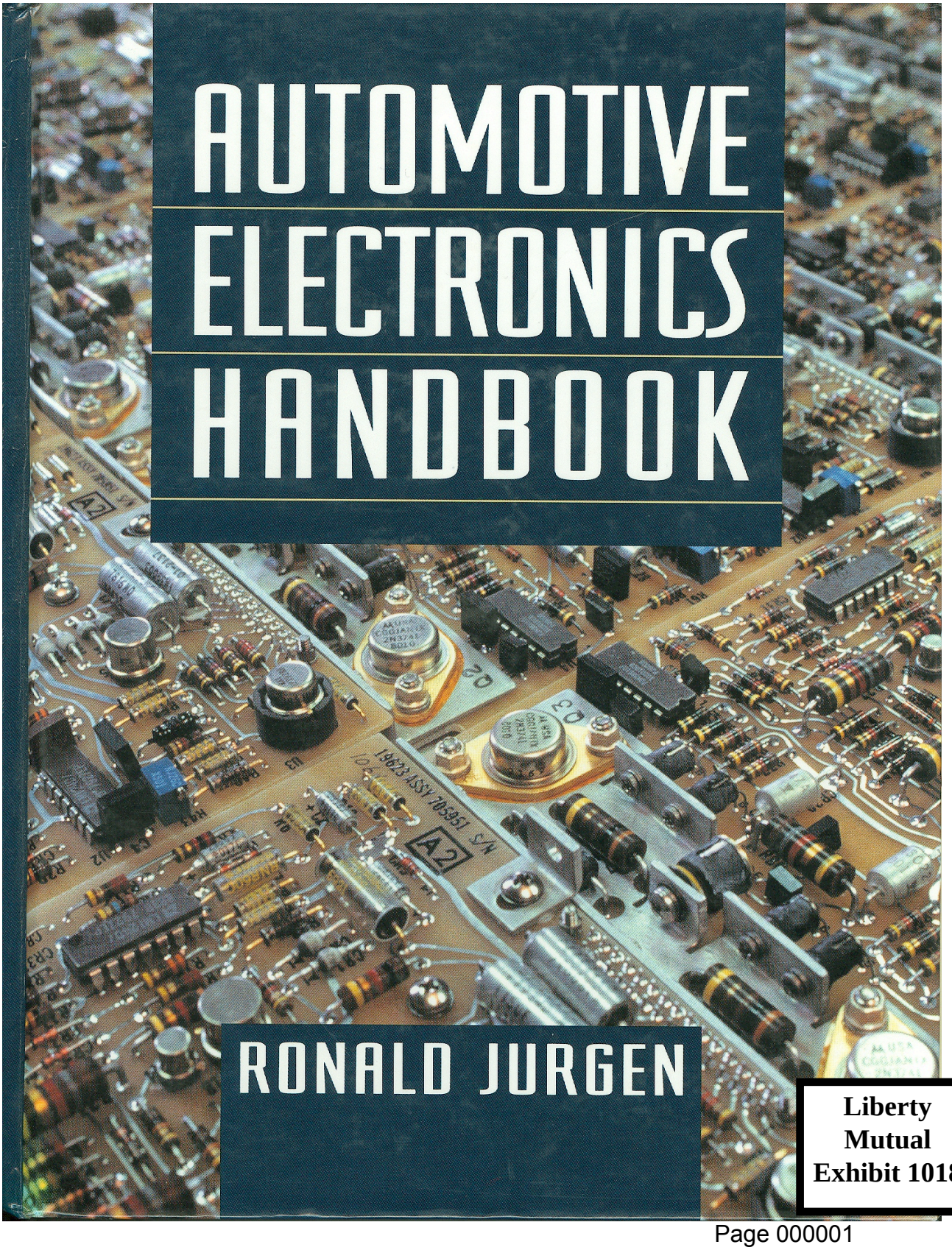

AUTOMOTIVE ELECTRONICS HANDBOOK

RONALD JURGEN

AUTOMOTIVE ELECTRONICS HANDBOOK

Ronald K. Jurgen Editor in Chief

McGraw-Hill, Inc.

New York San Francisco Washington, D.C. Auckland Bogotá
Caracas Lisbon London Madrid Mexico City Milan
Montreal New Delhi San Juan Singapore
Sydney Tokyo Toronto

Library of Congress Cataloging-in-Publication Data

Automotive electronics handbook / Ronald Jurgen, editor in chief.
p. cm.
Includes index.
ISBN 0-07-033189-8
1. Automobiles—Electronic equipment. I. Jurgen, Ronald K.
TL272.5.A982 1994
629.25'49—dc 94-39724
CIP

1 2 3 4 5 6 7 8 9 0 AGM/AGM 9 0 9 8 7 6 5 4

ISBN 0-07-033189-8

The sponsoring editor for this book was Stephen S. Chapman, the editing supervisor was Virginia Carroll, and the production supervisor was Suzanne W. B. Rapcavage. It was set in Times Roman by North Market Street Graphics.

Printed and bound by Arcata Graphics/Martinsburg.

This book is printed on acid-free paper.

Ambient temperature under bias (TA) refers to the temperature range that the microcontroller is guaranteed to operate at within a given application. While powered-up or operating, a microcontroller must not be subjected to temperatures that exceed its specified ambient temperature range. The most common ambient temperature ranges in industry are:

Commercial	0 to +70 °C
Extended	−40 to +85 °C
Automotive	−40 to +125 °C

11.2 MEMORY

Microcontrollers execute customized programs that are written by the user. These programs are stored in either on-chip or off-chip memory and are often referred to as the *user's code*. On-chip memory is actually integrated onto the same piece of silicon as the microcontroller and is accessed over the internal data bus. Off-chip memory exists on a separately packaged piece of silicon and is typically accessed by the microcontroller over an external address/data bus.

A memory map shows how memory addresses are arranged in a particular microcontroller. Figure 11.19 shows a typical microcontroller memory map.

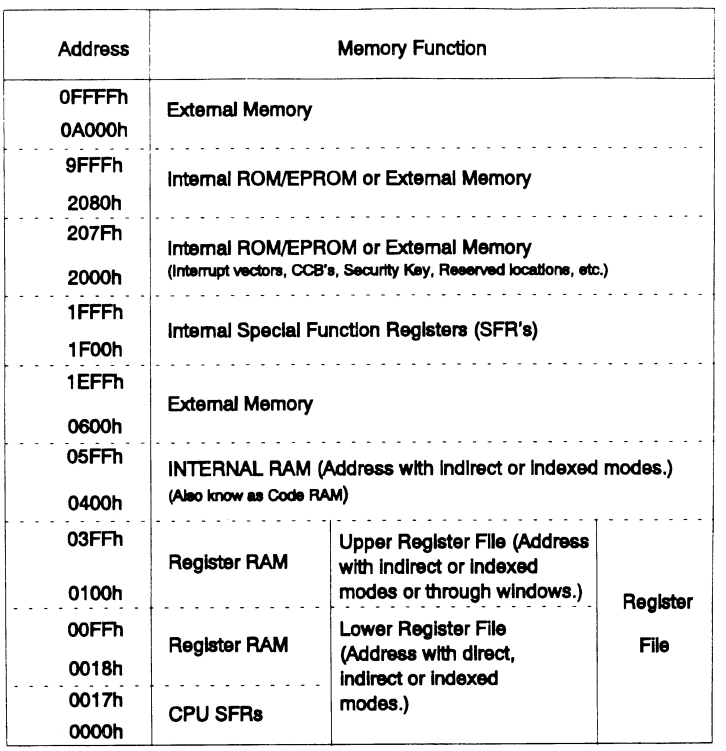

FIGURE 11.19 Microcontroller memory map.

Memory is commonly referred to in terms of Kbytes of memory. One Kbyte is defined as 1024 bytes of data. Memory is most commonly arranged in bytes which consist of 8 bits of data. For instance, a common automotive EPROM is referred to as a "256k × 8 EPROM". This EPROM contains 256-Kbytes 8-bit memory locations or 2,097,152 bits of information.

11.2.1 On-Chip Memory

On-chip microcontroller memory consists of some mix of five basic types: random access memory (RAM), read-only memory (ROM), erasable ROM (EPROM), electrically erasable ROM (EPROM), and flash memory. RAM is typically utilized for run-time variable storage and SFRs. The various types of ROM are generally used for code storage and fixed data tables.

The advantages of on-chip memory are numerous, especially for automotive applications, which are very size and cost conscious. Utilizing on-chip memory eliminates the need for external memory and the "glue" logic necessary to implement an address/data bus system. External memory systems are also notorious generators of switching noise and RFI due to their high clock rates and fast switching times. Providing sufficient on-chip memory helps to greatly reduce these concerns.

RAM. RAM may be defined as memory that has both read and write capabilities so that the stored information can be retrieved (read) and changed by applying new information to the cell (write). RAM found on microcontrollers is that of the static type that uses transistor cells connected as flip-flops. A typical six-transistor CMOS RAM cell is shown in Fig. 11.20. It consists of two cross-coupled CMOS inverters to store the data and two transmission gates, which provide the data path into or out of the cell. The most significant characteristic of static memory is that it loses its memory contents once power is removed. After power is removed, and once it is reapplied, static microcontroller RAM locations will revert to their default state of a logic "0". Because of the number of transistors used to construct a single cell, RAM memory is typically larger per bit than EPROM or ROM memory.

Although code typically cannot be executed from register RAM, a special type of RAM often referred to as *code RAM* is useful for downloading small segments of executable code. The difference between code and register RAM is that code RAM can be accessed via the

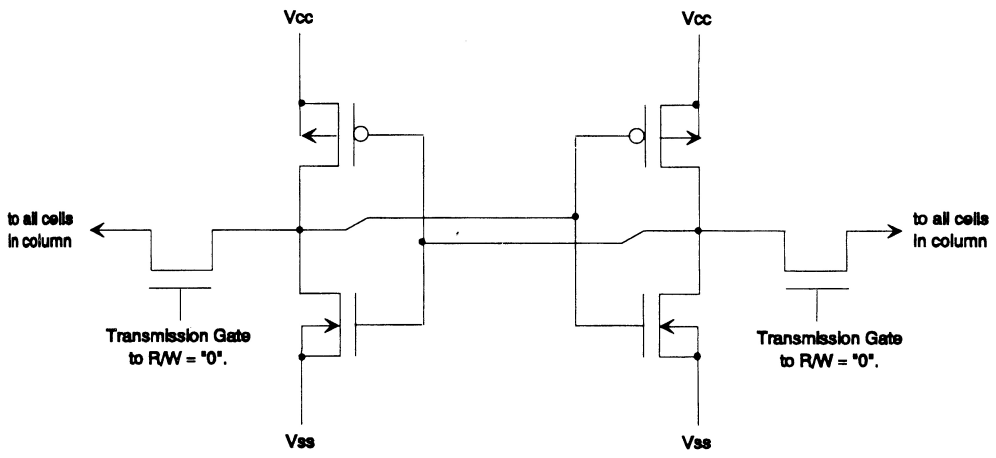

FIGURE 11.20 CMOS RAM memory cell.

memory controller, thus allowing code to be executed from it. Code RAM is especially useful for end-of-line testing during ECU manufacturing by allowing test code to be downloaded via the serial port peripheral.

ROM. Read-only memory (ROM), as the name implies, is memory that can be read but not written to. ROM is used for storage of user code or data that does not change since it is a non-volatile memory that retains its contents after power is removed. Code or data is either entered during the manufacturing process (masked ROM, or MROM) or by later programming (programmable ROM, or PROM); either way, once entered it is unalterable.

A ROM cell by itself (Fig. 11.21) is nothing more than a transistor. ROM cells must be used in a matrix of word and bit lines (as shown in Fig. 11.22) in order to store information. The word lines are connected to the address decoder and the bit lines are connected to output buffers. The user's code is permanently stored by including or omitting individual cells at word and bit line junctions within the ROM array. For MROMs, this is done during wafer fabrication. For PROMs, this is done by blowing a fuse in the source/drain connection of each cell. To read an address within the array, the address decoder applies the address to the memory matrix. For any given intersection of a word and bit line, the absence of a cell transistor allows no current to flow and causes the transistor to be off. This indicates an unprogrammed ROM cell. The presence of a complete cell conducts and is sensed as a logical "0", indicating a programmed cell. The stored data on the bit lines is then driven to the output buffers.

MROMs are typically used for applications whose code is stable and in volume production. After the development process is complete and the user's program has been verified, the user submits the ROM code to the microcontroller manufacturer. The microcontroller manufacturer then produces a mask that is used during manufacturing to permanently embed the program within the microcontroller. This mask layer either enables or disables individual ROM cells at the junctions of the word and bit lines. An advantage of MROM microcontrollers is that they come with user code embedded, which saves time and money since post-production programming is not necessary. A disadvantage of MROM devices is that, since the mask with the user code has to be supplied early in the manufacturing process, throughput time (TPT) is longer.

Some versions of ROM (such as Intel's Quick-ROM) are actually not ROMs, but rather EPROMs, which are programmed at the factory. These devices are packaged in plastic devices, which prevents them from being erased since ultraviolet light cannot be applied to the actual EPROM array. Throughput time for QROMs is faster since the user code isn't required until after the actual manufacturing of the microcontroller is complete. As with

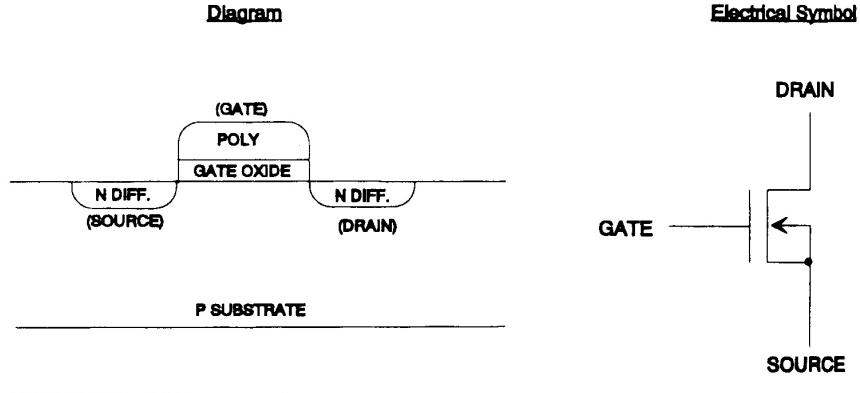

FIGURE 11.21 ROM memory cell.

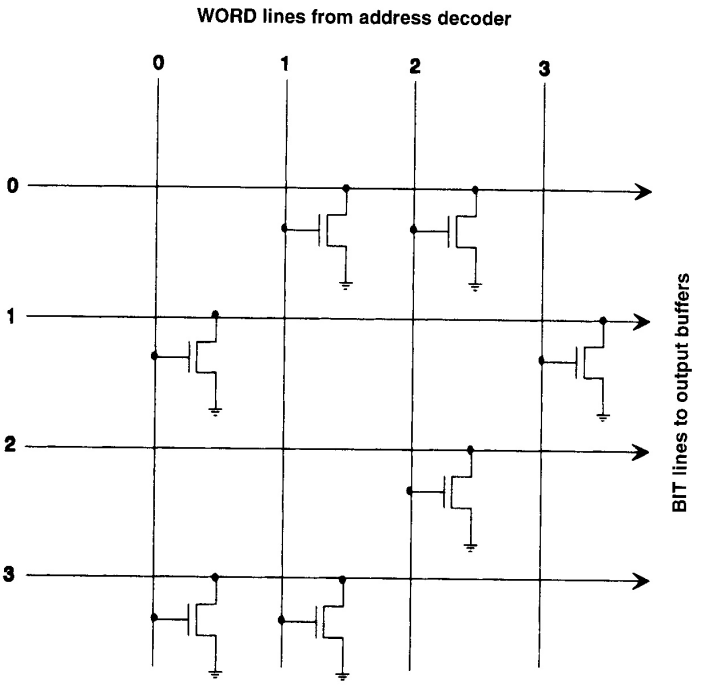

FIGURE 11.22 Simplified ROM memory matrix.

MROMs, the user supplies the ROM code to the microcontroller manufacturer. Instead of creating a mask with the ROM code, the manufacturer programs it into the device just prior to final test.

EPROM. EPROM devices are typically used during application development since this is when user code is changed often. EPROMs are delivered to the user unprogrammed. This allows the user to program the code into memory just prior to installation into an ECU module. Many EPROM microcontrollers actually provide a mechanism for in-module programming. This feature allows the user to program the device via the serial port while it is installed in the module. EPROM devices come assembled in packages either with or without a transparent window. Windowed devices are true EPROM devices that allow the user to erase the memory contents by exposing the EPROM array to ultraviolet light. These devices may be reprogrammed over and over again and thus are ideally suited for system development and debug during which code is changed often. EPROM devices assembled in a package without a window are commonly referred to as *one-time programmable devices* or OTPs. OTPs may only be programmed once, since the absence of a transparent window prevents UV erasure. OTPs are suited for limited production validation (PV) builds in which the code will not be erased.

A typical EPROM cell is shown in Fig. 11.23. It is basically an N-channel transistor that has an added poly1 floating gate to store charge. This floating gate is not connected and is surrounded by insulating oxide that prevents electron flow. The mechanism used to program an EPROM cell is known as *hot electron injection.* Hot electron injection occurs when very high drain (9-V) and select gate (12-V) voltages are applied. This gives the negatively charged electrons enough energy to surmount the oxide barrier and allows them to be stored on the gate.

This has the same effect as a negative applied gate voltage and turns the transistor off. When the cell is unprogrammed, it can be turned on like a normal transistor by applying 5 V to the poly2 select gate. When it is programmed, the 5 V will not turn on the cell. The state of the cell is determined by attempting to turn on the cell and detecting if it turns on. Erasure is performed through the application of ultraviolet (UV) light, which gives just the right amount of energy necessary for negatively charged electrons to surmount the oxide barrier and leave the floating gate.

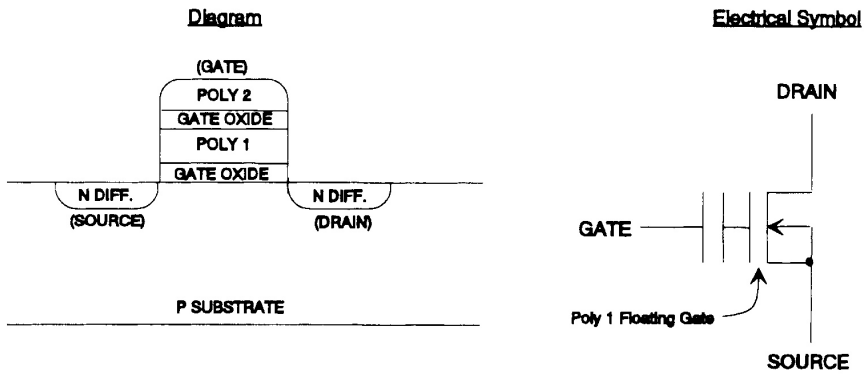

FIGURE 11.23 EPROM memory cell.

Flash. Flash memory is the newest nonvolatile memory technology and is very similar to EPROM. The key difference is that flash memory can be electrically erased. Once programmed, flash memory contents remain intact until an erase cycle is initiated via software. Like EEPROM, flash memory requires a programming and erase voltage of approximately 12.0 V. Since a clean, regulated 12-V reference is not readily available in automotive environments, this need is often provided for through the incorporation of an on-chip charge pump. The charge pump produces the voltage and current necessary for programming and erasure from the standard 5-V supply voltage. The advantage of flash is in its capability to be programmed *and* erased in-module without having to be removed. In-module reprogrammability is desirable since in-vehicle validation testing doesn't always allow for easy access to the microcontroller. Flash also allows for last-minute code changes, data table upgrades, and general code customization during ECU assembly. Since a flash cell is nearly identical in size to that of an EPROM cell, the high reliability and high device density capable with EPROM is retained. The main disadvantage of flash is the need for an on-chip charge pump and special program and erase circuitry, which adds cost.

A flash memory cell is essentially the same as an EPROM cell, with the exception of the floating gate. The difference is a thin oxide layer which allows the cell to be electrically erased. The mechanism used to erase data is known as *Fowler-Nordheim tunneling,* which allows the charge to be transferred from the floating gate when a large enough field is created. Hot electron injection is the mechanism used to program a cell, exactly as is done with EPROM cells. When the floating gate is positively charged, the cell will read a "1", when negatively charged, the cell will read a "0".

EEPROM. EEPROM (electrically erasable and programmable ROM, commonly referred to as E^2ROM) is a ROM that can be electrically erased and programmed. Once programmed, EEPROM contents remain intact until an erase cycle is initiated via software. Like flash, programming and erase voltages of approximately 12 V are required. Since a clean, regulated 12-V reference is not readily available in automotive environments, this requirement is satisfied using an on-chip charge pump as is done for flash memory arrays. Like flash, the advantage of EEPROM is its

capability to be programmed and erased in-module. This allows the user to erase and program the device in the module without having to remove it. EEPROM's most significant disadvantage is the need for an on-chip charge pump. Special program and erase circuitry also adds cost.

An EEPROM cell is essentially the same as an EPROM cell with the exception of the floating gate being isolated by a thin oxide layer. The main difference from flash is that Fowler-Nordheim electron tunneling is used for *both* programming and erasure. This mechanism allows charge to be transferred to or from the floating gate (depending upon the polarity of the field) when a large enough field is created. When the floating gate is positively charged, the cell will read a "1"; when negatively charged, the cell will read a "0".

11.2.2 Off-Chip Memory

Off-chip memory offers the most flexibility to the system designer, but at a price; it takes up additional PCB real estate as well as additional I/O pins. In cost- and size-conscience applications, such as automotive ABS, system designers almost exclusively use on-chip memory. However, when memory requirements grow to sizes in excess of what is offered on-chip (such as is common in electronic engine control), the system designer must implement an off-chip memory system. Off-chip memory is flexible because the user can implement various memory devices in the configuration of his choice. Most microcontrollers on the market today offer a wide variety of control pins and timing modes to allow the system designer flexibility when interfacing to a wide range of external memory systems.

Accessing External Memory. If circuit designers must use external memory in their applications, the type of external address/data bus incorporated onto the microcontroller should be considered. If external memory is not used, this will have, if any, impact upon the application. There are two basic types of interfaces used in external memory systems. Both of these are parallel interfaces in which bits of data are moved in a parallel fashion and are referred to as *multiplexed* and *demultiplexed* address/data buses.

Multiplexed Address/Data Buses. As the name implies, multiplexed address/data buses allow the address as well as the data to be passed over the same microcontroller pins by multiplexing the two in time. Figure 11.24 illustrates a typical multiplexed 16-bit address/data bus system as is implemented with Intel's 8XC196Kx family of microcontrollers.

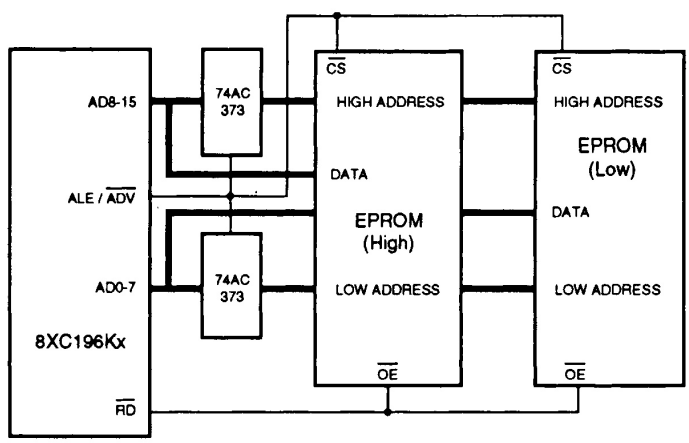

FIGURE 11.24 Multiplexed address/data bus system.