. In an advanced cruise control 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 was discussed in general terms in Chapter 1. Recall that the components of a 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 cruise control for a vehicle traveling along a straight road (along the x axis in our ECEF coordinate system). In the case of a cruise control, the variable being regulated is the vehicle speed:

$$V = \frac{\mathrm{d}x}{\mathrm{d}t}$$

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 powertrain) 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 powertrain 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 powertrain has sufficient power capability of maintaining constant vehicle speed on an up-sloping grade.

The plant being controlled consists of the powertrain (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 down-hills.

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 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 lock-up mode (see Chapter 7). This assumption removes some powertrain 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

$$T_w = g_A T_b \tag{1}$$

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 disabled, 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:

$$T_b = K_a u \tag{2}$$

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 Chapter 1 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 (i.e. along the direction of travel and vehicle fore/aft axis) is given by

$$M\dot{V} + D + F_{rr} = \frac{g_A T_b}{r_w} - W_V \sin\theta \tag{3}$$

where

M = vehicle mass
W_V = vehicle weight (gM)
r_w = drive wheel effective radius
$F_{rr} = \mu_r W_V$
μ_r = coefficient of tire-rolling resistance
$.02 \leq \mu_r \leq 0.04$ typically
θ = angle of the road surface relative to a horizontal plane
$D = \frac{\rho}{2} C_D S_{\text{ref}} (V + V_w)^2$
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 Eqn (3), the first term on the right-hand side (RHS) is the force acting on the vehicle due to the applied road torque acting at the tire/road interface due to the powertrain. 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:

$$\rho \frac{C_D S_{\text{ref}}}{2} V_C^2 + \mu_r W_V = g_a \frac{T_b}{r_w} \tag{4}$$

This equation permits a determination of engine brake torque vs. 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:

$$g_A \frac{T_b}{r_W} = \rho \frac{C_D S_{\text{ref}} V_C^2}{2} + \mu_r W_{\text{V}} + Mg \sin\theta \tag{5}$$

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 $V_w = 0$:

$$D = D_C + \delta D \tag{6}$$

$$V(t) = V_C + \delta V$$

where D_C is the drag at speed V_C:

$$\delta D = \left.\frac{\mathrm{d}D}{\mathrm{d}V}\right|_{V_C} \delta V$$

$$= \rho C_D S_{\text{ref}} V_C \delta V$$

$$= K_D \delta_{\text{V}}$$

where K_D is a constant for a given initial steady cruise speed V_C and constant ρ.

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

$$M\delta\dot{\text{V}} + \rho C_D S_{ref} \text{V}_C \delta\text{V} - Mg\theta = g_A \frac{\delta T_b}{r_W} = g_A \frac{K_a u}{r_w} \tag{7}$$

The operational transfer function $H_p(s)$ for the "plant" for zero disturbance (i.e., $\theta = 0$) is given by

$$H_p(s) = \frac{\delta \mathrm{V}(s)}{u(\mathrm{s})} = \frac{K_a g_A/(M r_w)}{s + \rho \frac{C_D S_{ref} \mathrm{V}_c}{M}} \tag{8}$$

The configuration for a representative automotive cruise control is shown in Figure 8.1.

When the vehicle reaches the desired speed under normal driver accelerator pedal regulation of the throttle, to activate cruise control at that speed the driver pushes a momentary contact switch 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 (pushbutton) switch that sets the command speed is denoted S_1 in Figure 8.1.

Also shown in this figure is a disable 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 Figure 8.1 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 disables the cruise control any time that the brake pedal is moved from its rest position. The throttle actuator opens and closes 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.

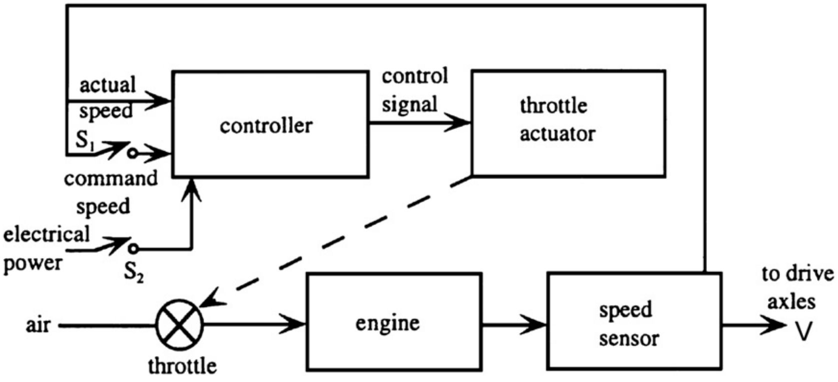


Figure 8.1:
Cruise control configuration.

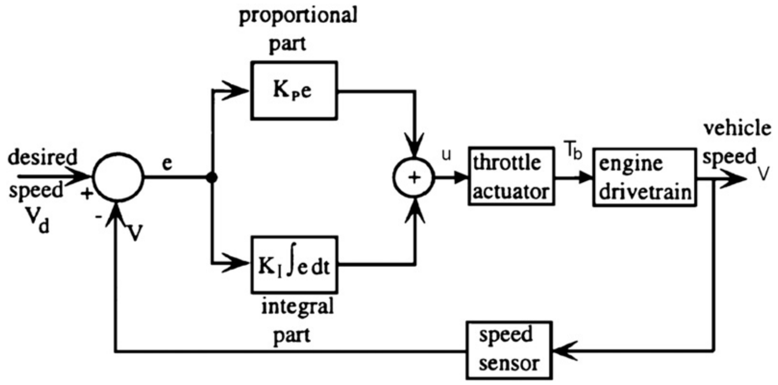


Figure 8.2:
Cruise control block diagram.

A block diagram of a cruise control system is shown in Figure 8.2. 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 which 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 Chapter 1). In this strategy, an error e is formed by subtracting (electronically) the actual speed V from the desired speed V_d:

$$e = V_d - V \tag{9}$$

It should be noted that the speed differential from V_c 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_pe) and a term proportional to the integral of the error:

$$K_1 \int e\mathrm{d}t \tag{10}$$

The actuator signal u is given by

$$u = K_pe + K_I \int e\mathrm{d}t \tag{11}$$

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. 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

$$u = K_p e + K_I \int e\mathrm{d}t \tag{12}$$

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):

$$\delta T_b = K_a u$$

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

$$\begin{aligned} M\delta\dot{\mathrm{V}} + K_D\delta\mathrm{V} + Mg\theta &= \frac{g_A K_a u}{r_w} \\ &= g_A \frac{K_a}{r_W}\left[K_p e + K_I \int e\mathrm{d}t\right] \end{aligned} \tag{13}$$

Taking the Laplace transform of the above equation and solving for the speed differential yield

$$\delta V(s) = -\frac{sg\theta(s)}{\left[s^2 + \left(\frac{K_D}{M} + \frac{g_A K_a K_p}{Mr_w}\right) s + \frac{g_A K_a K_I}{Mr_w}\right]} \tag{14}$$

A computer simulation of this simplified cruise control was done for a step change in grade of $\theta = 0.03$ starting at 2 s into the simulation for the following parameters in English units:

$W_V = 3100 lb$
$C_D = 0.3$
$S_{\mathrm{ref}} = 18\ \mathrm{ft}^2$
$\rho = 0.0024 \mathrm{slug}/ft^3$ (i.e. sea level on a standard day)
$K_A = 10$
$K_p = 10$
$K_I = 50$
$r_w = 1\ ft$

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). Figure 8.3 shows the response for the car initially traveling under cruise control at 60 MPH. 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

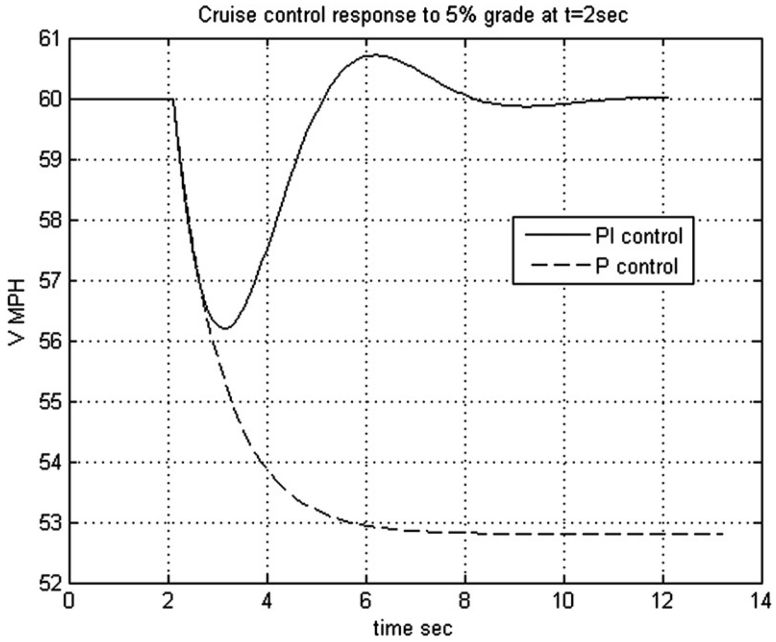


Figure 8.3:
Cruise control speed performance.

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 Chapter 1)).

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 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 Chapter 1, 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 Figure 8.3 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

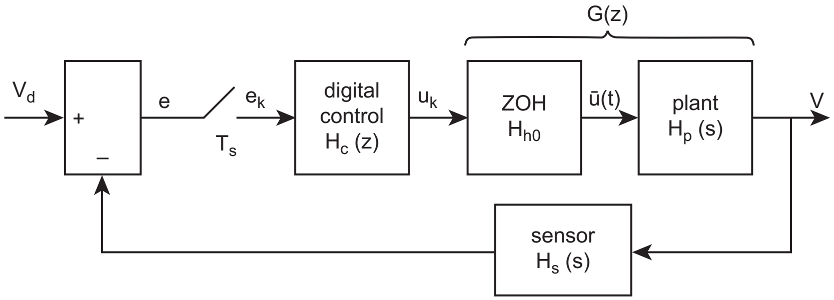


Figure 8.4:
Digital speed control block diagram.

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 Figure 8.4.

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 which is the ZOH output $\overline{\boldsymbol{u}}(t)$ is a piecewise continuous signal (see Chapter 2):

$$\begin{aligned} H_p(s) &= \frac{V(s)}{\overline{u}(s)} \\ &= \frac{g_A K_a}{M r_w (s + K_D/M)} \\ &= \frac{K}{s + s_0} \end{aligned} \tag{15}$$

where

$$K = \frac{g_A K_a}{M r_\omega}$$

$$s_o = K_D / M$$

Using the same parameters as were used for the analog version of the cruise control, this model is given numerically by the following transfer function:

$$H_p(s) = \frac{0.4129}{(s + 0.0118)} \tag{16}$$

As explained in Chapter 2, the z-transfer function for the combination of ZOH and plant ($G(z)$) is given by

$$G(z) = (1 - \mathrm{z}^{-1}) \mathscr{Z} \left(\frac{H_p(s)}{s} \right) \tag{17}$$

From the methods of Chapter 2, the z-transform above can be found by expanding $H_p(s)$/s in a partial fraction series and then using the tables of Chapter 2. Then it is left as an exercise to show that for sample period $T_s = 0.01$ s, $G(z)$ is given by

$$G(z) = \frac{K}{s_o} \left[\frac{(1 - z_0)(z - 1)}{z^2 - (z_0 + 1)z + z_0^2} \right] \tag{18}$$

where

$K_1 = .4129$
$s_o = .0018$
$z_0 = e^{-s_o T}$

The continuous time PI control law is given by

$$u(t) = K_p e(t) + K_I \int e \mathrm{d}t \tag{19}$$

In Chapter 7 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 rate) is given by

$$\mathscr{Z} \left[K_I \int e \mathrm{d}t \right] = \frac{K_I T_s (z + 1)}{2(z - 1)}$$

The z-operational transfer function for the controller is given by

$$H_c(z) = \frac{u(z)}{e(z)} \tag{20}$$

$$H_c(z) = K_p + \frac{K_I T_s(z+1)}{2(z-1)}$$

$$H_c(z) = \frac{\left(K_p + \frac{K_I T}{2}\right)z - \left(K_p - \frac{K_I T}{2}\right)}{(z-1)} \tag{21}$$

Using the same gains ($K_p = 10$ and $K_I = 50$) as for the continuous time control, one obtains

$$H_c(z) = \frac{10.25z - 9.75}{(z-1)} \tag{22}$$

Chapter 2 also showed that the forward path z-transfer function $H_F(z)$ for a discrete time control system as shown in Figure 8.4 is given by

$$\begin{aligned} H_F(\mathrm{z}) &= \frac{\delta V(\mathrm{z})}{e(\mathrm{z})} \\ &= H_c(z)G(z) \\ &= \frac{0.0423\mathrm{z}^2 - 0.0826\mathrm{z} + .0403}{\mathrm{z}^3 - 2.999\mathrm{z}^2 + 2.998\mathrm{z} - 0.9999} \end{aligned} \tag{23}$$

Assuming an ideal sensor for which $H_s(s) = 1$, the closed-loop gain z-transform function $H_{CL}(z)$ is given by

$$\begin{aligned} H_{CL}(\mathrm{z}) &= \frac{H_F(\mathrm{z})}{1 + H_F(\mathrm{z})} \\ &= \frac{0.0423\mathrm{z}^2 - 0.0826\mathrm{z} + .0403}{\mathrm{z}^3 - 2.9576\mathrm{z}^2 + 2.9172\mathrm{z} - 0.9596} \end{aligned} \tag{24}$$

The poles of this closed-loop transfer function are

$$\begin{aligned} \mathrm{z}_1 &= 1.000 \\ \mathrm{z}_2 &= 0.9788 + 0.0402i \\ \mathrm{z}_3 &= 0.9788 - 0.0402i \end{aligned}$$

Since all poles are either on or inside the unit circle ($|z| = 1$), the closed-loop cruise control system is stable.

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 to 70 MPH. This system set point is given by

$$\begin{aligned} V_d &= 60 \quad t < 2 \\ &= 70 \quad t \geq 2 \end{aligned} \tag{25}$$

The z-transform for this system input is given by

$$V_d(z) = 60 + \frac{10z}{z-1} \tag{26}$$

The output z-transform $V(z)$ is given by

$$V(z) = H_{CL}(z) V_d(z) \tag{27}$$

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 Chapter 2, the time response at $t = t_k$ is shown in Figure 8.5.

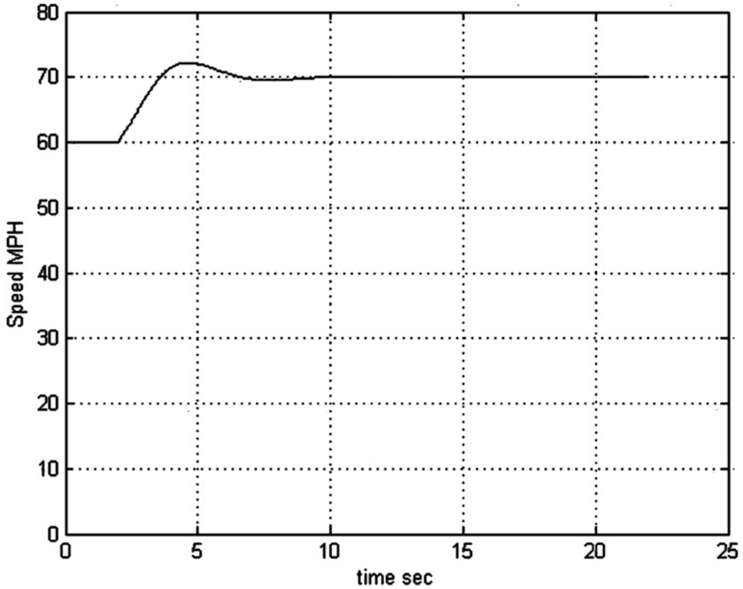


Figure 8.5:
Response of digital cruise control to step change in set speed.

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 either be modeled as 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.

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:

$$u_n = K_p e_n + K_1 \sum_{m=1}^{M} e_{n-m} \tag{28}$$

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 $\overline{\boldsymbol{u}}(t)$ suitable to operate the actuator (via a ZOH). It should be noted that $\overline{\boldsymbol{u}}(t)$ 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. Figure 8.6a 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 angular speed proportional to vehicle speed. A speed sensor driven by this cable generates a pulsed electrical signal (Figure 8.6b) 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 6, so we hypothesize an optical sensor for the

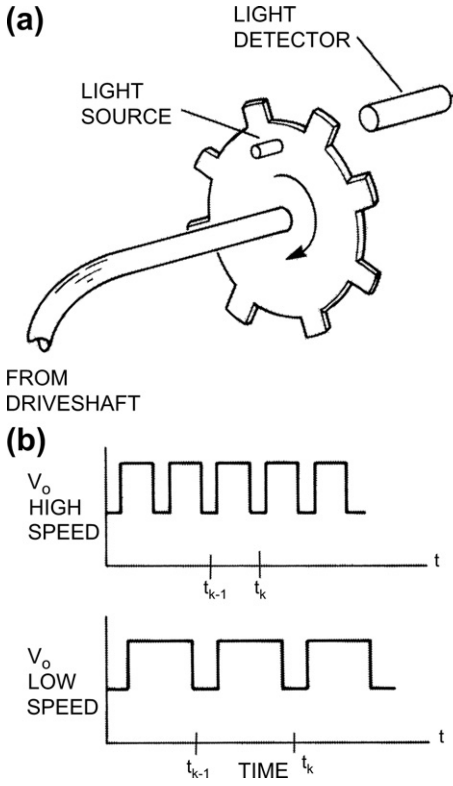


Figure 8.6:
Example speed sensor configuration.

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:

$$f = NVK$$

where f is the frequency in pulses per second, N is the number of slots in the sensor disk,

V is the vehicle speed, K is the proportionality constant that accounts for differential gear ratio and wheel size.

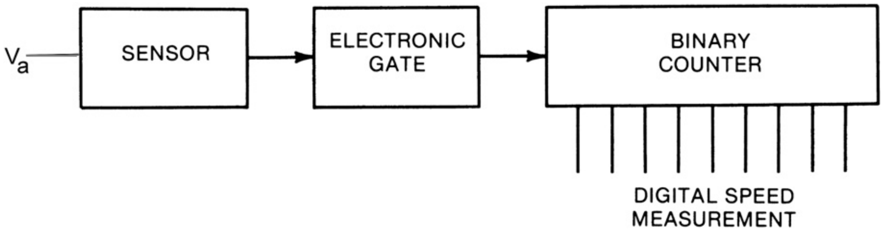


Figure 8.7:
Digital speed measurement system.

The sampled pulse frequency f_k is computed from measurements of the time of each low to high transition denoted t_k in Figure 8.6b:

$$f_k = \frac{1}{t_k - t_{k-1}}$$

The output pulses are passed through a sample gate to a binary counter (Figure 8.7).

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

$$P(n) = T_g(n)NVK \tag{29}$$

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.

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 7 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 6). The stepper motor displacement causes a change in throttle plate angle $\delta\theta_t(n)$ (see Chapter 5) 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

$$N_p(n) = f_p T_a(n) \tag{30}$$

where $T_a(n)$ is the actuator time during actuation cycle.

The actuation time interval is proportional to u_n:

$$T_a(n) = K_T u_u \tag{31}$$

where K_T is a constant for the control system.

The throttle plate angular displacement $\delta\theta_t(n)$ is proportional to $N_p(n)$:

$$\delta\theta_t(n) = K_\theta N_p(n) \tag{32}$$

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 $d\theta_t$ at cruise condition):

$$\begin{aligned}\delta T_b(n) &= K_b \delta\theta_t(n)\\ &= K_b K_\theta K_T f_p u_n\end{aligned} \tag{33}$$

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 $\delta T_b(t)$. 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 EGR valve described in Chapter 7 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 Figure 8.8.

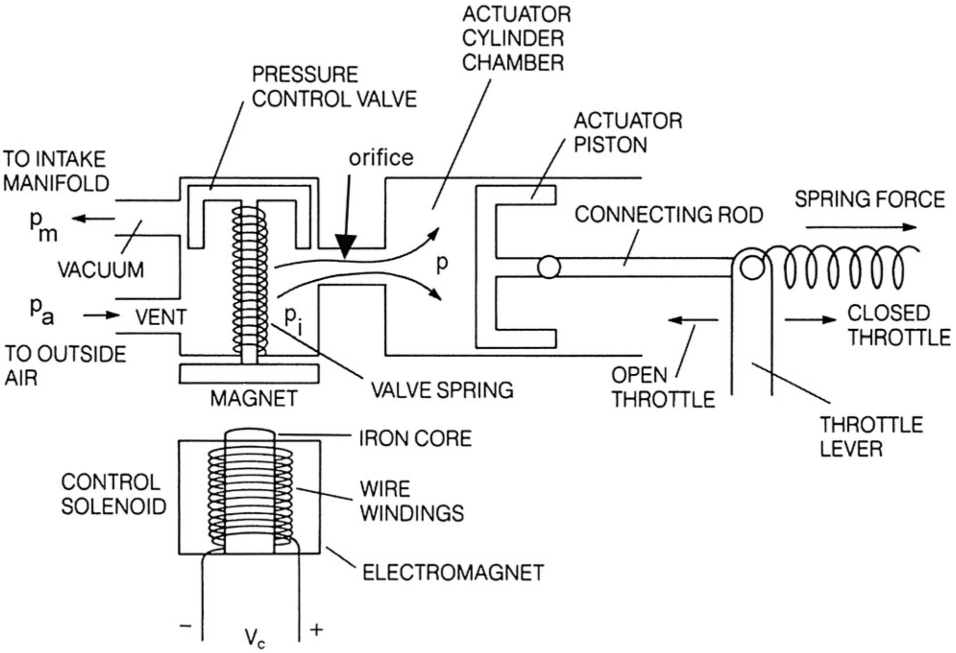


Figure 8.8:
Vacuum-operated throttle actuator.

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 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 6). 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; 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 Figure 8.8 is given by

$$\begin{aligned} p_i &= p_m \quad V_c = V_H \\ &= p_a \quad V_c = 0 \end{aligned} \tag{34}$$

where p_m is the manifold pressure and p_a the atmospheric pressure.

The cruise control computer generates actuator control signal

$$\begin{aligned} V_c(t) &= V_H \quad t_k \leq t \leq t_k + \tau \\ &= 0 \quad t_k + \tau < t < t_{k+1} \end{aligned}$$

The duty cycle δ_p is given by

$$\delta_p = \frac{\tau}{(t_{k+1} - t_k)} \tag{35}$$

where t_κ 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

$$\begin{aligned} p_{a\mathrm{v}}(t) &= \frac{1}{T_{av}} \int\limits_{t-T_{av}}^{t} p_i(t')\mathrm{d}t' \\ &= p_a + (p_m - p_a)\delta_p \end{aligned} \tag{36}$$

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 with analog electronics. It can also be implemented (in proportional control strategy alone) with an electromechanical speed governor.

The physical configuration for a digital, microprocessor-based cruise control is depicted in Figure 8.9. A system such as is depicted in Figure 8.9 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

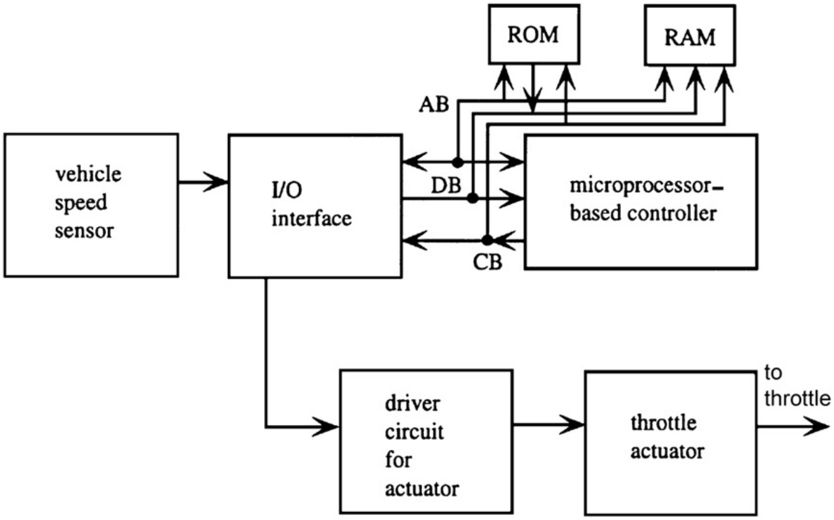


Figure 8.9:
Digital cruise control configuration.

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 above 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 Figure 8.10.

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 down-count binary counter. The direction of movement is in the form of the sign bit (SB of Figure 8.10). 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 B signal to coil 2. To retard the throttle these phases are each switched to the opposite coil. The amount of movement in

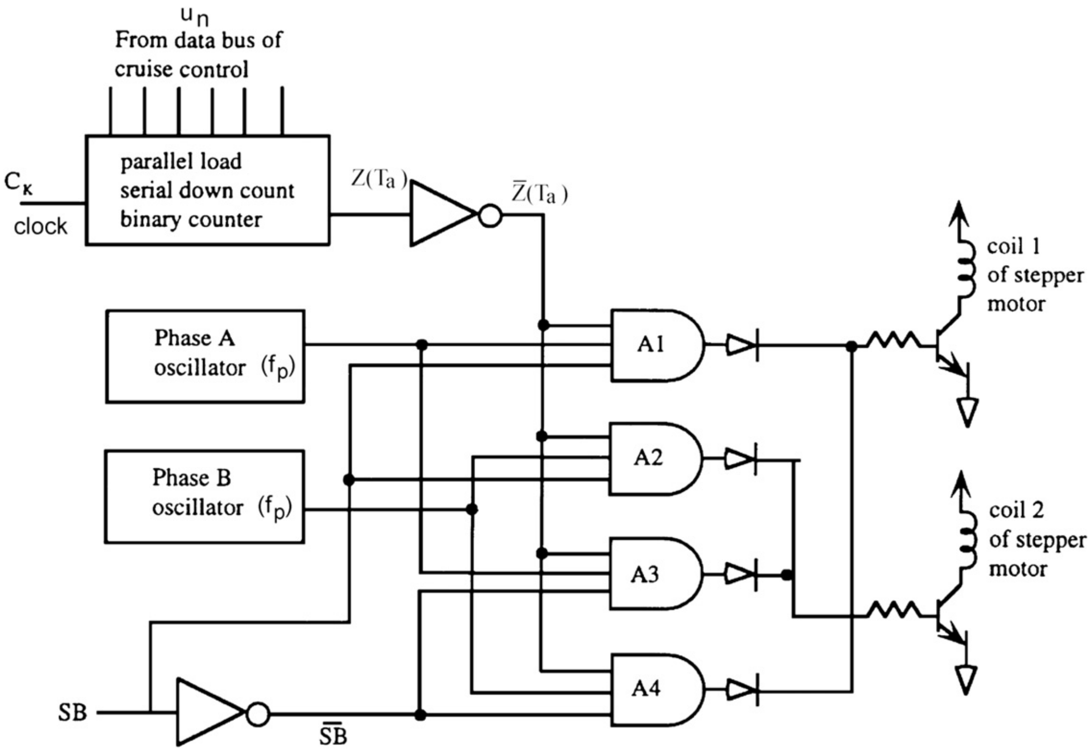


Figure 8.10:
Stepper motor actuator electronics for cruise control.

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 Figure 8.10. This logical signal is switched low such that $\overline{Z}(T_a)$ 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 $\overline{Z}$ 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., $\overline{Z}$) 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 Figure 8.10). When the countdown reaches zero, logical variable Z switches high (and $\overline{Z}$ switches low) and the gates are disabled and the stepper motor stops moving.

The time required to count down 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 3 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:

$$V_o = K_g V \tag{37}$$

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 Figure 8.11.

The vehicle speed sensor of Figure 8.11a 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

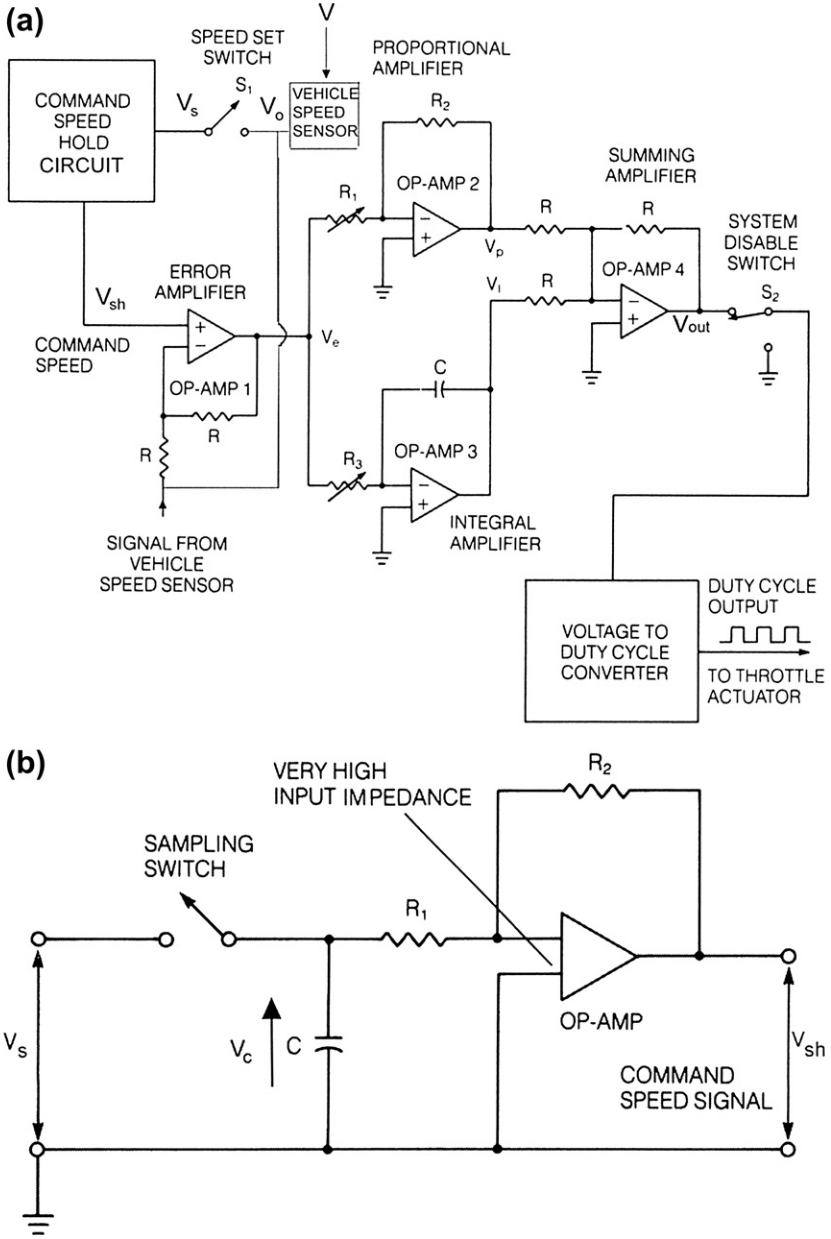


Figure 8.11:
Analog cruise control configuration.

as was described in Chapter 3. 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 3 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 $K_P = -R_2/R_1$. 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 $V_{I,}$ which is given by

$$V_I = -\frac{1}{R_3C}\int V_e \mathrm{d}t \tag{38}$$

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 which generates an output wave form at frequency f_p with duty cycle which is proportional to V_{out}.

Two switches, S_1 and S_2, are shown in Figure 8.11a. Switch S_1 is operated by the driver to set the desired speed. It signals the sample-and-hold electronics (Figure 8.11b) 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:

$$V_{\text{sh}}(t) = V_s(t_a) \tag{39}$$

where t_a is the time driver activating S_1.

Switch S_2 (Figure 8.11a) 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 advanced cruise control (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 Figure 8.12, emphasizing the automatic braking portion.

This system consists of a conventional brake system with master cylinder wheel cylinders, vacuum boost (power brakes), and various brake lines. Figure 8.12 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.

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) as well as the brake system via electrical output signals and in response to inputs, including the vehicle speed sensor and set cruise speed switch. The ACC

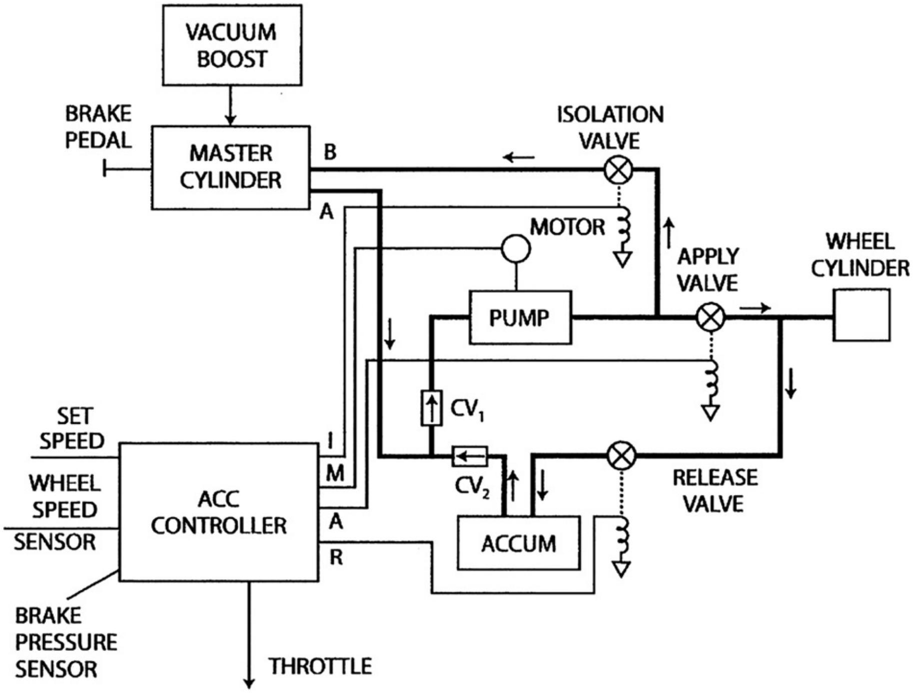


Figure 8.12:
ACC system configuration.

system functions as described above until the maximum available deceleration with closed throttle is inadequate. Whenever there is greater deceleration 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

$$M\dot{V} + D + Mg\sin\theta = \frac{g_A T_{bo}}{r_w} - \frac{T_B}{r_w} \tag{40}$$

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:

$$T_B = K_B p_B \tag{41}$$

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

$$\begin{aligned} M\delta\dot{V} + K_D \delta V + Mg\theta &= -K_B p_B / r_w \\ &= -K_B K_A u / r_w \end{aligned} \tag{42}$$

where K_A is the brake pressure actuator constant V_d is the cruise speed set point, and u the ACC control signal.

If a PI control law is assumed for this ACC automatic braking mode, the control signal is given by

$$u = K_p e + K_I \int e\mathrm{d}t \tag{43}$$

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:

$$\left(s + \frac{K_D}{M}\right)\delta V(s) + g\theta = -\frac{K_B K_A}{r_w M}\left[K_p + \frac{K_I}{s}\right]\delta V(s) \tag{44}$$

Solving for $\delta V(s)$ yields

$$\delta \mathrm{V}(s) = \frac{-gs|\theta|}{s^2 + \left(\frac{K_D}{M} + \frac{K_B K_A K_p}{r_w M}\right) s + \frac{K_B K_A K_I}{r_w M}} \tag{45}$$

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 g 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 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). Figure 8.13 is a plot of vehicle speed for P-only control as well as PI control. The same coefficients are assumed for the controller and K_B is taken to be four.

Figure 8.13 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, then with applied brakes decreases with small undershoot reaching the desired cruise speed of 60 MPH. The action of various control laws was described in Chapter 1. The present simulation confirms the predicted behavior.

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.

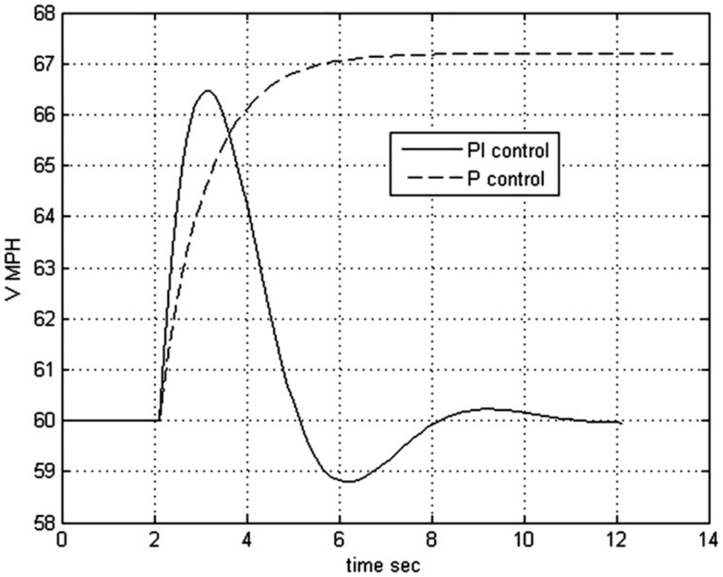


Figure 8.13:
Vehicle speed with ACC on hill with long downgrade.

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 Figure 8.14. 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 wheel, an electronic control module, and a hydraulic brake pressure modulator (regulator). For simplicity in the drawing, only a pair of brake pressure modulators are shown. However, in practice there is a separate modulator for each brake.

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

$$\omega_w = \frac{\pi \, \mathrm{RPM}_w}{30} \tag{46}$$

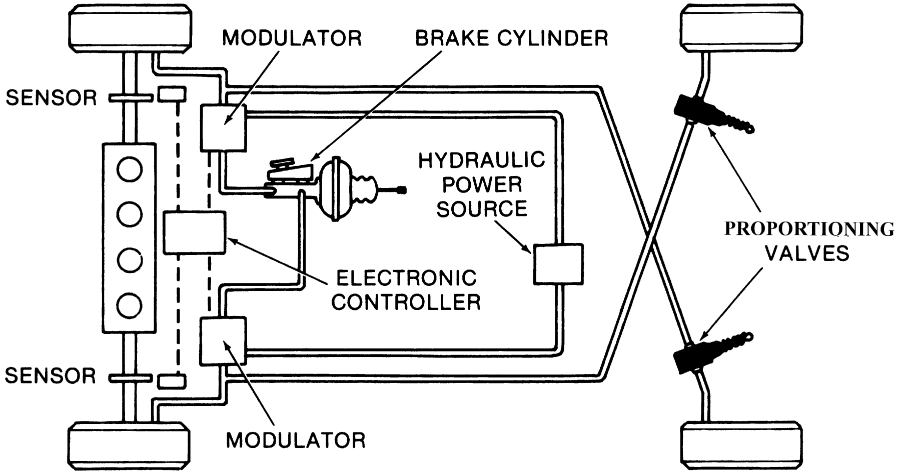


Figure 8.14:
Antilock braking system.

and where RPM_w is the RPM of the wheel in revolutions per minute. When the wheel is rolling (no applied brakes),

$$U = r_w \omega_w \tag{47}$$

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 Figure 8.15a. Figure 8.15b illustrates the forces applied to the wheel by the road during braking. This pressure causes a force which 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 Figure 8.15b. The lateral force that maintains directional control of the car is shown as F_L in Figure 8.15b.

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 percentage 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 Figure 8.15b) and the friction coefficients for braking force (F_b) and lateral force (F_L):

$$\begin{aligned} F_b &= N\mu_b \\ F_L &= N\mu_L \end{aligned} \tag{48}$$

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 Figure 8.16. 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.

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

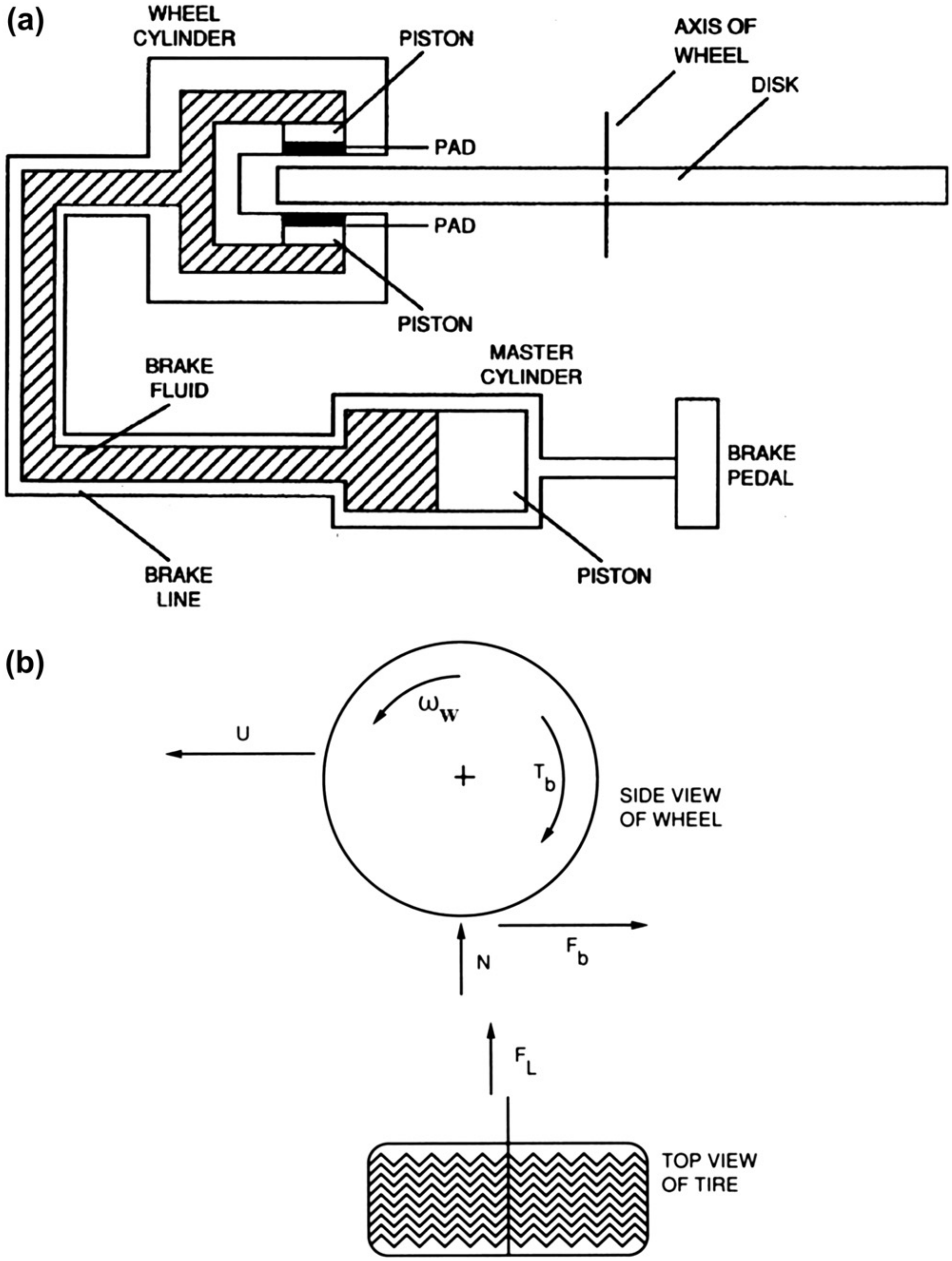


Figure 8.15:
Brake configuration and forces acting on wheel.

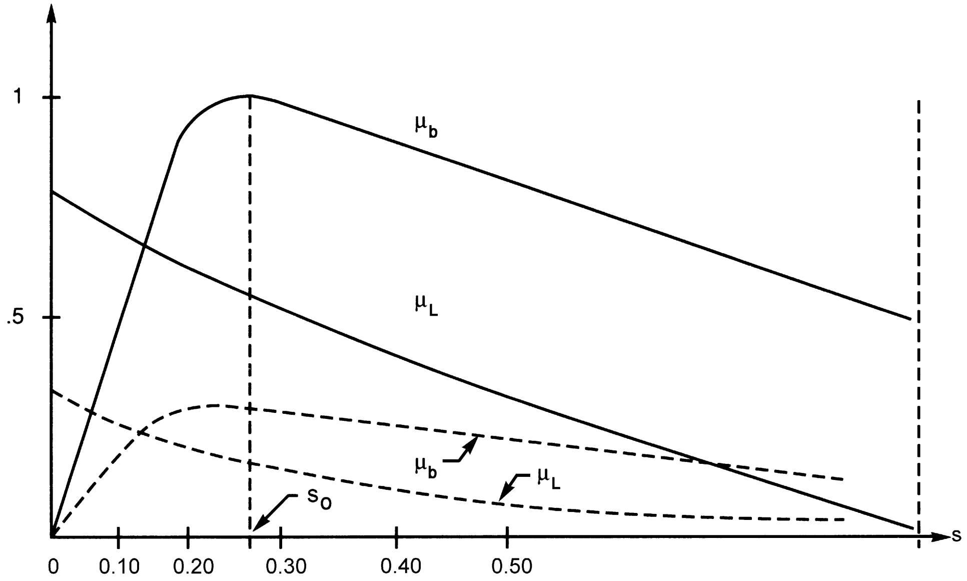


Figure 8.16:
Exemplary variation in friction coefficients with slip.

equivalent to pumping the brakes (as done by 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 Figure 8.14. This ABS regulates or modulates brake pressure to maintain slip as near to optimum for as much time as possible (e.g. at s_o in Figure 8.16). 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:

$$T_{\mathrm{w}} = r_w F_b \tag{49}$$

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:

$$T_b = f(p_b) \tag{50}$$

Although it is not necessary for ABS application, it is convenient to simplify the model for T_b to the following:

$$T_b \cong k_b p_b \tag{51}$$

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:

$$T_{\mathrm{w}} = T_b + I_{\mathrm{w}}\dot{\omega}_w$$

where I_w is the wheel moment of inertia about its rotational axis and $\dot{\omega}_w$ is the wheel deceleration $(\mathrm{d}\omega_w/\mathrm{d}t)$, that is, the rate of change of wheel speed.

During heavy braking under marginal conditions, sufficient braking force is applied to cause wheel lock-up (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 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. $\frac{\mathrm{d}\mu_b}{\mathrm{d}s} < 0$) 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.

Figure 8.17 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 Figure 8.17 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.

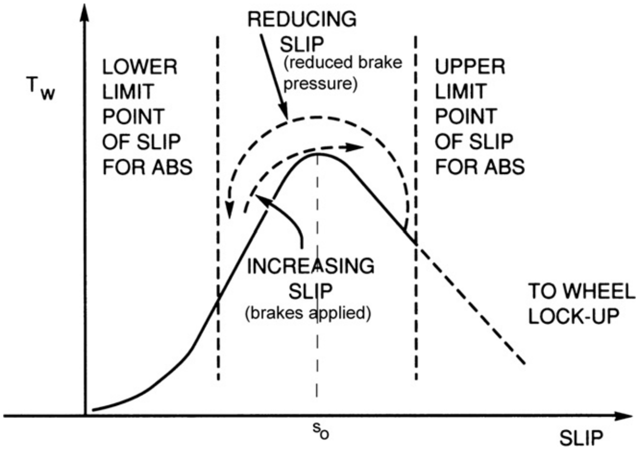


Figure 8.17:
Wheel torque vs. slip under ABS action.

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 lock-up.

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 which has demonstrated successful ABS operation in laboratory (wheel dynamometer) tests. This discussion can be considered exemplary of much of the mechanical dynamics as well as 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 Figure 8.17. 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 Chapter 1 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

$$\begin{aligned}\delta T_b &= T_b - T_w \\ &= -I_w\dot{\omega}_w\end{aligned} \tag{53}$$

Rewriting Eqn (53) yields a form from which the SMO can be readily derived:

$$\dot{\omega} = -\frac{\delta T_b}{I_w} \tag{54}$$

The goal for the SMO for this application is to calculate an estimate $(\delta\hat{T}_b)$ 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:

$$\dot{\hat{\omega}}_w = -m\,\mathrm{sgn}(\hat{\omega}_w - \omega_w) \tag{55}$$

Where m is the SMO gain that must satisfy the following inequality:

$$m \geq \max|\delta T_b| \tag{56}$$

The SMO requires an accurate, precise measurement of wheel angular speed (ω_w). The desired estimate $(\delta\hat{T}_b)$ is the solution to the following first-order differential equation:

$$\tau \frac{d\delta\hat{T}_b}{dt} + \delta\hat{T}_b = -m\,\mathrm{sgn}(\hat{\omega}_w - \omega_w) \tag{57}$$

Effectively, $\delta\hat{T}_b$ is a first-order low-pass-filtered version of the right-hand side 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 low-pass filter. Experiments and simulations have been run with 2nd-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 $(\delta\hat{T}_b)$. 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 $\mathrm{s_o}$ 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.

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 Figure 8.17), 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 Figure 8.17. 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 re-applied.

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.

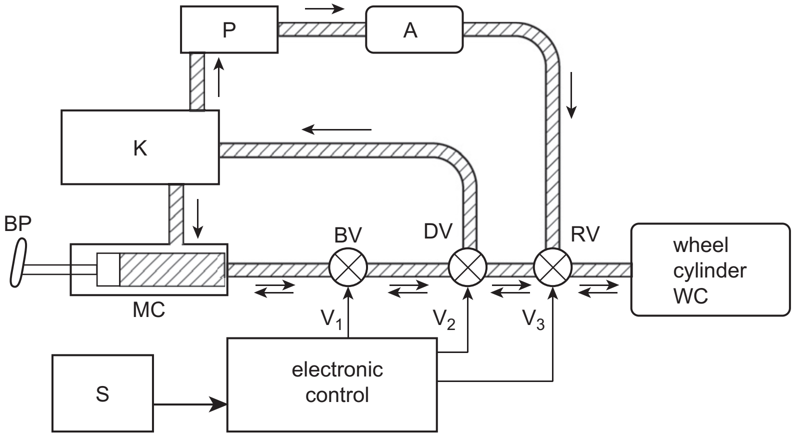


Figure 8.18:
Schematic illustration of ABS.

The mechanism for modulating brake pressure is illustrated in Figure 8.18.

In Figure 8.18 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 Figure 8.18 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 Figure 8.18 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 which leads to the repressurization valve. This latter valve passes the pressurized brake fluid to the wheel cylinder which thereby applies brake torque to the corresponding wheel.

Whenever the ABS control detects a potential wheel lock-up owing to slip $s > s_o$ (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 which 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 which switches it to its second position. In this position the line to the RV and wheel cylinder are 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 which 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 represurization is repeated. The cycling of the ABS normally continues until the wheel speed with brakes "off" is below a pre-set value (e.g. 1−5 MPH) or until the driver releases the brake pedals.

Figure 8.19 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 Figure 8.19a. The solid curve of Figure 8.19a 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 lock-up and ABS reduces brake pressure. The cycle continues until the vehicle is slowed sufficiently.

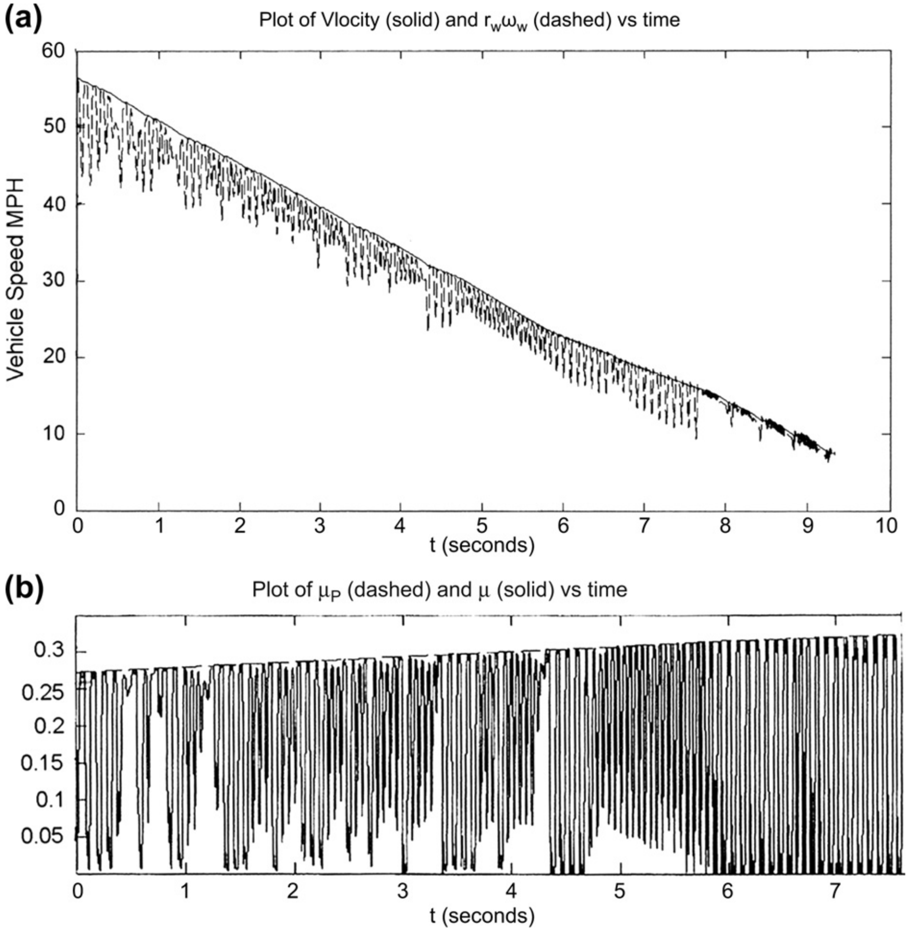


Figure 8.19:
Illustration of ABS action.

Figure 8.19b 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 Figure 8.19 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 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 antilock brake systems, 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.

Tire-Slip Controller

Another benefit of the ABS is that the brake pressure modulator can be used for ACC as explained earlier as well as for tire-slip control. Tire slip is effective in moving the car forward just as it is in braking. Under normal driving circumstances with powertrain 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 7 and Figure 7.26), 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.

Antilock braking can also be achieved with electrohydraulic brakes. An electrohydraulic brake system was described in the section of this chapter devoted to advanced cruise control (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. Figure 8.20 depicts, 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

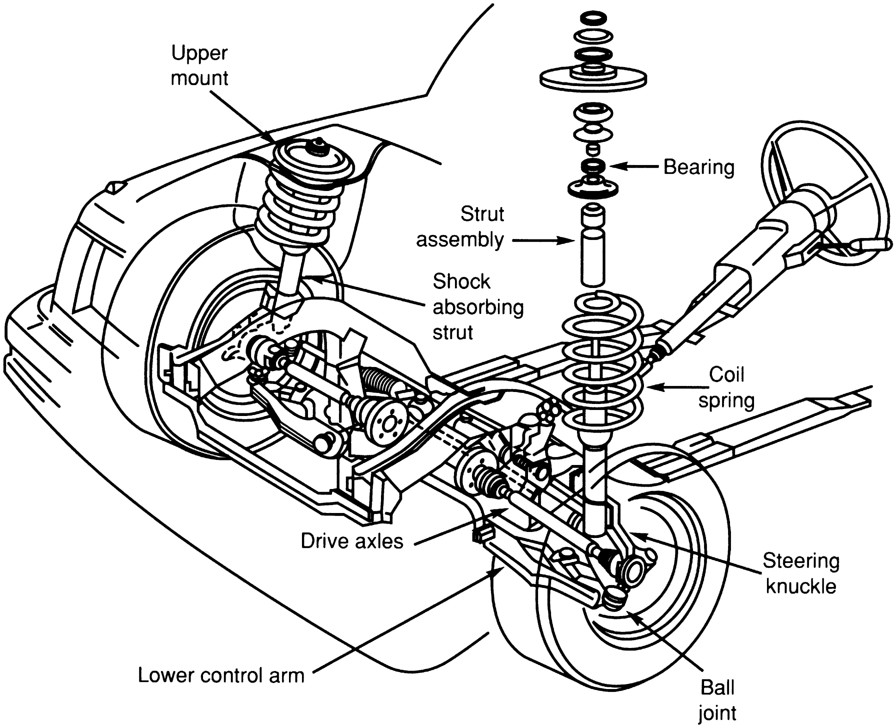


Figure 8.20:
Illustration of front suspension system.

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 Figure 8.23. 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 semi-active. The semi-active 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 semi-active. In this chapter, we explain the semi-active system first, then the active one.

The primary purpose of the semi-active 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 either be a stationary or a nonstationary process. For the following discussion we assume a stationary random process. A semi-active 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 Figure 8.21a 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).

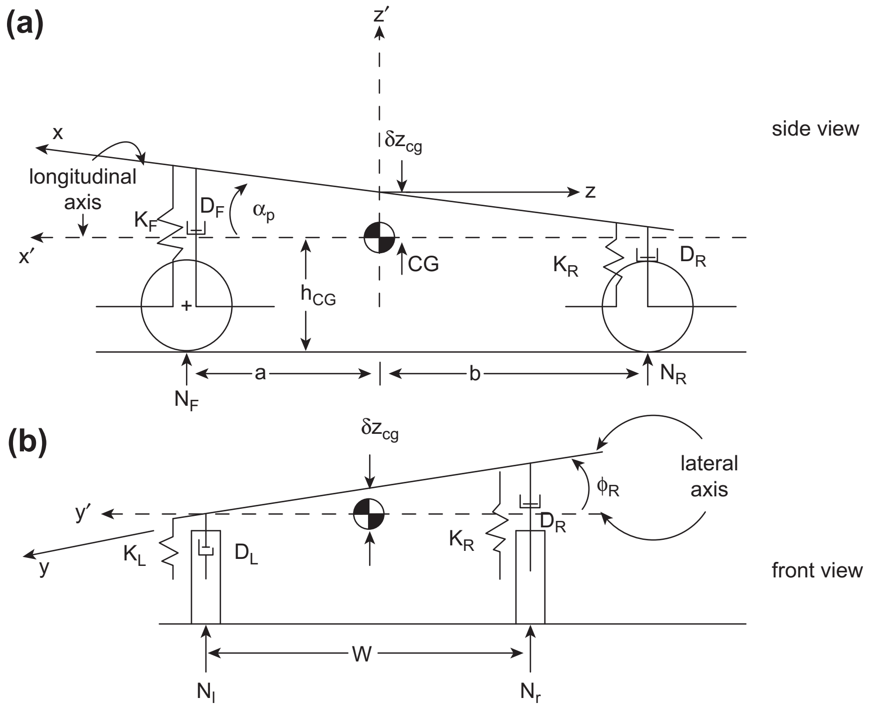


Figure 8.21:
Schematic illustration of suspension.

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.

Figure 8.21 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 which 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 and left. 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).

Figure 8.21b 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.

It is beyond the scope of this book to present a full discussion of vehicle dynamics which 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 the 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 as well as 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

$$\begin{aligned} F_F &= -(N_{Fr} + N_{F\ell})\mu_F \\ F_R &= -(N_{Rr} + N_{R\ell})\mu_R \end{aligned} \tag{58}$$

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

$$T_b = (F_F + F_R)h_{CG} \tag{59}$$

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

$$\begin{aligned} N_F &= N_{Fr} + N_{Fr} \\ N_R &= N_{Rr} + N_{R\ell} \end{aligned} \tag{60}$$

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 Figure 8.21a).

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:

$$\begin{aligned} N_F &= -\big(K_F \delta z_F + D_F \delta \dot{z}_F\big) \\ N_R &= -\big(K_R \delta z_R + D_R \delta \dot{z}_R\big) \end{aligned} \tag{61}$$

where δz_F is the deflection of front spring and δz_R the deflection of rear spring.

$$\begin{aligned} K_R &= K_{Rr} + K_{R\ell} = \text{rear spring rate} \\ K_F &= K_{Fr} + K_{F\ell} = \text{front spring rate} \\ D_F &= D_{Fr} + D_{F\ell} = \text{front damping coefficient} \\ D_R &= R_{Rr} + D_{R\ell} = \text{rear damping coefficient} \end{aligned} \tag{62}$$

Note that in the absence of any vertical motion of the CG (i.e. it is assumed here that $\delta \ddot{z}_g = 0$), the normal forces sum to the vehicle weight (W_V):

$$N_F + N_R = W_V \tag{63}$$

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

$$\begin{aligned} F &= F_F + F_R \\ &= -\mu W_\mathrm{V} \end{aligned}$$

The moment acting around the CG due braking (T_b) is given by

$$T_b = -W_\mathrm{V} \mu h_{CG}$$

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:

$$I_{yy} \ddot{\alpha}_p = -\mu W_\mathrm{V} h_{CG} + T_n \tag{64}$$

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:

$$\begin{aligned}
\delta z_F &= \mathrm{a}\alpha_p && \text{(front displacement)}\\
\delta z_R &= -\mathrm{b}\alpha_p && \text{(rear displacement)}\\
\delta \dot{z}_F &= \mathrm{a}\dot{\alpha}_p && \text{(front vertical velocity)}\\
\delta \dot{z}_R &= -\mathrm{b}\dot{\alpha}_p && \text{(rear vertical velocity)}
\end{aligned} \tag{65}$$

Substituting these relationships into the pitch dynamic Eqn (64) yields

$$I_{yy}\ddot{\alpha}_p = (F_F + F_R)h_{CG} - \left[\mathrm{a}(K_F\mathrm{a}\alpha_p + D_F\mathrm{a}\dot{\alpha}_p) + \mathrm{b}(K_R\mathrm{b}\alpha_p + D_R\mathrm{b}\dot{\alpha}_p)\right] \tag{66}$$

Simplifying and rearranging terms in this equation yield the following second-order differential equation in α_p:

$$I_{yy}\ddot{\alpha}_p + D\dot{\alpha}_p + K\alpha_p = Fh_{CG} \tag{67}$$

where

$$\begin{aligned}
D &= \mathrm{a}^2 D_F + \mathrm{b}^2 D_R\\
K &= \mathrm{a}^2 K_F + \mathrm{b}^2 K_R
\end{aligned}$$

The operational transfer function ($H_\alpha(s)$) relating braking force to pitch angle is given by

$$\begin{aligned}
H_\alpha(s) &= \frac{\alpha_p(s)}{F(s)}\\
&= \frac{h_{CG}}{I_{yy}s^2 + Ds + K} = \frac{h_{CG}}{I_{yy}}\left[\frac{1}{s^2 + 2\zeta\omega_n s + \omega_n^2}\right]
\end{aligned} \tag{68}$$

where

$$F(s) = F_F(s) + F_R(s)$$

and

$$\omega_n = \sqrt{\frac{K}{I_{yy}}}$$

$$\zeta = D/(2I_{yy}\omega_n) = \text{damping ratio}$$

Solution to this equation for the pitch dynamics due to an arbitrary braking force function $F(t)$ is found using the methods of Chapter 1 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

$$\alpha_{pss} = -\frac{h_{CG}F_0}{K} \tag{69}$$

Depending on the damping ratio ζ, there may be overshoot in α_p before settling to α_{pss} where, with the sign convention of Figure 8.21a, $\alpha_{pss} < 0$.

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

$$H_\alpha(j\omega) = \frac{h_{CG}/I_{yy}}{(\omega_n^2 - \omega^2) + 2j\zeta\omega_n\omega} \tag{70}$$

The peak response occurs at $\omega = \omega_n$ and has magnitude

$$|H_\alpha(j\omega_n)| = \frac{h_{CG}}{2\zeta K} \tag{71}$$

and a 90° 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.

Figure 8.22 illustrates qualitatively a representative tire normal force variation as a function of frequency of excitation for a fixed-amplitude, variable-frequency sinusoidal excitation (see Chapter 1 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 Figure 8.22 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 approximately 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

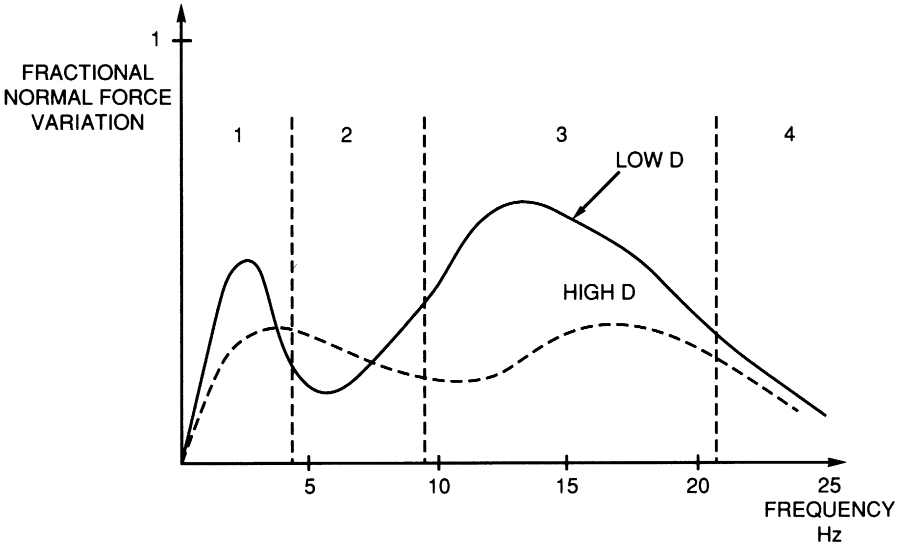


Figure 8.22:
Normal force variation due to sinusoidal excitation vs. frequency.

important for good subjective ride. Good ride is achieved for relatively low damping (low D in Figure 8.22).

For low damping, the unsprung mass moves relatively freely due to road input while the sprung mass motion remains relatively low. Note from Figure 8.22 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 Figure 8.22, 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 center of gravity 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:

$$r = \dot{\psi} \tag{72}$$

Similarly, the notation for $\dot{\phi}_R$ is taken to be p:

$$p = \dot{\phi}_R \tag{73}$$

The lateral velocity component of the CG is denoted v:

$$\mathrm{v} = \dot{y} \tag{74}$$

The inertial forces due to the motion of the car along the curve create a rolling moment T_R about the CG given by

$$T_R = -Mh_{CG}(\dot{\mathrm{v}} + ru_0) \tag{75}$$

where u_o is the vehicle speed (assumed constant) and M the vehicle mass.

The sum of the moments about the CG for this maneuver yields the following approximate differential equation:

$$I_{xx}\ddot{\phi} + Mh_{CG}(\dot{v} + ru_0) = -(L_\phi \phi_R + L_p p) \tag{76}$$

where I_{xx} is the moment of inertia of the sprung mass structure about the body longitudinal axis

$$L_{\phi=}(K_F + K_R)w^2$$

$$L_p = (D_F + D_R)w^2 \tag{77}$$

where w is the distance between right and left tire planes of symmetry (Figure 8.21a).

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

$$I_{xx}\ddot{\phi}_R + L_p\dot{\phi}_R + L_\phi \phi = T_R \tag{78}$$

where

$$T_R = -Mh_{cg}(\dot{\mathrm{v}} + ru_o)$$

If the curve is a segment of a constant radius circle, then during the constant turn maneuver the moment T can be given as

$$\begin{aligned} T_R(t) &= 0 \qquad t < 0 \\ &= T_0 \qquad t \geq 0 \end{aligned} \tag{79}$$

It can be shown that for a vehicle traveling along a curve of constant radius R at a constant speed u_o, T_0 is given by

$$T_0 = -Mh_{CG}\, u_o^2/R$$

The operational transfer function for the roll dynamics $H_\phi(s)$ is given by

$$\begin{aligned} H_\phi(s) &= \frac{\phi_R(s)}{T_R(s)} \\ &= \frac{1}{I_{xx}\left(s^2 + \dfrac{sL_p}{I_{xx}} + \dfrac{L_\phi}{I_{xx}}\right)} \end{aligned} \tag{80}$$

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 which 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

$$\phi_{\mathrm{RSS}} = \frac{T_0}{L_\phi} \tag{81}$$

That is, the suspension spring rate 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 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 Chapter 1, we discussed the dynamics of a spring/mass/damping system, identifying resonant frequency and unity damping D_c (i.e. $\zeta = 1$):

$$D_c = 2\sqrt{KM}$$

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 Figure 8.23. Physically, this strut consists of a closed cylinder with a movable

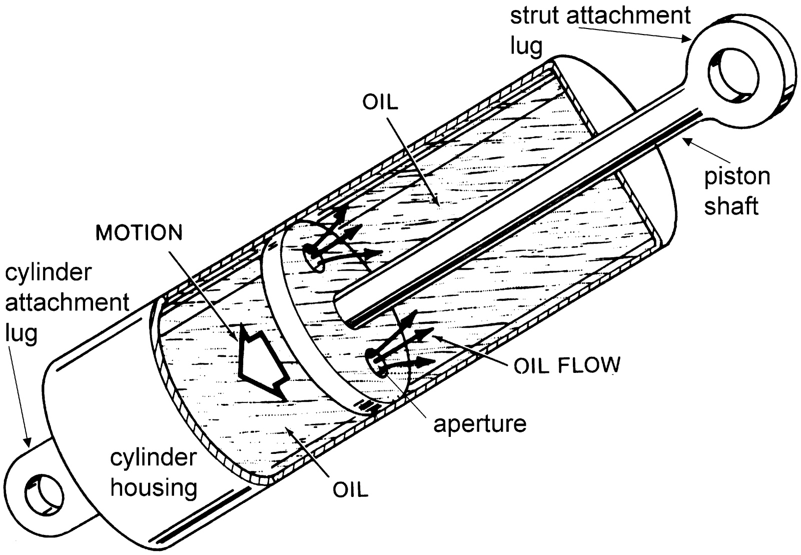


Figure 8.23:
Strut physical configuration.

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 which 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.

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 center of gravity (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 Figure 8.24.

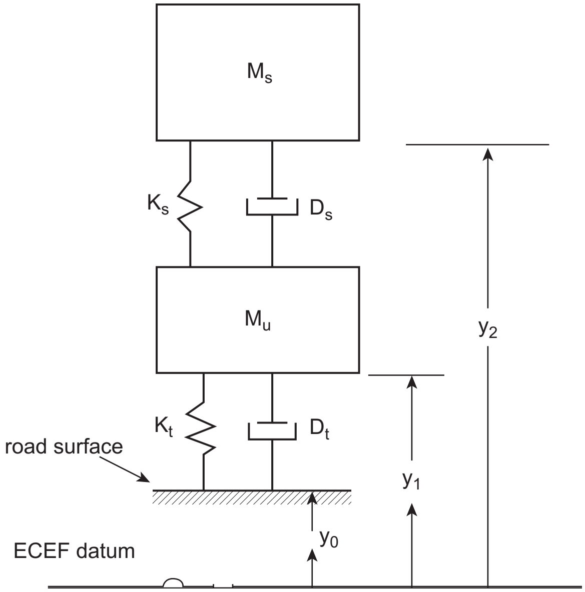


Figure 8.24:
QCM car suspension configuration.

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

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

$$M_u\ddot{y}_1 + D_s(\dot{y}_1 - \dot{y}_2) + K_t(y_1 - y_0) + K_s(y_1 - y_2) = 0 \tag{82}$$

$$M_s\ddot{y}_2 + D_s(\dot{y}_2 - \dot{y}_1) + K_s(y_2 - y_1) = 0 \tag{83}$$

Solution of Eqn (84) can be found in two ways. The first way we consider leads to a closed-form analytical 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

$$x = [\mathrm{v}_1, \mathrm{v}_2, y_1, y_2]^T$$

where

$$\begin{aligned} \mathrm{v}_1 &= \dot{y}_1 \\ \mathrm{v}_2 &= \dot{y}_2 \end{aligned}$$

Taking the Laplace transform of Eqns (82) and (83) (with zero initial conditions) yields a pair of coupled algebraic equations in complex frequency s:

$$\left[M_u s^2 + D_s s + (K_t + K_s)\right] y_1(s) - (D_s s + K_s) y_2 = K_t y_0(s) \tag{84}$$

$$(M_s s^2 + D_s s + K_s) y_2(s) - (D_s s + K_s) y_1(s) = 0.$$

In matrix form, this pair of equations can be written in the form

$$A \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = K_t \begin{bmatrix} y_0 \\ 0 \end{bmatrix} \tag{85}$$

where

$$A = \begin{bmatrix} M_u s^2 + D_s s + (K_s + K_t) & -(D_s s + K_s) \\ -(D_s s + K_s) & M_s s^2 + D_s s + K_s \end{bmatrix} \tag{86}$$

The two-dimensional state vector $[y_1, y_2]^T$ is found using matrix methods yielding

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = K_t A^{-1} \begin{bmatrix} y_0 \\ 0 \end{bmatrix} \tag{87}$$

The 2 × 2 matrix A is readily inverted using standard methods from matrix algebra yielding an analytical 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 Chapter 1. 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

$$a_s = \ddot{y}_2 \tag{88}$$

The operational transfer function relating $a_s(s)$ to $y_0(s)$ can be shown to be given by

$$\begin{aligned} H_a(s) &= \frac{a_s(s)}{y_o(s)} \\ &= \frac{s^2 y_2(s)}{y_o(s)} \\ &= \frac{2\zeta\omega_1^2\omega_2 s^3 + \omega_1^2\omega_2^2 s^2}{s^4 + 2\mu\zeta\omega_2 s^3 + (\omega_1^2 + \mu\omega_2^2)s^2 + 2\zeta\omega_1^2\omega_2 s + \omega_1^2\omega_2^2} \end{aligned} \tag{89}$$

where

$$\begin{aligned} \omega_1^2 &= \frac{K_t}{M_u} \\ \omega_2^2 &= \frac{K_s}{M_s} \\ \mu &= \frac{M_s + M_u}{M_u} \\ \zeta &= \frac{D_s}{2\sqrt{M_s K_s}} \end{aligned}$$

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

$$\begin{aligned} H_H(s) &= \frac{d(s)}{y_0(s)} \\ &= 1 - \frac{y_1(S)}{y_0(S)} \end{aligned} \tag{90}$$

The solution for $y_1(s)$ from the matrix equation yields the following:

$$H_H(s) = \frac{s^4 + 2\mu\zeta\omega_2 s^3 + \mu\omega_2^2 s^2}{s^4 + 2\mu\zeta\omega_2 s^3 + (\omega_1^2 + \mu\omega_2^2)s^2 + 2\zeta\omega_1^2\omega_2 s + \omega_1^2\omega_2^2} \tag{91}$$

As an illustration of the variation in relative displacement d vs. road displacement y_o vs. frequency, a plot of the sinusoidal frequency response for $H_H(j\omega)$ is given in Figure 8.25. 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
$\zeta = 0.8$

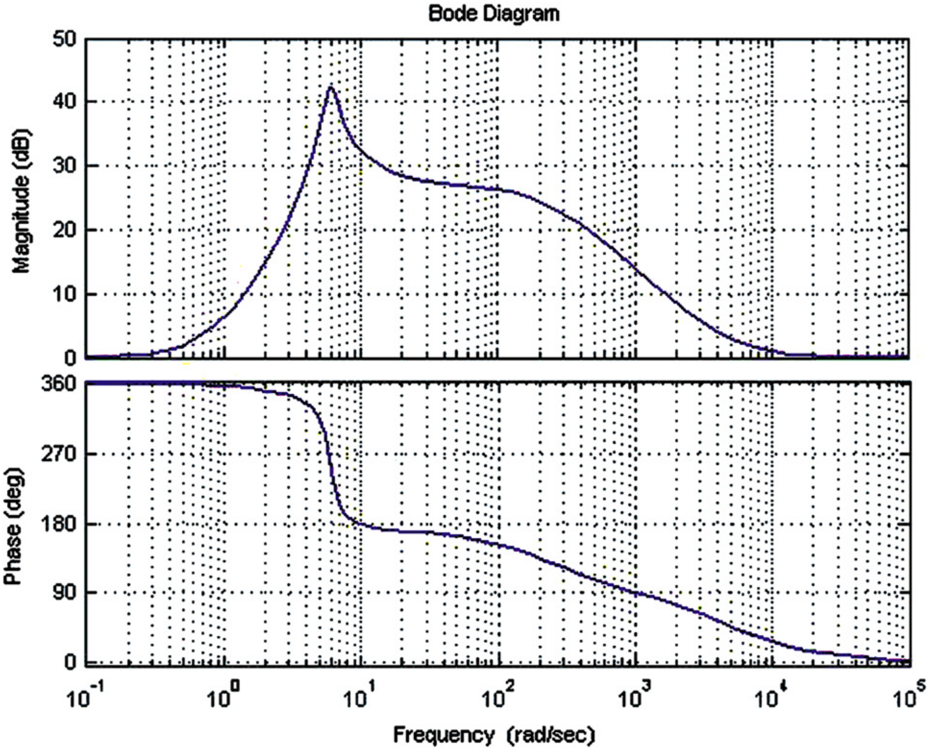


Figure 8.25:
Frequency response for $H_H(s)$.

Figure 8.25 presents the magnitude of this frequency response (in dB) as 20 log $|H_H(\mathrm{j}\omega)|$ and the phase of $H_H(j\omega)$ vs. log(ω). It can be seen that this QCM has a relatively sharp resonance at $\omega \cong 7$ rad/s rad/s or about 1.1 Hz. This resonance is primarily due to the dynamic response of the unsprung mass.

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 which is denoted $W_o(f)$ and is given by

$$W_o = |Y_o(j\omega)|^2 \tag{92}$$

where

$$Y_o(j\omega) = \underset{T\rightarrow\infty}{Lim} \int_{-T}^{T} y_o(t)e^{j\omega t}\mathrm{d}t \tag{93}$$

However, road profiles are functions of distance along the road surface (i.e. $y_{\mathrm{o}}(x)$). This road profile random process can be converted to a time function by considering motion at a constant speed u_o where

$$u_0 = \frac{\mathrm{d}x}{\mathrm{d}t}$$

The time function $y_{\mathrm{o}}(t)$ 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 $W_{\mathrm{o}}(f)$, 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 $\tilde{a}_s$:

$$\tilde{a}_s = \left[\underset{T\rightarrow\infty}{Lim}\left(\frac{1}{\mathrm{T}}\int_0^T a_s^2(t)\mathrm{d}t\right)\right]^{\frac{1}{2}}$$

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

$$\tilde{a}_s^2 = \int_0^\infty W_a(f)\mathrm{d}f$$

For a stationary random process, the power spectral density $W_a(f)$ is given by

$$W_a(f) = |H_a(j2\pi f)|^2 W_o(f) \tag{94}$$

Thus, the ride quality can be represented by the integral

$$\tilde{a}_s = \left[\int_0^\infty |H_a(j2\pi f)|^2 W_o(f)df\right]^{\frac{1}{2}} \tag{95}$$

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 as well as their ratio $\left(\frac{M_s}{M_u}\right)$, the strut and tire spring rates, as well as 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

$$\tilde{d} = \left[\int_0^\infty |H_H(j2\pi f)| W_o(f)\mathrm{d}f\right]^{\frac{1}{2}} \tag{96}$$

Clearly, both ride and handling are influenced by suspension parameters as well as M_s and M_u.

Considerable research and development has gone into determining optimum strut damping over the years. Table 8.1 is a summary of some of the results of those studies vs. running condition, control objective, optimum condition, as well as 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

Table 8.1: Summary of Optimum Suspension System Parameters

			Optimum Damping Ratio	
Running Condition	**Control Objective**	**Optimum Condition**	**Theoretical Value ζ**	**For Compact Car**
Ordinary driving	Ride improvement	To minimize sprung overall acceleration	$\dfrac{D_2}{2\sqrt{M_s K_s}} = \dfrac{1}{2}\sqrt{\dfrac{\mu(K_s/K_t)}{\mu - 1}}$	0.16
Roll	Roll reduction when turning	To suppress dynamic roll angle to a level below static roll angle	$\dfrac{(L_\phi/I_{xx})D}{4\sqrt{KI_{xx}}} = \dfrac{1}{\sqrt{2}}$	0.71
Pitch	Pitch reduction when accelerating, decelerating and braking	To suppress dynamic pitch angle to a level below static pitch angle	$\dfrac{a^2 D_F + b^2 D_R}{\sqrt{2I_{yy}(a^2 K_F + b^2 K_R)}} = \dfrac{1}{\sqrt{2}}$	0.71
Bouncing	Reduction of bouncy feeling and ride improvement	To suppress light bouncy vibrations within a range where ride quality does not deteriorate	$\dfrac{D_s}{2\sqrt{M_s K_s}} = \dfrac{\sqrt{2}}{\mu} + \dfrac{1}{\mu}\sqrt{\dfrac{\mu(K_s/K_t)}{\mu - 1}}$	0.43
Tough road	Road holdability improvement	To minimize the root-mean-square value of unsprung relative displacement	$\dfrac{D_s}{2\sqrt{M_s K_s}} = \dfrac{1}{2}\left[\dfrac{\mu^3 r_K^2 - 2\mu(\mu - 1)r_K + (\mu - 1)^2}{\mu^2(\mu - 1)r_k}\right]^{1/2}$	0.44
			where $r_K = K_s/K_t, \qquad D = D_F + D_R$	

by the size of the aperture through which the fluid flows (see Figure 8.23) 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. 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 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 which 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. Figure 8.26 illustrates such a system in which the threshold for switching to firm damping (i.e. higher damping) is 0.35 g. A separate curve of similar shape of steering angle vs. vehicle speed exists for each lateral acceleration threshold. A simple model for acceleration for a vehicle maneuvering with constant lateral acceleration is presented later in this chapter. There it is shown that the steering angle vs. velocity are given by

$$\delta_F = \frac{K}{u_o} \tag{97}$$

where K is proportional to lateral acceleration. Thus, each curve of $\delta_F(u_o)$ has a shape such as is given in Figure 8.26. The strut damping is switched from the soft damping region (i.e. relatively large strut aperture) where it is for normal driving to the firm damping region whenever the lateral acceleration exceeds the threshold (e.g. 0.35 g in the present example).

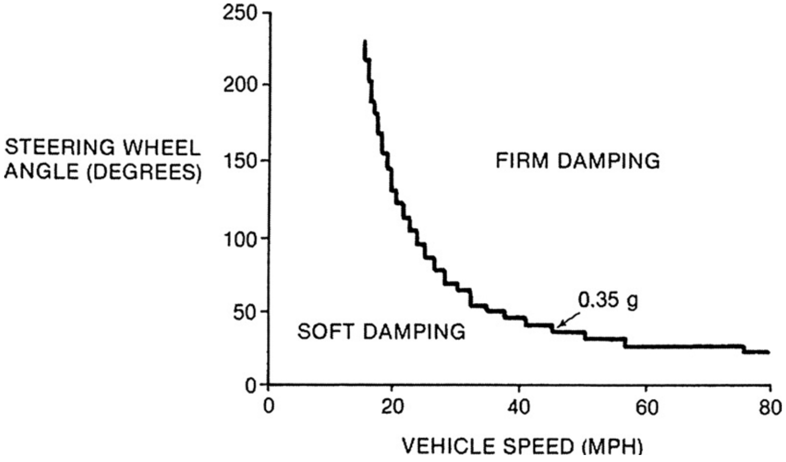


Figure 8.26:
Illustration of switching threshold for switched type variable strut damping.

In this example, a sensor for measuring the lateral acceleration (called an accelerometer) is commercially available at relatively low cost. The signal from this sensor can be compared with the threshold acceleration value to determine which of the two aperture settings are to be selected and to generate an appropriate actuator signal. Figure 8.27 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 Figure 8.27. 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 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:

$$d = \int_{o}^{t} \int_{o}^{\tau} a_u(t\prime)\mathrm{d}t\prime\mathrm{d}\tau \tag{98}$$

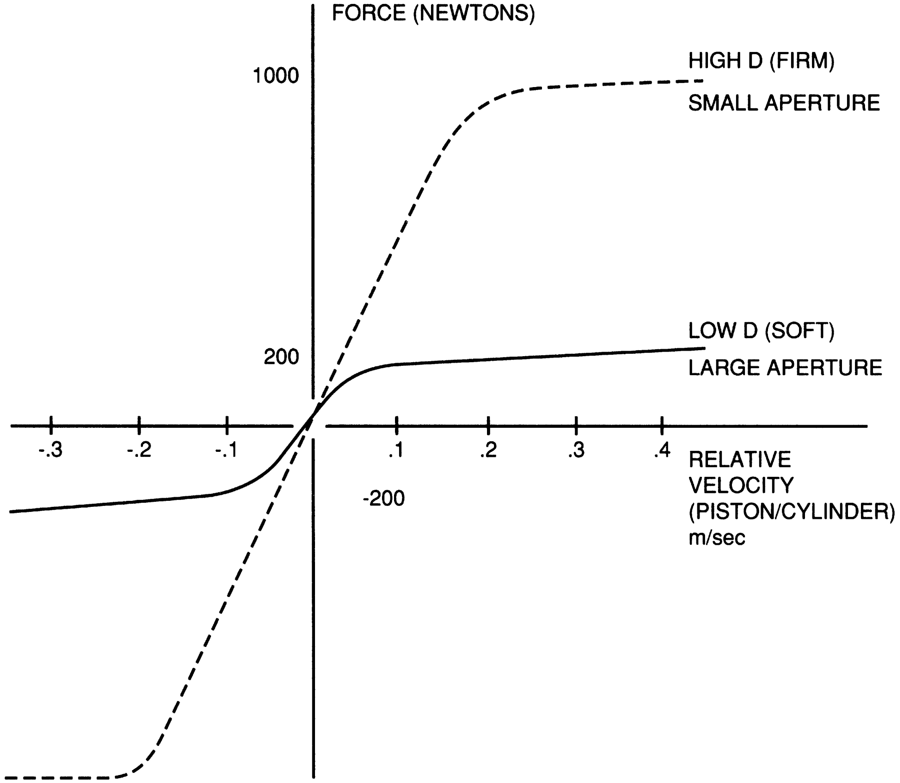


Figure 8.27:
Strut force velocity relationship for variable-aperture strut.

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 magneto-rheological fluid (MR). 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 as well as 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 air 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 which contains it in accordance with the adiabatic gas law:

$$pV^{\gamma} = K \tag{99}$$

where K is the constant and $\gamma$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

$$\mathrm{V} = A_p(\ell - x) \tag{100}$$

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 vs. displacement function is given by

$$\begin{aligned} F &= pA_p \\ &= \frac{KA_p}{\left[A_p(\ell - x)\right]^{\gamma}} \end{aligned} \tag{101}$$

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 vs. 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, as well as to increase spring "stiffness."

Electronic Suspension Control System

The control system for an exemplar electronic suspension system is depicted in the block diagram of Figure 8.28.

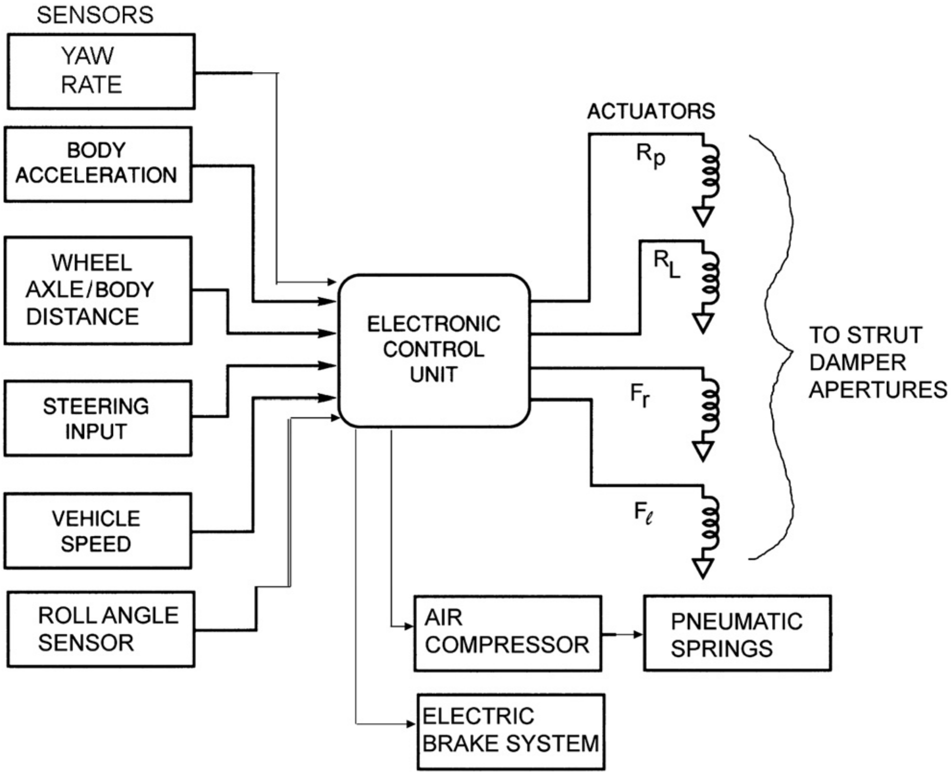


Figure 8.28:
Example electronic suspension system configuration.

The control system configuration in Figure 8.28 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) for stability enhancement purposes.

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 which 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 is beyond the scope of this book, but the interested reader should refer to Society of Automotive Engineers (SAE) publications on this subject.

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 center of gravity) 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 Figure 8.29.

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, 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.

The adverse effect of this wheel alignment is an increase in steering effort for the driver relative to zero camber and caster, which is undesirable. Rather than compromise on alignment to achieve lower steering effort, car manufacturers have found it desirable to provide a power assist via a hydraulic system as depicted in Figure 8.29. An engine-driven pump P provides hydraulic fluid (power steering fluid) under pressure. This pressurized fluid is sent via hydraulic lines to a control valve (CV) mounted at some point on the steering shaft. This CV 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. Figure 8.30 shows qualitatively the desired boost which 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) which is a function

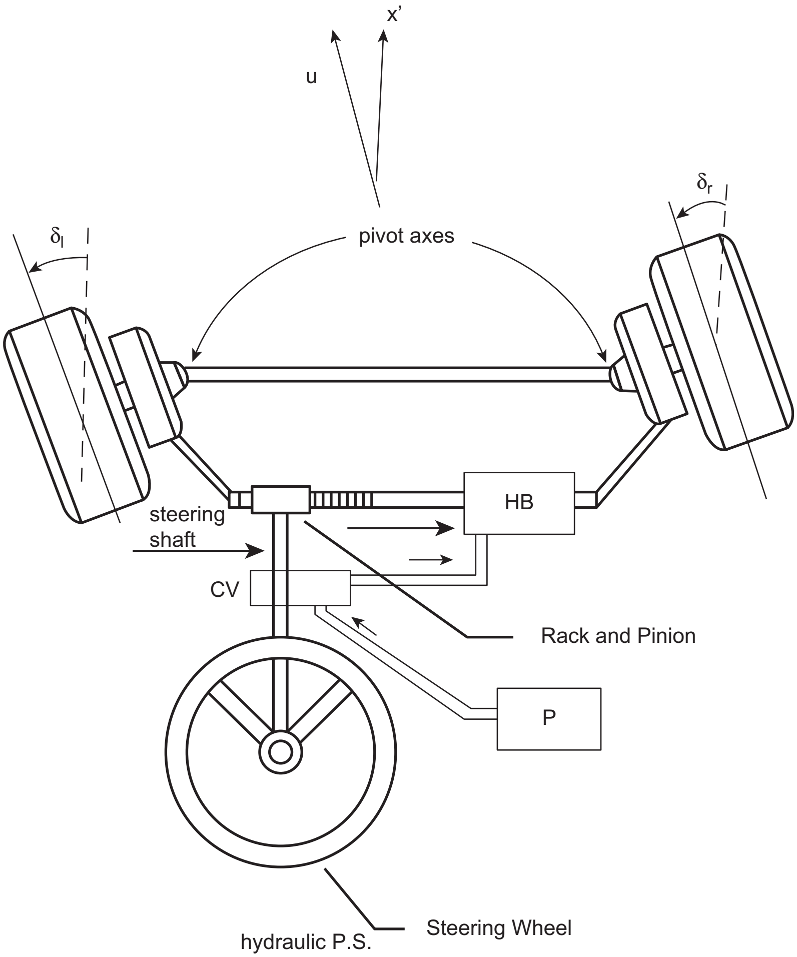


Figure 8.29:
Basic steering mechanism with power assist.

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 vs. 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 6).

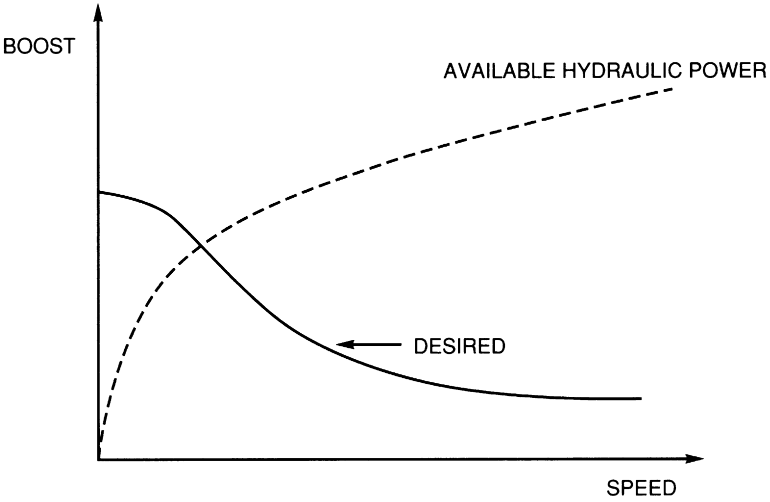


Figure 8.30:
Power steering boost vs. speed.

An alternative power steering scheme uses a special electric motor to provide the boost required instead of the hydraulic boost as depicted in Figure 8.31.

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

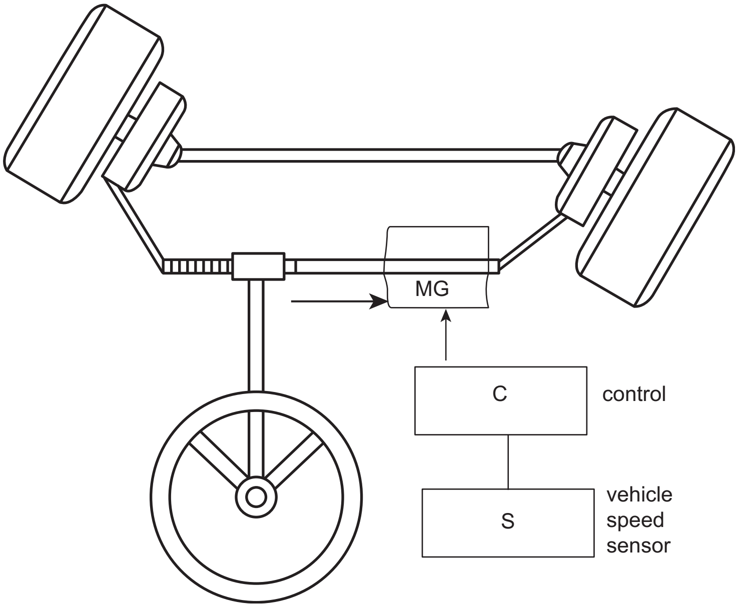


Figure 8.31:
Electric power steering boost.

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 an electric motor must be rotating to produce meaningful torque. 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

Electronically controlled power steering also has the capability for four-wheel steering (4WS). As will be shown later 4WS 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 four-wheel steering systems (4WS). 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. Figure 8.32 is an illustration of the 4WS configuration.

In Figure 8.32, the notation is as follows:

$x' =$ vehicle longitudinal axis
$x =$ inertial (ECEF) reference axis (i.e. initial direction of x)
$\psi =$ angle between x and x'
$\delta_F =$ angle between x' and the front tire plane of symmetry
$\delta_R =$ angle between x' and the rear tire plane of symmetry
$\delta_u =$ angle between x' and u_o
$u_o =$ car instantaneous velocity vector
$a =$ longitudinal distance CG to front wheel axis
$b =$ longitudinal distance CG to rear wheel axis
$\mathrm{v} =$ car lateral velocity
$r = \dot{\psi}$
$\mathrm{v}_F =$ lateral velocity of front wheel relative to road surface
$\mathrm{v}_R =$ lateral velocity of rear wheel relative to road surface

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 $\cos(\delta_u) \cong 1$, $\sin(\delta_u) \cong \delta_u$.

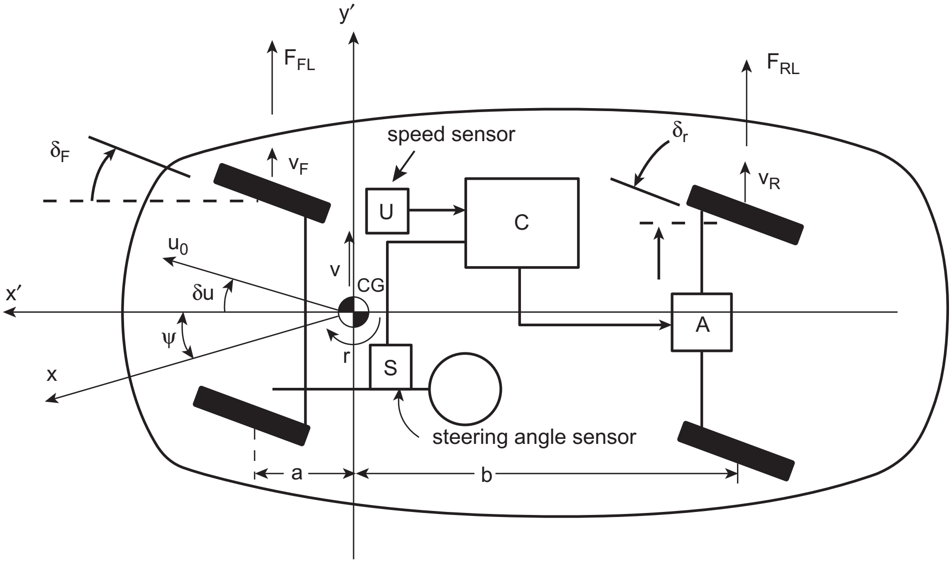


Figure 8.32:
4WS basic configuration.

Under these conditions, the lateral velocities of the front and rear tires respectively are given by

$$\begin{aligned} \mathrm{v}_F &= \mathrm{v} + ra \\ \mathrm{v}_R &= \mathrm{v} - br \end{aligned} \tag{102}$$

The models for the tire lateral forces at the front and rear tires F_{FL} and F_{RL} respectively 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

$$\begin{aligned} \alpha_F &= \delta_F - \tan^{-1}\left(\frac{\mathrm{v} + ra}{u_o}\right) \\ \alpha_R &= \delta_R - \tan^{-1}\left(\frac{\mathrm{v} - br}{u_o}\right) \end{aligned} \tag{103}$$

In these equations right-and-left symmetry is assumed which is valid for relatively small δ_F, 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

$$\begin{aligned} \alpha_F &\cong \delta_F - \left(\frac{\mathrm{v} + ar}{u_o}\right) \\ \alpha_R &\cong \delta_R - \frac{\mathrm{v} - br}{u_0} \end{aligned} \tag{104}$$

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

$$\begin{aligned} F_{FL} &= 2C_F\left[\delta_F - \left(\frac{\mathrm{v} + ar}{u_o}\right)\right] \\ F_{RL} &= 2C_R\left[-\left(\frac{\mathrm{v} - br}{u_o}\right)\right] \end{aligned} \tag{105}$$

where C_F is the front tire concerning stiffness, C_R 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. $\mathrm{N_F}$, $\mathrm{N_R}$). 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:

$$M(\dot{\mathrm{v}} + u_o r) = 2C_F\left[\delta_F - \left(\frac{\mathrm{v} + ar}{u_o}\right)\right] - 2C_R\left(\frac{\mathrm{v} - br}{u_o}\right) \tag{106}$$

where M is the vehicle mass.

Similarly, the model for the rotational motion about the CG is found by summing all moments about the CG and is given by

$$I_{zz}\dot{r} = 2aC_F\left[\delta_F - \left(\frac{\mathrm{v} + ar}{u_o}\right)\right] - 2bC_R\left(\frac{\mathrm{v} - br}{u_o}\right) \tag{107}$$

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 Chapter 1. In this case the independent variables v,r are put in state vector (x) form where the state vector is given by

$$x = \begin{bmatrix} \mathrm{v} \\ r \end{bmatrix} \tag{108}$$

The state variable model for the pair of equations is in the form

$$\dot{x} = Ax + Bu \tag{109}$$

where the state transition matrix A is given by

$$A = \begin{bmatrix} -2\dfrac{(C_F + C_R)}{Mu_o} & -2\dfrac{(\mathrm{a}C_F - bC_R)}{Mu_o} - \dfrac{u_o}{M} \\ -2\dfrac{(\mathrm{a}C_F + bC_R)}{I_{zz}u_o} & -2\dfrac{(\mathrm{a}^2C_F - b^2C_R)}{I_{zz}u_o} \end{bmatrix}$$

and the input matrix B is given by

$$B = \begin{bmatrix} \dfrac{2C_F}{M} \\ \dfrac{-2\mathrm{a}C_F}{I_{zz}} \end{bmatrix} \tag{110}$$

The input u for front wheel steering is given by

$$u = \delta_F \tag{111}$$

Taking the Laplace transform of the state variable Eqn (109) and solving for $x(s)$ yields

$$x(s) = (sI - A)^{-1}Bu(s) \tag{112}$$

where I is an identity matrix which, 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 Chapter 1 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:

$$M(\dot{\mathrm{v}} + u_o r) + M_s h_{CG}\dot{p} = 2C_F\left[\delta_F - \left(\frac{\mathrm{v} + ar}{u_o}\right)\right] + 2C_R\left[\delta_R - \left(\frac{\mathrm{v} - br}{u_o}\right)\right] \tag{113}$$

where the possibility of nonzero δ_R is explicitly taken. Summing moments about the vertical axis through the CG yields the following equation:

$$I_{zz}\dot{r} - I_{zx}\dot{p} = 2aC_F\left[\delta_F - \left(\frac{\mathrm{v} + ar}{u_o}\right)\right] - 2bC_R\left[\delta_R - \left(\frac{\mathrm{v} - br}{u_o}\right)\right] \tag{114}$$

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:

$$I_{xx}\dot{p} - I_{xz}\dot{r} + M_s h_{CG}(\dot{\mathrm{v}} + u_o r) = -(L_{PF} + L_{PR})p - (L_{\phi F} + L_{\phi R})\phi_R \tag{115}$$

where

$p = \dot{\phi}_R$

and

I_{xx} = moment of inertia about the x axis.
$I_{xz} = I_{zx}$ = product of inertia in x and z

The above set of coupled linear differential equations can be written in state variable form with a four-dimensional state vector

$$x = \begin{bmatrix} \mathrm{v} \\ r \\ p \\ \phi_R \end{bmatrix} \tag{116}$$

and input vector

$$u = \begin{bmatrix} \delta_{\mathrm{F}} \\ \delta_{\mathrm{R}} \end{bmatrix}$$

This equation is four dimensional requiring four coupled differential equations. In addition to the three given above, the 4th differential equation is given by

$$p = \dot{\phi}_R$$

The state vector equation is given by

$$G\dot{x} = Hx + Du \tag{117}$$

The matrix G is given by

$$\begin{bmatrix} M & 0 & M_s h_{CG} & 0 \\ 0 & I_{zz} & -I_{zx} & 0 \\ M_s h_{CG} & -I_{xz} & I_{xx} & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{118}$$

The matrix H is given by

$$\begin{bmatrix} -2(C_F + C_R)/u_o & -2(C_F a - C_R b)/u_o - Mu_o & 0 & 0 \\ -2(C_F a - C_R b)/u_o & -2(a^2 C_F + b^2 C_R)/u_o & 0 & 0 \\ 0 & -M_s h_{CG} u_o & -(L_{PF} + L_{PR}) & -(L_{\phi F} + L_{\phi R}) \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

and matrix D is given by

$$D = \begin{bmatrix} 2C_F & 2C_R \\ 2aC_F & -2bC_R \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

The above state variable equation can be put in standard form by multiplying Eqn (117) by the inverse of G yielding

$$\begin{aligned} \dot{x} &= G^{-1}Hx + G^{-1}Du \\ &= Ax + Bu \end{aligned}$$

where

$$\begin{aligned} A &= G^{-1}H \\ B &= G^{-1}D \end{aligned}$$

In the illustration of Figure 8.32, 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:

$M = 1000$ kg
$M_s = 890$ kg
$I_{xx} = 300$ kg m^2
$I_{zz} = 1200$ kg m^2
$I_{xz} = I_{zx} = -11.25$ kg m^2
$a = 1.28$ m
$b = 0.92$ m
$h_{CG} = 0.3$ m
$L_{PF} = L_{PR} = 1045$ N m s/rad
$L_{\phi F} = L_{\phi R} = 15450\ Nm\text{sec}^2$
$C_F = 2 \times 10^4\ N/rad$
$C_R = 2.9 \times 10^4\ N/rad$
$u_o = 30$ m/s

Figure 8.33 qualitatively depicts the car position during alone change maneuver for 2Ws and for 4Ws.

From the simulation, Figure 8.34 plots the steering wheel angle in radians for this lane change-type maneuver as well as 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 twice as much for 4WS as for 2WS during the steering input time interval. Alternatively a given lateral displacement can be achieved in half the time with 4WS as for 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 to 2WS. In fact, such a control strategy is not necessarily desirable for passenger car

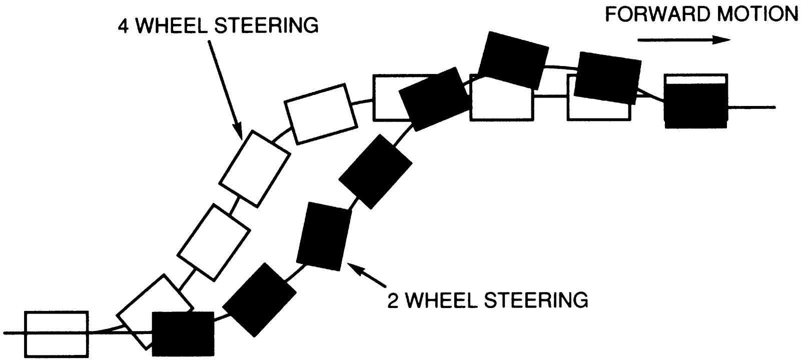


Figure 8.33:
Lane change maneuver (qualitative sketch).

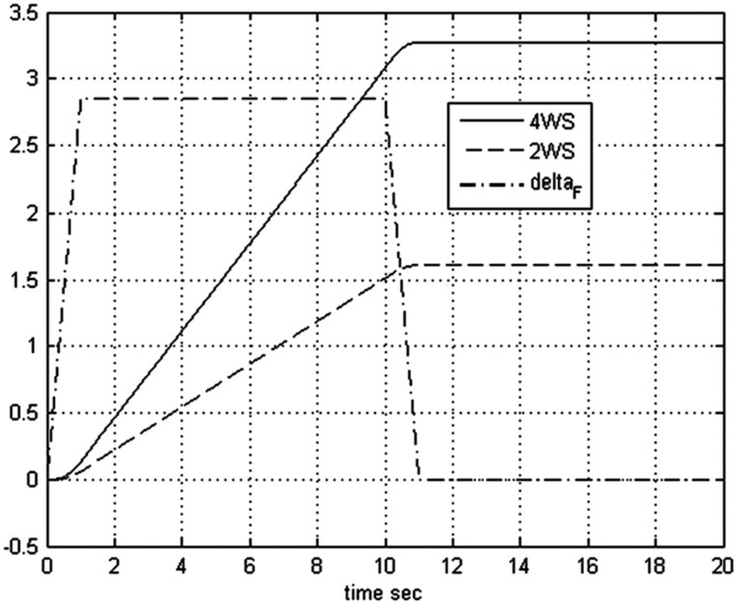


Figure 8.34:
Lane change maneuver 4WS vs. 2WS from simulation.

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 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 un-modeled 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.

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.

This page intentionally left blank

CHAPTER 9

Automotive Instrumentation and Telematics

Chapter Outline

This chapter describes electronic instrumentation and the relatively new field of telematics. 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. By the term *telematics*, we refer to communication of all forms within the vehicle as well as communication to and from the vehicle. Communication within the vehicle takes the form of digital data links between various electronic subsystems. Communication to and from the vehicle spans all communication from voice and digital data via cell or satellite phone

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-08-097097-4.00009-6

systems to digital data sent from land or satellite. Internet connections to an on-board PC (or the like) are included in those categories listed above. This chapter begins with a discussion of electronic instrumentation and concludes with telematics.

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 1950s, however, 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 a loss of signal from a sensor, a lamp illuminates the “Check Engine” message on the instrument panel. Such warning messages alert the driver to seek repairs from authorized technicians who have the expertise and special equipment to perform necessary maintenance.

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 block diagram for instrumentation (first given in Chapter 1), which is repeated here as Figure 9.1. 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 nonelectrical signal to an equivalent voltage or current. Electronic signal processing is then performed on the sensor

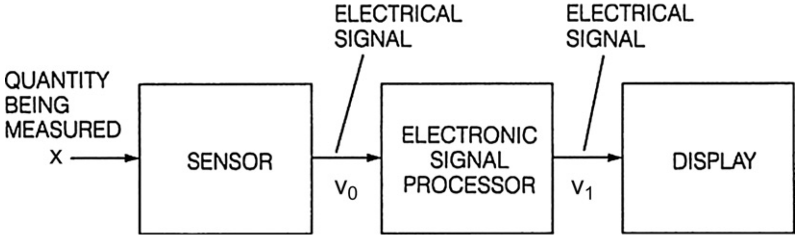


Figure 9.1:
General instrumentation block diagram.

output to produce an 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), then this signal can be used directly and no sensor is required.

As explained in Chapter 1 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:

$$\mathrm{v}_0 = f(x) \tag{1}$$

As explained in Chapter 6 $f(x)$ can take many forms. The simplest and often the most desirable form is a linear transformation for which the sensor model is given by

$$\mathrm{v}_0 = K_s x \tag{2}$$

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

$$\mathrm{v}_0 = K_s \frac{dx}{dt} \tag{3}$$

Signal processing might include integration of the sensor voltage to obtain $x(t)$:

$$x(t) = \frac{1}{K_s} \int_0^t \mathrm{v}_0(\tau) d\tau \tag{4}$$

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 is a linear analog device, then the signal-processing operation can be given by the operation transfer function $(H_{sp}(s))$ where

$$H_{sp}(s) = \frac{v_1(s)}{v_0(s)} \tag{5}$$

where v_0 is the sensor output and v_1 is the input signal to the display (see Figure 9.1).

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 Chapter 2:

$$y_n = \sum_{k=0}^{K} a_k v_0(t_{n-k}) - \sum_{i=1}^{L} b_i y_{n-i} \tag{6}$$

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 for specific measurements are presented throughout this chapter. The hardware for various types of display is discussed later in this chapter.

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 instrument panel. A block diagram for such an instrumentation system is shown in Figure 9.2.

All measurements from the various sensors and switches are processed in a special-purpose digital computer, i.e. the instrumentation computer. The processed signals are routed to the appropriate display or warning message. It is common practice in modern automotive

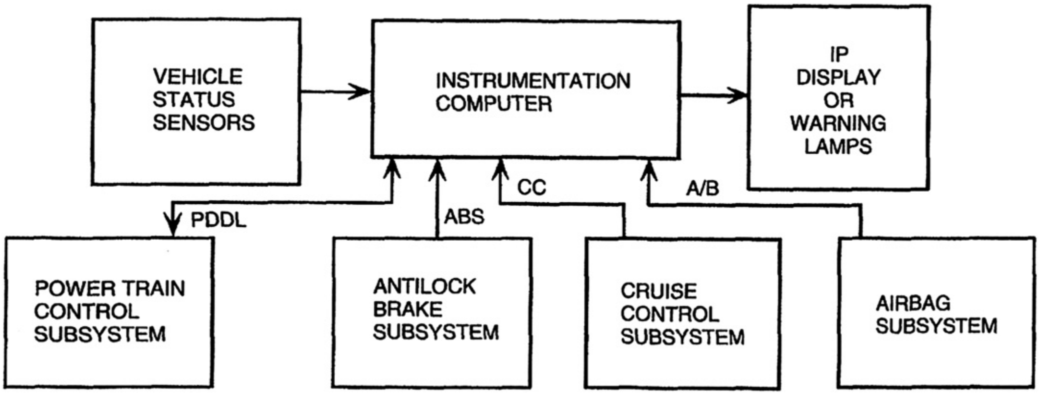


Figure 9.2:
Computer-based instrumentation system.

instrumentation to integrate the display or warning in a single module that may include both 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 as well as 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, and
10. brake fluid quantity.

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 powertrain control system, for example, continuously performs self-diagnosis operations (see Chapter 10). 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 a powertrain digital data line (PDDL in Figure 9.2). 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 10.

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

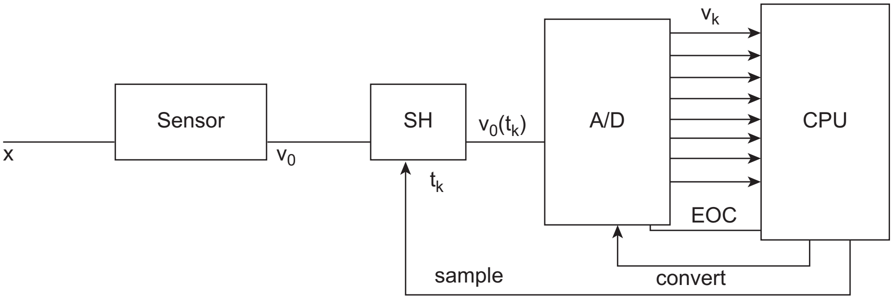


Figure 9.3:
Digital instrumentation input system.

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 8 uses a sensor described in Chapter 6 is an example of a measurement that is already in digital format.

The analog inputs must all be converted to digital format using an analog-to-digital (A/D) converter as explained in Chapter 4 and illustrated in Figure 9.3. In the example of Figure 9.3, a quantity x being measured uses an analog sensor with output voltage $v_o(x)$. The instrumentation computer causes a sample of v_o to be taken at time t_k via a sample and hold (SH) circuit. The sampled voltage $v_o(t_k)$ is, then, converted to a digital input v_k by the A/D converter (see Chapter 4) and is input to the CPU 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 and could, for example, be between 8 and 32). 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 Figure 9.3) that is connected to the computer. (Recall the use of interrupts for this purpose, as discussed in Chapter 4.) 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 digital signal processing are complete, the digital output is transferred to 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 4).

Figure 9.4 illustrates a typical D/A converter used to transform digital computer output to an analog signal.

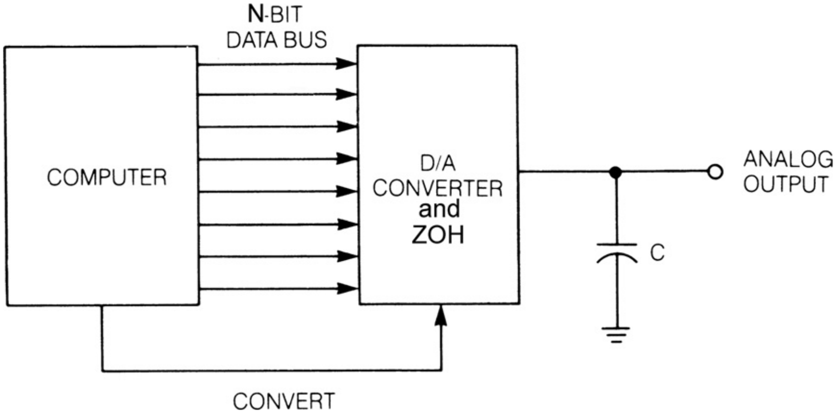


Figure 9.4:
Digital instrumentation analog output.

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 Chapter 2, the D/A conversion often includes a zero-order hold circuit (ZOH). A low-pass filter (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, digital signal processing, 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 shortly.

Multiplexing

Of course, the computer 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 digital signal-processing 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 Figure 9.5.

In the configuration of Figure 9.5, a set of N analog sensors generates output voltages $v_n(t)$. Each of these is connected to an electronic switch (S_n), which, e.g. can be implemented

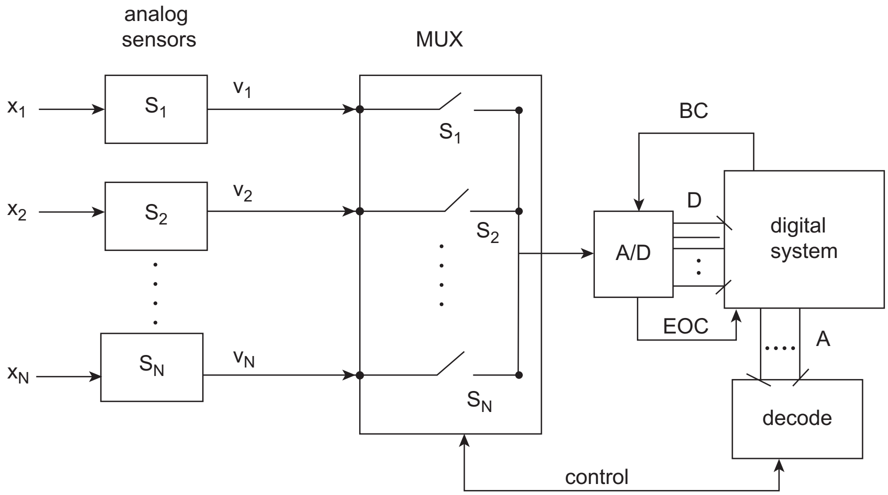


Figure 9.5:
Analog multiplexing system.

using a transistor as described in Chapters 3 and 4 located within an electronic module denoted MUX. The MUX performs the multiplexing function as well as a sampling function. Not shown in this figure are the electrical connections which activate (i.e. close) the normally open switches. In the configuration of Figure 9.5, 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 which activates that switch effectively connecting voltage v_n to the A/D converter. 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 4 for a discussion of sampling).

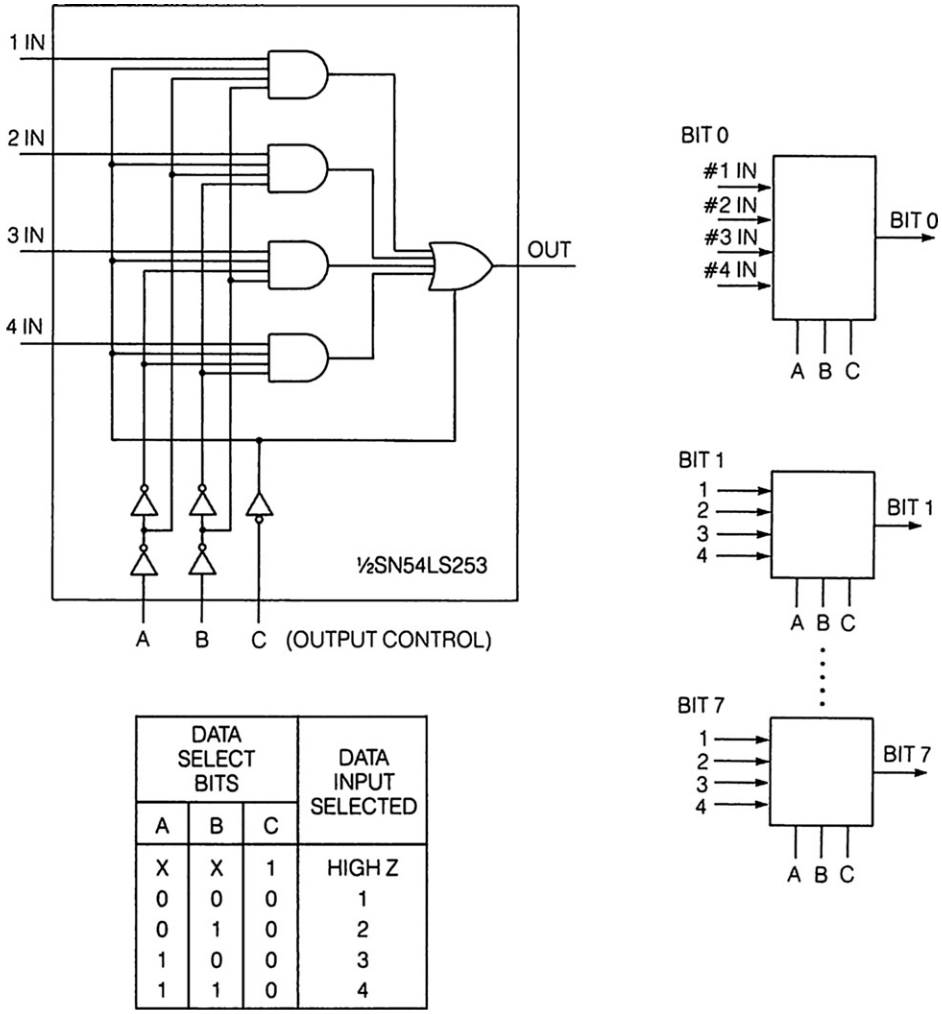
In the configuration of Figure 9.5, it is assumed here that each sensor signal is assigned a time slot within a larger period. In this case, the sample time t_{nk} for sensor n during the kth MUX cycle is given by

$$\begin{aligned} t_{nk} &= T_k + n\delta T \qquad\qquad n = 1, 2 \cdots N, \\ \delta T &= \frac{T_c}{N} \\ T_c &= T_{k+1} - T_k = \text{cycle period} \end{aligned} \tag{7}$$

where N is the 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 8) or from an analog sensor with its own dedicated A/D converter. Figure 9.6 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



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

Figure 9.6:
Digital data multiplexer.

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 line in the digital computer (see Chapter 4). The digital system controls sequencing by generating data select line signals as shown in Figure 9.6. 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 Figure 9.5 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 Figure 9.7) has a time associated with it such that the data y_n is 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.

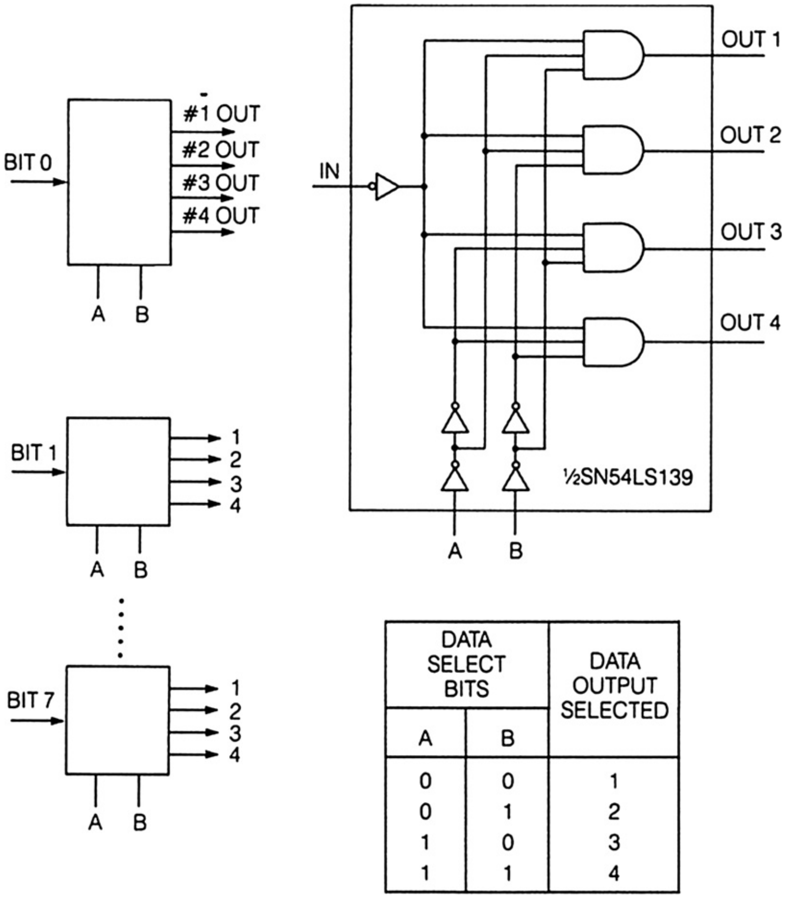
Multirate Sampling

As explained above with respect to the configuration of Figure 9.5, 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 Figure 9.8.

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 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 Figure 9.5 for each subset of variables at a given sample rate. However, it is also possible to achieve multirate sampling with a single MUX which samples signal sensors at the highest rate required. Then, rather than store and process every sample for the low rate variable, the



DATA SELECT BITS		DATA OUTPUT SELECTED
A	B	
0	0	1
0	1	2
1	0	3
1	1	4

Figure 9.7:
Digital data demultiplexer.

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 major advantages of computer-based instrumentation is its great flexibility. 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 ROM

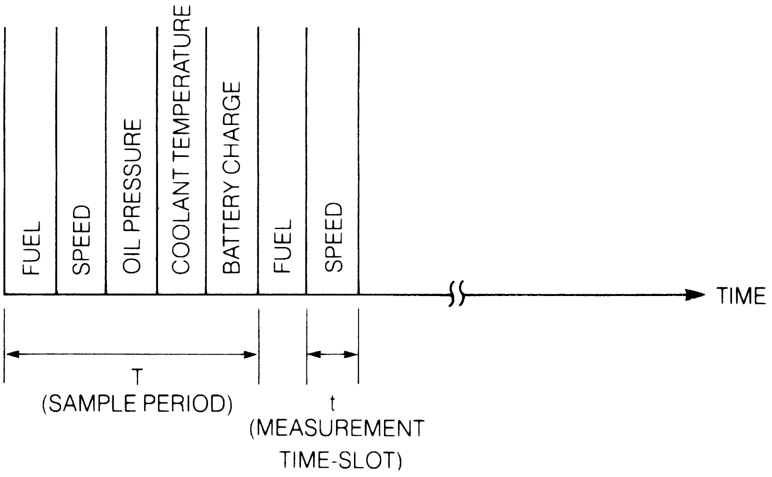


Figure 9.8:
Sequential sampling.

(read-only memory) with another. Remember that the program is permanently stored in a ROM that is typically packaged in a single integrated circuit package (see Chapter 4).

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 geometrical 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. In general, the electronic fuel quantity measurement maintains calibration over essentially the entire range of automotive operating conditions. 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 display devices are in common use in contemporary vehicles.

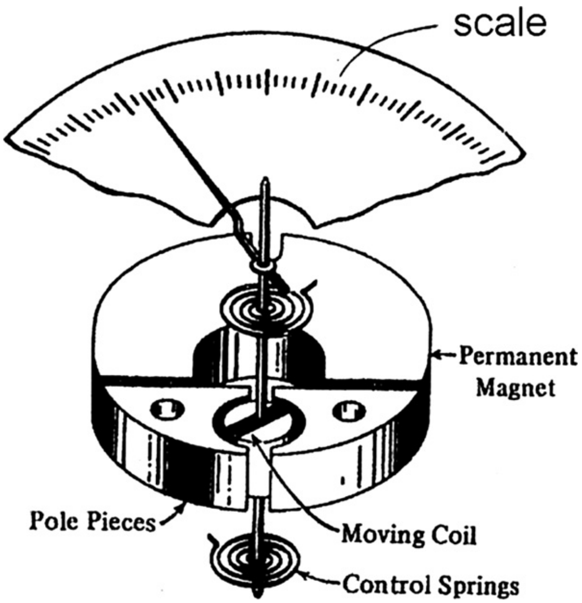


Figure 9.9:
Galvanometer configuration.

Even in certain models of contemporary vehicles, analog display devices are sometimes used (e.g. to display fuel quantity, coolant temperature, oil pressure, and engine RPM). The most common analog electromechanical display is the galvanometer. The physical configuration for a galvanometer is shown in Figure 9.9.

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. The rod to which the pivot is attached is also attached to a coil of wire having N turns as well as one or more springs. A permanent magnet is the source of a magnetic field which 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.

The pole pieces and cylindrical core are designed such that the magnetic flux density vector $\overline{\boldsymbol{B}}$ is directed radially from the pivot axis, inward on the left pole piece and outward on the right. The magnitude of this magnetic flux density B_r is approximately constant over the entire region of coil movement as shown in Figure 9.10.

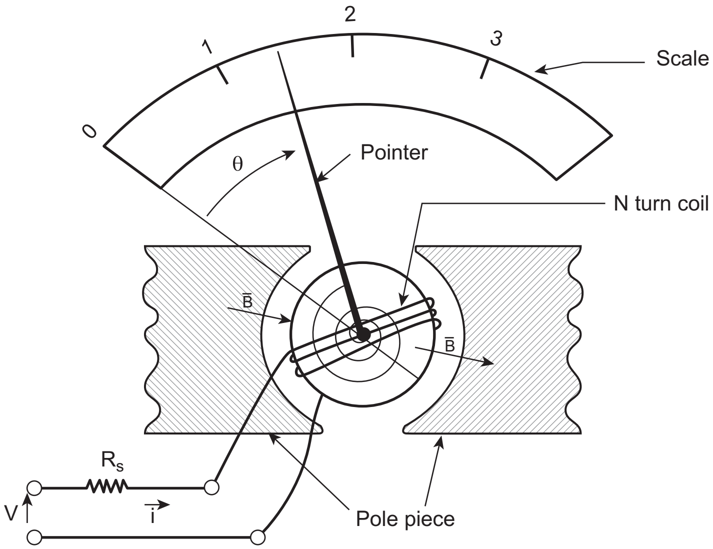


Figure 9.10:
Galvanometer magnetic field configuration.

From basic magnetic field theory (as introduced in Chapter 6), it is known that whenever a current flows through the movable coil, a torque T_c acts on the coil, which is given by

$$T_c = K_c N B_r i \tag{8}$$

where K_c is the constant for the configuration, N is the number of turns on the coil, and i is the 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:

$$\begin{aligned} J\ddot{\theta} + D\dot{\theta} + K\theta &= T_c \\ &= K_c N B_r i \end{aligned} \tag{9}$$

where J is the moment of inertia of the moving assembly about the pivot of axes, D is the viscous damping coefficient for the movable elements, and K is the spring rate of the torsional spring.

From Chapter 1 it can be shown that the operational transfer function for the above model ($H_i(s)$) is given by

$$\begin{aligned} H_i(s) &= \frac{\theta(s)}{i(s)} \\ &= \frac{K_c N B_r}{Js^2 + Ds + K} \end{aligned} \tag{10}$$

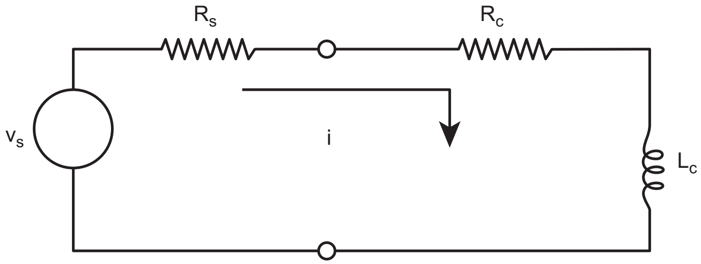


Figure 9.11:
Galvanometer circuit diagram.

The electrical input to the galvanometer is represented by its terminal voltage and source impedance R_s (assumed to be purely resistive). A circuit diagram for the galvanometer is shown in Figure 9.11.

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

$$\mathrm{v}_s = (R_s + R_c)\ i + L_c \frac{di}{dt} \tag{11}$$

Using the Laplace transform methods of Chapter 1, the operational transfer function for the galvanometer $H_g(s)$ can be shown to be

$$\begin{aligned} H_g(s) &= \frac{\theta(s)}{\mathrm{v}_s(s)} \\ &= \frac{K_c B_r N}{(Js^2 + Ds + K)[(R_s + R_c) + sL_c]} \end{aligned} \tag{12}$$

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. 0.1 rad/sec). With such low bandwidth, the fuel quantity display will indicate the time average of the sensor signal. In this case, the relatively rapid fluctuations in sensor output (due, e.g. to fuel sloshing) are 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) 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 $H_g(s)$.

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. This 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. 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 electronic displays for automotive instrumentation. We consider three representative technologies here: light-emitting diode (LED), liquid crystal display (LCD), and vacuum-fluorescent display (VFD). Each of these 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 character. The ultimate application of these electro-optic devices is in the form of a rectangular arrangement of individual devices yielding the so-called flat panel display. This sophisticated display and its operation are explained after the following explanation of the electro-optic technologies. We consider these three 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.

LED

The light-emitting diode is a semiconductor diode that is constructed in a manner and of a material so that light is emitted when an electrical current is passed through it. The semiconductor material most often used for an LED that emits red light is gallium arsenide phosphide. Light is emitted at the diode's p—n junction when the positive carriers combine with the negative carriers at the junction (see Chapter 3 for a discussion of p—n 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 bandgap of the carriers in the junction.

Physically, an LED consists of a chip of semiconducting material doped with impurities to create a p—n junction as depicted in Figure 9.12. The LED configuration and excitation voltage (V_{ex}) source are depicted in Figure 9.12. The polarity for this voltage source forward biases the junction. The majority charge carriers flow readily across the junction region which 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 bandgap energy ΔE_g.

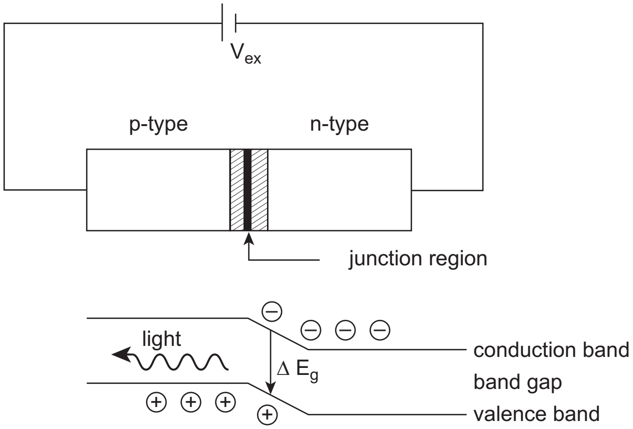


Figure 9.12:
LED configuration and energy bands.

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:

$$\nu = \frac{\Delta E_g}{h} \tag{13}$$

where h is the Planck's constant. The wavelength (λ) of the photon is given by

$$\lambda = \frac{c}{\nu} = \frac{ch}{\Delta E_g} \tag{14}$$

where c is the speed of light.

The color of the emitted light is determined by its wavelength which, in turn, depends upon ΔE_g. The energy bandgap (and hence the color of the emitted light) is determined by the semiconductor material as well as 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 silicon (Si) $n = 4.24$ relative to air (1.000)). The index of refraction for any material n is given by

$$n = \frac{c_o}{c}$$

where c is the speed of light in the material and c_0 is the 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 Si material at which it can leave the material. Angles of incidence greater than θ_c result in total internal reflection with no light leaving the material. For the LED, θ_c is given by

$$\begin{aligned}\theta_c &= \sin^{-1}(1/n_i)\\ &= \sin^{-1}(1/4.24)\\ &= 13.6^\circ\end{aligned}$$

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.

LCD

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 Figure 9.13. 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 onto 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° 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.

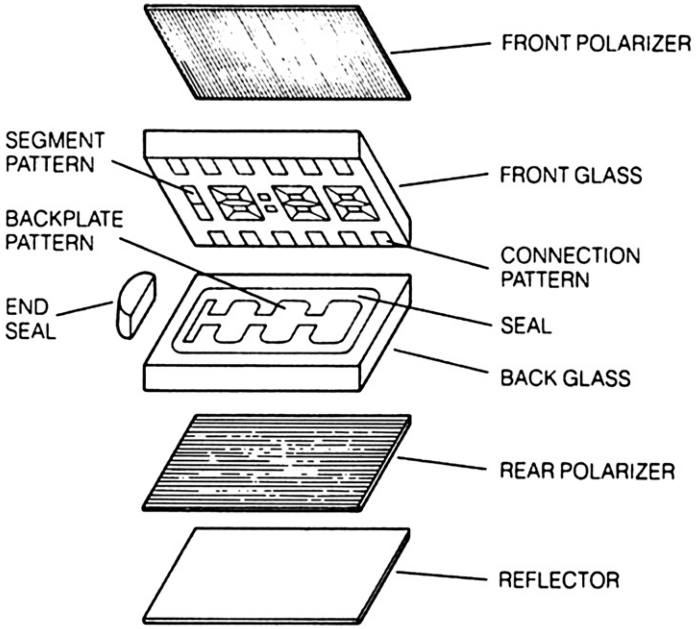


Figure 9.13:
Typical LCD construction.

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 an electromagnetic wave (e.g. radio wave) at a single frequency $\omega = 2\pi\text{v}$ can be represented locally 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 $\overline{\boldsymbol{E}}$ is assumed to be x-directed and is given by

$$\overline{\boldsymbol{E}}(z,t) = E_x\hat{\boldsymbol{x}}e^{j(\omega t-kz)} \tag{15}$$

where

$$k = \frac{2\pi}{\lambda}$$

$\hat{\boldsymbol{x}}$ is the unit vector in x direction and where λ is the wavelength:

$$\lambda = \frac{c}{\text{v}}$$

where c is the speed of light in the medium of propagation:

$$c = \frac{c_o}{n}$$

c_o is the vacuum speed of light and n is the index of refraction of the medium.

The magnetic field intensity vector $\overline{\boldsymbol{H}}$ is given by

$$\overline{\boldsymbol{H}}(z,t) = H\hat{\boldsymbol{y}}e^{j(wt-kz-\pi/2)} \tag{16}$$

where $\hat{\boldsymbol{y}}$ is a unit vector in the y direction.

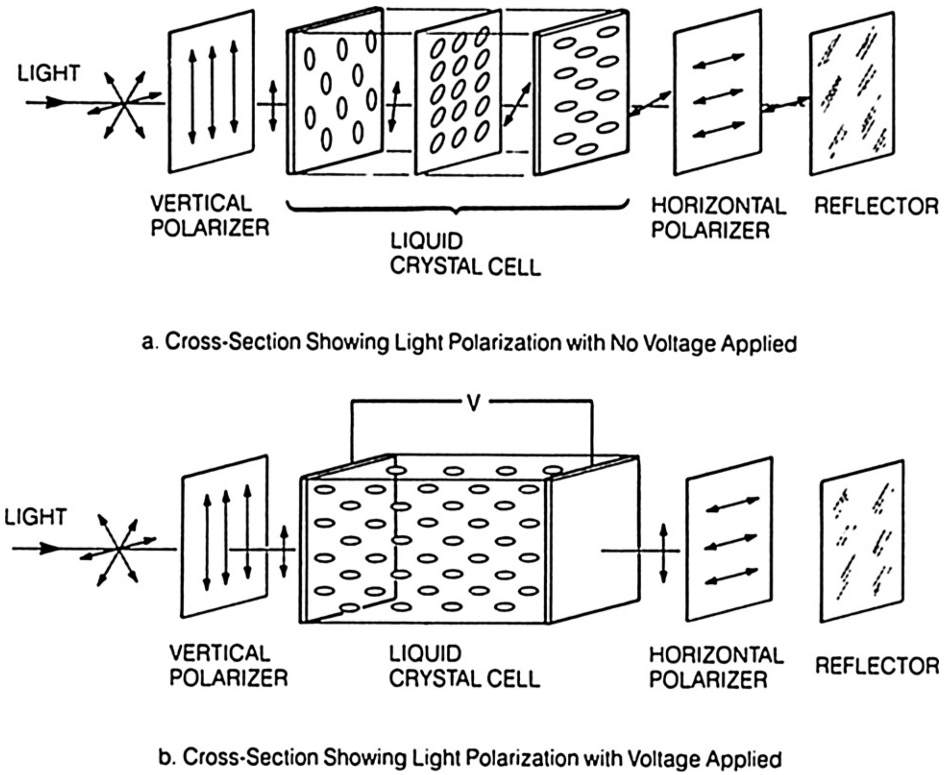
The above electromagnetic wave is said to be linearly polarized because the field intensities $\overline{\boldsymbol{E}}, \overline{\boldsymbol{H}}$ have the directions $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$, respectively. Light from the sun and from most artificial light sources is not polarized, 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 which has the property of having two different indices 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 polarization 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 Figure 9.14a. 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° 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 Figure 9.14b. 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



a. Cross-Section Showing Light Polarization with No Voltage Applied

b. Cross-Section Showing Light Polarization with Voltage Applied

Figure 9.14:
Liquid-crystal polarization.

rather than twisted. In this case, the light that enters the liquid crystal in the vicinity of the segments passes through the crystal structure 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 multisegment 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 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. nighttime). 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 displays. Such displays are capable of presenting complex programmable alphanumeric messages to the driver and can also present graphical 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 book devoted to the discussion of the flat panel solid-state display.

VFD

The VFD has been widely used in automotive instrumentation, although the multicolor LCD is becoming the preferred choice for this application. This 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.

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 Figure 9.15. The filament is a special type of resistance wire and is heated by passing an electrical 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 such a device.

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

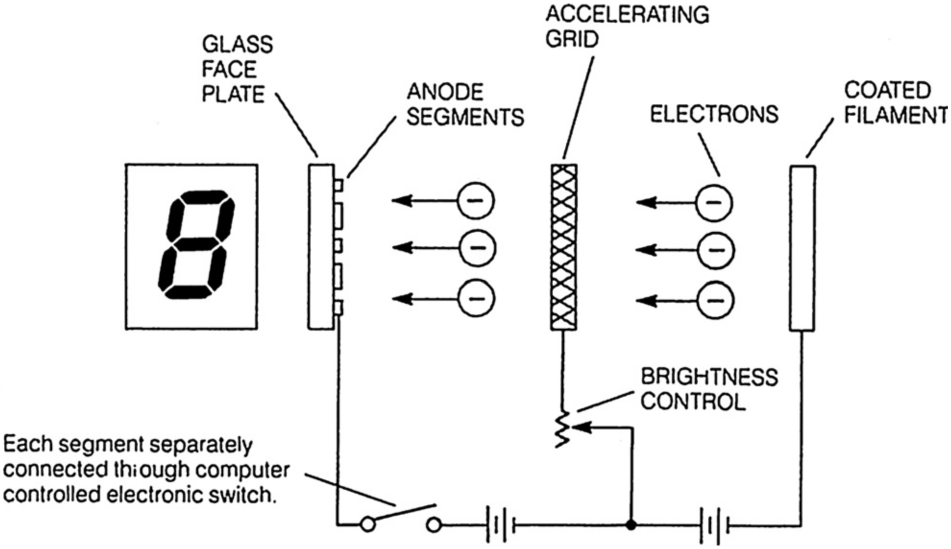


Figure 9.15:
Simplified vacuum-fluorescent display configuration.

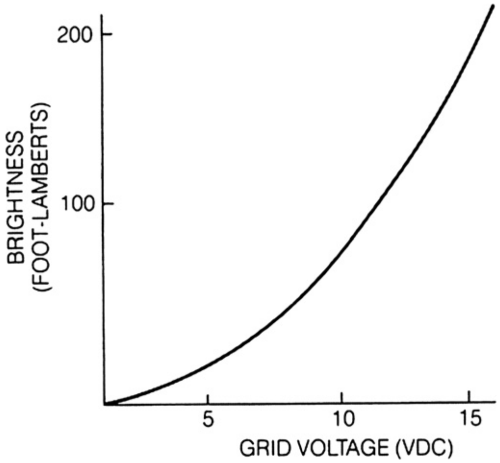


Figure 9.16:
Brightness control range for vacuum-fluorescent display.

the phosphor and the brightness of the light is an increasing function of the grid voltage. Figure 9.16 shows the brightness characteristics for a typical VFD device.

A brightness of 200 fL (foot-lamberts) 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 ambient light. A photoresistor is a device whose resistance varies as a function of the amount of light striking it (see discussion of optical sensors in Chapter 6).

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.

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 further in the next section of this chapter.

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.

Flat Panel Display

Arguably, the display device with the greatest flexibility for presenting all types of data (including pictorial representations) is the so-called flat panel display. 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 flat panel (FPD) display has great potential for automotive instrumentation display. The FPD can be implemented with various electro-optical technologies as described above. We assume for convenience that LCD technology is employed in the following discussion.

A solid-state LCD display consists of an array of LCDs arranged in a matrix format as depicted in Figure 9.17. The individual elements in such a display are termed pixels. In this

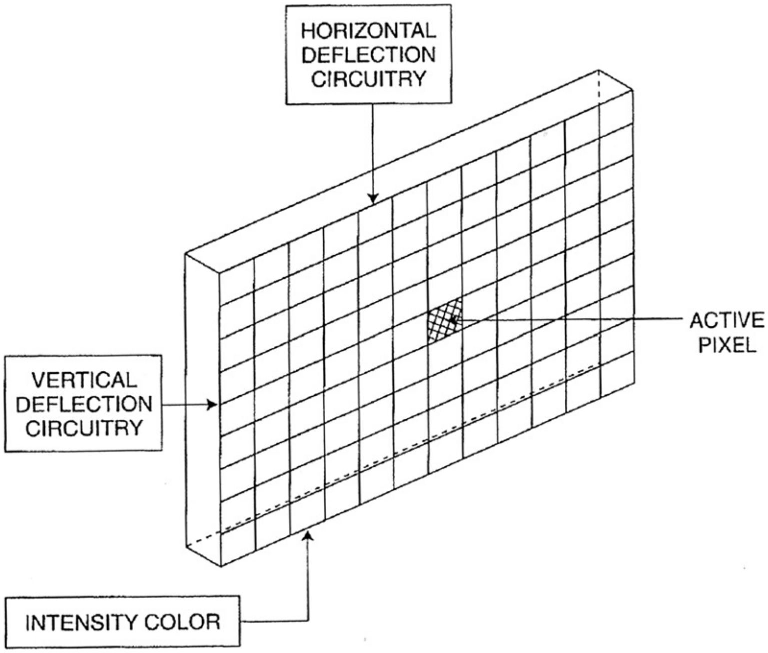

HORIZONTAL DEFLECTION CIRCUITRY
ACTIVE PIXEL
VERTICAL DEFLECTION CIRCUITRY
INTENSITY COLOR

Figure 9.17:
Solid-state array-type display.

structure, only one LCD is active at any time. The intensity of the active LCD is controlled by circuitry connected within the microstructure. The active LCD is selected via horizontal and vertical control circuitry. In the 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 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 LED, VFD, or LCD, as depicted in Figure 9.17. 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 graphical 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 is 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 is 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. top to bottom). For such a display, the pixel located at $x_m y_n$ is selected at time $t_{m,n}$ where

$$t_{m,n} = nT_{py} + mT_{px} \qquad n = 1, 2 \ldots N, m = 1, 2 \ldots M \tag{17}$$

and where

$$T_{py} = MT_{px}$$

and

$$T_c = NT_{py}$$

where T_c is the period required to scan the entire display:

$$T_c = \frac{1}{f_c}$$

$$f_c = \text{picture cycle(or refresh rate)}$$

$$T_{px} = \frac{1}{f_x} \tag{18}$$

$$f_x = \text{frequency at which columns are scanned}$$

$$T_{py} = \frac{1}{f_y} \tag{19}$$

$$f_y = \text{frequency at which rows are scanned}$$

In a raster scan type of display, the scanning is done for one row at a time from left to right beginning at the top row during each complete scanning cycle. For the *n*th 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 from 30 to 60 Hz typically).

One hypothetical configuration for implementing this raster-type scan is shown schematically in Figure 9.18. 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 *m*th column electrical lead and the *n*th row lead are simultaneously at high voltage the pixel at

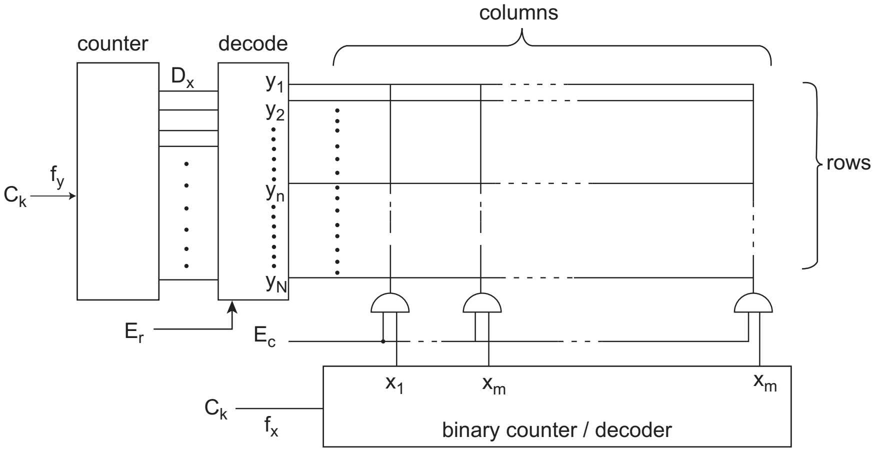


Figure 9.18:
Schematic illustration of representative pixel drive circuits.

$x_{m,}y_n$ is active. By active we mean that it is illuminated if it is an LED or VFD type or made dark in an otherwise illuminated matrix if it is an LCD. The control of whether a pixel is active or not in this hypothetical configuration is controlled by the digital display via logical signals E_r and E_c, which either enable the signal for a given column or disable 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 (C_k) pulses at frequency f_x. For each pulse received, the column counter is incremented by one continuing modulo M. Similarly, the row counter receives clock pulses at frequency f_y and is incremented by one 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 4). 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 as well as the illumination period. It has long been known from analog TV that a refresh frequency of 60 Hz is sufficient to satisfy the visual persistency requirement.

In the above-described raster-type scan operation of a flat panel display, the counters and "clocks" at frequencies $f_x f_y$ 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

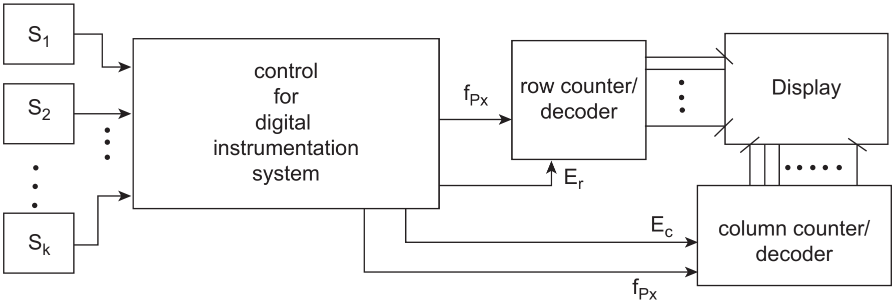


Figure 9.19:
Block diagram for flat panel display controller.

be inactive, this signal is set to logical low, thereby inhibiting the column select signal from establishing the voltage output column lead and, in turn, thereby rendering that pixel inactive.

A block diagram of the complete hypothetical flat panel display is shown in Figure 9.19. In this system, the digital instrumentation system, which is microprocessor based and which 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 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.

The logical value for each pixel i.e. 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 graphical, 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.

An alternative to a raster-type flat panel display 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 Figure 9.20. In the display depicted in Figure 9.20, the logical variables $E_{m,n}$ for each pixel are stored at memory

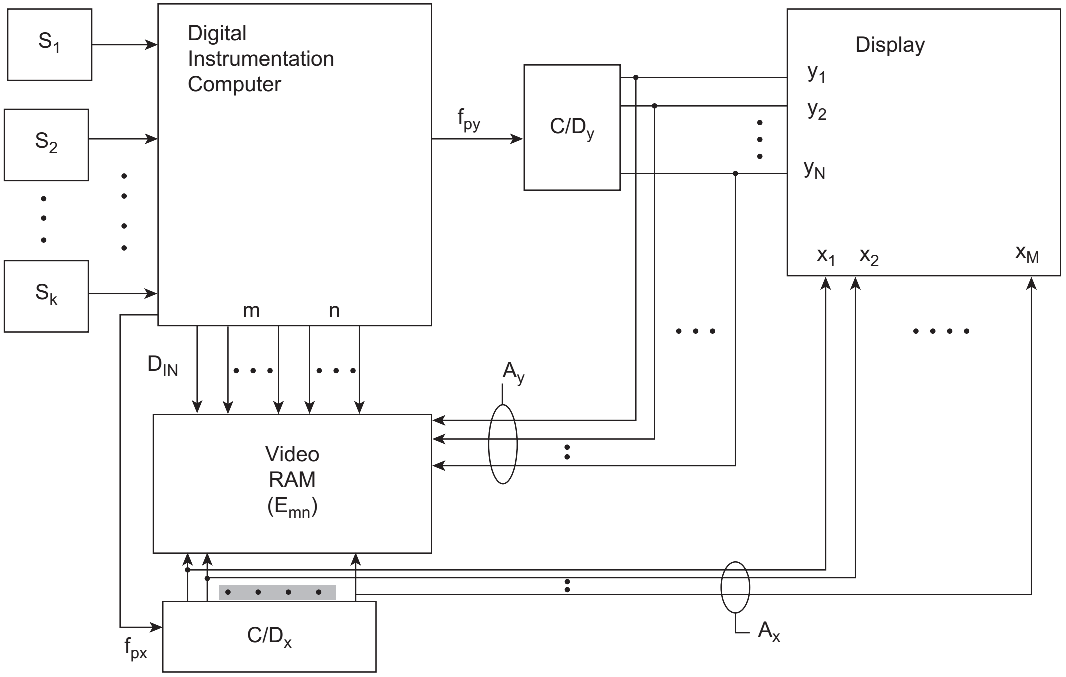


Figure 9.20:
Random access display block diagram.

locations corresponding to pixel locations $x_{m,}y_n$ via address buses carrying addresses m and n corresponding to position x_m and y_n respectively. The logical value (E_{mn}) for each pixel is supplied via D_{IN} during a data write operation by the computer. 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 Figure 9.20, 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 which continuously refreshes the display.

The digital system can "write" data (see Chapter 4) 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 flat panel display 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.

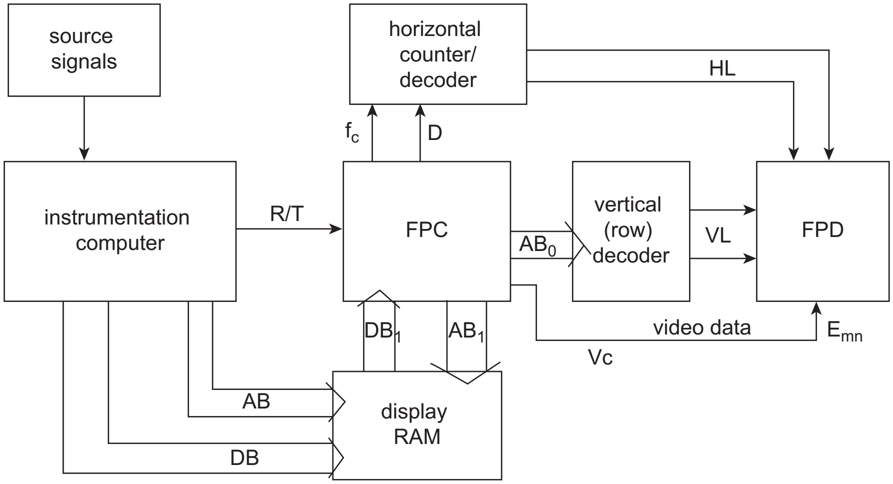


Figure 9.21:
Block diagram of automotive instrumentation with FPD.

A simplified block diagram for a system incorporating an FPD-type display with the associated controller is depicted in Figure 9.21.

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 Figure 9.2. The output of the instrumentation computer controls the flat panel display, working through flat panel display controller (FPC).

In the example architecture of Figure 9.21, 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 flat panel display 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 (Figure 9.21) itself incorporates an MPU for controlling the FPD. The data to be displayed are stored in the display RAM via the system buses under 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. 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 control of the instrumentation computer. This computer transfers data that are to be displayed to the video RAM, via address bus AB and data bus DB and signals the display controller via control R/T. The FPC outputs a clock signal at frequency f_c to the horizontal counter/decoder and initiates counting via control D in Figure 9.21. 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 which 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 shown in Figure 9.21, the display is assumed to be an array of LED, VFD, or LCD elements arranged in 240 rows vertically by 480 columns horizontally. Figure 9.22 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 $E_{mn} = 0$.

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

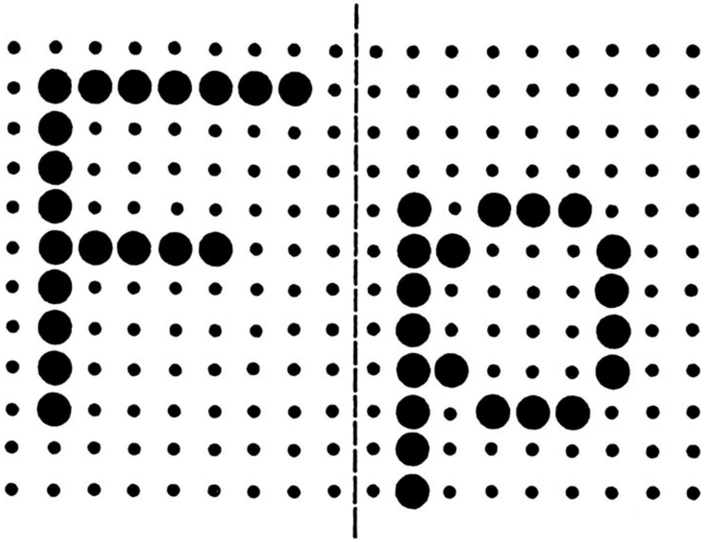


Figure 9.22:
Display of characters F and P.

parameters that have traditionally been available to the driver, the FPD-type display can present engine data for diagnostic purposes (see Chapter 10), 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.

Fuel Quantity Measurement

Having described the various components and implementation of automotive instrumentation, we now present some specific measurement examples. During a measurement of fuel quantity, the MUX switch functionally connects the computer input to the fuel quantity sensor, as shown in Figure 9.23. 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.)

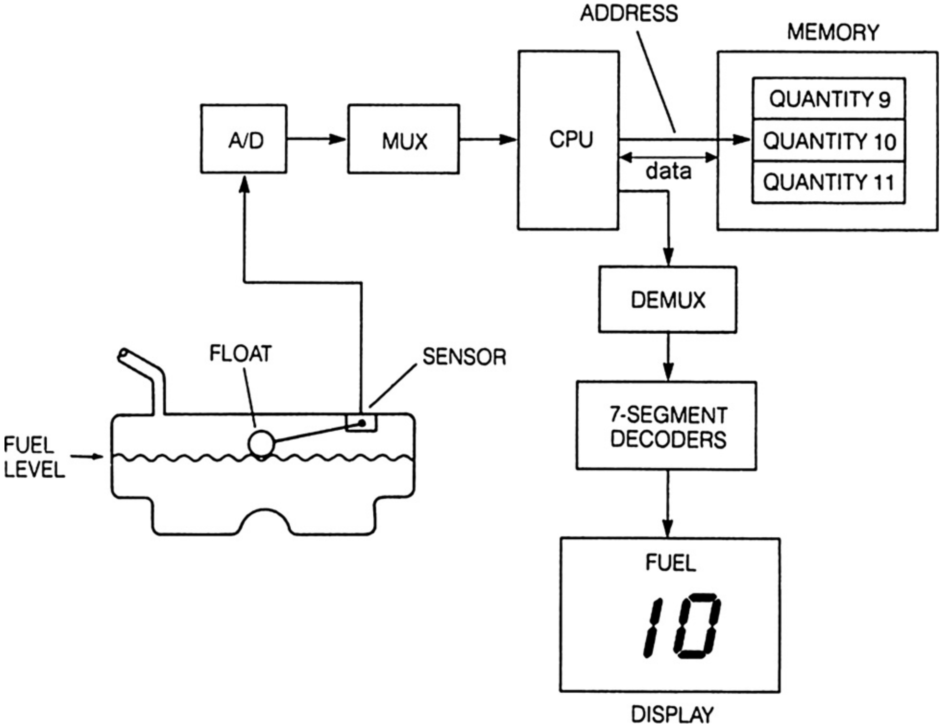


Figure 9.23:
Fuel quantity measurement system.

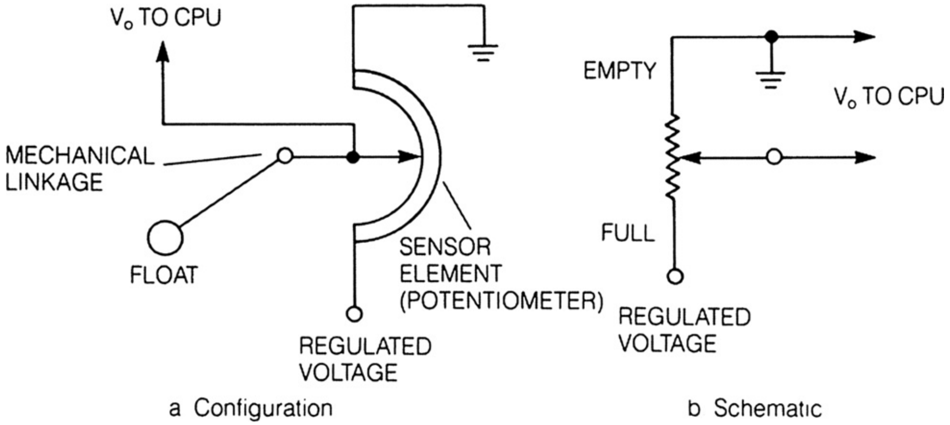


Figure 9.24:
Fuel quantity sensor configuration.

Several fuel quantity sensor configurations are available. Figure 9.24 illustrates the type of sensor to be described, which is a potentiometer connected via mechanical linkage to a float. In Chapter 6, 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.

The computer reads the binary number from the A/D converter (see Figure 9.23) 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 Figure 9.24). 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 $\mathrm{v}_n = \mathrm{v}_0(t_n)$ be the fuel sensor voltages at the nth sample time. 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:

$$\mathrm{v}_0 = f_F(F) \tag{20}$$

The instrumentation computer (under program control) computes the sampled measurement F_n from v_n:

$$F_n = f_F^{-1}(\mathrm{v}_n) \tag{21}$$

The short-term time average of fuel quantity F_{av} is given by

$$F_{\mathrm{av}}(n) = \frac{1}{N}\sum_{m=1}^{N} F_{n-m} = \frac{1}{N}\sum_{m=1}^{N} f_F^{-1}(\mathrm{v}_{n-m}) \tag{22}$$

The sloshing effect of fuel on fuel gauge indicated value can be reduced by filtering the fuel sensor output. A block diagram of a fuel measurement instrumentation configuration is shown in Figure 9.25.

In this figure, the sensor output voltage is given by Eqn (20). The digital signal processing (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 Eqn (21). The DSP also includes a low-pass filter LPF which has z-transfer function $H_{\mathrm{sp}}(z)$. 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.

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

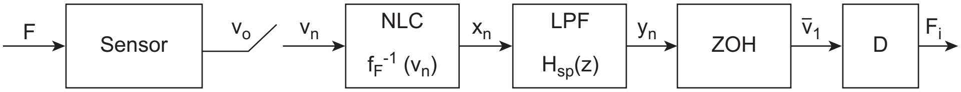


Figure 9.25:
Filtering fuel sensor signal.

time transfer function for this type of display $(H_0(s))$ was shown earlier (Eqn (12)) to be given by

$$
\begin{aligned}
H_D(s) &= \frac{\theta(s)}{\mathrm{v}_1(s)} \\
&= \frac{K_c N B_r}{(Js^2 + sD + K)[(R_c + R_s) + sL_c]}
\end{aligned}
\tag{23}
$$

where v_1 is the signal processing output of Figure 9.25 and is the source voltage for the galvanometer display.

A typical galvanometer display is designed such that the torque component proportional to the moment of inertia (*J*) is insignificant compared to the damping and spring torques. Thus, the transfer function is given approximately by its dominant pole factor:

$$
H_0(s) \cong \frac{K_c N B_r}{D L_C (s + s_0)(s + s_L)} \tag{24}
$$

where

$$
s_0 = \frac{K}{D}
$$

$$
s_L = \frac{R_s + R_c}{L_c}
$$

Using numerical values for a representative automotive analog fuel gauge, these frequency parameters are the approximate ranges given below:

$$
\begin{aligned}
s_0 &\cong 0.5 \text{ to } 2.0 \\
s_L &\cong 10^5 \text{ to } 10^6
\end{aligned}
$$

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

$$
\begin{aligned}
H_D(s) &\cong \frac{K_c N B_r}{D(R_s + R_c)(s + s_0)} \\
&= \frac{K_D}{s + s_0}
\end{aligned}
\tag{25}
$$

where K_D is a constant for the display that is given by

$$K_D = \frac{K_c N B_r}{D(R_s + R_c)} \tag{26}$$

Using the methods of Chapter 2 it can be shown that the z-transfer function for the combination ZOH and display $H_D(z)$ is given by

$$H_D(z) = (1 - z^{-1})\mathscr{Z}\left(\frac{H_D(s)}{s}\right) \tag{27}$$

Representative values for K and S_0 are given by

$$\begin{aligned} K_D &= 0.5 \\ s_0 &= 0.5 \end{aligned}$$

For a sample period of $T = 0.001$ sec, the z-transfer function is given by

$$H_D(z) = \frac{(1 - z_1)}{(z - z_1)} \tag{28}$$

where $z_1 = \mathrm{e}^{-s_0 T} = 0.9995$

The digital filter is chosen as a second-order Butterworth filter having a digital corner frequency $\Omega_c = 0.001$. It can be shown that the z-transfer function for this filter is

$$H_{\mathrm{sp}}(z) = 10^{-5}\frac{[0.2462z^2 + 0.4924z + 0.2462]}{z^2 - 1.9956z + 0.9956} \tag{29}$$

The dynamic performance of this digital fuel measurement system can readily be demonstrated via simulation. The SIMULINK simulation model block diagram is shown in Figure 9.26. The fuel tank is assumed to be ½ full ($F = .5$) 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 ½ full tank. The block labeled discrete transfer function is the second-order Butterworth digital filter having transfer function $H_{\mathrm{sp}}(z)$ of Eqn (29). The final continuous time transfer function represents the display dynamic response having gain $K_D = 0.5$ and bandwidth $s_o = 0.5$.

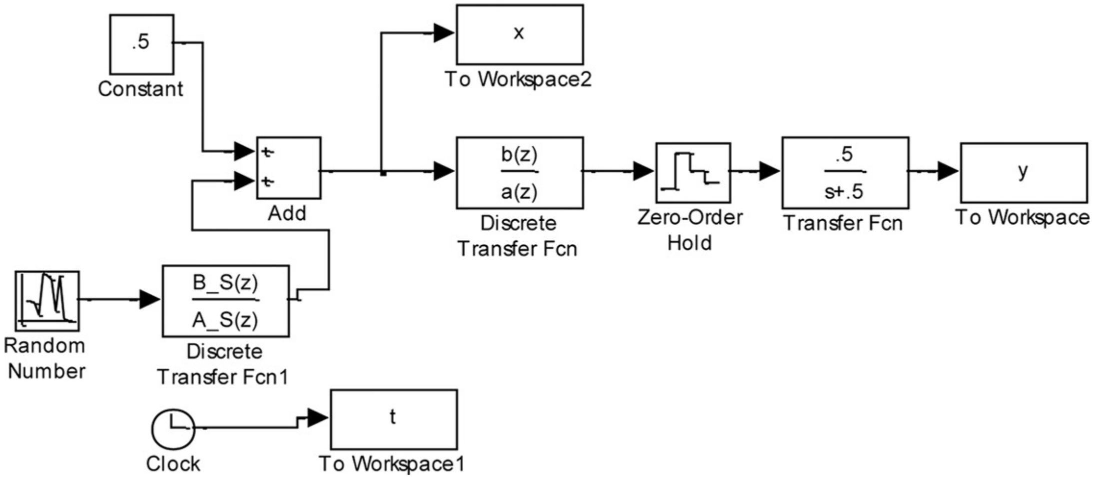


Figure 9.26:
SIMULINK model for fuel quantity instrument subsystem.

A sample of the system response is shown in Figure 9.27 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.

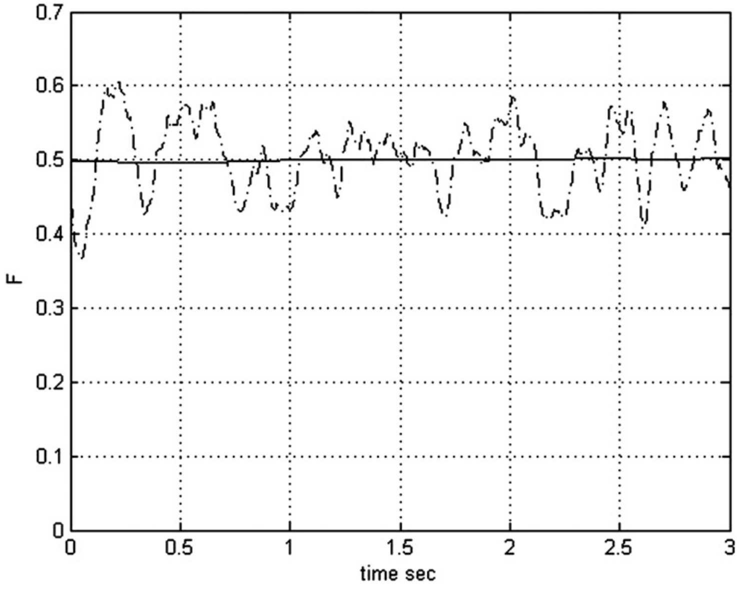


Figure 9.27:
Filtered fuel quantity F (solid line) and unfiltered fuel quantity (dashed line).

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 that the coolant temperature is less than a critical value. A block diagram of the measuring system is shown in Figure 9.28.

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 6, where it was shown that the resistance of this sensor decreases with increasing temperature. Figure 9.29 shows the circuit connection and a sketch of a typical sensor output voltage (v_0) versus temperature (T) curve.

The sensor output voltage is sampled during the appropriate time slot and is sampled (S) 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, an output signal is generated that activates the

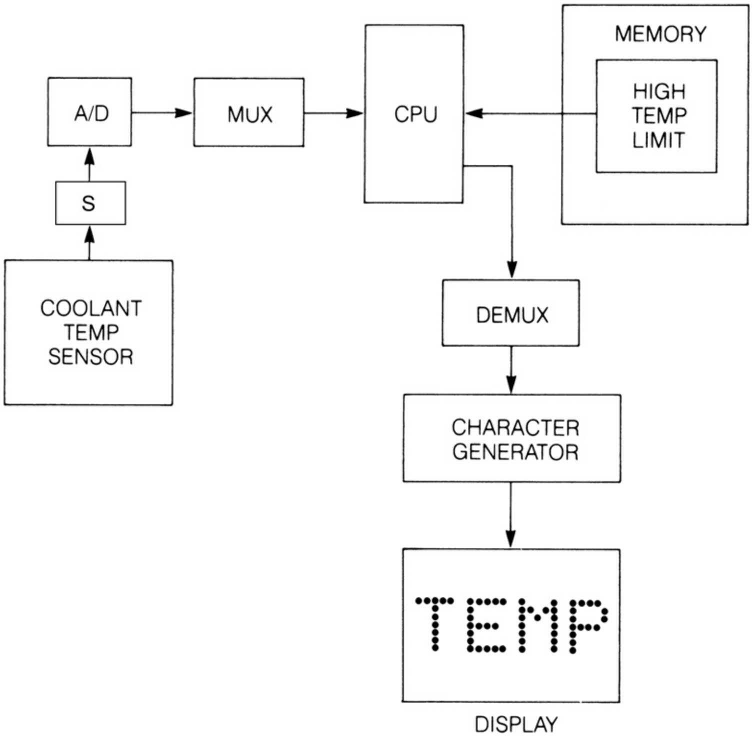
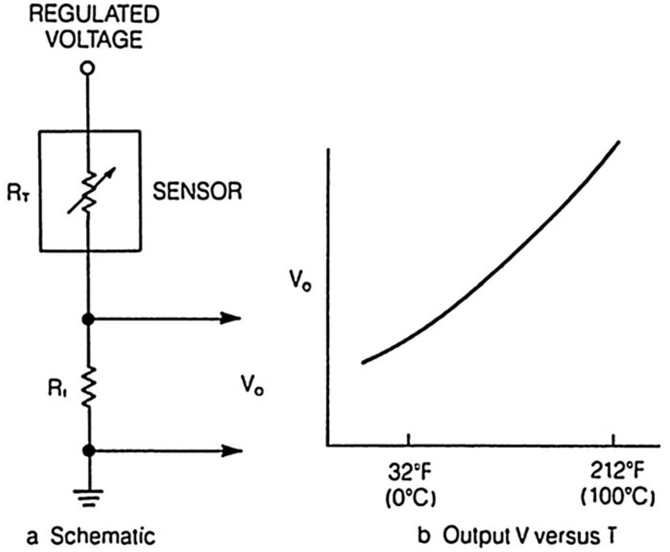


Figure 9.28:
Coolant temperature measurement.



a Schematic

b Output V versus T

Figure 9.29:
Coolant temperature sensor circuit.

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, many manufacturers do not include a 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 Figure 9.30. This system uses a variable-resistance oil pressure sensor (e.g. piezoresistive) such as shown in Figure 9.31. Sensors of this type were discussed in Chapter 6. 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.

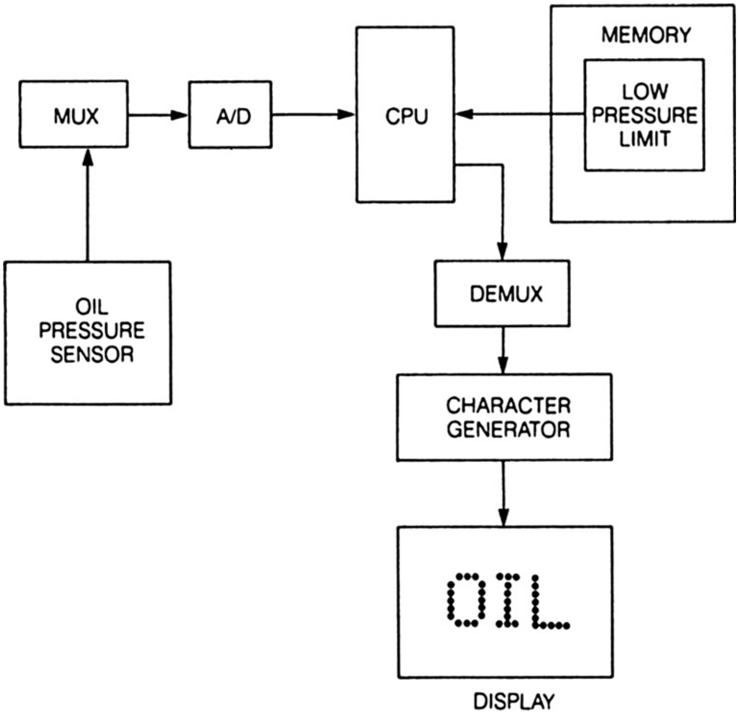


Figure 9.30:
Oil pressure measurement instrumentation.

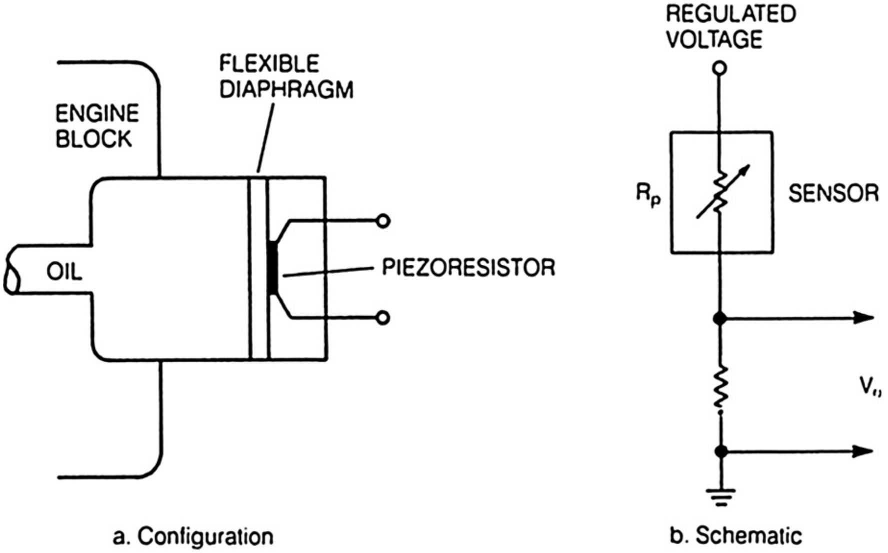


Figure 9.31:
Oil pressure sensor.

During the appropriate measurement time slot, the oil pressure sensor voltage is sampled through the analog MUX switch and converted to a binary number 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 Figure 9.5. 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 Figure 9.7).

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.

Vehicle Speed Measurement

An example of a digital speed sensor has already been described in Chapter 8 for a cruise control system. The speed sensor is assumed to be of a structure such as is depicted in Figure 6.10 or 6.13. 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 car speed V:

$$f_p = k_s V \tag{30}$$

where k$_s$ is a constant for the sensor system.

A block diagram of the digital system (including the instrumentation computer) that determines vehicle speed from the speed sensor is depicted in Figure 9.32. 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 4). 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 Figure 9.32 is functionally an electronically controlled switch (e.g. implemented by an FET; see Chapter 3) 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:

$$\begin{aligned} G &= 1 \quad t_k \leq t \leq t_k + \tau \\ &= 0 \quad t_k + \tau < t < t_{k+1} \\ t_{k+1} - t_k &= T_s = \text{sample period} \end{aligned} \tag{31}$$

During the period in which $G = 1$, each sensor pulse causes the counter to increment by one. Thus, at time $t_k + \tau$ the counter contains count P where

$$P = \left\{ \lfloor f_p \tau \rfloor \right\} \tag{32}$$

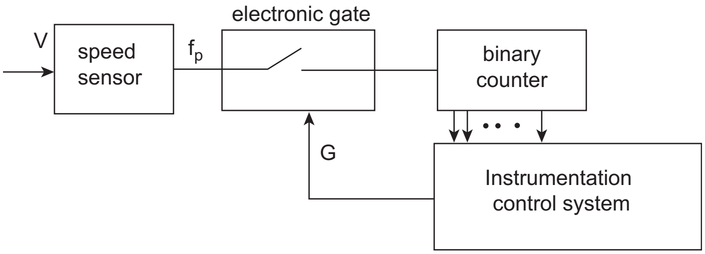


Figure 9.32:
Vehicle speed instrument subsystem.

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:

$$V = \frac{P}{k_s \tau} \tag{33}$$

where k_s is the speed sensor constant given above in Eqn (30).

The computer reads the number P in the binary counter, 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 flat panel display is now commonly used for displaying such measurements. This display has sufficient flexibility and detailed resolution that graphical 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 a high-speed digital data (HSDD) link between the various on-board electronics systems. In the next section, we discuss high-speed intermodule digital communication systems.

High-Speed Digital Communications (CAN)

As has been shown in the preceding sections of this book, there are multiple electronic subsystems on board any vehicle. It is normal for various electronic subsystems to make use of the same variable. For example, vehicle speed measurements are used by the instrumentation subsystem, powertrain control, and possibly vehicle suspension or steering subsystems. In order to share such measurements or to exchange other data (e.g. subsystem status) it has become necessary to provide a high data rate digital communication, which is often denoted HSDD for high-speed digital data link network in the vehicle.

Figure 9.33 is a block diagram of an integrated vehicle instrumentation system in which all on-board electronic systems are coupled by an HSDD link.

This system requires 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

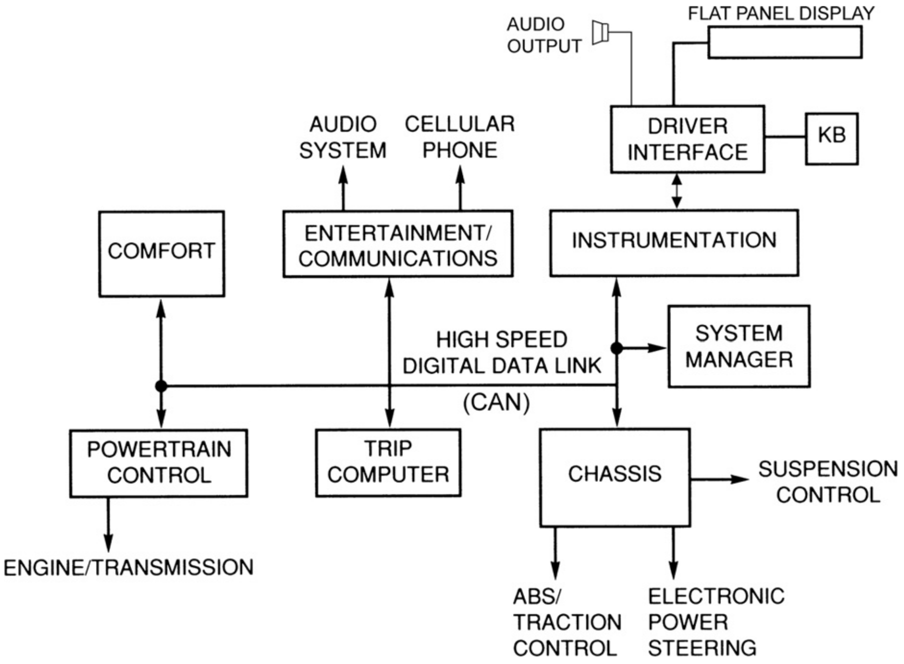


Figure 9.33:
Block diagram for vehicle digital data system

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 on-board electronic system (see Chapter 10).

In Figure 9.33, 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 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.

The various subsystems in Figure 9.33 have all been identified in other sections of this book and will not be discussed further here, except for the system manager. This subsystem is responsible for coordinating data transfer and regulating the use of the data bus so that no two systems are transmitting simultaneously.

CAN Network

Essentially, the digital data link 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 an HSC which it terms a "Controller Area Network" or CAN. This CAN operates at a data rate of 500 kilobits/sec (KBSC) and can be implemented with wire or optical fiber. Originally developed for passenger car applications, CAN is a form of local area network that permits data to be shared.

Some form of network arbitration is required for determining priority of the use of the link whenever there is conflict between subsystems for its use. This feature is typically handled by the system manager subsystem (see Figure 9.33).

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 varies). 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) *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 (i.e. subsystem) 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 as well as 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 gateway (subsystem) can 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. 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 in terms of its physical layer and portions of the data link layer. It primarily focuses on a minimum standard level of performance from the HSDD CAN 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 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 between CAN_H and CAN_L that is approximately 0. The recessive state represents a logical 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 logical 0-bit value.

The SAE J_2284-3 standard gives a number of definitions of terms by which the CAN HSC can be understood. The term "media" refers to the physical structure/configuration which conveys the electrical transmission between ECUs on the network and may, for example, be 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 as well as its priority. The message also includes a field for the information being sent in the form of eight 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 Figure 9.34 which illustrates a CAN with N ECUs.

The configuration of the CAN shown in Figure 9.34 includes a connection to an off-board diagnostics tool (ECU_{N-2}) via a data link connector (DLC). Each ECU is connected to the CAN via a stub whose length (L_1) must satisfy

$$0 < L_1 \leq 1\text{m}$$

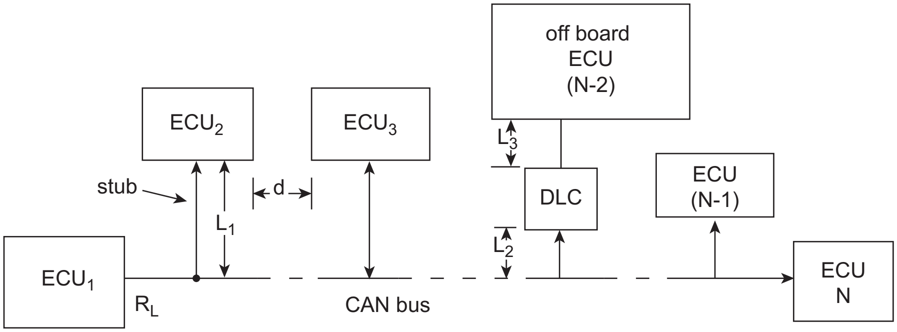


Figure 9.34:
CAN bus architecture

The stub length to the DLC L_2 has the same requirement as L_1. The off-board stub length (L_3) must satisfy

$$0 < \mathrm{L}_3 \leq 5\mathrm{m}$$

The distance between and two ECUs including cable stubs (d) must satisfy

$$0.1 \leq \mathrm{d} \leq 33\mathrm{m}$$

The CAN must be terminated at either end with a resistance R_L which has tolerance range

$$118 \leq \mathrm{R_L} \leq 132\Omega$$

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

$$108 \leq \mathrm{z}_0 \leq 132\Omega$$

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 nsec/m.

The CAN is an arbitrating protocol which requires a very precise control of various timing events in the exchange of inter-ECU information. It is essential to maintain synchronization between modules at all times. There are several time delays that must be taken into account in the CAN, including delays associated with ECU transmitter and receiver and logic delays. Propagation delays between modules must take into account the time for a signal to make a complete round trip from one module to another.

In addition, the basic CAN bit time requirements are a critical specification. In SAE J-2284-3, the bit time (t_{bit}) must satisfy $1990 \leq t_{\mathrm{bit}} \leq 2010$ nsec. 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-3394-3 also has specific requirements concerning electromagnetic compatibility. The electromagnetic radiation from the CAN as well as 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.

Trip Information Computer

One of the most popular electronic instruments for automobiles is the trip information system. 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,
9. time of day,
10. engine RPM,
11. engine temperature, and
12. average fuel cost per mile.

The trip information computer analyzes fuel flow, vehicle speed, and fuel tank quantities, and then calculates information such as miles to empty tank, average fuel economy, and estimated arrival time. In the present chapter, English units are used because in the USA, these are the preferred units.

Additional functions can be performed, which no doubt will be part of future developments. However, we will discuss a representative system having features that are common to most available systems.

A block diagram of this system is shown in Figure 9.35. Not shown in the block diagram are MUX, DEMUX, and A/D converter components, which are normally part of a computer-based instrument.

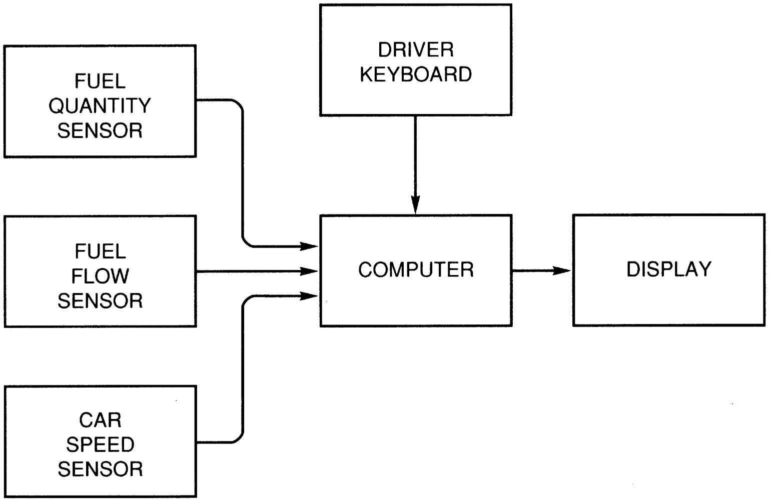


Figure 9.35:
Trip information system block diagram.

This system can be implemented as a set of special functions of the main automotive instrumentation system, or it can be a stand-alone system employing its own computer.

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, and
3. vehicle speed.

Other inputs that are obtained by the computer from other parts of the control system are

1. odometer mileage and
2. time (from clock in the computer).

The driver enters inputs to the system through the keyboard. At the beginning of a trip, the driver initializes the system and enters the total trip distance and fuel price. At any time during the trip, the driver can use the keyboard 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

$$\text{MPG} = V/\dot{F} \tag{34}$$

where V is the speed in miles per hour 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

$$D = \text{MPG x } Q \tag{35}$$

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 Eqn (35) by using short-term time average values of MPG. However, the calculation of Eqn (31) gives a correct estimate of the miles to empty for steady cruise along a highway in which operating conditions are constant.

Still another pair of parameters that can be calculated and displayed by this system are distance to destination, D_d, and estimated time of arrival, ETA. These can be found by computing

$$D_d = D_T - D_P \tag{36}$$

$$\text{ETA} = T_1 + (D_d/V) \tag{37}$$

where D_T is the trip distance (entered by the driver), D_P is the present position (in miles traveled since start), V is the present vehicle speed, and T_1 is the start time.

The computer can calculate the present position 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 present odometer mileage. Alternatively the variables D_T D_P can be obtained automatically (once a destination has been selected by the driver) via a GPS as explained later in this chapter.

The average fuel cost per mile (on any given trip) C can be found by calculating

$$C = (D_P/\text{MPG}) \cdot \text{fuel price per gallon}$$

There are many other useful and interesting operations that can be performed by the variety of available systems. Actually, the number of such functions that can be performed is limited primarily by cost and by the availability of sensors to measure the required variables.

Telematics

Communications to and from an automobile has become routine as a result of both cell phone and satellite technology. In addition, the technology is evolving for area broadcast of road condition information on radio station subcarriers. Technology is also evolving that will permit Internet connections via cell phones, making the car in effect on Internet node. Automobile Internet connectivity opens a limitless range of services for the driver, from on-line navigation help to on-line diagnostic and/or road service for mechanical problems.

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 on-board 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 in which the cell phone user verbally gives the phone number, speaking each digit separately. Included within the cell phone 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 on-line 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 on-line navigation service that transmits data to the car for display on an electronic map.

The telematics technology is presently in its infancy and is certain to grow spectacularly in capability and flexibility, providing the motorist with virtually limitless services. Telematics functionality is probably limited more by imagination and by potential driver distraction than by technology.

The use of satellite communication with automobiles provides many significant applications. These include satellite radio, navigation, and safety applications. Satellite communications are conducted in the microwave portion of the electromagnetic spectrum requiring a special automotive antenna compatible with these frequency bands that is normally mounted on the roof of the vehicle. Satellites are inherently far away from terrestrial receivers so the signal strengths are relatively weak and require receivers of high sensitivity and also incorporate very sophisticated signal processing.

A satellite system such as "OnStar" provides the capability of completely hands-free telephone connection. 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 satellite operator 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.

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° spaced 60° apart in longitude and at a nominal altitude of 11,000 *n* miles above the local surface (i.e. orbital semi-major 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 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:

$$R = c\delta t = \sqrt{(x - x_s)^2 + (y - y_s)^2 + (z - z_s)^2} \tag{38}$$

where c is the speed of propagation of the satellite-transmitted signal, x, y, z are the receiver location coordinates, and x_s, y_s, z_s are the coordinates of the satellite. Both receiver and satellite coordinate systems are in ECEF coordinates (see Chapter 8). If the receiver and satellite clocks were perfectly synchronized then, in principle, 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 (x,y,z) from these measurements. However, it is, in practice, impossible to exactly synchronize these two clocks. The actual measured time difference between satellite i (i = 1, 2, . . . I) clock and receiver clock time yields an estimate of R (denoted R_i) called pseudo-range. Because of the receiver clock uncertainty, at least four measurements are required to estimate position and receiver clock error. The pseudo-range model is given by

$$R_i = \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2} + B \tag{39}$$

where 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 propagation speed of the GPS signal.

The user position can be determined (i.e. the navigation problem can be solved) by measurement of the pseudo-range to at least four satellites by triangulation or trilateration. However, the accuracy of this position solution is influenced by many factors, including the geometry of the satellites in relation to the user and various error sources. These errors include intentional degradation by DOD (significantly reduced in the year 2000) meant to reduce the accuracy to unfriendly users, propagation errors, clock random errors, orbital perturbations, and satellite ephemeris errors. Consequently, many more than four measurements are made to reduce errors.

In addition to estimations of position and clock bias, GPS receives also estimate user velocity and clock drift rate. These additional estimates are required on application since, in this case, the GPS receiver is presumed to be moving and to have changed its position during the time required to obtain the pseudo-range measurements.

The procedure for estimating vehicle position and clock bias is to assume an initial position for the receiver and clock bias (x_0, y_0, z_0, B_0) and to find the vehicle-estimated position at time$t_k(t_k = kT, k = 1, 2, \ldots)$.

The model upon which vehicle position and velocity estimates are based represents vehicle vector position and clock bias as a four-dimensional vector X_k at time t_k. A similar four-dimensional vector represents the initial estimates (X_o). A second represents errors in initial position estimates and is denoted δX_k:

$$\delta X_k = [\delta x,\ \ \delta y,\ \ \delta z,\ \ \delta B]^T$$

where $\delta X_k = X_k - X_0$:

$$X_k = [x_k,\ \ y_k,\ \ z_k,\ \ B_k]^T$$

$$X_0 = [x_o,\ \ y_o,\ \ z_o,\ \ B_o]^T$$

Finally, the rate of change in vector position (i.e. $\dot{X}$) is given by a third four-dimensional vector:

$$\dot{X} = [\dot{x},\ \ \dot{y},\ \ \dot{z},\ \ \dot{B}]^T$$

In all of these four-dimensional vectors, the superscript T represents transpose of the row vector.

The model for four-dimensional vector estimates is given by

$$X_k = \begin{bmatrix} x_k \\ y_k \\ z_z \\ B_k \end{bmatrix} = \begin{bmatrix} x_o \\ y_o \\ z_o \\ B_o \end{bmatrix} + \begin{bmatrix} \delta x \\ \delta y \\ \delta z \\ \delta B \end{bmatrix} + (k-1)T \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{B} \end{bmatrix} \tag{40}$$

where $\delta x, \delta y, \delta z, \delta B$ are errors in the initial estimates, $(\dot{x}, \dot{y}, \dot{z})$ are the estimated user velocity vector components, and $\dot{B}$ is the user clock drift rate.

The GPS navigation problem can now be formulated as a state estimation problem in which the state vector, $\boldsymbol{X}$, is eight dimensional and is given by

$$\boldsymbol{X} = [\delta x, \delta y, \delta z, \delta B, \dot{x}, \dot{y}, \dot{z}, \dot{B}]^T$$

The standard method for solving this problem is to linearize the pseudo-range equation:

$$R_i = R_{io} + \alpha_{i1}\delta x + \alpha_{i2}\delta y + \alpha_{i3}\delta z + \delta B, \tag{41}$$

where R_{i0} is the initial pseudo-range estimate from satellite i ($i = 1,2 \ldots I$):

$$= \sqrt{(x_0 - x_i)^2 + (y_0 - y_i)^2 + (z_0 - z_i)^2} + B_0 \tag{42}$$

and where

$$\alpha_{i1} = \left.\frac{\partial R_i}{\partial x}\right|_{R_{io}} = \frac{x - x_i}{R_{io} - B_o}$$

$$\alpha_{i2} = \left.\frac{\partial R_i}{\partial y}\right|_{R_{io}} = \frac{y - y_i}{R_{io} - B_o}$$

$$\alpha_{i3} = \left.\frac{\partial R_i}{\partial z}\right|_{R_{io}} = \frac{z - z_i}{R_{io} - B_o} \tag{43}$$

The parameters $\alpha_{i1}, \alpha_{i2}, \alpha_{i3}$ are the direction cosines of the angles between the line of sight from the user to satellite i and the coordinate axes. The linearized pseudo-range equation can be written in terms of δR_i where

$$\delta R_i = R_i - R_{i0} = \alpha_{i1}\delta x + \alpha_{i2}\delta y + \alpha_{i3}\delta z + \delta B \tag{44}$$

Figure 9.36 is a simplified illustration of the geometry for a vehicle moving at a constant velocity beginning at true position, x_T, y_T. In figure 9.36, S1 and S2 denote satellites i=1 and i=2 respectively and x_0, y_0 are the initial position estimates associated with pseudo ranges R_{01}, R_{02}

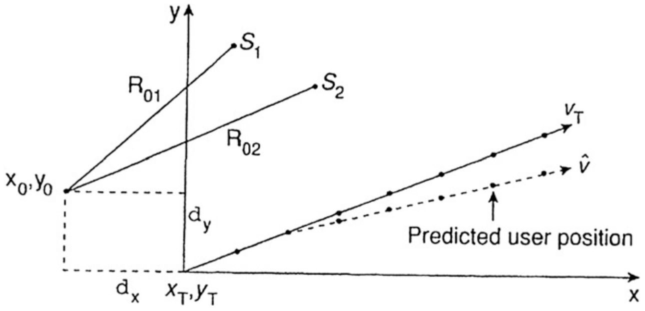


Figure 9.36:
GPS navigation geometry (simplified).

For an understanding of the basic GPS in vehicles it is convenient to simplify the navigation problem by assuming that the vehicle is moving at constant speed. The measurement model for vehicle traveling at a constant speed from an initial estimated position (x_0, y_0, z_0) for four satellites is given by

$$\delta R = HX + e,$$

where

$$\delta R = \begin{bmatrix} [\delta R_{11} \quad \delta R_{21} \quad \delta R_{31} \quad \delta R_{41}]^T \\ \vdots \\ [\delta R_{n1} \quad \delta R_{n2} \quad \delta R_{n3} \quad \delta R_{n4}]^T \end{bmatrix} \tag{45}$$

In this expression n is the number of measurements made to each of the four satellites, and the matrix H is given by

$$H = \begin{bmatrix} H_1 \\ H_2 \\ \vdots \\ H_{n-1} \end{bmatrix},$$

where

$$H_k = \begin{bmatrix} \alpha_{11} & \alpha_{12} & \alpha_{13} & 1 & kT\alpha_{11} & kT\alpha_{12} & kT\alpha_{13} & kT \\ \alpha_{21} & \alpha_{22} & \alpha_{23} & 1 & kT\alpha_{21} & kT\alpha_{22} & kT\alpha_{23} & kT \\ \alpha_{31} & \alpha_{32} & \alpha_{33} & 1 & kT\alpha_{31} & kT\alpha_{32} & kT\alpha_{33} & kT \\ \alpha_{41} & \alpha_{42} & \alpha_{43} & 1 & kT\alpha_{41} & kT\alpha_{42} & kT\alpha_{43} & kT \end{bmatrix} \in R^{4n\times 8} \tag{46}$$
$$K = 1, \ldots, n$$

The state vector X is given by

$$X = [\,\delta x \quad \delta y \quad \delta z \quad \delta B \quad \dot{x} \quad \dot{y} \quad \dot{z} \quad \dot{B}\,]^T$$

and both δR and e are $4n \times 1$ dimensional error vectors.

The solution to this problem for finding X can be obtained in a "batch" mode, in which data are collected for n samples and found as the ordinary "least-squares" (OLS) solution, or the solution can be found recursively. The OLS solution is found using the pseudo-inverse of H and is given by

$$X = (H^T H)^{-1} H^T \delta R \tag{47}$$

Once a solution has been reached for X, the position of the vehicle for all time x_k, y_k, z_k is given by

$$\begin{bmatrix} x_k \\ y_k \\ z_k \\ B_k \end{bmatrix} = \begin{bmatrix} x_o \\ y_o \\ z_o \\ B_o \end{bmatrix} + \begin{bmatrix} \delta_x \\ \delta_y \\ \delta_z \\ \delta_B \end{bmatrix} + (k-1)T \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \\ \dot{B} \end{bmatrix} \tag{48}$$

for a vehicle moving at a constant velocity vector $\mathrm{v} = [\dot{x}, \dot{y}, \dot{z}]^T$. For maneuvering vehicles the solution involves time-varying estimates of v.

The recursive solution to the GPS navigation problem is a Kalman filter which continuously estimates X. The *k*th estimate of X (i.e. X_k) is based upon K previous measurements and estimates and is of the form

$$X_k = X_{k-1} + K_k(\delta R_k - H_k X_{k-1}), \tag{49}$$

where K_k is the Kalman filter gain and δR_k, H_k are samples of *δR and H as* defined above at times $\mathrm{t_k}$. The Kalman filter gain is computed in a multistep process that is derived from known statistics of process and measurement noise, the basic dynamics of the vehicle motion, and the relationship between the state and the measurements in the zero noise ideal case. The details of these relations are beyond the scope of this book but can be found in any of the many excellent books which have been written about optimal filtering as well as those concerning GPS theory.

The GPS System Structure

The structure of the GPS navigation system consists of three major segments: (1) the space segment (the satellites), (2) the control segment, and (3) 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 this ephemeris data are required owing to orbital perturbations and changes due to lunar−solar perturbations, air drag, asphericityof Earth's gravitational potential, and magnetic, static-electric forces in orbit.

The control segment configuration is depicted in Figure 9.37. The monitor stations receive the GPS signals (same as the 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 pseudo-range values form the satellites as they come into view. These measurements are used to determine ephemeris and clock errors. In addition, these stations monitor local meteorological data that are useful for correcting for tropospheric delays. The data and corrections obtained by these monitor stations are sent to the master control.

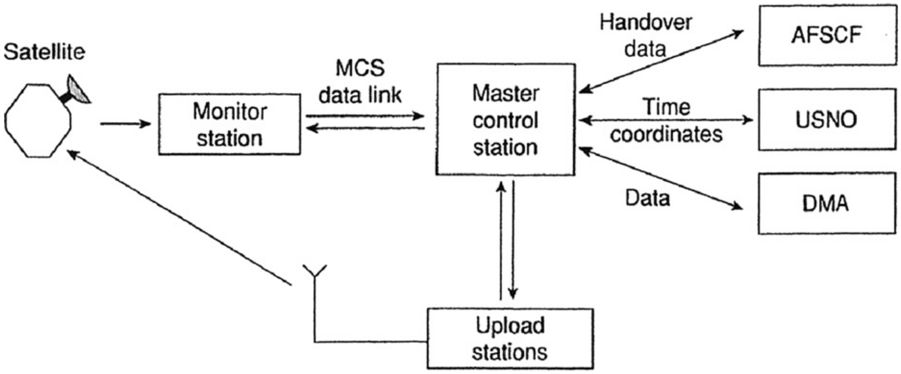


Figure 9.37:
Control segment configuration.

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 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

$$n = \frac{c_0}{v_\phi} \tag{50}$$

where v_ϕ is the phase velocity.

At any carrier frequency, f, the index of refraction is given by

$$n = \sqrt{1 - \left(\frac{f_c}{f}\right)^2}, \tag{52}$$

where f_c is the plasma frequency:

$$f_c = \frac{1}{2\pi}\sqrt{\frac{N_c e^2}{m\varepsilon_0}} \cong 9\sqrt{N_e} \tag{53}$$

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:

$$n \cong 1 + \frac{K_1}{T}\left(p + \frac{K_2 p_w}{T}\right) \tag{53}$$

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

$$\Delta L = \int_0^R (n-1)ds, \tag{54}$$

where $R =$ distance to the satellite

$\cong$ pseudo-range

$s =$ 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:

$$\Delta L = \int_{h_0}^H \frac{n-1}{\sin\phi_0} dh, \tag{55}$$

where H is the satellite elevation above the Earth. If the atmosphere is assumed to be exponential, then

$$n - 1 \cong (n_0 - 1)e^{-bh}, \tag{56}$$

where (for a standard day)

$$n_0 \cong 1.00032$$
$$b \cong 0.000145/\text{m}$$

Typical values are

$$\begin{aligned} \Delta L &\cong 2.2\text{m} \quad \text{for} \quad \phi_0 = 90^\circ \\ &\cong 25\text{m} \quad \text{for} \quad \phi_0 = 5^\circ \end{aligned}$$

The influence of satellite geometry is given via the GDOP. The GDOP can be computed from the matrix of direction cosines to the satellite, H_i, where the *i*th row of H is

$$H_i = [\alpha_{i1}, \alpha_{i2}, \alpha_{i3}, -1] \tag{57}$$

$$\text{GDOP} = \sqrt{\text{trace}\left\{ \left[H_i^T H_i \right]^{-1} \right\}} \tag{58}$$

With a given value for GDOP, the rms error is given by

$$\sigma = \text{GDOP}\sigma_0 \tag{59}$$

where σ_0 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.

Automotive Diagnostics

In certain automobile models, the instrumentation computer can perform the important function of diagnosis of the electronic engine control system as well as other subsystems. This diagnosis takes place at several different levels. One level is used during manufacturing to test the system, and another level is used by mechanics or interested car owners to diagnose engine control system problems. Some levels operate continuously and others are available only on request from an external device that is connected to the car data link for diagnostic purposes by a technician. This application is explained in the next chapter.

CHAPTER 10

Diagnostics and Occupant Protection

Chapter Outline

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) as well as 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.

Understanding Automotive Electronics. http://dx.doi.org/10.1016/B978-0-08-097097-4.00010-2

Then, when the car is delivered to a repair station (e.g. at the dealer), the technician can retrieve the diagnostic information electronically.

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 pre-emission 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 as well as 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 “On-Board 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. A serial data link transmits digital data in a binary time sequence along a pair of wires. Before discussing the details of onboard and off-board diagnosis, it is perhaps worthwhile to discuss briefly automotive digital communications.

It was shown in the previous chapter that the various electronic subsystems (ECUs) in a contemporary vehicle are connected together via the CAN network. For example, in Figure 9.34 one of the connections to the 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.

Figure 10.1 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 Figure 10.1

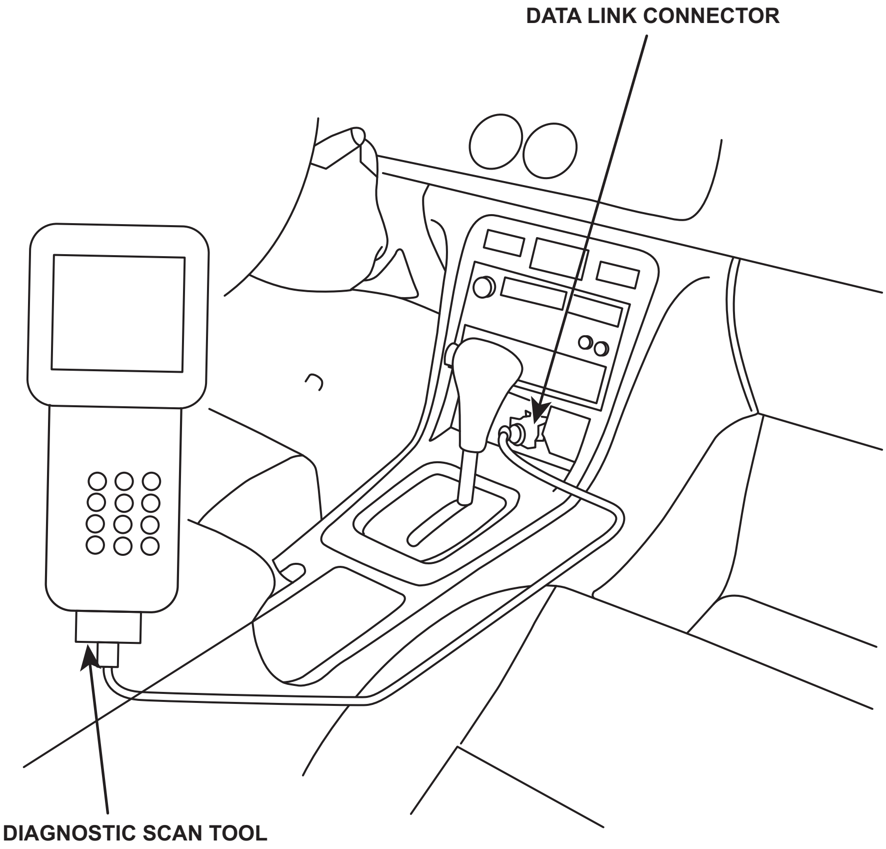


Figure 10.1:
Illustration of diagnostic scan tool connection to vehicle.

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 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) which 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 as well as recommended procedures for diagnosing specific problems from the stored fault codes as well as 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 systems 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 e.g. 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 which can adversely affect exhaust emissions during periods of degraded performance. For example, the heated exhaust gas oxygen concentration sensor (HEGO; see Chapter 6) which 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 (see Figure 9.34) 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 as well as 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 7 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 7). 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 M_a into the intake system is given by:

$$\dot{M}_a = \dot{V}_a \delta_i \tag{1}$$

where $\dot{V}_a$ = volume flow rate

δ_i = intake air density.

Using tables of volumetric efficiency (η_V) for the engine as a function of throttle angle and RPM, the volume flow rate $\dot{V}_a$ is given by:

$$\dot{V}_a = \frac{D_e n \eta_V(\theta_t, R)}{2} \tag{2}$$

where D_e = engine displacement

$n = R/60$

R = RPM

θ_t = throttle angle.

The intake air density is given by

$$\delta_i = \frac{\delta_0 p_i T_0}{p_0 T_i} \tag{3}$$

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 7) that power train control uses numerous angular speed sensors. As shown in Chapter 6 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 $t_k(k = 1, 2, \cdots K)$ 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

$$\delta\, t_k = t_k - t_{k-1}, \quad k = 1,\ 2 \cdots K$$

If a sensor failure of the types described above has occurred then there will be at least one differential time which is significantly different from the others in a full sequence during an engine cycle. The measurements of angular speed for this particular time differential is termed an outlyer. Several standard algorithms exist for detecting and removing these erroneous measurements from the data collected during each cycle of operations as well as setting a fault code.

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 electrical 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 9). 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 major limitation—they cannot detect intermittent failures reliably. For the system to detect and isolate a failure, the failure must 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.

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 10.1 as an example of a subset of fault codes. Manufacturer-specific codes can have a format P1xxx as illustrated in Table 10.1.

Table 10.1: 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)

Table 10.1: Fault code sample.—cont'd

FIL Indication	Affected Component
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 of* $\begin{array}{l}\text{P0301}\\\text{P0302}\\\text{P0303}\\\text{P0304}\\\text{P0305}\\\text{P0306}\end{array}\begin{pmatrix}71\\72\\73\\74\\75\\76\end{pmatrix}$	*Misfire of Multiple Cylinders* Detected
$\begin{array}{l}\text{P0301}\\\text{P0302}\\\text{P0303}\\\text{P0304}\\\text{P0305}\\\text{P0306}\end{array}\begin{pmatrix}71\\72\\73\\74\\75\\76\end{pmatrix}$	Misfire Detected —Cylinder 1 —Cylinder 2 —Cylinder 3 —Cylinder 4 —Cylinder 5 —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

Continued

Table 10.1: Fault code sample.—cont'd

FIL Indication	Affected Component
P0500 (17)	Vehicle Speed Sensor Circuit Malfunction
P0505 (14)	Idle Control System Malfunction
P0715 P0720 P0725 P0730 (70) P0740 P0753 P0758	Automatic Transaxle
P1107 (13)	Barometric Pressure Circuit Low Input
P1108 (13)	Barometric Pressure Circuit High Input
P1297 (20)	Electrical Load Detector Circuit Low Input
P1298 (20)	Electrical Load Detector Circuit High Input
P1361 (8)	Top dead Center Sensor Intermittent Interruption
P1362 (8)	Top Dead Center Sensor No Signal EGR - exhaust gas recirculation
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

Table 10.1: Fault code sample.—cont'd

FIL Indication	Affected Component
P1508 (14)	Idle Air Control Valve Circuit Failure
P1607 (—)	Engine Control Module Internal Circuit Failure A

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. With the ignition switch on and this 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 graphical/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 7 for a review of the role played by this important sensor in closed-loop fuel control.

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 10.1). 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 Figure 10.2.

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

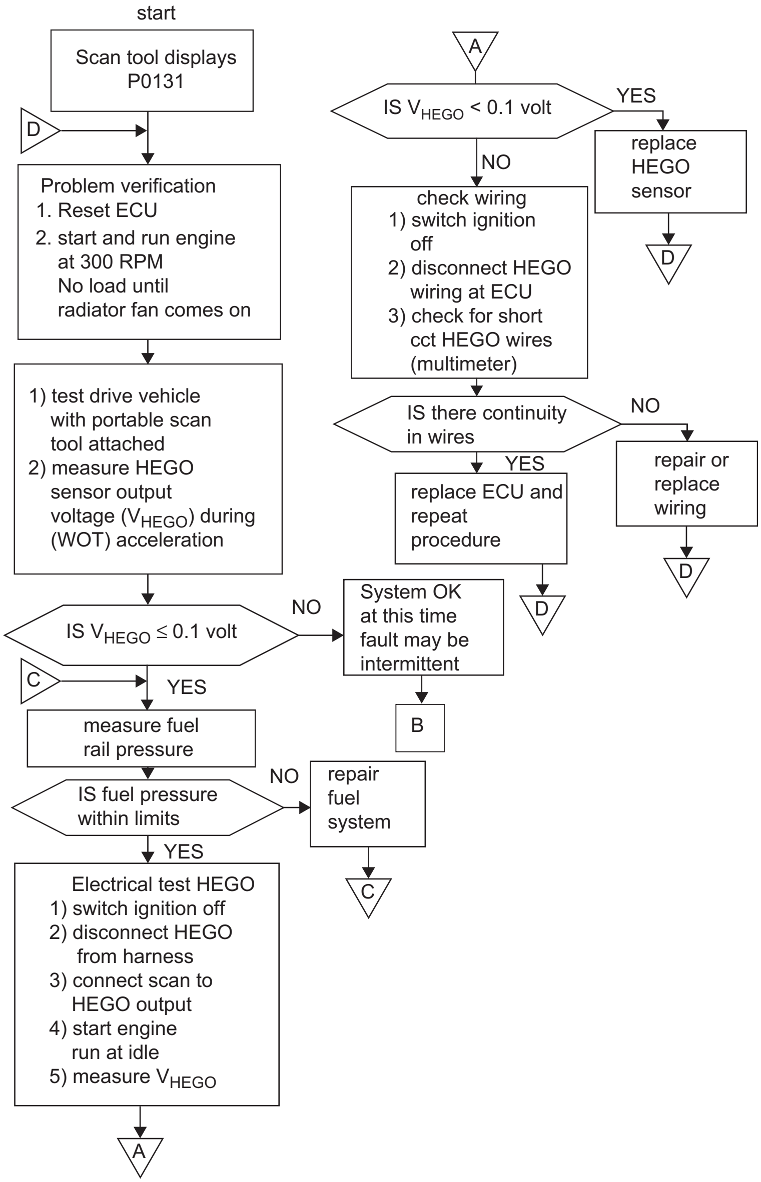

start
Scan tool displays P0131
D
Problem verification
1. Reset ECU
2. start and run engine at 300 RPM No load until radiator fan comes on
1) test drive vehicle with portable scan tool attached
2) measure HEGO sensor output voltage (V_{HEGO}) during (WOT) acceleration
IS $V_{HEGO} \leq 0.1$ volt
NO
System OK at this time fault may be intermittent
B
C
YES
measure fuel rail pressure
IS fuel pressure within limits
NO
repair fuel system
C
YES
Electrical test HEGO
1) switch ignition off
2) disconnect HEGO from harness
3) connect scan to HEGO output
4) start engine run at idle
5) measure V_{HEGO}
A
A
IS $V_{HEGO} < 0.1$ volt
YES
replace HEGO sensor
D
NO
check wiring
1) switch ignition off
2) disconnect HEGO wiring at ECU
3) check for short cct HEGO wires (multimeter)
IS there continuity in wires
NO
repair or replace wiring
D
YES
replace ECU and repeat procedure
D

Figure 10.2:
Flowchart for diagnosing fault in primary HEGO.

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 Figure 10.2) 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 $V_{\mathrm{HEGO}} \leq 0.1$ 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: for a HEGO, there is also a pair of connectors for the heating element; see Chapter 6). 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 $V_{\mathrm{HEGO}} \leq 0.1$ 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.

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 approximately 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 approximately 0.5 V, and that the electronic control system will not go into closed-loop operation (see Chapters 5 and 7) 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 multi meter). The desired voltage is displayed on the scan tool.

Using this voltage, the service technician follows a procedure outlined in Figures 10.3 and 10.4. If the voltage is less than 0.37 V or greater than 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 multi meter). If this voltage is less than 0.05 V, the control unit is functioning correctly and the HEGO sensor must be investigated for defects. If the indicated sensor voltage is greater than 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 Figure 10.4. 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 7, 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 less than 0.3 V 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

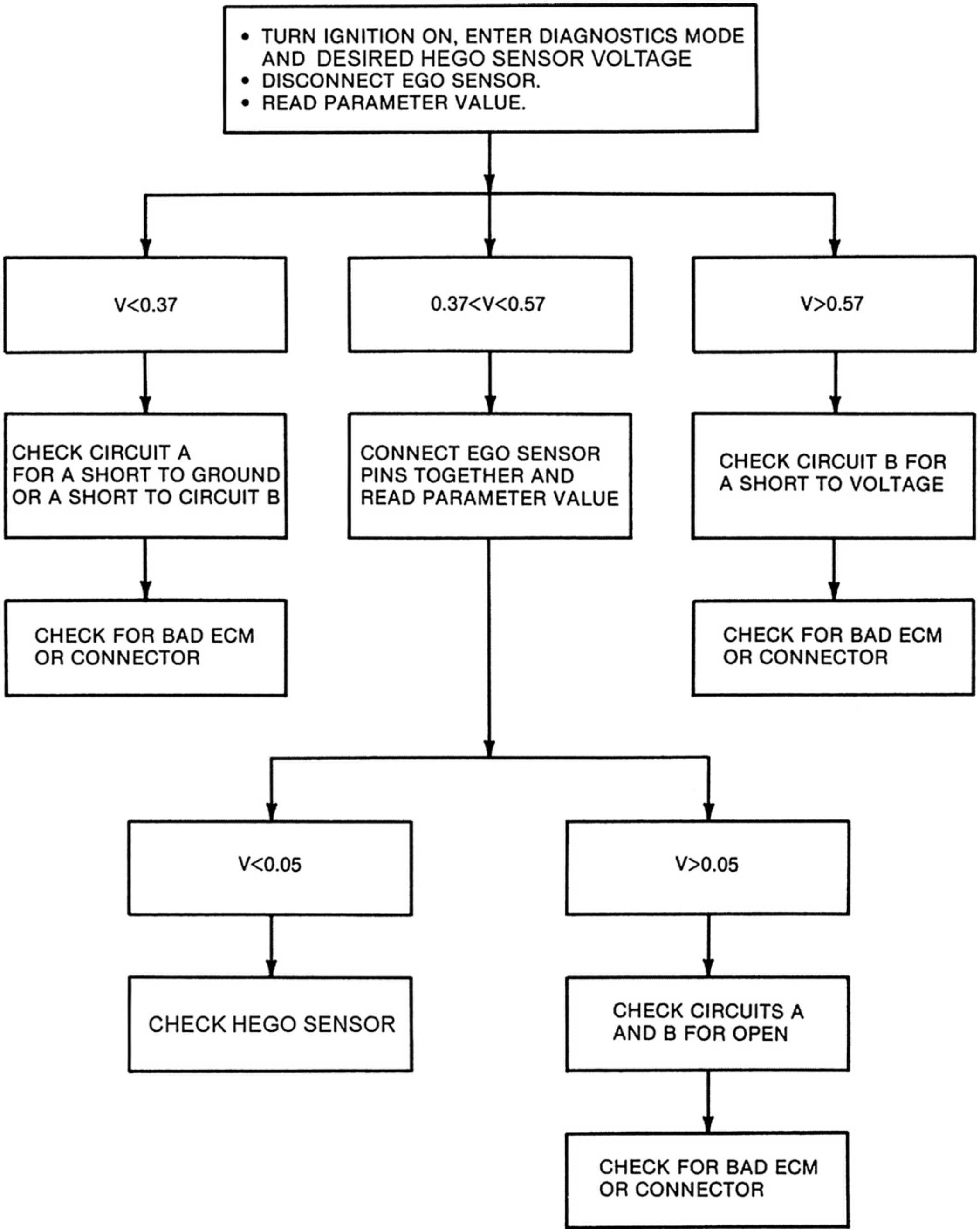

• TURN IGNITION ON, ENTER DIAGNOSTICS MODE AND DESIRED HEGO SENSOR VOLTAGE
• DISCONNECT EGO SENSOR.
• READ PARAMETER VALUE.
V<0.37
0.37<V<0.57
V>0.57
CHECK CIRCUIT A FOR A SHORT TO GROUND OR A SHORT TO CIRCUIT B
CONNECT EGO SENSOR PINS TOGETHER AND READ PARAMETER VALUE
CHECK CIRCUIT B FOR A SHORT TO VOLTAGE
CHECK FOR BAD ECM OR CONNECTOR
CHECK FOR BAD ECM OR CONNECTOR
V<0.05
V>0.05
CHECK HEGO SENSOR
CHECK CIRCUITS A AND B FOR OPEN
CHECK FOR BAD ECM OR CONNECTOR

Figure 10.3:
Flowchart for diagnosis of HEGO sensor output voltage problems.

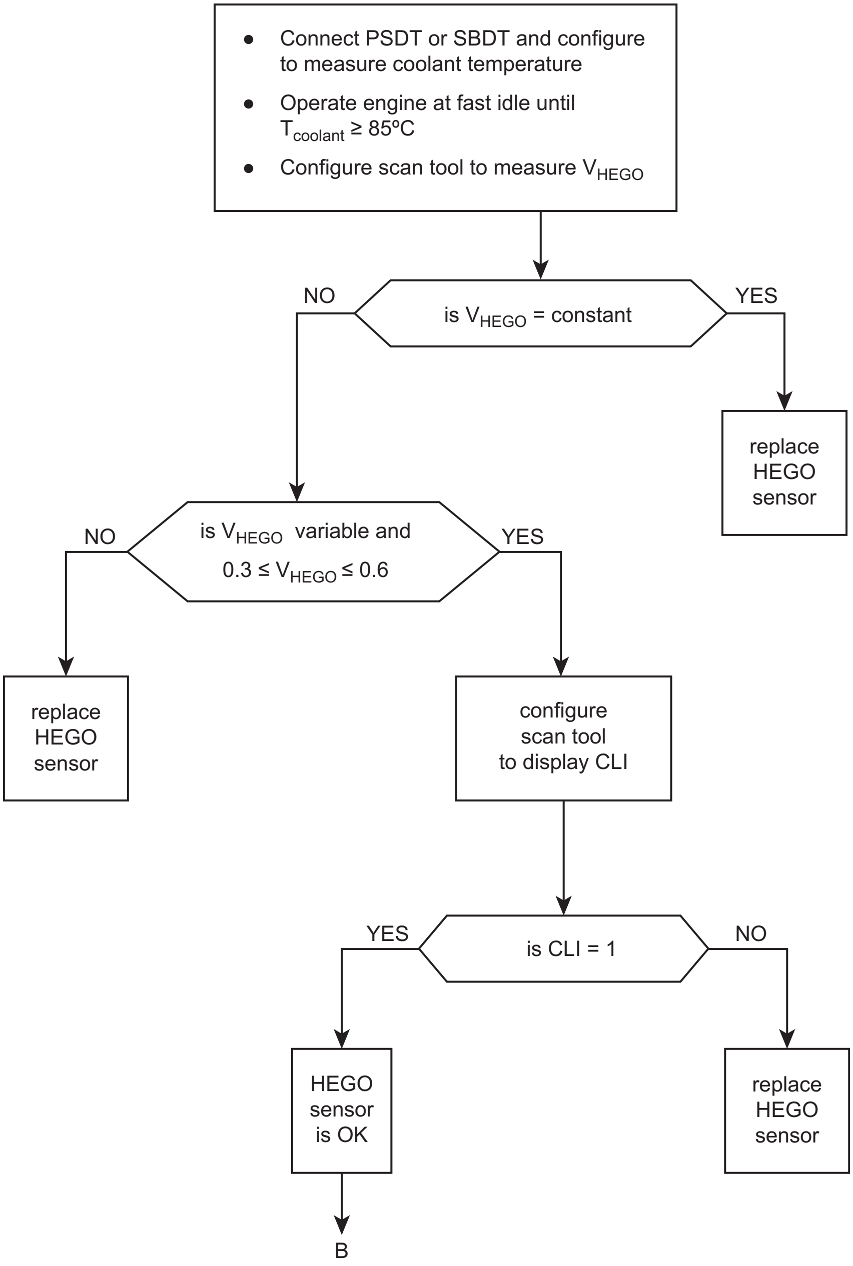

Connect PSDT or SBDT and configure to measure coolant temperature
Operate engine at fast idle until $T_{coolant} \geq 85°C$
Configure scan tool to measure V_{HEGO}
is V_{HEGO} = constant
NO
YES
replace HEGO sensor
is V_{HEGO} variable and $0.3 \leq V_{HEGO} \leq 0.6$
NO
YES
replace HEGO sensor
configure scan tool to display CLI
is CLI = 1
YES
NO
HEGO sensor is OK
replace HEGO sensor
B

Figure 10.4:
Flowchart for HEGO sensor proper switching sensor test.

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 which provides an important input to the electronic engine control as explained in Chapters 5, 6 and 7. The onboard diagnostic can detect out-of-limits values for this sensor and display fault codes, e.g. P0122 for voltage below a lower limit or P0123 for voltage above a high limit. However, 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 7). 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. 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 which 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 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 8) 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

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

Figure 10.5:
Switch test sequence.

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 Figure 10.5. 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 Figure 10.5.

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 Figure 10.6) 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. Figure 10.7 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.

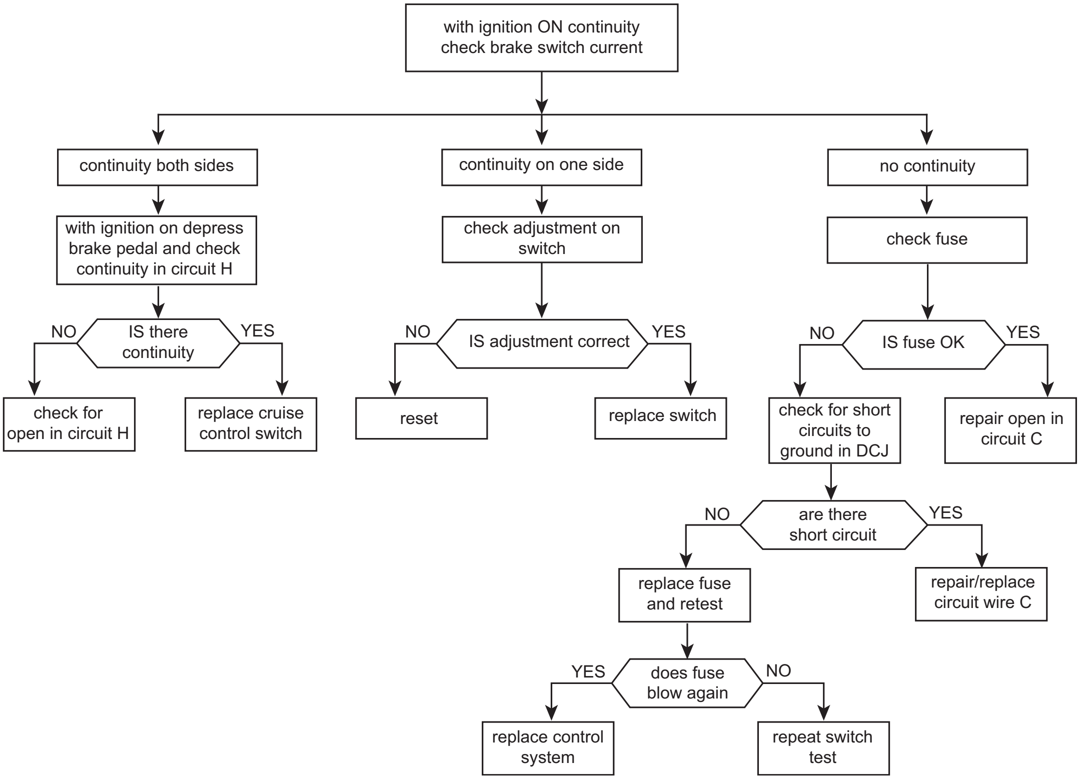


Figure 10.6:
Cruise control brake circuit.

(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.
(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.

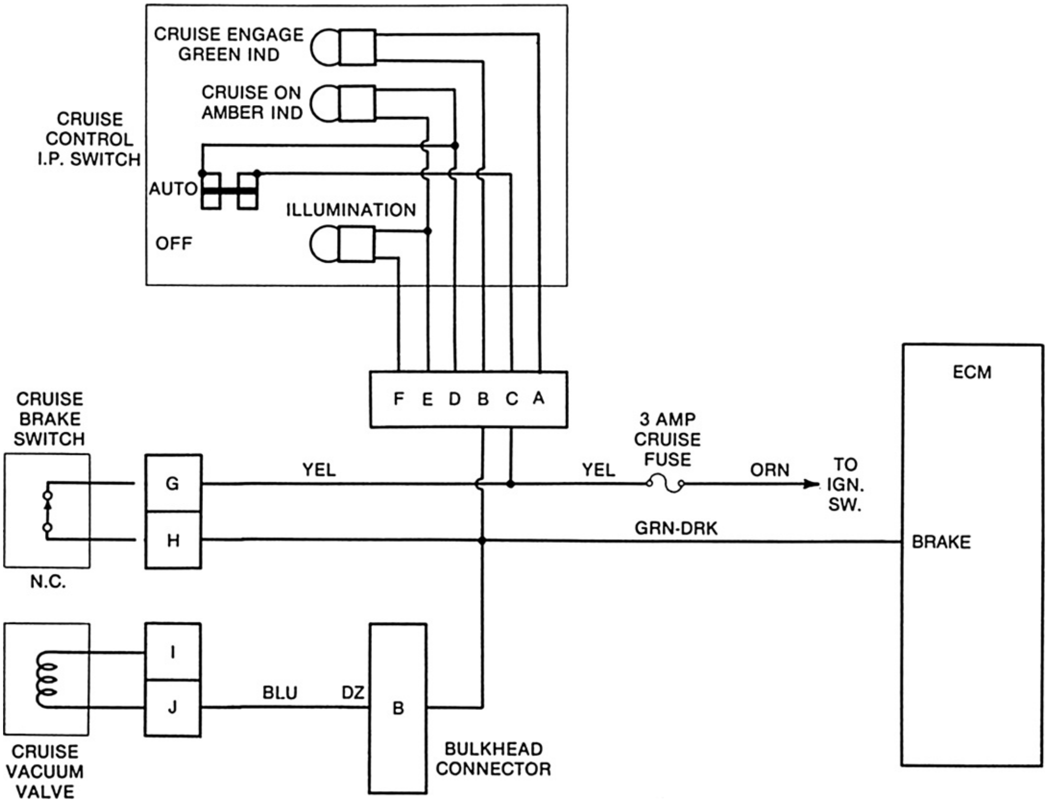


Figure 10.7:
Cruise control brake circuit.

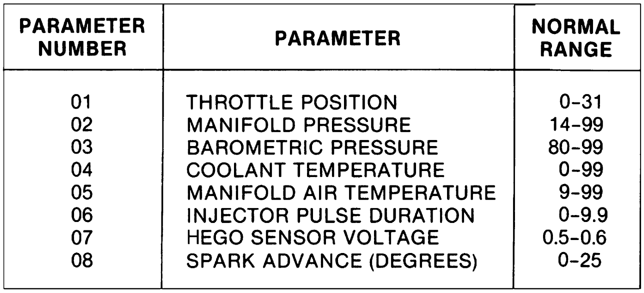
(9) With 79 displayed, depress and release the reset button on the TIC panel. If the reset button is working normally, the code will advance to 80.

(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 as well as the normal range for the parameter.

Figure 10.8 shows the parameter values in sequence for an exemplary vehicle. Parameter 01 is the angular deflection of the throttle in degrees from idle position.

Parameter 02 is the manifold absolute pressure in kilopascals. The range for this parameter is 14-99, with 14 representing about the maximum manifold vacuum. Parameter 03 is the absolute atmospheric pressure in kPa. Normal atmospheric pressure is roughly 90-100 kPa at



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

Figure 10.8:
Chart of exemplary engine parameters with normal ranges.

sea level. Parameter 04 is the coolant temperature and Parameter 05 is the intake manifold temperature.

Parameter 06 is the duration of the fuel injector pulse in milliseconds. Refer to Chapters 5, 6, and 7 for an explanation of the injector pulse widths and the influence of these pulse widths on 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 Vas 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. This value should agree with that obtained using a SBDT configured in the engine analyzer mode. Although it is not shown in Figure 10.8, parameter 09 is the number of ignition cycles that have occurred since a trouble code was set in memory. If 20 such cycles have occurred without a fault, this counter is set to zero and all trouble codes are cleared.

Parameter 10 (not shown in Figure 10.8) 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 6 and 7. Parameter 11 is the battery voltage.

Onboard Diagnosis (OBD II)

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 5 and 7, consist of fuel and ignition control for the three-way catalytic converter, as well as 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).

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.

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 which has been tested under laboratory conditions as well as 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:

$$T_n\left[\theta_e(t)\right] = T_i\left[\theta_e(t)\right] + T_R\left[\theta_e(t)\right] + T_{Fp}\left[\theta_e(t)\right] - T_l\left[\theta_e(t)\right] \qquad (4)$$

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 5). 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:

$$T_m(\theta_{\mathrm{e}}) = A_{\mathrm{p}}R_{\mathrm{c}}(p_{\mathrm{c}} - p_{\mathrm{o}})f_m(\theta_{\mathrm{e}}) \tag{5}$$

where $$f_m(\theta_{\mathrm{e}}) = \sin(\theta_{\mathrm{e}} - \theta_m)\left[1 + \frac{(R_{\mathrm{c}}/L_{\mathrm{c}})\cos\,(\theta - \theta_m)}{\sqrt{1 - (R_{\mathrm{c}}/L_{\mathrm{c}})^2\sin^2(\theta_{\mathrm{e}} - \theta_{\mathrm{w}})}}\right]$$

and where $L_{\mathrm{c}} =$ connecting rod length

$R_{\mathrm{c}} =$ crankshaft throw

$p_{\mathrm{c}} =$ cylinder pressure

$p_{\mathrm{o}} =$ atmospheric pressure

where $\theta_m = \theta_{\mathrm{e}}$ 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

$$T_{\mathrm{i}}(\theta_{\mathrm{e}}) = \sum_{m=1}^{M} T_m(\theta_{\mathrm{e}})$$

The reciprocating torque associated with the mth cylinder are given by:

$$T_{\mathrm{R}m}(\theta_{\mathrm{e}}) \cong M_{\mathrm{eq}} R_{\mathrm{c}}^2 f_{\mathrm{T}}(\theta_{\mathrm{e}}) \left[f_{\mathrm{T}}(\theta_{\mathrm{e}}) \dot{\omega}_{\mathrm{e}} + f_{\mathrm{B}}(\theta) \omega_{\mathrm{e}}^2 \right] \tag{6}$$

where $\omega_{\mathrm{e}} = \dfrac{\pi \,\mathrm{RPM}}{30}$

$$f_{\mathrm{T}}(\theta_{\mathrm{e}}) = \sin(\theta_{\mathrm{e}} - \theta_m) + \frac{(R_{\mathrm{c}}/L_{\mathrm{c}}) \sin[2(\theta_{\mathrm{e}} - \theta_m)]}{2\sqrt{1 - (R_{\mathrm{c}}/L_{\mathrm{c}})^2 \sin^2(\theta_{\mathrm{e}} - \theta_m)}}$$

$$\begin{aligned} f_{\mathrm{B}}(\theta_{\mathrm{e}}) = {} & \frac{R_{\mathrm{c}}}{L_{\mathrm{c}}} \left\{ \frac{\cos[2(\theta_{\mathrm{e}} - \theta_m)]}{\sqrt{1 - (R_{\mathrm{c}}/L_{\mathrm{c}})^2 \sin^2(\theta_{\mathrm{e}} - \theta_m)}} \right\} + \left(\frac{R_{\mathrm{c}}}{L_{\mathrm{c}}} \right)^3 \frac{\sin^2(\theta_{\mathrm{e}} - \theta_m)}{4\sqrt{[(1 - R_{\mathrm{c}}/L_{\mathrm{c}})^2 \sin^2(\theta_{\mathrm{e}} - \theta_m)]^3}} \\ & + \cos(\theta_{\mathrm{e}} - \theta_m) \end{aligned} \tag{7}$$

where: M_{eq} = sum of the mass of the piston, wrist pin and 1/3 of the connecting rod.

The combined reciprocating torque T_{R} is given by:

$$T_{\mathrm{R}}(\theta_{\mathrm{e}}) = \sum_{m=1}^{M} T_{\mathrm{R}m}(\theta_{\mathrm{e}}) \tag{8}$$

For the purposes of modeling the engine for misfire detection, it is possible to approximate $T_{\mathrm{Fp}}(\theta_{\mathrm{e}})$ with a linearized model as given below:

$$T_{\mathrm{Fp}}(\theta_{\mathrm{e}}) \cong R_{\mathrm{e}} \omega_{\mathrm{e}}$$

where R_{e} = linearized friction coefficient.

The net torque applied to the crankshaft is the sum of the components:

$$T_{\mathrm{n}}(\theta_{\mathrm{e}}) = T_{\mathrm{i}}(\theta_{\mathrm{e}}) + T_{\mathrm{R}}(\theta_{\mathrm{e}}) \tag{9}$$

The present method of misfire detection in an engine is based upon a metric which represents the nonuniformity in torque generation (i.e. in δT_k). If every cylinder produced exactly the same torque during a given engine cycle, the fluctuations in δT_k 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 as well as 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 δT_k 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 for a given engine cycle for an M cylinder engine, which is denoted $\bar{n}$ and is given by:

$$\bar{n} = \delta\bar{T} - \delta T_{\mathrm{av}}\bar{u} \tag{10}$$

where $\overline{T} = [T_1, T_1', \cdots T_M, T_M']^T \in \; R^{2M}$

where $T_m = T_m(\theta_{\mathrm{e}}^m)$

$= \text{relative maximum of } T_m$

$\theta_{\mathrm{e}}^m = \text{crankshaft angle at which } \; T_m \text{ occurs}$

and where $T_m' = T_m(\theta_{\mathrm{e}m})$

$= \text{relative minimum of } T_m$

$\theta_{\mathrm{e}m} = \theta_{\mathrm{e}} \text{ at which } \; T_m' \text{ occurs.}$

That is, the extremal values for δT are characterized by

$$\left.\begin{aligned} &\frac{\mathrm{d}T_{\mathrm{n}}}{\mathrm{d}\theta_{\mathrm{e}}}|_{\theta_{\mathrm{e}}^m} = 0 \\ &\frac{\mathrm{d}^2T_{\mathrm{n}}}{\mathrm{d}\theta_{\mathrm{e}}^2}|_{\theta_{\mathrm{e}}^m} < 0 \\ &\frac{\mathrm{d}T_{\mathrm{n}}}{\mathrm{d}\theta_{\mathrm{e}}}|_{\theta_{em}} = 0 \\ &\frac{\mathrm{d}^2T_{\mathrm{n}}}{\mathrm{d}\theta_{\mathrm{e}}^2}|_{\theta_{em}} > 0 \end{aligned}\right. \quad , \quad m = 1, 2 \ldots M \tag{11}$$

where

$$T_{\mathrm{av}} = \frac{1}{2M}\sum_{m=1}^{M}\left[T_m + T_m'\right] \quad \text{average of extrema per cycle} \tag{12}$$

$$\delta\overline{T} = \left[\delta T_1 \delta T_1' \cdots \delta T_m \delta T_m' \cdots \delta T_M \delta T_M'\right]^T$$

$$\delta T_m = T_m - T_{av}$$

$$\delta T_m' = T_m' - T_{\mathrm{av}}$$

$$\delta T_{av} = \frac{1}{2M} \sum_{m=1}^{M}\left[\delta T_m - \delta T_m'\right] \quad (\textit{average torque deviation})$$

and where $\overline{u}$ is a $2M$ dimensional vector given by:

$$\overline{u} = [1, \ -1, \ 1, \ -1 \ldots 1, \ -1]^T \qquad \in R^{2M}$$

Figure 10.9 illustrates (qualitatively) the nonuniformity vector samples for a hypothetical torque waveform. Note that for perfectly uniform torque waveform $\delta T'_m = \delta T_m$ with the result that n is a zm dimensional vector with all elements zero.

The presence of a misfire can readily be detected by a scalar n derived from a norm of the vector $\overline{n}$

$$\begin{aligned} n &= \| \, \overline{n} \, \|_1 \qquad\qquad \ell_1 \text{ norm} \\ &\qquad\quad \text{or} \\ &= \|\overline{n} \, \|_2 \qquad\qquad \ell_2 \text{ norm} \end{aligned} \tag{13}$$

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, H_2 where

$H_0 \rightarrow$ normal engine operation

$H_1 \rightarrow$ misfire in a single cylinder within an engine cycle

$H_2 \rightarrow$ 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 which obtained the ℓ_1

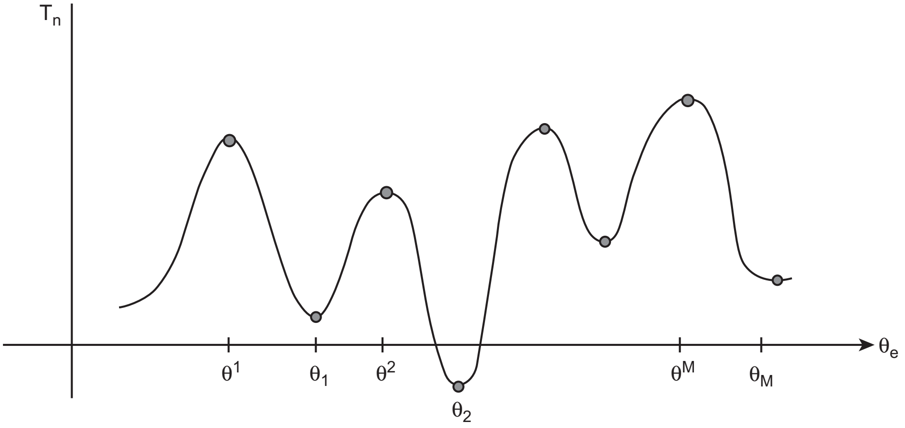


Figure 10.9:
Illustrative torque waveform and its extrema.

norm of n (n_1) for each of several thousand engine cycles. Figure 10.10 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, N_{av} (H_1) be the mean value for n_1 under H_1. A threshold n_t is chosen such that

$$n_t = [N_{av}(H_0) + N_{av}(H_1)]/2 \tag{14}$$

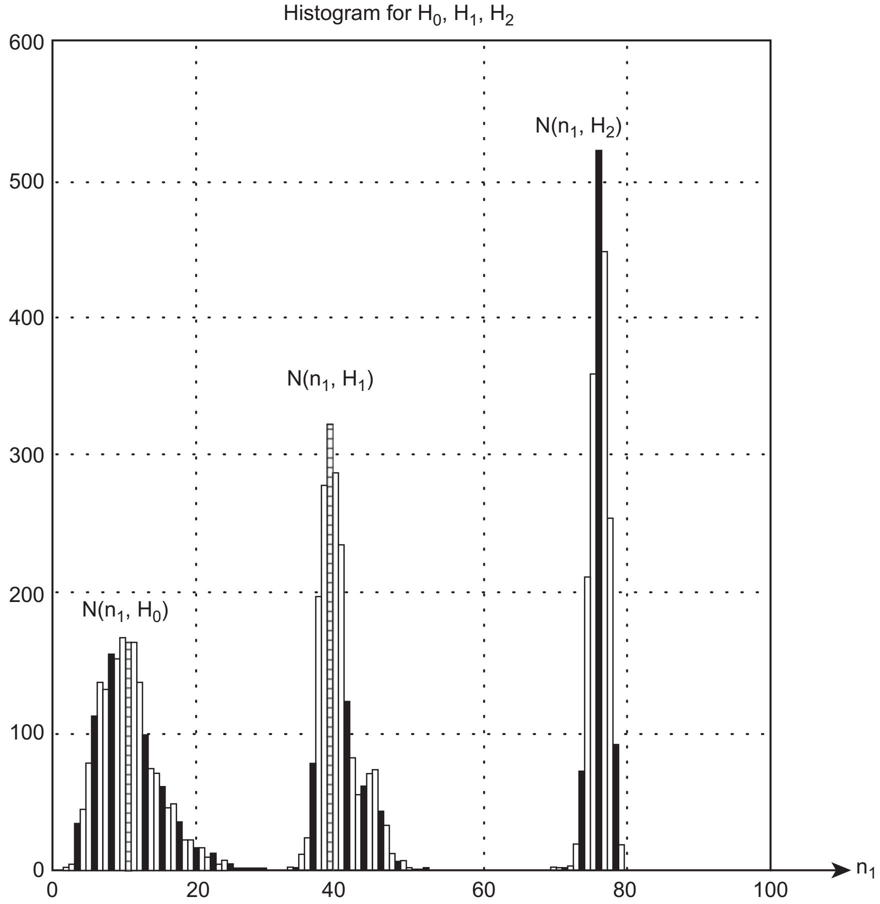


Figure 10.10:
Histograms of nonuniformity index.

The criterion for misfire is as follows:

$n_1 > n_t \rightarrow$ misfire

$n_1 < n_t \rightarrow$ no misfire.

There are two types of error associated with the above misfire criterion:

$n_1 < n_t$ for an actual misfire (missed detection)

$n_1 > n_t$ 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 does not, by itself, identify the cylinder(s) that is (are) misfiring. The nonuniformity index vector $\overline{n}_1$ 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 ($\overline{n}$) 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 $\overline{n}(m)$. This "signature" can be obtained by running the engine with cylinder m purposely disabled (i.e. via fuel or spark). Data for the nonuniformity vector $\overline{n}(m)$ are given by the statistical average of $\overline{n}$ over a sample of K engine cycles:

$$\overline{n}\;(m) = \frac{1}{K}\sum_{k=1}^{K}\overline{n}_k, \qquad m = 1, 2 \ldots M \tag{15}$$

where $\overline{n}_k$ = nonuniformity vector for the kth engine cycle.

Each of these M vectors is directed to a point in a $2M$ dimensional space. The isolation of the misfiring cylinder is done by finding the shortest "distance" from a nonuniformity vector $\overline{n}$ to these vectors. This vector distance (for the kth engine cycle) in $2M$ dimensional space $\overline{\delta}_k(m)$ is given by

$$\overline{\delta}_k(m) = \overline{n}(m) - \overline{n}_k \tag{16}$$

where $\overline{n}$ 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

$$\min_m(\|\overline{\delta}_m\|_2) = \|\overline{\delta}_{m_o}\|_2 \tag{17}$$

That is, cylinder $\mathrm{m_o}$ ($\mathrm{m_o} = 1, 2, \ldots M$) has the minimum $\| \bar{\delta}_{m_o} \|_2$ and is identified as the misfiring cylinder.

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 which are measurable with a simple, inexpensive non-contacting 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.

A close enough estimate of δT_n for misfire detection purposes can be obtained from a sliding mode observer (SMO) based upon a relatively straightforward system for measuring crankshaft angular speed (ω_e). The model from which this SMO is built for an automatic transmission-equipped vehicle with unlocked torque converter is given below

$$J\dot{\omega}_e = T_i(\theta_e) - T_R(\theta_e) - R_e\omega_e - T_l(\theta_e) \tag{18}$$

where $J =$ moment of inertia of engine rotating parts

and where $T_l =$ 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 for which T_l is a constant. This term can be neglected in the computation of torque fluctuations (as is done here).

Combining Eqns (10.4 through 10.8) with Eqn (10.18) yields the following model for $\dot{\omega}_e$

$$\dot{\omega}_e = \frac{1}{J + M_{eq}R_c^2 f_T^2(\theta_e)}\{(p_c - p_o)A_p R_c f_T(\theta_e) - M_{eq}R_c^2 f_T(\theta_e) f_B(\theta_e)\omega_e^2 - R_e\omega_e\} \tag{19}$$

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

$$x = [x_1\ x_2]^T, \quad x_1 = \theta_e \quad x_2 = \omega_e$$

yields $\dot{x}_1 = x_2$

$$\dot{x}_2 = \frac{1}{J + M_{\mathrm{eq}}R_{\mathrm{c}}^2 f_{\mathrm{T}}^2(x_1)}\left\{(p_{\mathrm{c}} - p_{\mathrm{o}})A_{\mathrm{p}}R_{\mathrm{c}}f_{\mathrm{T}}(x_1) - M_{\mathrm{eq}}R_{\mathrm{c}}^2 f_{\mathrm{T}}(x_1)f_{\mathrm{B}}(x_1)x_2^2 - R_{\mathrm{e}}x_2\right\} \tag{20}$$

It is shown below that both x_1 and x_2 are measurable with inexpensive non-contacting 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:

$$\begin{aligned}\hat{x}_2 = {} & \frac{1}{J + M_{\mathrm{eq}}R_{\mathrm{c}}^2 f_{\mathrm{T}}^2(y_1)}\{ - A_{\mathrm{SMO}}\mathrm{sgn}\;\;[f_{\mathrm{T}}(y_1)(\hat{x}_2 - y_2)]A_{\mathrm{p}}R_{\mathrm{c}}f_{\mathrm{T}}(y_1) \\ & -M_{\mathrm{eq}}{}^2R_{\mathrm{c}}^2 f_{\mathrm{T}}(y_1)f_{\mathrm{R}}(y_1)y_2^2 - R_{\mathrm{e}}y_2\}\end{aligned} \tag{21}$$

where $A_{\mathrm{SMO}} =$ SMO gain

sgn( ) = sign function of argument.

The SMO gain requirement is that it be larger than the maximum value that can occur for $(P_{\mathrm{e}} - P_{\mathrm{o}})$:

$$A_{\mathrm{SMO}} > \max\,(P_{\mathrm{c}} - P_{\mathrm{o}})$$

The estimate of indicated torque is obtained as the output of the first order filter given by

$$\tau\,\dot{v} + v = -A_{\mathrm{SMO}}\mathrm{sgn}\,\left[f_{\mathrm{T}}(y_1)(\hat{x}_2 - y_2)\right]A_{\mathrm{p}}R_{\mathrm{c}}f_{\mathrm{T}}(y_1) \tag{22}$$

$$\hat{T}_{\mathrm{n}} = v$$

Using this SMO to estimate $\hat{T}_{\mathrm{n}}$ it is possible to form δT_{n} and the vector $\overline{T}$ from which misfire detection is possible as explained above.

The measurement of crankshaft angular position and speed can readily be made using a non-contacting sensor such as that depicted in Figure 10.11 and as explained in Chapter 6. In Figure 10.11, the ferromagnetic disk (with lugs) is attached to the crankshaft. However, for the accuracy in measurements of θ_{e} and we 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 which 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 $\omega_{\mathrm{e}}(t)$ is given by:

$$\omega_{\mathrm{e}}(t) = \Omega_{\mathrm{e}} + \delta\,\omega_{\mathrm{e}}(t) \tag{23}$$

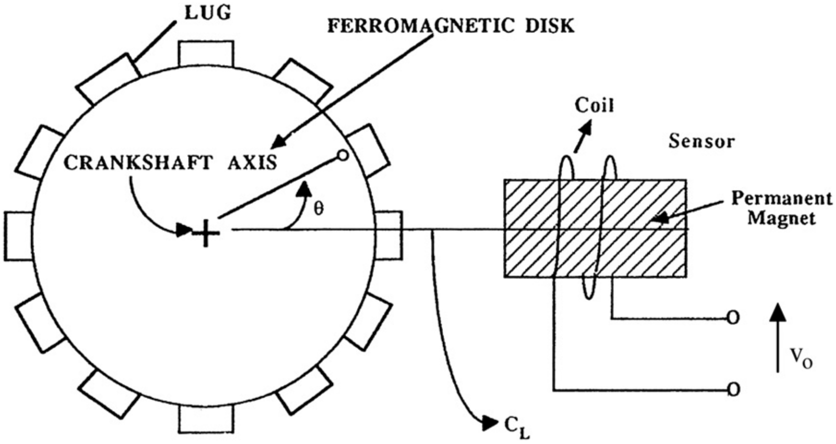


Figure 10.11:
Non-contacting crankshaft angular speed sensor.

where $\Omega_e = \dfrac{\pi\ \text{RPM}}{30} = \text{short-term time average of } \omega_e$

$\delta\omega_e(t) = \text{variation in } \omega_e \text{ due to } \delta T_n$

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 $\Omega_e \gg \max(\delta\omega_e)$.

The crankshaft instantaneous angular position $\theta_e(t)$ is given by

$$\theta_e(t) = \theta_o + \int_o^t \omega_e\left(t'\right)dt' \tag{24}$$

where $\theta_o = \theta_e(0) = \text{phase reference}$.

The phase reference can be established relative to the engine cycle via a camshaft once/revolution non-contacting sensor (see Chapter 7).

The sensor output signal $v_0(t)$ is given by

$$v_0(t) = f[M_d\theta_e(t) + \psi] \tag{25}$$

where $M_d = \text{number of lugs on disk}$

$\psi(t) = \text{random process (error in the sensor output)}$.

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

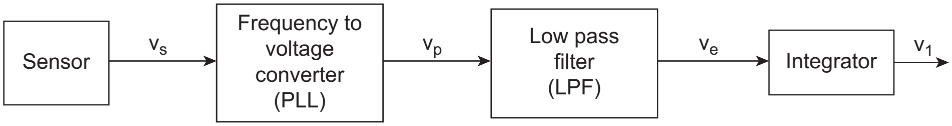


Figure 10.12:
Block diagram for ω_e measurement.

digital electronic signal processing. Figure 10.12 shows a block diagram for an analog signal processing.

The "frequency to voltage converter" is in effect an FM demodulator which can be implemented with a circuit known as a "phase-locked loop" (PLL). The PLL is an electronic closed-loop system. It is output voltage $\mathrm{v_p}(t)$ is given by

$$\mathrm{v_p} = K_\mathrm{p}\left(M_\mathrm{d}\omega_\mathrm{e}(t) + \dot{\psi}\right)$$

The low-pass filter (LPF) passes the first term and suppresses that portion of the spectrum of $\dot{\psi}$ which lies outside the LPF pass bands thereby yielding the measurement of ω_e needed for the SMO to compute $\hat{T}_n$. 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 3 the integrator output voltage V_I is given by

$$V_1 = \frac{1}{\tau_\mathrm{i}}\int_0^\mathrm{t} V_\mathrm{e}(t')\mathrm{d}t'$$

$$= \frac{K_\mathrm{p}\,M_\mathrm{d}}{\tau_\mathrm{i}}[\theta_\mathrm{e}(t) + \theta_\mathrm{o}] \tag{26}$$

where $\tau_\mathrm{i} =$ integrator time constant

Of course, digital integration as explained previously (e.g. see Chapter 8) can also be used to obtain V_I. The phase origin for this measurement of $\theta_\mathrm{e}(t)$ is established via the once/revolution camshaft sensor. The measurement of θ_e is required as part of the computation of the nonuniformity index $\overline{n}$.

In a contemporary implementation the measurement of $\omega_\mathrm{e}(t)$ is done in discrete time based upon successive samples of $v_\mathrm{o}(t)$. As explained in Chapter 6, a sensor such as is depicted in Figure 10.12 generates an output waveform which 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 $\omega_e(t)$ which is denoted $\omega_e(k)$ and is given by:

$$\omega_e(k) = \frac{1}{M_d \delta\, t_k} \tag{27}$$

If M_d is sufficiently large, the sequence $\{\omega_e(k)\}$ will be an un-aliased sample of $\omega_e(k)$.

The instantaneous crankshaft angular position $\theta_e(k)$ is given by:

$$\theta_e(k) = \theta_e(t_k), \quad k = 1, 2 \cdots 2M_d \tag{28}$$

This sampled crankshaft angular position is readily obtained by passing the sensor through a zero crossing detector (ZCD) and counting the output pulses using a binary counter (see Chapter 4 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 Figure 10.12 modulo $2M_d$.

Using the instrumentation above for measuring ω_e and θ_e provides the necessary values for a calculation of $\hat{T}_n$ using the SMO as well as the nonuniformity index $\overline{n}$. The misfire detection proceeds using the estimate of T_n 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 less than 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 Figure 10.13.

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.

Figure 10.14 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 10.2.

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.

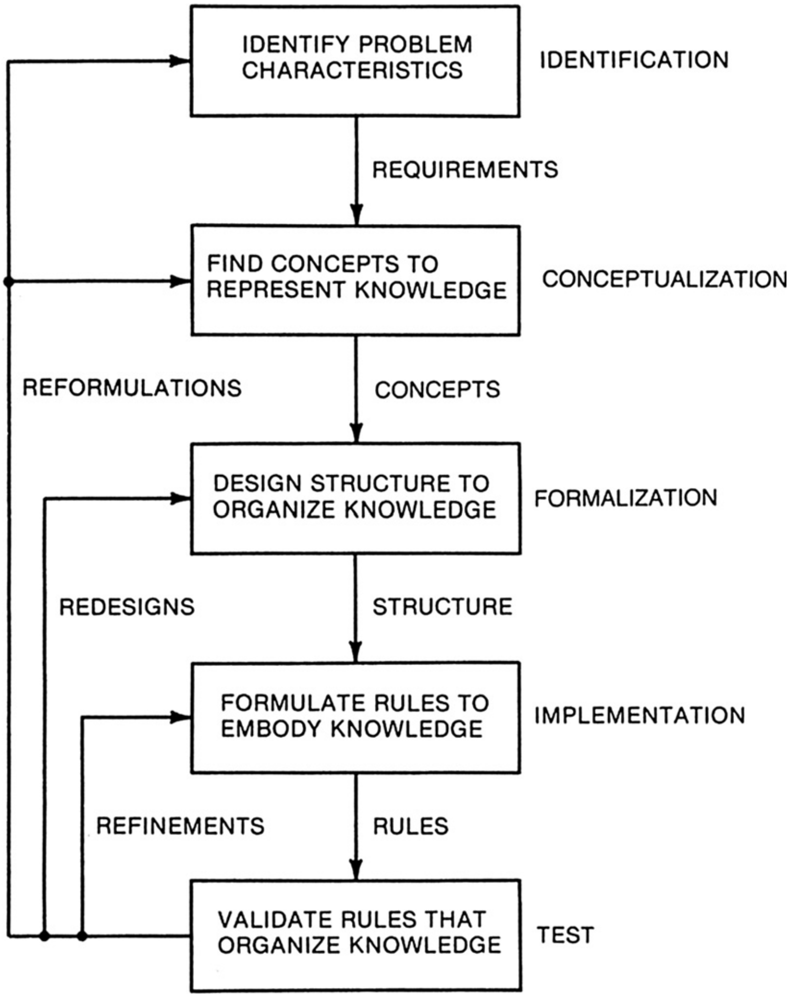

IDENTIFY PROBLEM CHARACTERISTICS
IDENTIFICATION
REQUIREMENTS
FIND CONCEPTS TO REPRESENT KNOWLEDGE
CONCEPTUALIZATION
REFORMULATIONS
CONCEPTS
DESIGN STRUCTURE TO ORGANIZE KNOWLEDGE
FORMALIZATION
REDESIGNS
STRUCTURE
FORMULATE RULES TO EMBODY KNOWLEDGE
IMPLEMENTATION
REFINEMENTS
RULES
VALIDATE RULES THAT ORGANIZE KNOWLEDGE
TEST

Figure 10.13:
Expert system development procedure.

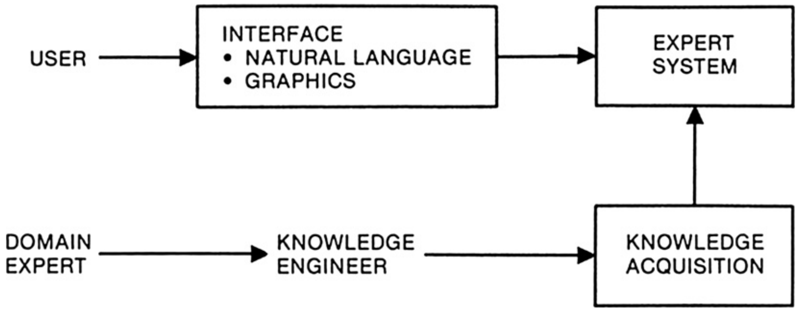


Figure 10.14:
Environment of an expert system.

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.

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

Table 10.2: 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

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 to 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 Figure 10.15.

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 Figure 10.16.

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 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 Figure 10.17. 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).

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

```
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 degrees Fahrenheit
    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)
```

Figure 10.15:
Simple automobile diagnostic rule base.

```
gas gauge works
starter turns engine but it fails to start
gas gauge is not on empty
can smell gas
```

Figure 10.16:
Starting database of known facts.

```
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)
```

Figure 10.17:
Final resulting database of known facts.

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, including, 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 e-mail.

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.

Occupant Protection Systems

Occupant protection during a crash has evolved dramatically since about the 1970s. Beginning with lap seat belts, and motivated partly by government regulation and partly by market demand, occupant protection has evolved to passive restraints and airbags. We will discuss only the latter since airbag deployment systems can be implemented electronically, whereas other schemes are largely mechanical. Whereas the first airbag occupant protection systems were intended for occupant protection in crashes that were mostly along the vehicle longitudinal axis, contemporary vehicles provide side impact protection.

Occupant protection by an airbag is conceptually quite straightforward. The airbag system has a means of detecting when a crash occurs that is essentially based on exceptional deceleration along a car axis. A collision that is serious enough to injure car occupants involves deceleration in the range of tens or hundreds of gs, whereas normal driving involves acceleration/deceleration less than 1 g.

Once a crash has been detected, a flexible bag is rapidly inflated with a gas that is released from a container by electrically igniting a chemical compound. Ideally, the airbag inflates in sufficient time (e.g. typically $\leq$50 ms) to act as a cushion for the driver (or passenger) as he or she is thrown forward or sideways during the crash.

On the other hand, practical implementation of the airbag has proven 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 10.3 lists required airbag deployment times for a variety of test crash conditions.

A typical airbag will require about 30 ms to inflate, meaning that the crash must be detected within about 20 ms. With respect to the speed of modern digital electronics, a 20-ms time interval is not considered to be short. The complicating factor for crash detection is the many crash-like accelerations/decelerations experienced by a typical car that could be interpreted by airbag electronics as a crash, such as impact with a large pothole or driving over a curb.

The configuration for an airbag system has also evolved from electromechanical implementation (using switches) to electronic systems employing sophisticated signal processing. 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 Figure 10.18. Each of these switches is in the form of a mass suspended in a tube with the tube axis aligned parallel to the longitudinal car axis. Figure 10.18b is a circuit diagram for the 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 ignitor that activates the charge. A gas is produced (essentially explosively) that inflates the airbag.

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 locate a switch under the driver's seat on the floor pan).

Table 10.3: 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
30 mph 550 hop road, panic stop	ND
30 mph 629 hop road, panic stop	ND
30 mph 550 tramp road, panic stop	ND
30 mph 629 tramp road, panic stop	ND
30 mph square block road, panic stop	ND
40 mph washboard road, medium braking	ND
25 mph left-side pothole	ND
25 mph right-side pothole	ND
60 mph chatter bumps, panic stop	ND
45 mph massoit bump	ND
5 mph curb impact	ND
20 mph curb drop-off	ND
35 mph Belgian blocks	ND

Note: ND = nondeployment.

Referring to the sketch in Figure 10.18, 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:

$$M\ddot{x} + D\dot{x} + F_c + Kx = 0 \tag{29}$$

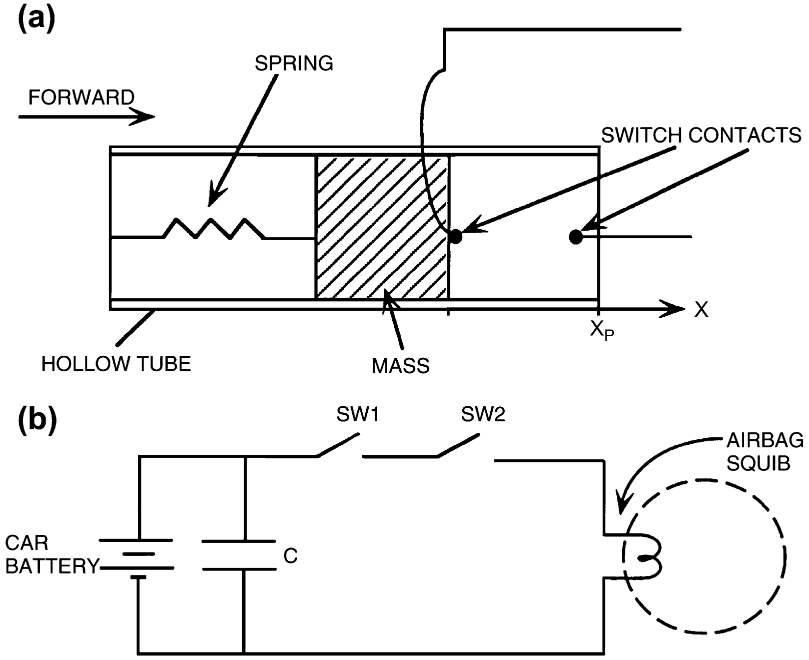


Figure 10.18:
Airbag deployment system.

where M = mass of the movable element

D = viscous friction coefficient

F_c = coulomb friction force (stiction)

K = spring constant.

x = mass displacement of M from stop.

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:

$$D\dot{x} + F_c + Kx = Ma \tag{30}$$

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.

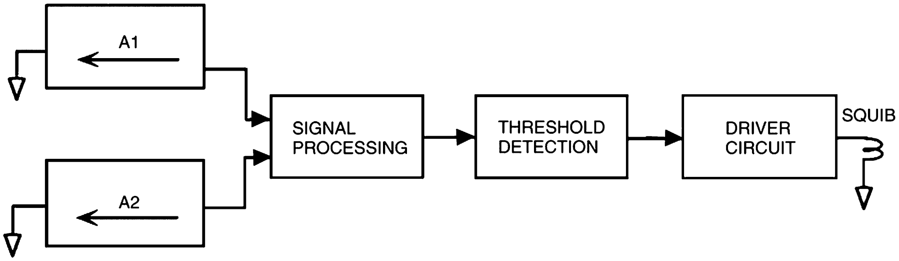


Figure 10.19:
Accelerometer-based airbag system.

Figure 10.18 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.

In recent years, there has been a trend to implement electronic airbag systems. 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. Figure 10.19 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 illustration of the characteristic waveform from an accelerometer, Figure 10.20 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 less than 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 Figure 10.20 for this crash with the deployment requirements of Table 10.3 illustrates the complexity of the signal processing necessary to properly deploy the airbag.

After being processed, the deceleration signal is compared with a threshold level. As long as the processed signal is less than this threshold the driver circuit remains deactivated. However, when this signal exceeds the threshold, the driver circuit sends a current of sufficient strength to activate the chemical and inflate the airbag.

Typically, the threshold is set so that airbag deployment occurs for a crash into a barrier at or above a specific speed. Depending on the system design, this speed can be anywhere between

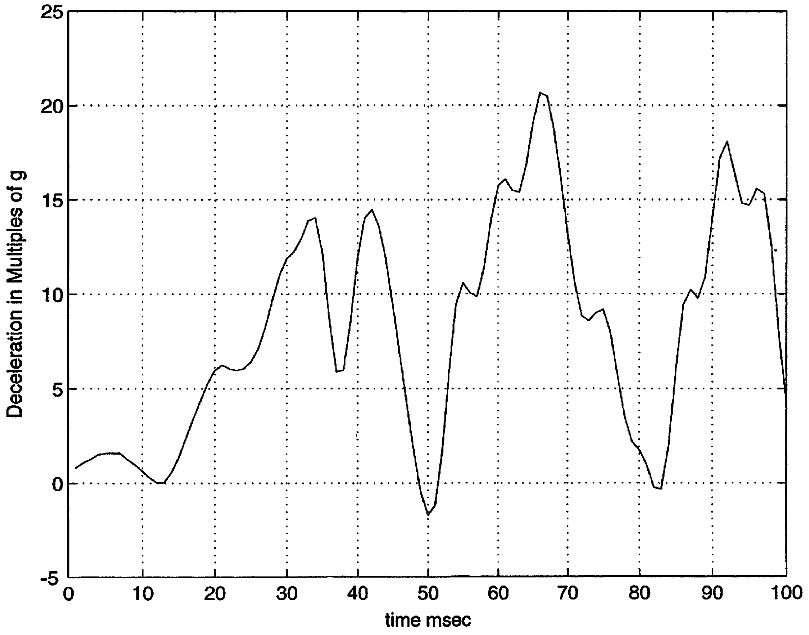


Figure 10.20:
Acceleration data for 30 mph crash.

8 and 12 mph. This speed range is chosen by the manufacturer to optimize the protection offered to the car occupants while minimizing (or completely eliminating) false deployment (that is, deployment when there is no crash).

In addition to airbags for protecting the driver and front seat passenger against frontal collision, airbags have become available for occupant protection against other types of collisions. Airbags now are available for protection against side impact. Conceptually, these occupant protection systems operate in ways similar to the type described above.

There will continue to be new developments in airbag technology in order to improve performance. Complicating this task is the fact that the signature of a crash differs depending on the crash configuration and vehicle design. For example, there is one class of signature for a crash into a rigid barrier (i.e. a nonmoving and incompressible object) and another for a crash between a pair of cars (particularly when vehicle curb weights are different). In spite of technical difficulties in implementation, the airbag is finding broad application for occupant protection and has achieved broad acceptance by the driving public.

In addition to airbags, contemporary vehicles employ passive restraint systems consisting of lap and over-the-shoulder belts. These passive restraints combined with airbags offer a high level of occupant protection. Although it is not an electronic system, another aspect of occupant protection comes from an optimal vehicle structural design. Contemporary vehicles are structurally designed to absorb crash energy in such a way as to minimize intrusion of damage into the passenger compartment.

This page intentionally left blank

Glossary

Accumulator (Computer component): The basic work register of a computer.

Actuator: A device which performs an action in response to an electrical signal.

A/D (also ADC): Analog-to-digital converter; an electronic circuit device that generates binary or binary coded digital output that is proportional to an analog input voltage.

A/F: *See* Air/Fuel Ratio.

Analog Circuits: Continuous time linear electronic circuits in which voltages are proportional to some physical quantities.

Assembly Language: An abbreviated computer language in which basic computer instructions and addresses are represented by pneumonic alphabetic symbol groups.

BDC: Bottom dead center; the extreme lowest position of the piston during its stroke.

Bit: A binary digit; a single digit in a binary number system.

Block Diagram: A system diagram which shows all of the major parts and their interconnections normally including an analytical model for each component.

BSCO: Brake specific CO: the ratio of the rate at which carbon monoxide leaves the exhaust pipe to the brake horsepower.

BSFC: Brake specific fuel consumption; the ratio of fuel consumption to the brake horsepower being generated.

BSHC: Brake specific HC; the ratio of the rate at which unburned hydrocarbons leave the exhaust pipe to the brake horsepower.

BSNO$_x$: Brake specific NO$_x$; the ratio of the rate at which oxides of nitrogen leave the exhaust pipe to the brake horsepower.

Byte: An 8-bit binary number.

CAFE: Corporate-Average-Fuel-Economy. The government mandated fuel economy which is averaged over the production for a year for any given manufacturer.

Capacitor: A two-terminal circuit component that stores charge and has a terminal voltage that is proportional to stored charge.

Catalytic Converter: A device which enhances certain chemical reactions which help to reduce the levels of undesirable exhaust gases from the tailpipe relative to their levels at the output of the engine.

Closed-Loop Control: A control system in which the control signal that regulates the plant output is a function of the difference between the desired and actual values of the output variable being regulated.

Closed-Loop Fuel Control: A fuel control mode where input air/fuel ratio is controlled by metering the fuel response to the rich-lean indications from an exhaust gas oxygen sensor.

CO: Carbon monoxide; an undesirable chemical combustion product due to imperfect combustion.

Combinational Logic: Logic circuits whose outputs depend only on the present logic inputs.

Combustion: The burning of the fuel-air mixture in the cylinder.

Comparator, Analog: An electronic circuit in which the binary valued ooutput voltage is determined by the relative amplitude of its two input voltages.

Compression Ratio: The ratio of the cylinder volume at BDC to the volume at TDC.

Control Variable: The output of the controller which is the input to the plant that regulates the output variable in accordance with the control law.

Conversion Efficiency (Catalytic Converter): the ratio of flow rate of remaining regulated exhaust gas in the converter output to their input flow rates.

CPU: Central processing unit; the portion of a computer that performs all arithmetic, logic and data transfer operations.

Cutoff: A transistor operating mode for which very little (ideally zero) current flows between the collector and emitter (with proper supply voltages) due to zero base current.

D/A (also DAC): Digital-to-analog converter; a device which produces a voltage which is proportional to the digit input number.

Damping Coefficient: The coefficient of the first order time derivative in the linear differential equation model for a second order system.

DEMUX: Demultiplexer; a type of electronic switch used to select one of several input lines for its output.

Diesel: A class of internal combustion engine (also known as a compression ignition engine) in which combustion is initiated by the high temperature of the compressed air/fuel mixture in the cylinder during a compression stroke.

Differential Gain: The coefficient multiplying the time derivative of the error in a PID control law.

Digital Circuits: Electronic circuit systems made up of combinations of individual circuits whose output voltages switch between two levels corresponding to 1 and 0 of a binary or a binary coded number system.

Diode: A two-terminal circuit component that ideally functions as an electronic switch whose state is a function of the polarity of the applied voltage.

Display: A device which indicates in human readable form the result of measurement of some variable.

Drivetrain: The combination of mechanisms connecting the engine to the driving wheels including transmission, driveshaft, and differential.

Dwell: The portion of an engine cycle during which current flows through the primary circuit of the ignition coil for each spark generation.

Dynamometer: A device for applying a mechanical load to an engine output having the capability of measuring engine performance.

Efficiency: A variable or parameter that serves as a metric for the performance of a system or component in achieving a desired result.

EGO: Exhaust gas oxygen; the concentration of oxygen in the exhaust of an engine.

EGO Sensor: A sensor is used in closed-loop fuel control systems to indicate rich or lean A/F.

EGR: Exhaust gas recirculation; a mechanism in which a portion of exhaust is introduced into the intake of an engine.

Electronic Carburetor: A fuel metering actuator in which the air/fuel ratio is controlled by continual variations of the metering rod position in response to an electronic control signal (now obsolete).

Engine Calibration: The values for air/fuel, spark advance, and EGR at any operating condition.

Engine Crankshaft Position: The angular position of the crankshaft relative to a reference point.

Engine Mapping: A procedure of experimentally determining the performance of an engine at selected operating points and recording the results.

Equivalence Ratio: Actual air/fuel ratio divided by the air/fuel ratio at stoichiometry.

Evaporative Emissions: Evaporated fuel from the fuel system which can mix with the surrounding air unless collected by a subsystem.

Foot-Pound: A unit of torque corresponding to a force of one pound acting on a one-foot level arm.

Frequency Response: A graph of the magnitude and phase of a system transfer function for sinusoidal input as a function of frequency.

Gain: The ratio of a system's output magnitude to its input magnitude.

Gasoline: The fuel for spark ignited internal combustion (IC) engines consisting of a mixture of various chemical compounds of hydrogen and carbon (with various additives).

Hardware (Computer): The Electronic digital circuits and peripheral devices that constitute the physical structure of a computer.

Harness: The wire bundles that interconnect all electrical and electronic components/systems in an automobile.

HC: Abbreviation for hydrocarbon chemicals, such as gasoline, formed by the union of carbon and hydrogen.

Ignition Timing: The time of occurrence of ignition measured in degrees of crankshaft rotation relative to TDC.

Inductor: A two-terminal magnetic circuit component that stores energy in a magnetic field produced by current flowing in it and having a terminal voltage that is proportional to the time rate of change of that current.

Instrumentation: Apparatus (often electronic) which is used for measurement or control, and for display of measurements or conditions.

Integral Gain: The coefficient multiplying the integral of the error in a PI or PID control law.

Integrator: An electronic circuit or computer function whose output is the time integral of its input.

Integrated Circuit: A semiconductor device which contains many circuit functions on a single chip interconnected in such a way as to perform high level functions.

Interrupts: An input to a microprocessor that provides an efficient method of quickly providing a signal that a particular external event has occurred, normally resulting in an interruption in the execution of a sequence of computer instructions.

Lead/Lag Term: A control system phase compensation that provides a phase advance (lead) or delay (lag) in the control law.

Limit Cycle: A mode of control system operation in which the controlled variable switches between a pair of limits with the average near the desired value.

Linear Region: A transistor operating mode where the collector current is proportional to the base current.

Logic Circuits: Digital electronic circuits that perform logical operations such as NOT, AND, OR, and combinations of these.

Lookup Table: A table in the computer memory of selected data points for a dependent variable at corresponding points for an independent variable from which the value of the dependent variable can be computed at intermediate values of the independent variable by interpolation

MAP: Manifold absolute pressure; the absolute pressure in the intake manifold of an engine.

Mathematical Model: An equation or set of equations expressing the functional relationship between the variables of a given system from which the dynamic response of the system outputs can be computed for various inputs.

Microcomputer: A small computer which uses an integrated circuit microprocessor and related components capable of discrete time control applications

MUX: Multiplexer; a type of electronic switch used to select one of several input lines and functionally connect that input to the MUX output.

NOx: The various oxides of nitrogen.

Op Code: A binary number that causes a specific microprocessor operation to occur when placed in the instruction decode circuit

Open-Loop Control: A control mode in which the control or actuator signal is computed as a function of a measurement of an input.

Open-Loop Fuel Control: A mode where engine input air/fuel ratio is controlled by measuring the mass flow rate of input air and adding the proper mass flow rate of fuel to obtain the desired air/fuel ratio.

Operational Amplifier: A standard analog building block with two inputs, one output, and a very high voltage gain between the differential input and the output.

Optimal Damping: The damping of a second order system that produces the fastest time response without overshoot.

Peripheral: An external input-output device that is connected to a computer.

Phase Shift: The time delay between the output of a system and the sinusoidal input expressed as a fraction of the period of the sinusoid either in degrees or radians.

Plant: The system that is to be controlled in a control system.

Port Fuel Injection: A fuel metering system in which fuel from a fuel injector is directed into the intake of a cylinder.

Proportional Gain: The component of a control (or actuating) signal that is proportional to the difference between the desired and actual output of a closed-loop control system.

Qualitative Analysis: A study that reveals how a system works expressed intuitively rather than analytically.

Quantitative Analysis: An analytical performance analysis for a system.

RAM: Random access memory; read/write memory.

Random Error: A stochastic measurement error that is characterized by its statistics.

Reference Input: The input to a control system that corresponds to the desired plant output.

Resister: A two-terminal circuit component whose terminal voltage is proportional to the current passing through it

ROM: Read only memory; permanent memory used to store programs and parameters.

RPM: Revolutions per minute; the angular speed of rotation of the crankshaft of an engine or other rotating shaft.

Sample and Hold: The act of measuring a voltage at a particular time and storing that voltage until a new sample is taken.

Sampling: The act of obtaining a measurement of a time varying voltage at discrete times (mostly done periodically).

Saturation: A transistor operating mode for which the collector-emitter current is the maximum possible for a given supply voltage and collector/emitter resistance (functionally equivalent to a closed switch between collection and emitter).

Semiconductor: A material that has an electrical conductivity somewhere between that of a conductor and an insulator.

Sensor: An energy conversion device which measures some physical quantity and converts it to an electrical quantity (e.g., voltage).

Sequential Logic: Logic circuits whose output depends on the particular sequence of the input logic signals often requiring a timing (clock) input.

SI Engine: Abbreviation for spark-ignited, gasoline-fueled, piston-type, internal-combustion engine.

Signal Processing: The alteration of an electrical signal by electronic circuitry used to achieve a desired result (e.g., filtering).

Skid: A condition in which the tires are sliding over the road surface rather thin rolling; usually associated with braking.

Slip: The ratio of the difference between a given speed (either translational or rotational) and a corresponding reference speed to the reference speed.

Software: The computer program instructions to perform the desired computer operations.

Spark Advance: The number of degrees of crankshaft rotation before TDC where the spark plug is fired. (See ignition timing.)

Spark Timing: The process of firing the spark plugs at the proper moment to ignite the combustible mixture in the engine cylinders.

Stoichiometry: The air/fuel ratio for perfect combustion for which all of the hydrogen and carbon in the fuel are oxidized to H_2O and CO_2.

System: A collection of interacting parts.

Systematic Error: A measurement error in instrumentation system which is predictable and correctable.

TBFI: Throttle-body-fuel-injector; a fuel metering actuator in which the air/fuel ratio is controlled by injecting precisely controlled spurts of fuel into the air stream entering the intake manifold (now obsolete).

TDC: Top dead center; the extreme highest point of the piston during its stroke.

Throttle Angle: The angle between the throttle plate and a reference line.

Torque Converter: A form of fluid coupling used in an automatic transmission in which the coupling between the input and output shafts is achieved by momentum transfer of a fluid between an input shaft and an output shaft.

Torque: The twisting force of the crankshaft or other driving shaft.

Transfer Function: The ratio of the Laplace transforms of the output to the input of a linear system with zero initial conditions for which complex frequency **s** is the independent variable.

Tranformer: An electrical circuit component capable of inductively coupling circuits for time-varying electrical signals and changing a-c voltage levels.

Transistor: A three-terminal active semiconductor circuit component capable of amplifying currents or voltages.

Transport Delay (IC Engine): The time required for a given mass of fuel and air to travel from the intake manifold through the engine to the EGO sensor in the exhaust manifold.

Volumetric Efficiency: The pumping efficiency of the engine as air is pumped into the cylinders.

Quiz questions

Chapter 1

1. Explain the purpose of the following systems:

   a) control system

   b) instrumentation system

   c) communication system.

2. Given a system having input $x(\mathrm{t})$ and output $y(\mathrm{t})$ which are related by the following differential equation:

$$2\frac{\mathrm{d}^2y}{\mathrm{d}t^2} + 1.7\frac{\mathrm{d}y}{\mathrm{d}t} + 4.3y = 1\frac{\mathrm{d}x}{\mathrm{d}t} + 10x$$

   a) define the operational transfer function for this system $H(s)$

   b) find $H(s)$.

3. Find the poles and zeros for the system of Problem 2.

4. The input to a system $x(t)$ is a unit step:

$$\begin{aligned} x(t) &= 0\ t < 0 \\ &= 1\ \mathrm{t} \geq 0 \end{aligned}$$

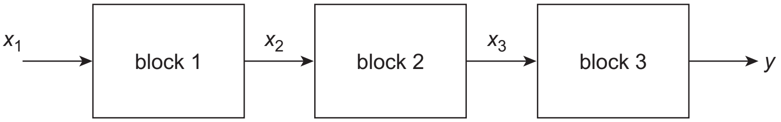
   Show that the Laplace transform $x(s)$ for this input is given by:

$$x(s) = \frac{1}{s}$$

5. Find the unit response for the system of Problem 2 assuming:

$$x(0) \ = \ 0, \quad y(0) = 0$$

6. A system consists of three functional blocks as depicted below:



   Assume that these blocks are modeled by the following equations:

   a) $\dfrac{3\mathrm{d}x_1}{\mathrm{d}t} + x_1 = x_2$

b) $0.5\dfrac{\mathrm{d}^2x_2}{\mathrm{d}t^2} + 2.1\dfrac{\mathrm{d}x_2}{\mathrm{d}t} + \dot{v}yx_2 = \dfrac{\mathrm{d}x_3}{\mathrm{d}t} + 1.1x_3$

c) $\dfrac{\mathrm{d}y}{\mathrm{d}t} + 0.1y = 2x_3$

Assume zero initial conditions and find the sinusoidal frequency response for this system.

7. Let a system be described by a two-dimensional state vector $x = [x, x_2]^T$ and a one-dimensional input $U(t)$ and a one-dimensional output. The system is described by the following set of equations:

$$\begin{aligned} \dot{x}_1 &= -3x_1 + x_2 \\ -\dot{x}_2 &= -2x_1 \\ y &= 2x_1 \end{aligned}$$

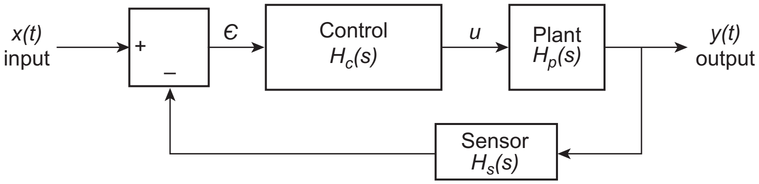
a) Find the system matrix A, the input matrix B and the output matrix C for a stable variable formulation of the form:

$$\begin{aligned} \dot{x} &= Ax \;+\; Bu \\ y &= Cx \end{aligned}$$

b) Find the operational transfer function $H(s)$ where:

$$H(s) = \frac{y(s)}{H(s)}$$

8. Given the PI control system depicted below:



assume the following component models:

$$\text{control: } H_c(s) = \frac{u(s)}{\in(s)} = K_p + \frac{\dot{K}_1}{s}$$

$$\text{plant: } H_p(s) = \frac{0.04}{s^2 + 2s + 0.75}$$

$$\text{sensor: } H_s(s) = 1$$

a) Show that the error $\in(x)$ = next gain.

b) Let $K_I/K_p - 3.15$ and with K_p as a variable, find the closed loop transfer function for this control system.

c) Using the root locus method of MATLAB for the equivalent, show that the closed loop system becomes unstable for sufficiently large K_p.

d) Find the overshoot (%) of the step response of the system for $K_p = 1.32$ (hint, use MATLAB).

e) Find the gain and phase margins.

9. a) Show that a control system for a plant having a transfer function $Hp(s)$ given by:

$$H_p(s) = \frac{10}{s(s+0.5)}$$

and a proportional only controller is stable for any gain provided the feedback sensor is ideal (i.e. $H_s(s) = 1$).

b) However, if the sensor has a dynamic response modelled by a first order transfer function given by:

$$H_s(s) = \frac{1}{s+2}$$

The closed loop system becomes unstable for a sufficiently large gain.

c) Find the value of proportional gain for which the system becomes unstable.

10. Explain the role of sensor dynamics in the systematic errors of an instrumentation system.

11. A sensor having a resistive source impedance $R = 10$ kΩ is connected to a signal processing filter having a transfer function $H_{sp}(s)$ given by:

$$H_{sp}(s) = \frac{1000}{s+1000}$$

If the sensor is at temperature $T = 400°K$, find the rms random error voltage $\tilde{v}$ in the output of the filter. Note: Boltzmans constant is $k = 1.4 \times 10^{-23}$ Joule/°K.

Chapter 2

1. A voltage $v(t)$ is given by:

$$\mathrm{v}(t) = 100(1 - \cos(2\pi t))$$

This voltage is to be sampled at a period $T = 0.01$.

a) Find the value of the 15th sample, i.e. $\mathrm{v}(k)|_{k=15}$

where $\mathrm{v}(k) = \mathrm{v}(t_k)$

and

where $t_k = kT$

The input x_k to a discrete time system is to be an 8-bit representation of $v(t_k)$ rounded to the nearest binary integer

$$x_k = \{1\ v(t_k)\}$$

b) Find the 8-bit binary representation of x_{15}

c) What is the lowest theoretical sample rate (i.e. the Nyquist rate) for sampling this voltage?

2. Given a function of time $f(t)$ of the form:

$$\begin{aligned} f(t) &= 0 \qquad t > 0 \\ &= e^{-at} \quad t > 0 \end{aligned}$$

This function is sampled at a period T_s.

a) Find the z-transform $F(z)$ of $f(t_k)$ where:

$$t_k = k\,T_s$$

b) What is the region of convergence for $F(z)$

c) Find the pole(s) for $F(z)$

3. Find the inverse z-transform of the function $Y(z)$ (i.e. find y_k) where $Y(z)$ is given by:

$$Y(z) = \frac{1}{14.3z^{-1} - 0.4z^{-2}}$$

4. Find the z-transfer function and recursive algorithm for a 3rd order Butterworth low-pass filter having the following design parameters:

$$\begin{array}{rl} \text{sample frequency:} & F_s = 20\,\text{kHz} \\ \text{corner frequency:} & F_c = 3.5\,\text{kHz} \end{array}$$

5. For the open loop control system of Figure 2.6, assume the following:

$$\text{digital control:} \quad U_k = K_p x_k$$

$$\text{plant transfer function:} \quad H_p(s) = Y(z)/U(z) = \frac{K_a}{s + s_0}$$

parameters $K_p = 10$
$K_a = 50$
$s_0 = 2.5$
sample period $T = 0.01$ sec

a) Find the z-transfer function $H(z)$

$$H(z) = \frac{Y(z)}{X(z)}$$

b) Is the closed loop control system stable?

c) If the system is stable, find the unit step response.

6. For the closed loop feedback control system of Figure 2.9, assume the parameters given for the example and show that the closed loop z-transfer function $H_{Cl}(z)$ is given by:

$$H_{Cl}(z) = \frac{0.029z + 0.0257}{z^2 - 1.668z + 0.7226}$$

7. What affect does the sample period have on the accuracy of the discrete time model of a digital control system performance analysis?

8. What is the influence of a ZOH on the dynamic response of a discrete time digital system?

9. What is the effect of sampling a continuous time signal on the spectrum of that signal (i.e. compare spectra at original and sampled signals)?

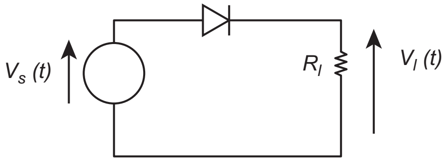
10. How can the original version of a sampled signal be recreated from the sampled signal and what are the limitations on the accuracy of such reconstruction?

Chapter 3

1. How do holes contribute to electrical conduction in a semiconductor material?

2. What is the source of the junction potential in a *p-n* semiconductor diode?

3. Sketch the wave form of voltage v_e for the following diode circuit:



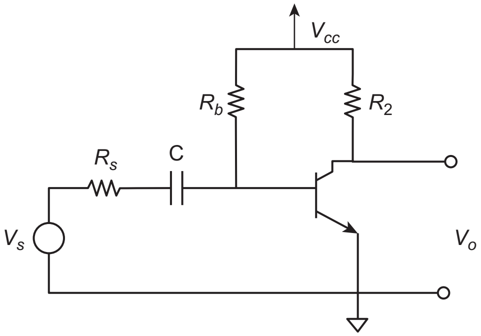
Where $V_s(t)$ is a symmetric square wave having the following model:

$$V_s(t) = 5 \qquad t_n \leq t \leq t_n + \frac{T}{2}$$

$$= -5 \quad t_n + \frac{T}{2} < t < t_{n+1}$$

$$t_n = n\,T \qquad n = 1,\ 2\ldots$$

4. Given the following single stage NPN transistor amplifier circuit:



Let $V_s(t)$ be given by:

$$V_s(t) = V_s \sin \omega_s t$$

assuming the following:

a) the transistor is biased via R_b to its linear range

b) $R_s \gg 1/\omega_s\, c$

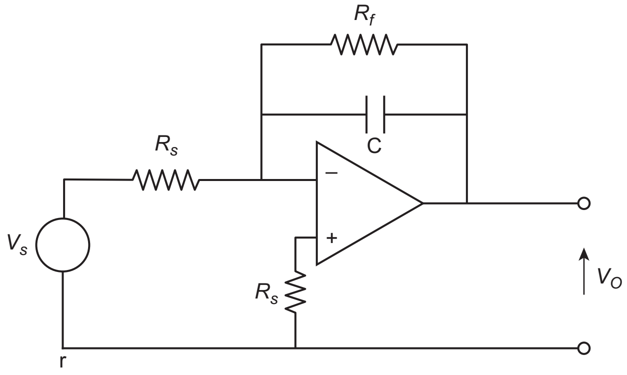
c) $V_{cc} > V_s$

d) $V_o(t) = V_{co} + V_o \sin \omega_s t$

Find the voltage gain A for this circuit:

$$A = \frac{V_o}{V_s}$$

5. Given the operational amplifier circuit shown below:



$$\text{If} \quad R_s = 1\ \text{K}\,\Omega$$

$$R_f = 10\ \text{K}\,\Omega$$

$$C = 1\mu\ \text{fd}$$

a) find the closed loop gain $A_{c\ell}$ expressed as a transfer function.

b) find A_{ce} as a complex sinusoidal frequency response.

6. Find the 8-bit binary equivalent to decimal 74.

7. Using the truth table of the XOR circuit of Figure 3.17a, verify that the truth table for the half adder of Figure 3.17b is correct and show that this circuit configuration performs the sum of two 1 bit binary numbers.

8. Show via a truth table that the circuit of Figure 3.17c is a full adder.

Chapter 4

1.

a) Find the product of 27 and 15 by writing the numbers in binary form and by using binary multiplication.

b) How many bits are required to represent this product?

2. For the D/A converter circuit of Figure 4.17, let $D_n = 2.5000\,A_n$ for $n = 1,2 \ldots 8$. Let the input digital data be given by:

$$A_8 \ldots A_1 = 01101001$$

find the output voltage V_{out} for this input.

3. Let the contents of an 8-bit left shift register at clock time t_1 be given by:

$$C = 10100110$$

Find the contents of the shift register for t_2, t_7 and t_8.

4. Repeat Problem 3 for a shift right register.

5. Repeat Problem 3 for a rotate left register.

6. A 4 bit binary number $B_4B_3B_2B_1$ stored at memory location $M(B) = 10000001$ is to be multiplied by a second 4 bit binary number $C_4C_3C_2C_1$ stored at memory location $M(C) = 10000010$ using an 8-bit microprocessor.

   Using the assembly language mnemonics of Table 4.1, write an assembly language program for this operation for which the product D is to be stored at memory location $M(D) = 10000011$.

7. Liquid cooled engines that power most contemporary vehicles require cooling (via the radiator) to remove heat from the engine and prevent coolant temperature from exceeding a pre-determined threshold temperature (T_{th}). Depending upon the ambient temperature, a fan is required to remove heat from the radiator. In contemporary vehicles an electric motor powers the cooling fan. Although a relatively simple thermally activated switch can be used to switch power on to the motor, for the present problem assume that the cooling fan is to be regulated by the engine control microprocessor. Assume further that a coolant temperature sensor is available (as explained in Chapter 6) that produces an output voltage $V_0 = K_s\,(T_c - T_{ref})$ where T_c is the coolant temperature, T_{ref} and K_s are constants for the sensor. Sketch a block diagram for a subsystem (including the microprocessor as a block) that switches the coolant fan on whenever $T_c \geq T_{th}$. Assume memory mapped I/O and denote the input memory address M_I and the output memory address is M_o. Show all necessary blocks and assume that a binary output signal operates the fan switch.

8. Using the assembly language mnemonics of Table 4.1, write a program for regulating the coolant fan of Problem 7.

9. Given an 8-bit A/D converter such as is depicted in Figure 4.18, having an input voltage range $0 \leq v_{in} \leq 5$ volt where 0 volts corresponds to binary 0:

   a) find the resolution of this A/D converter in volts/bits

   b) find the output binary number for an input of 1.47 volts

   c) The maximum conversion time is to be 10 msec (*i.e.* 0.01 sec). Find the minimum clock frequency (f_c of Figure 4.19).

Chapter 5

1. A four cylinder engine has a displacement V_D. The clearance volume for each cylinder is 10% of the cylinder contained volume V_C. Find the theoretical maximum compression ratio for a gasoline/air mixture assuming an ideal adiabatic cycle.

2. Calculate the brake torque in lb.ft for an engine having brake HP of 25 HP at 2000 RPM.

3. Find the inlet manifold air density as a fraction of sea level standard day density P_0^-. For an inlet air pressure that is 30% of sea level standard day pressure at a temperature that is 100°F above T_0 where $T_0 = 518.7°R$ in English units.

4. An engine having displacement of 270 in^3 is operated part throttle at 2400 RPM. If the inlet air density is 0.018 slug/ft^3 and the volumetric efficiency under these conditions is 62%, find the mass flow rate in slugs/sec of the inlet mixture.

5. For the same engine and operating conditions as in Problem 3, assume the engine is operated at a stoichiometric mixture with pure iso-octane fuel having chemical formula C_8H_{18}. Assume further that the ratio of mass of water vapor to dry air is 0.05. Find the mass flow rate of dry air into the engine in slugs/sec.

6. For the example idle speed control system of this chapter, show that derive Eqn 47 using Eqns 41 and 46.

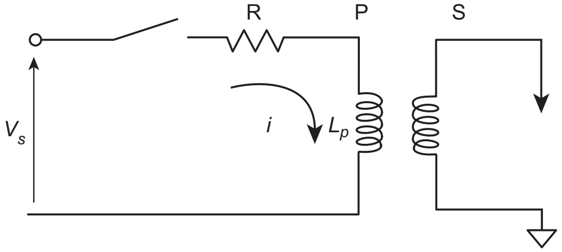
7. An engine is producing 50 HP and consuming 2.6 gal/hr of fuel while operating at stoichiometry.

   a) assume that gasoline has a density of 6 lb/gal and compute the brake specific fuel consumption.

   b) Find the mass flow rate of air into the engine.

8. A 4 cylinder port fuel injected engine having a displacement of 270 in^3 is operating at part throttle at a constant of 2400 RPM. The volumetric efficiency is 37% and the inlet air density is 1 d $\times$ 10^{-3} slug/ft^3. Each fuel injector can deliver 0.2 in^3/sec of fuel at the operating rad pressure when activated. Each fuel injector is operating in pulse mode. Find the duty cycle for the fuel injectors.

9. For the ignition coil circuit depicted below:



Find the current as a function of time during the dwell period. (NOTE: the circuit model during the spark interval is highly nonlinear and not susceptible to analyses using the theory covered in this book.) Assume the following:

$$V_s = 12.8 \text{ volt}$$

$$R = 2.2\ \Omega$$

$$L_p = .018 \text{ henry}$$

Chapter 6

1. Independently using the models given beginning with Eqn 6.3, derive Eqn 6.15.

2. A six cylinder engine operating at a steady load at 2200 RPM that has a piezoresistive intake MAP sensor such as is depicted in Figure 6.4. The pressure sensor is connected in a circuit with schematic shown in Figure 6.5a. With MAP (p) in English units (psi) the model for sensor element resistance R$_n$ ($n = 1$, 3) is given by:

$$\mathrm{R(p)} = 1000 + 1.4\ \mathrm{p}$$

The instantaneous MAP is given by:

$$P(t) = P_M + \delta p \sin\left(\frac{M_d.W_e\ t}{z}\right)$$

where P_M = time average MAP

δp = amplitude of MAP fluctuations due to dynamic pumping of intake mixture.

Let $P_M = .65\ P_o$

$P_o =$ sea level standard day barometer pressure

$\delta p = .1\ P_M$

$V_S = 5$ volt

$G_A = 10$

Find the instantaneous sensor circuit output voltage $V_o(t)$.

3. Given an engine that is equipped with a magnetic sensor mounted on the crankshaft such as depicted in Figure 6.7. There are M_d lugs on the disk. The magnetic circuit for this sensor is depicted in Figure 6.9 with lateral dimensions h_c and w_c as described in the text for this chapter. Let the magnetic flux linking each turn of the N turncoil be given by:

$$\Phi(\theta) = \Phi_o\left\{1 + \gamma\cos\left[\frac{2\pi\ (\theta_m - \theta)}{\Delta\theta_L}\right]\right\} \quad 1\theta_m - \theta 1 \leq \frac{\Delta\theta_2}{2}$$

$$= \Phi_o\ 1\theta_m - \theta 1 > \frac{\Delta\theta_2}{2}$$

Where $\theta =$ angular position of the crankshaft

$$\theta_M = \frac{2\pi m}{M_d} \quad m = 1,\ 2 \ldots M_d$$

$M_d =$ number of tabs on the dish

$\Delta\theta_L =$ angular width of each lug and is small compared to $2\pi/M_d$

a) Find the sensor open circuit terminal voltage $V_o(t)$

b) Find the values for θ for which

$$V_o(t) = 0$$

$$\text{and}\frac{dV_o(t)}{dt} \neq 0$$

c) Assume the existence of a zero crossing detector circuit and explain how this sensor can be used for crankshaft position measurements.

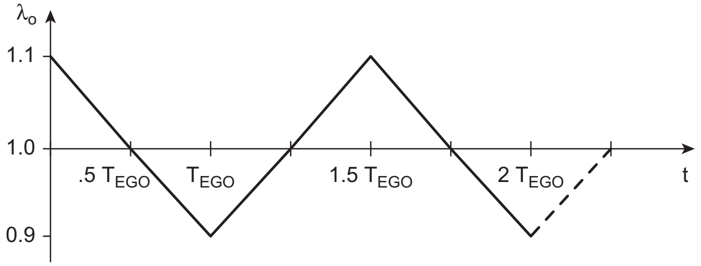
4. An optical crankshaft position sensor such as depicted in Figure 6.15a has four holes (as shown) in the disk. The radius of the circle of centers for the holes is 1.5 inch and each hole has a ¼ inch diameter. Assume that the fiber optical light pipe diameter is very small compared with the hole diameter and that its ends are located along the circle of hole centers.

   Let the optical power received by the phototransistor whenever a hole is present along the optical fibers be P_o. Develop a piece-wise continuous model for the output voltage (V_L) of the circuit of Figure 6.15b.

5. Using the model for thermistor resistance V_s temperature (Eqn 59) and Eqn 60 are correct, show that the equation for temperature T given in the text is correct.

6. An engine is operating at a steady load and angular speed. The engine control is in closed loop. Assume that the EGO sensor has ideal switching characteristics such as depicted in Figure 6.23 with the lean to rich transition at exhaust equivalence ratio (λ_o) of 1.000 and with rich to lean transition at $\lambda_o = 1.035$. Let the high output voltage be 1.00 volts and the low output be 0.10 volts.

Let $\lambda_o(t)$ be periodic with period T_{EGO} as shown in the figure below:



Sketch the waveform of the EGO sensor output voltage, clearly labeling all points.

7. Although the dynamic model for a solenoid actuator is nonlinear, as explained in the text, there are certain configurations for which the model for the moveable element motion (i.e. $x(t)$) can be linearized. One example of such a situation is the solenoid actuated vacuum actuator. For this configuration, the solenoid is designed such that the viscous damping force is the dominant mechanical force on the moveable element. With respect to the solenoid configuration of Figure 6.29, the approximate mechanical model for the moveable element is given by:

$$Fe = -B_s\left(\frac{dx}{dt}\right)^2$$

Furthermore, for this situation the velocity of the moveable element is such that the velocity term in the circuit portion of the model can be neglected such that this model becomes:

$$V_s = R_s i + \left(\frac{L_o}{1 + x_{o/g}}\right)\frac{di}{dt}$$

where X_o is the moveable element position against the upper mechanical stop.

Let the excitation voltage be a step such that

$$\begin{aligned} V_s(t) &= 0 \qquad t < 0 \\ &= V_s \qquad t \geq 0 \end{aligned}$$

Let T_s denote the time at which the moveable element reaches the center post such that

$$X(T_s) = 0$$

Show that the delocity of the moveable element (relative to the frame) is given by

$$\frac{dx}{dt} = -\frac{V\sqrt{L_o/(2\,g\,B_s)}}{R(1 + x/g)} \qquad 0 \leq x \leq x_o$$

Find the position of the moveable element $x(t)$ for $0 \leq t \leq T_s$ (corresponding to $0 \leq x \leq x_o$).

Chapter 7

1. A discrete time engine control is operating in a closed loop limit cycle mode of operation at a steady speed and load. Assume an ideal EGO sensor having a lean to rich transition at exhaust gas equivalence ratio $\lambda_o = 1$ and from rich to lean at $\lambda_o = 1.03$. For this steady operating condition, 1 λ_o is periodic and is given by

$$\lambda_o(b) = 1.05 \qquad k = 1,\ 2 \ldots 10$$

$$= 0.97 \qquad k = 11,\ 12 \ldots 20$$

$$\lambda_o(k+20) = \lambda_o(k)$$

$$\lambda_o(k) = \lambda_o(t_k)$$

where $t_k = kth$ sample

Sketch the waveform for the integral like term of the limit cycle control $I(k)$ assuming $I(0) = 0$.

2. Explain how the $I(k)$ term in the equation for fuel injector pulse duration ($I_F(k)$) for the limit cycle fuel control is analogous to the integral term in a PID control law.

3. An engine incorporates a MAF sensor (as explained in Chapter 6) in a discrete time control system. Find the algorithm for calculating $\dot{M}_{\mathrm{a}}(k)$ from the MAP sensor terminal voltage $v_o(k)$

Where $$\dot{M}_{\mathrm{a}}(k) = \dot{M}_{\mathrm{a}}(t_k)$$

$$\mathrm{v}_o(k) = \mathrm{v}_o(t_k)$$

$$t_n = k\text{th sample time}$$

4. Given an engine having a discrete time idle speed control system as depicted in Fig. 7.7, and having a plant continuous time model transfer function as given in Eqn 25 of Chapter 7.
   a) Complete the derivation of Eqn 35 for the discrete time forward transfer function $H_F(z)$.
   b) Complete the derivation of the closed loop transfer function $H_{CL}(z)$ of Eqn 37.
   c) For the example step change in command idle speed, find the discrete time system dynamic response numerical values y_k.

5. For the example VVP discrete time control system shown in Figure 7.10, complete the derivation of the closed loop control transfer function $H_{cl}(z)$ of Eqn 51.

6. An engine is equipped with a discrete time engine control having knock limited spark control. Assume that knock sensors as described in Chapter 6 are incorporated, sketch a complete block diagram for this type of ignition timing control and explain in detail the operation of your configuration.

7. A model for the ignition circuit was presented in Chapter 6. There it was shown that, during the dwell period, the coil current increases from 0 exponentially, asymptotically approaching a saturation current I_s where $I_s = V_s/(R_c + R_{on})$ where R_c is the coil resistance and R_{on} is the transistor emitter to collector resistance when the base current required to drive the transistor into saturation (see Chapter 3) is supplied by the engine control. Assume that it has been determined empirically that the coil current must reach 0.9 I_s to produce a spark of sufficient energy to reliably ignite the mixture. The dwell period is initiated by the engine control based upon measurement of crankshaft angular position. Let $R_c = 2.7\ \Omega$, $R_{on} = 1.5\ \Omega$ and coil inductance $L_c = 0.018$ henry.
   a) Find the coil angle relative to the angle at which spark is to occur for an engine that is operating at 5000 RPM to assure a reliable spark (*i.e.* dwell crankshaft angle to either radians or degrees).
   b) How does this dwell angle vary as a function to RPM?

8. A parallel hybrid vehicle as represented in Figure 7.28 is operating with electric propulsion from an induction motor. The motor is operating in the region for which the slope $\frac{\mathrm{d}T_e}{\mathrm{d}\omega_s} < 0$ and operating at a relatively small slip such that the approximate $T_e(s)$ is as given in this chapter for the small s case. Assume that the vehicle is being operated at a steady 60 MHP along a level curve. The vehicle weight is 2800 lb, the time rolling resistance

coefficient is 0.021, the local air density is 0.0021 slug/ft^3, the drag coefficient is 0.30, the reference area is 10 ft^2, the gear ratio from the motor to the wheel axis is 3.8, and there is no wind. The amplitude of the motor excitation voltage $V_s = 150$ v.

a) Find the motor excitation frequency such that the slip $s = 0.075$

b) The vehicle encounters a hill having a slope of 5° up. Assume that the vehicle is to continue at a steady state speed of 60 MPH along the hill slope. After a brief transient period which you are to ignore, find the amplitude at the excitation voltage such that the steady speed on the hill is maintained at 60 MPH.

Chapter 8

1. A vehicle, having a weight of 2850 lb and a manual transmission, is travelling a long road having a slope of +5° (i.e. upslope) at a steady speed of 55 MPH. Assume the following parameters:

$$\mu_r = 0.017 \text{ rolling resistance coefficient}$$
$$C_D = 0.31 = \text{drag coefficient}$$
$$S_{ref} = 21 \text{ ft}^2 = \text{reference area}$$

local air density = 0.0021 slug/ft3

gear ratio = 4.1

wheel effective radius $r_w = 1.2$ ft

Find the engine brake torque required to maintain this steady speed.

2. For the same vehicle as in Problem 1, find the linearized plant transfer function for the variation in vehicle speed (δV) for the vehicle on a level road (i.e. θ = 0)

$$H_p(s) = \frac{\delta V(s)}{u(s)}$$

3.

a) For the plant of Problem 2, find the closed loop transfer function $H_{CL}(s)$. For a PID control system in which the desired speed is denoted V$_d$:

$$H_{CL}(s) = \frac{V(s)}{V_d(s)}$$
$$K_p = 10$$
$$K_I = 50$$
$$K_D = 25$$

b) Is the closed loop system stable for these control gains?

4. For the digital speed control system depicted in the block diagram of Figure 8.4, derive equations 18 and 24 filling in the missing step 5.

5. For the vacuum operated cruise control throttle actuator of Figure 8.8, assume the following: barometric pressure = 14.0 psi, manifold pressure is 50% of barometric pressure, the actuator piston area is 1.7 in^2, the

throttle angle (θ_t)in radians is 0.4 times the force on the piston (i.e. force = pressure × area). The maximum throttle angle deflection (from closed throttle) is to be 60°. Find the throttle angle vs control signal duty cycle δp.

6. For a vehicle with a digital advanced cruise control (i.e. one with automatic braking for maintaining vehicle speed or long steep downgrades) develop a block diagram for the system when the vehicle is on this downgrade. Show all components of the controller and assume that any required D/A conversion is via zero order hold.

7. For the ACC of Problem 6, let the system parameters be the same as those used in the example presented with respect to Figure 8.2 and assume $K_B = 15$; find the closed loop z-transfer function $H_{CL}(z)$ of the form given in Eqn 24.

8. A vehicle having a suspension system as depicted in Figure 8.21 is travelling a long, straight, horizontal road at 70 MPH. Let the time origin be such that the braking force F is given by

$$\begin{aligned} F &= 0 & t &< 0 \\ &= F_o & t &\geq 0 \end{aligned}$$

   a) Find the dynamic pitch angle $\alpha_{p(t)}$ for variable damping ratio Z assuming the natural frequency $\omega_A = 7.9$ rad/sec for a spring rate $K = 2 \times 10^4$ lb/ft. The parameter Z should appear in the answer.

   b) Explain how electronically controlled strut damping affects the dynamic response of $\alpha_{p(t)}$.

9. Given a vehicle which is equipped with an electronic suspension having a quarter car model (QCM) as depicted in Figure 8.24:

   a) show that Eqn 89 is correct for the transfer function $H_a(s)$ and that Eqn 91 is correct for transfer function $H_H(s)$.

   b) Find the sinusoidal frequency response $H_H(j\omega)$ for this QCM using the parameters given on p. 436.

10. For the 4WS system depicted in Figure 8.32, compute the matrices A and B for the standard form state variable equation:

$$\dot{x} = Ax + Bu$$

Use the exemplary parameters given on p. 455.

Chapter 9

1. An automotive instrumentation system incorporates a multiplex input along with an 8-bit A/D converter, as shown in Figure 9.5. The A/D is designed such that all output bits are 0's for 0 volt input and all 1's for 5 volt analog input.

   a) Assuming that the conversion from analog to digital is perfectly linear, find the voltage resolution for the A/D in volts per bit. That is, what is the input voltage change for a 1-bit change in the A/D output.

   b) Assume that the A/D converter rounds to the nearest 8-bit binary number for the conversion of a continuous analog input, find the A/D output for the following set of voltages: 2.43 volts; 3.17 volts; 4.05 volts; 0.09 volts.

2. The multiplexed input of Problem 1 is to be used with $N = 10$ sensors. If each input variable is to be measured at a rate of 100 samples/sec, what is the maximum conversion time for the A/D if the system is to read each

sensor during an input cycle and remain synchronous, and to sample periodically with no wait time between successive cycles?

3. A galvanometer such as is depicted in Figure 9.10 is to be used to display vehicle speed. Assume that the inertial term in its dynamic model is negligible, that the electrical drive circuit is as depicted in Figure 9.11, and that v_s and R_s are the open circuit voltage and source impedance, respectively, of the analog output of the digital instrumentation system.

   a) Find the transfer function $H_g(s)$ for this display:

   $$H_g(s) = \frac{\theta(s)}{\mathrm{v}(s)}$$

   In deriving this transfer function, use the same symbols as in the text for all relevant parameters.

   b) Find the poles of the transfer function in terms of the given parameter notation.

4. An automotive flat panel display consists of an array of 128 rows and 256 columns of LED elements. It is operated in a raster scan pattern with a cycle frequency (or refresh rate) of 50 Hz. An element in the 57th row and 101st column is to be activated, so find the time from the beginning of the raster cycle at which this element is activated.

5. An automotive fuel quantity instrument is to be implemented with a potentiometer sensor such as is depicted in Figure 9.24 where the sensor voltage $\mathrm{V_o}(F)$ is given by:

   $$\mathrm{V_o} = 5\sqrt{F} \qquad 0 \leq F \leq 1$$

   The block diagram for this fuel measuring instrument is depicted in Figure 9.25. The signal processing block (LPF) is a second-order Butterworth filter having a z-transfer function given by Eqn 29.

   a) Find the output y_n of the LPF where the input x_n is given by:

   $$x = \mathrm{V_o}(t_n)$$
   $$t_n = n\,T$$
   $$T = 0.001 \text{ sec}$$

   b) Using the methods of Chapter 2, show that Eqn 29 is the correct z-transform for a second-order Butterworth digital filter having normalized corner frequency $\Omega_c = 0.001$.

6. Show that Eqn 28 is the correct z-transfer function for the display whose continuous time model has transfer function $H_D(s)$ given by Eqn 25 assuming sample period $T = 0.001$ sec, $K_D = 0.5$, and $s_0 = 0.5$ rad/sec.

7. A digital speed instrument has a block diagram as depicted in Figure 9.32. The speed sensor generates a sequence of pulses of frequency F_p for vehicles speed V (in ft/sec) given by:

   $$F_p = 100\,V$$

   The electronic gate is closed for a period $t = 10$ msec for each sample. At sample time t_k the vehicle is travelling at 72 MPH. Find the contents of the binary counter P (expressed in binary) for the period:

   $$t_k + \tau \leq t \leq t_{k+1}$$

8. An advanced vehicle navigation system combines the measurement of vehicle speed V and fuel consumption rate with GPS navigation and a detailed electronic road map. The electronic map contains all distances along each road segment for any given trip as well as legal speed limits for each segment. Assuming that the time

intervals for any unplanned stop (e.g. traffic congestion, traffic signal lights, etc.) are negligible and that the vehicle travels at exactly the legal speed limit:

a) explain how such a system could be used to estimate the total fuel consumption

b) let the fuel consumption rate $(\dot{\mathrm{f}})$ be a polenomial function of vehicle speed V of the form:

$$\dot{\mathrm{f}} = \sum_{m=0}^{M} \mathrm{a}_m V^m$$

Let a trip be planned that consists of N segments each having legal speed limit V_n and distance D_n $(n = 1, 2 \ldots N)$.

c) Find the estimated total fuel consumption for (F) for this trip.

9. For the trip of Problem 8, the actual vehicle speed can vary by ±5% of the legal speed limit. Let f_T be the true fuel consumption for this same trip. Find an expression for the maximum error in fuel consumption E_F:

$$E_F = F - F_T$$

Find the upper and lower limits for E_F.

Chapter 10

1. Explain in detail how a model-based diagnosis of problems with a mass air flow (MAF) sensor is possible if sensors are incorporated for measuring manifold pressure (p_m) and intake air temperature (T_i). What additional sensor(s) for this diagnosis and what tabulated data (taken during engine mapping) are required for the engine involved.

2. For Problem 1, assume that a MAF sensor as depicted in Figure 6.2, a piezoresistive MAP sensor as depicted in Figure 6.4, and a thermistor intake air temperature sensor as depicted in Figure 6.21, are employed in the engine control system. Assume that any additional sensor required for the MAF sensor model based diagnosis are available and that any required tabulated data is stored in ROM.

   Using the models developed for each necessary sensor as given in Chapters 6 and 7, develop an expression for the relationship between the terminal voltages of the MAF, MAP and T_i sensors for a properly operating MAF sensor.

3. A vehicle is travelling along a straight, level road in cruise control mode. The cruise control is a digital system incorporating a vehicle speed sensor (VSS) as shown in Figure 8.6. The VSS disk is driven by the transmission output shaft. The disk was made with M = 25 lugs. At some time during the trip, one of the lugs becomes detached owing to a manufacturing defect and is suddenly missing:

   a) Assuming all other vehicle components function without a failure, describe the change in sensor output signal due to this failure.

   b) What is the effect on vehicle speed?

   c) Describe an onboard diagnostic system that could detect this VSS failure. You may make the following simplifying assumptions:

      i) the transmission maintains a unity gear ratio.

      ii) the torque converter of the automatic transmission is in, and remains in, lock-up mode.

4. A thermistor coolant temperature sensor such as is depicted in Figure 6.20, and connected in a circuit configuration of Figure 6.21, experiences a calibration change. The calibration change results in an indicated temperature that is 25 °C cooler than actual temperature. While operating on a very hot day in dense urban traffic, the engine overheats significantly enough that a coolant hose fails and the vehicle must be towed for repairs.

a) Explain the failure mechanism.

b) Describe a diagnostic procedure by which this failure can be identified and proper repairs made.

Assume that both portable and service bay diagnostic tools are available.

c) Develop a flow chart for the diagnostic procedure.

NOTE: The solution to this problem is not necessarily unique.

5. The digital control system of a 4-cylinder engine is equipped with a misfire detection system based upon the nonuniformity index method of the present chapter. For the sake of simplifying calculations, assume that the net torque T_n during any given engine cycle (i.e. $0 \leq \theta_e \leq 4\,\pi$) is given by:

$$T_n(\theta_e) = T_o + \delta T_m \sin(2m\theta_e) \qquad\qquad m = 1,\ 2,\ 3,\ 4$$

Where $\theta_e =$ crankshaft angular position.

a) Show that for perfect combustion (i.e. no misfires) and δT_m is the same for all 4 cylinders, the non = uniformity vector $\bar{n}$ is an eight-dimensional vector with all elements exactly zero.

b) Let there be a partial misfire in cylinder 3 due to fuel injector calibration failure that is continuous such that δT_m given below is the same for all engine cycles:

$$\begin{aligned} \delta T_m &= 0.6\ T_o \quad m = 1,\ 2,\ 4 \\ &= 0.4\ T_o \quad m = 3 \end{aligned}$$

Find the nonuniformity vector $\bar{n}$ and the nonuniformity index $\mathrm{n} = ||\ \bar{n}\ ||_1$.

6. If a cylinder other than $m = 3$ were the only misfiring cylinder, explain how the nonuniformity vector $\bar{n}$ would change and how this could lead to a method of identifying the misfiring cylinder.

7. A movable mass, switch type crash sensor such as depicted in Figure 10.18 is to be employed in an airbag system. The deceleration of a vehicle during a crash is never a constant (e.g. see Figure 10.20). However, part of the specifications for this sensor call for the switch activation to be tested under a steady acceleration obtained by spinning the sensor on a centrifugal table. The specifications are that the switch only be closed for acceleration exceeding a threshold value (a_T) as given below:

$$\begin{aligned} &(\text{open switch})\ \mathrm{S} = 0 \qquad a < a_T \\ &(\text{closed switch})\ \mathrm{S} = 1 \qquad a \geq a_T \end{aligned}$$

a) Find the relationship between the mass M of the moveable element and the spring rate K to meet the above criteria.

b) For the sensor of part a, assume that $M = 8$ oz, $F_c = 6$ oz, $x_p = ½$ inch, and let the damping ratio for the second order system Z = 0.7 (see Chapter 1). If the threshold steady acceleration $a_T = 2.7$ g, find the response time for switch closure.

NOTE: A solution to this problem can be found either analytically via the methods presented in Chapter 1, or via computer simulation (e.g. using MATLAB/SIMULINK).

Index

Note: Page numbers with "*f*" denote figures; "*t*" tables.

This page intentionally left blank