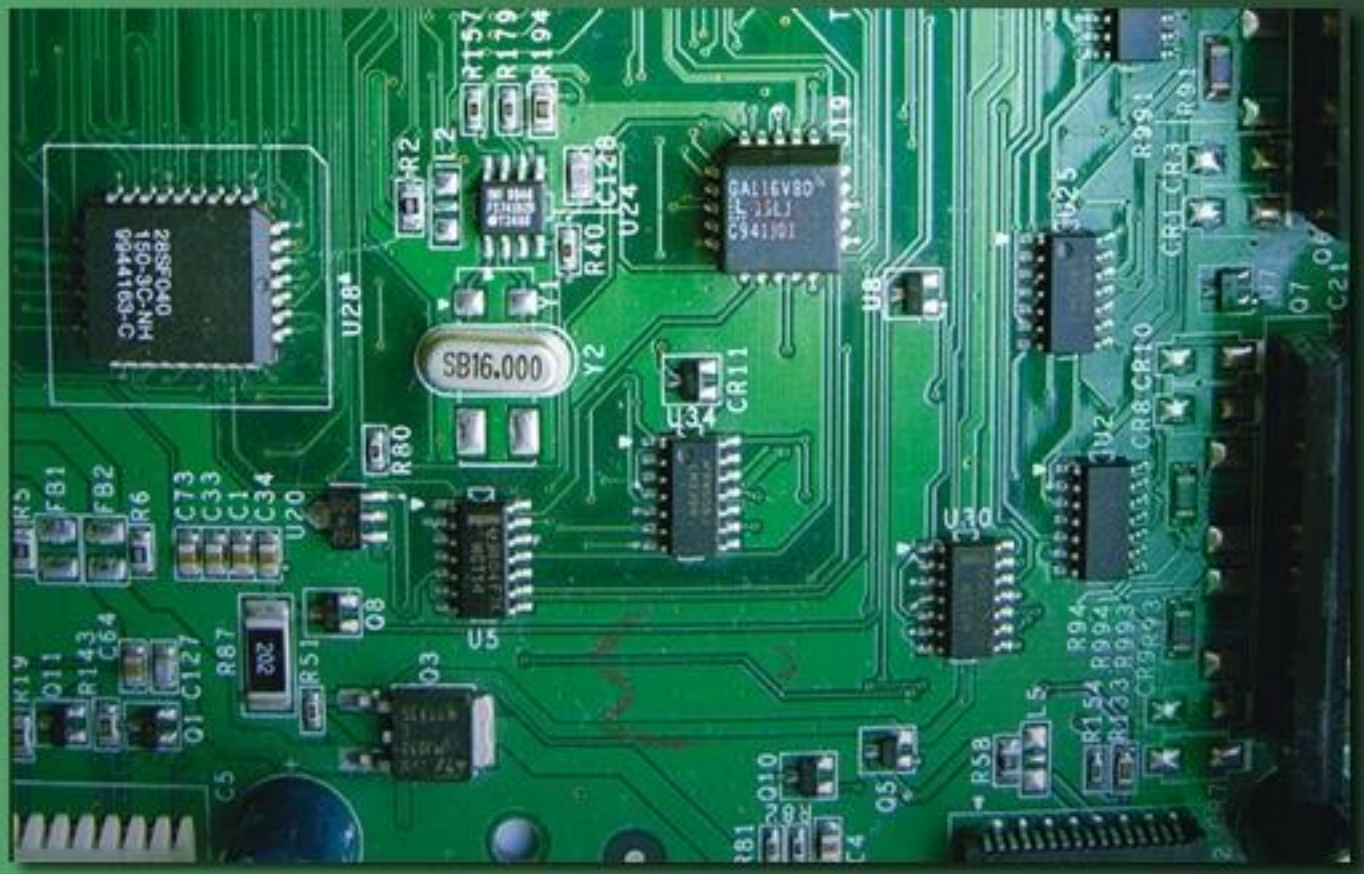
The Intel Microprocessors

8086/8088, 80186/80188, 80286, 80386, 80486 Pentium, Pentium Pro Processor, Pentium II, Pentium 4, and Core2 with 64-bit Extensions

Architecture, Programming, and Interfacing



EIGHTH EDITION

Barry B. Brey

PEARSON

Chapter 10: Memory Interface

Introduction

- Simple or complex, every microprocessor-based system has a memory system.
- Almost all systems contain two main types of memory: **read-only memory** (ROM) and **random access memory** (RAM) or read/write memory.
- This chapter explains how to interface both memory types to the Intel family of microprocessors.

Chapter Objectives

Upon completion of this chapter, you will be able to:

- Decode the memory address and use the outputs of the decoder to select various memory components.
- Use programmable logic devices (PLDs) to decode memory addresses.
- Explain how to interface both RAM and ROM to a microprocessor.

Chapter Objectives (*cont.*)

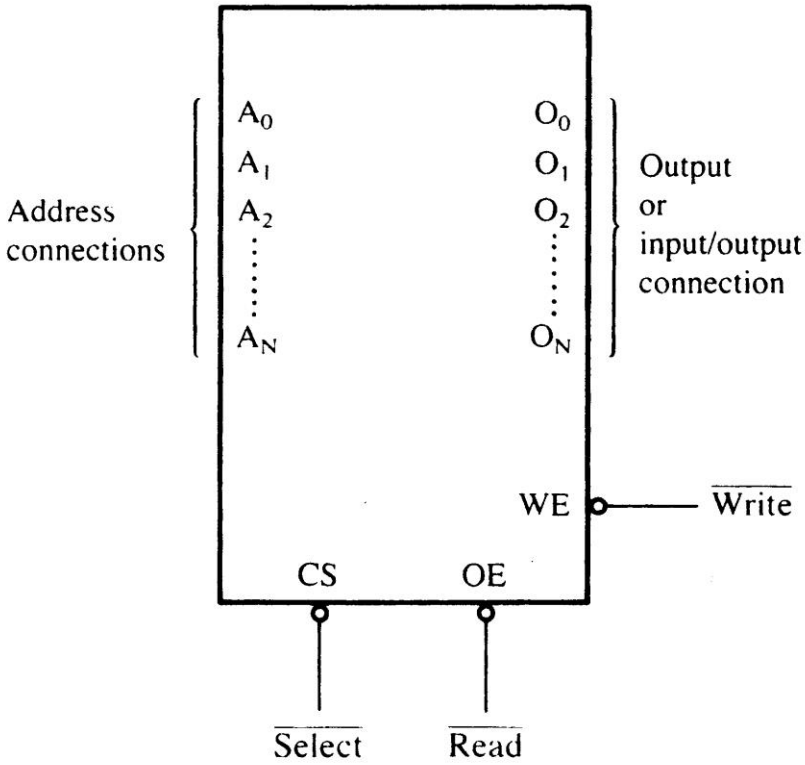
Upon completion of this chapter, you will be able to:

- Explain how error correction code (ECC) is used with memory.
- Interface memory to an 8-, 16-, 32-, and 64-bit data bus.

10–1 MEMORY DEVICES

- Before attempting to interface memory to the microprocessor, it is essential to understand the operation of memory components.
- In this section, we explain functions of the four common types of memory:
  - Read-only memory (ROM)
  - Flash memory (EEPROM)
  - Static random access memory (SRAM)
  - Dynamic random access memory (DRAM)

Memory Pin Connections



- address inputs
- data outputs or input/outputs
- some type of selection input
- at least one control input to select a read or write operation

Figure 10–1 A pseudo memory component illustrating the address, data, and control connections.

Address Connections

- Memory devices have address inputs to select a memory location within the device.
- Almost always labeled from A_0, the least significant address input, to A_n
  - where subscript n can be any value
  - always labeled as one less than total number of address pins
- A memory device with 10 address pins has its address pins labeled from A_0 to A_9.

Address Space (Main Memory: RAM)

- Address bus:10 bit →Address Space:1 Kbytes (2^{10})
- Address bus:11 bit →Address Space:2 Kbytes (2^{11})
- Address bus:16 bit →Address Space:64 KBytes (2^{16})
- Address bus:20 bit →Address Space:1 MBytes
  Address bus:32 bit →Address Space:4 GBytes
  Address bus:34 bit →Address Space:16GBytes
- Address bus:36 bit →Address Space:64GBytes
- Address bus:38 bit →Address Space:256GBytes
- Address bus:52 bit →Address Space:10^{15} Bytes

- The number of address pins on a memory device is determined by the number of memory locations found within it.
- A 1K memory device has 10 address pins.
  - therefore, 10 address inputs are required to select any of its 1024 memory locations

Data Connections

- All memory devices have a set of data outputs or input/outputs.
  - today, many devices have bidirectional common I/O pins
  - data connections are points at which data are entered for storage or extracted for reading
- Data pins on memory devices are labeled D_0 through D_7 for an 8-bit-wide memory device.

Selection Connections

- Each memory device has an input that selects or enables the memory device.
  - sometimes more than one
- This type of input is most often called a **chip select** ($\overline{CS}$) **chip enable** ($\overline{CE}$) or simply **select** ($\overline{S}$) input.

Control Connections

- All memory devices have some form of control input or inputs.
  - ROM usually has one control input, while RAM often has one or two control inputs
- Control input often found on ROM is the **output enable** or **gate** connection, which allows data flow from output data pins.
- The $\overline{OE}$ connection enables and disables a set of three-state buffers located in the device and must be active to read data.

- RAM has either one or two control inputs.
  - if one control input, it is often called R/$\overline{W}$
- If the RAM has two control inputs, they are usually labeled $\overline{WE}$ (or $\overline{W}$), and $\overline{OE}$ (or $\overline{G}$).
  - **write enable** must be active to perform memory write, and $\overline{OE}$ active to perform a memory read
  - when the two controls are present, they must never both be active at the same time
- If both inputs are inactive, data are neither written nor read.
  - the connections are at their high-impedance state

ROM Memory

- **Read-only memory** (ROM) permanently stores programs/data resident to the system.
  - and must not change when power disconnected
- Often called **nonvolatile memory**, because its contents *do not* change even if power is disconnected.
- programmed during fabrication at the factory

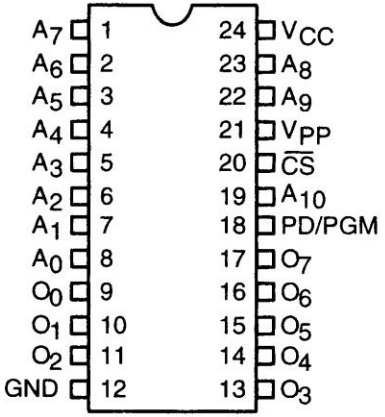
- The EPROM (**erasable programmable read-only memory**) is commonly used when software must be changed often.
  - or when low demand makes ROM uneconomical
  - for ROM to be practical at least 10,000 devices must be sold to recoup factory charges
- An EPROM is programmed in the field on a device called an EPROM programmer.
- Also erasable if exposed to high-intensity ultraviolet light.
  - depending on the type of EPROM

- PROM memory devices are also available, although they are not as common today.
- The PROM (**programmable read-only memory**) is also programmed in the field by burning open tiny NI-chrome or silicon oxide fuses.
- Once it is programmed, it cannot be erased.

- A newer type of **read-mostly memory** (RMM) is called the **flash memory**.
  - also often called an EEPROM (**electrically erasable programmable ROM**)
  - EAROM (**electrically alterable ROM**)
  - or a NOVRAM (**nonvolatile RAM**)
- Electrically erasable in the system, but they require more time to erase than normal RAM.
- The flash memory device is used to store setup information for systems such as the video card in the computer.

Figure 10–2 The pin-out of the 2716, 2K × 8 EPROM. (Courtesy of Intel Corporation.)

PIN CONFIGURATION



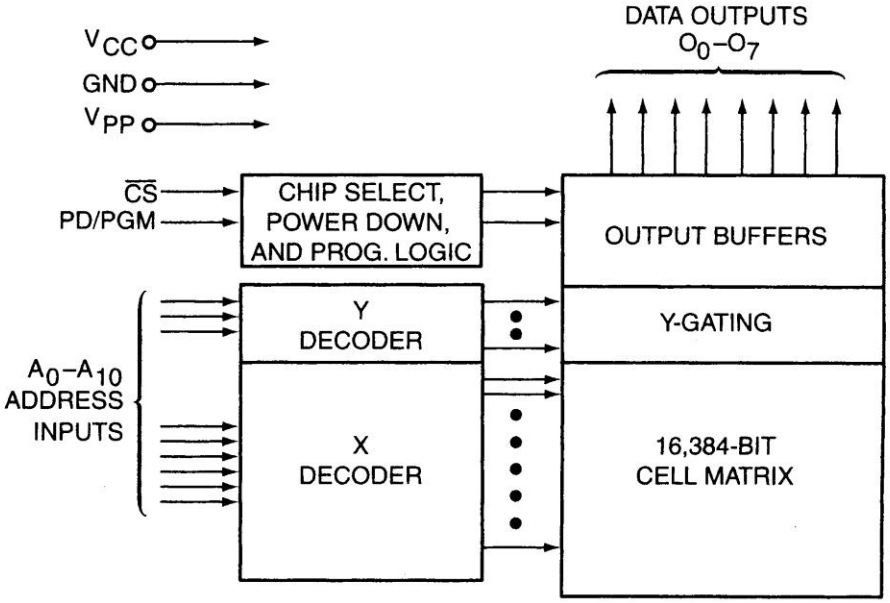
PIN NAMES

A_0–A_{10}	ADDRESSES
PD/PGM	POWER DOWN/PROGRAM
$\overline{CS}$	CHIP SELECT
O_0–O_7	OUTPUTS

MODE SELECTION

PINS / MODE	PD/PGM (18)	$\overline{CS}$ (20)	V_{PP} (21)	V_{CC} (24)	OUTPUTS (9-11, 13-17)
Read	V_{IL}	V_{IL}	+5	+5	D_{OUT}
Deselect	Don't care	V_{IH}	+5	+5	High Z
Power Down	V_{IH}	Don't care	+5	+5	High Z
Program	Pulsed V_{IL} to V_{IH}	V_{IH}	+25	+5	D_{IN}
Program Verify	V_{IL}	V_{IL}	+25	+5	D_{OUT}
Program Inhibit	V_{IL}	V_{IH}	+25	+5	High Z

BLOCK DIAGRAM

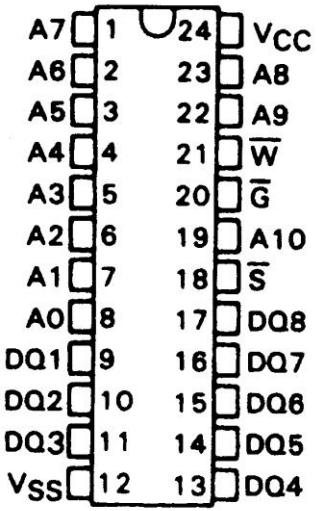


Static RAM (SRAM) Devices

- Static RAM memory devices retain data for as long as DC power is applied.
- Because no special action is required to retain data, these devices are called **static memory**.
  - also called **volatile memory** because they will not retain data without power
- The main difference between ROM and RAM is that RAM is written under normal operation, whereas ROM is programmed outside the computer and normally is only read.

Figure 10–4 The pin-out of the TMS4016, 2K × 8 static RAM (SRAM). (Courtesy of Texas Instruments Incorporated.)

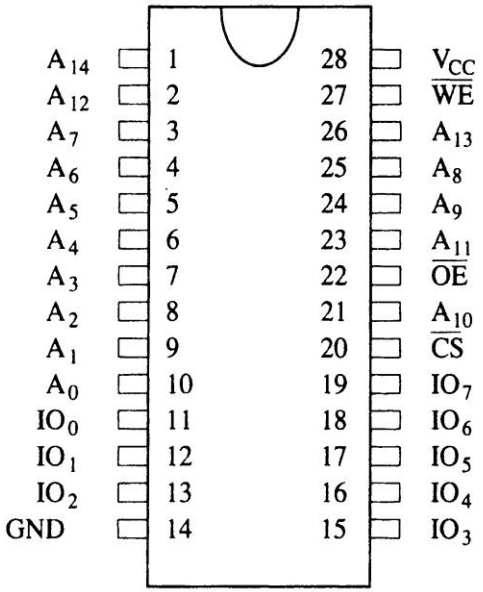
TMS4016 . . . NL PACKAGE
(TOP VIEW)



PIN NOMENCLATURE	
A0 – A10	Addresses
DQ1 – DQ8	Data In/Data Out
$\overline{G}$	Output Enable
$\overline{S}$	Chip Select
V_{CC}	+5-V Supply
V_{SS}	Ground
$\overline{W}$	Write Enable

- SRAM is used when the size of the read/write memory is relatively small
- Today?, a small memory is less than 1M byte

Figure 10–6 Pin diagram of the 62256, 32K × 8 static RAM.



PIN FUNCTION

A_0 - A_{14}	Addresses
IO_0 - IO_7	Data connections
$\overline{CS}$	Chip select
$\overline{OE}$	Output enable
$\overline{WE}$	Write enable
V_{CC}	+5V Supply
GND	Ground

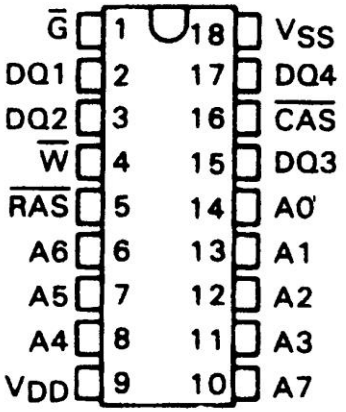
Dynamic RAM (DRAM) Memory

- Available up to 256M × 8 (2G bits).
- DRAM is essentially the same as SRAM, except that it retains data for only 2 or 4 ms on an integrated capacitor.
- After 2 or 4 ms, the contents of the DRAM must be completely rewritten *(refreshed)*.
  - because the capacitors, which store a logic 1 or logic 0, lose their charges

- In DRAM, the entire contents are refreshed with 256 reads in a 2- or 4-ms interval.
  - also occurs during a write, a read, or during a special refresh cycle
- DRAM requires so many address pins that manufacturers multiplexed address inputs.
- Figure 10–7 illustrates a 64K × 4 DRAM, the TMS4464, which stores 256K bits of data.
  - note it contains only eight address inputs where it should contain 16—the number required to address 64K memory locations

Figure 10–7 The pin-out of the TMS4464, 64K × 4 dynamic RAM (DRAM). (Courtesy of Texas Instruments Incorporated.)

TMS4464 . . . JL OR NL PACKAGE
(TOP VIEW)



(a)

PIN NOMENCLATURE	
A0-A7	Address Inputs
$\overline{CAS}$	Column Address Strobe
DQ1-DQ4	Data-In/Data-Out
$\overline{G}$	Output Enable
$\overline{RAS}$	Row Address Strobe
V_{DD}	+5-V Supply
V_{SS}	Ground
$\overline{W}$	Write Enable

(b)

- 16 address bits can be forced into eight address pins in two 8-bit increments
- this requires two special pins: the **column address strobe** ($\overline{CAS}$) and **row address strobe** ($\overline{RAS}$)

- First, A_0–A_7 are placed on the address pins and strobed into an internal row latch by $\overline{RAS}$ as the row address.
- Next, address bits A_8-A_{15} are placed on the same eight address inputs and strobed into an internal column latch by $\overline{CAS}$ as the column address
- The 16-bit address in the internal latches addresses the contents of one of the 4-bit memory locations.
  - $\overline{CAS}$ also performs chip selection input to DRAM

Figure 10–8 $\overline{\text{RAS}}$, $\overline{\text{CAS}}$ and address input timing for the TMS4464 DRAM. (Courtesy of Texas Instruments Incorporated.)

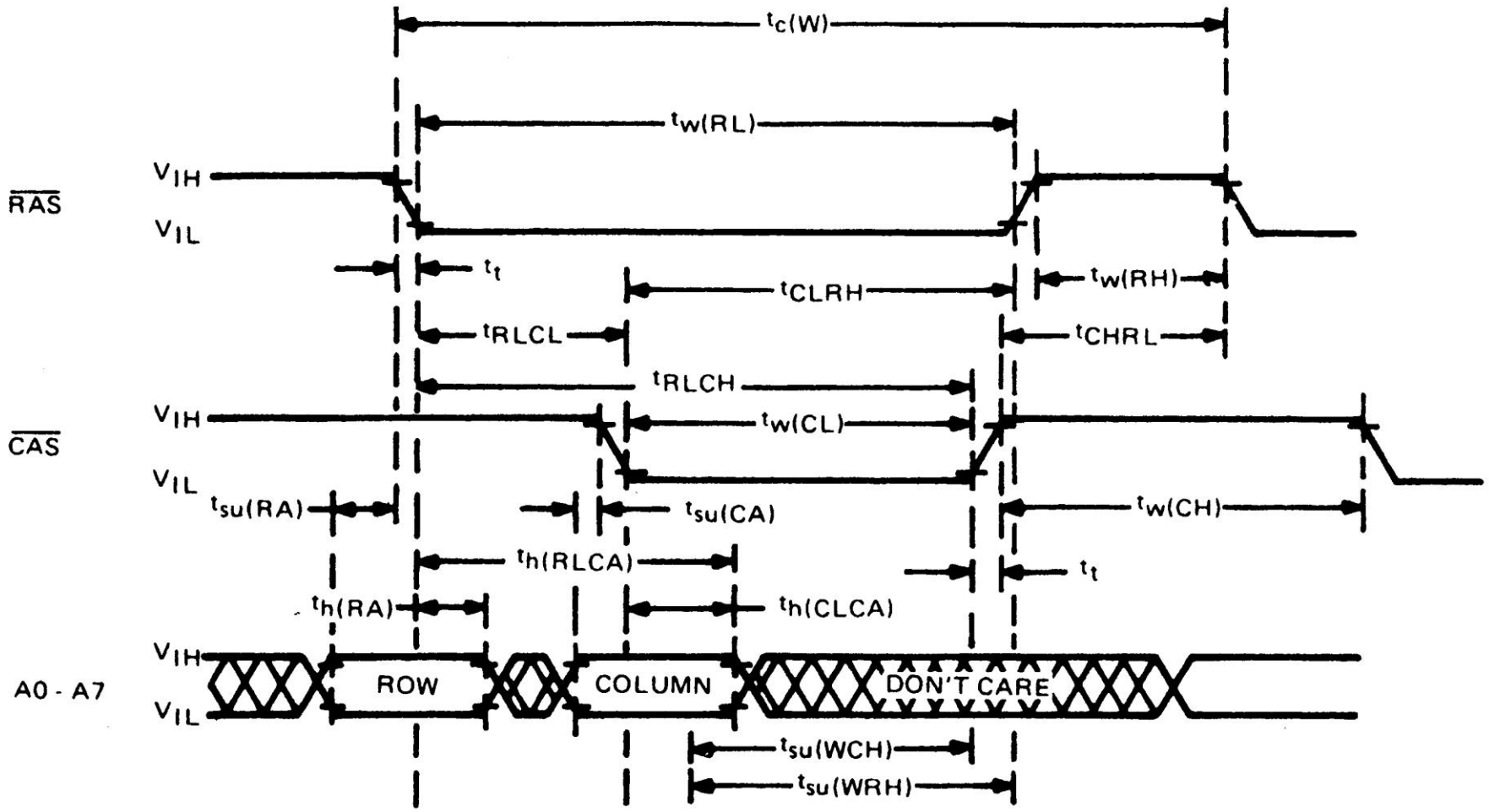
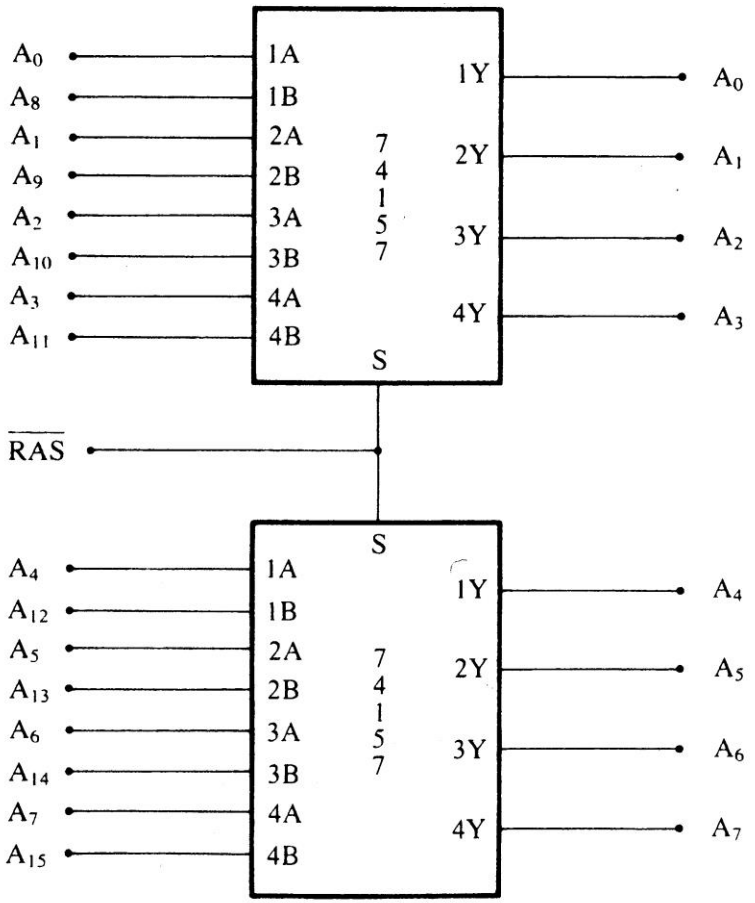
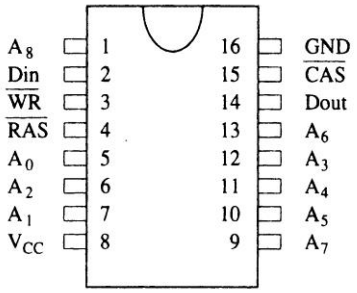


Figure 10–9 Address multiplexer for the TMS4464 DRAM.



- multiplexers used to strobe column and row addresses into the address pins on a pair of TMS4464 DRAMs.
- the $\overline{\text{RAS}}$ signal not only strobes the row address into the DRAMs, but it also selects which part of the address is applied to the address inputs.

Figure 10–10 The 41256 dynamic RAM organized as a 256K × 1 memory device.



PIN FUNCTIONS

A_0 - A_8	Addresses
Din	Data in
Dout	Data out
$\overline{CAS}$	Column Address Strobe
$\overline{RAS}$	Row Address Strobe
$\overline{WR}$	Write enable
V_{CC}	+5V Supply
GND	Ground

– the pin-out of the 41256 dynamic RAM
– this device is organized as a 256K × 1 memory
– requires as little as 70 ns to access data

10–2 ADDRESS DECODING

- In order to attach a memory device to the microprocessor, it is necessary to decode the address sent from the microprocessor.
- Decoding makes the memory function at a unique section or partition of the memory map.
- Without an address decoder, only one memory device can be connected to a microprocessor, which would make it virtually useless.

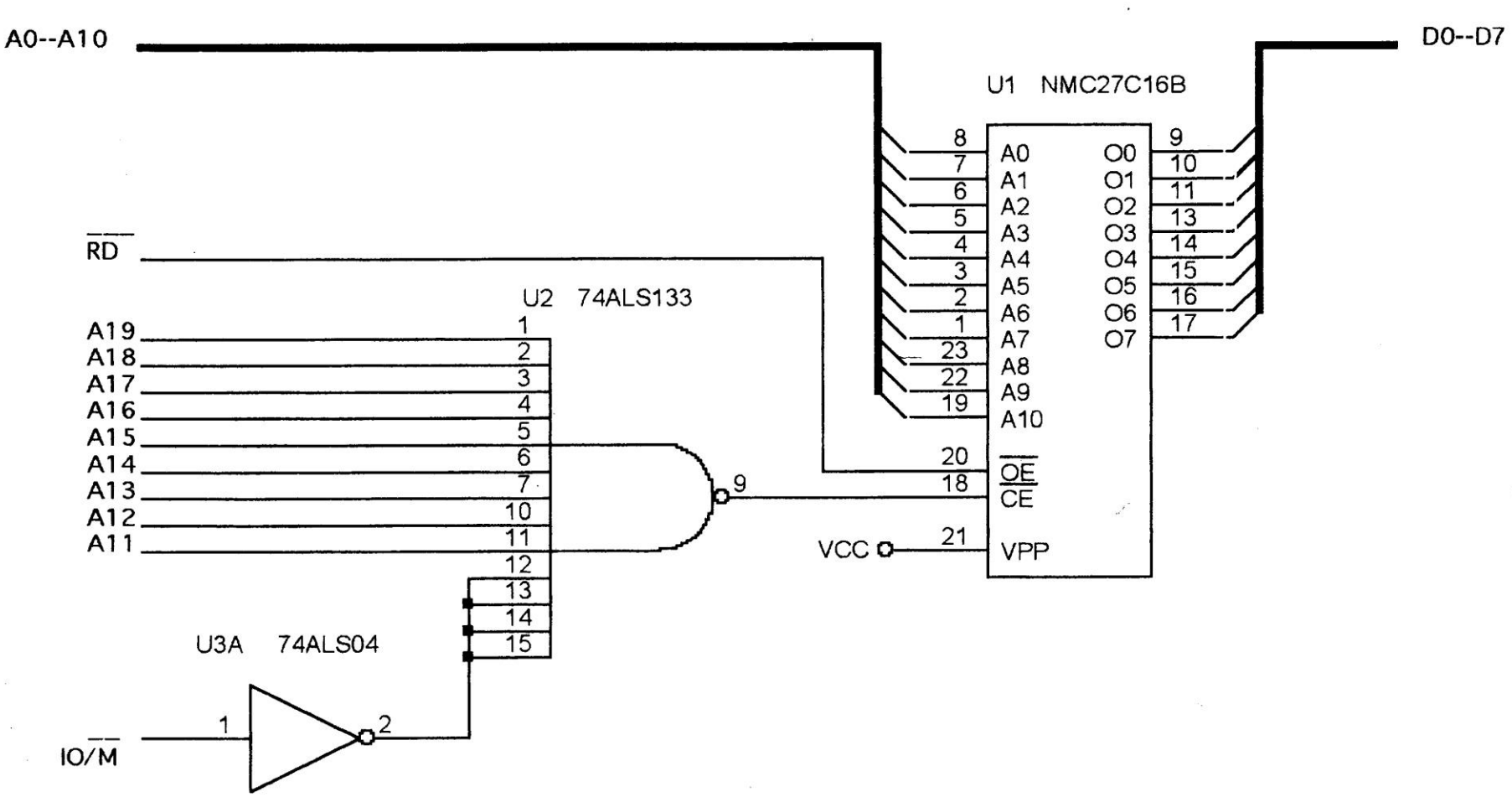
Why Decode Memory?

- The 8088 has 20 address connections and the 2716 EPROM has 11 connections.
- The 8088 sends out a 20-bit memory address whenever it reads or writes data.
  - because the 2716 has only 11 address pins, there is a mismatch that must be corrected
- The decoder corrects the mismatch by decoding address pins that do not connect to the memory component.

Simple NAND Gate Decoder

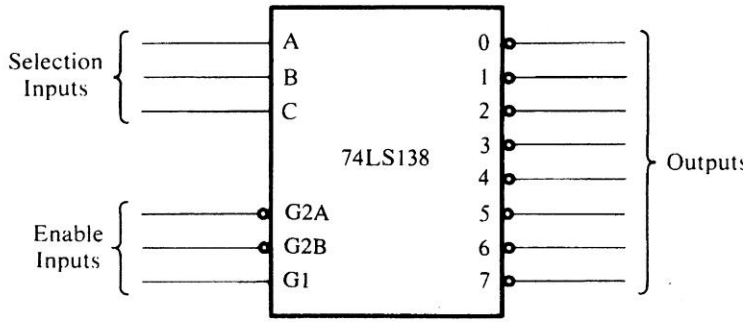
- When the 2K × 8 EPROM is used, address connections A_{10}–A_0 of 8088 are connected to address inputs A_{10}–A_0 of the EPROM.
  - the remaining nine address pins (A_{19}–A_{11}) are connected to a NAND gate decoder
- The decoder selects the EPROM from one of the 2K-byte sections of the 1M-byte memory system in the 8088 microprocessor.
- In this circuit a NAND gate decodes the memory address, as seen in Figure 10-13.

Figure 10–13 A simple NAND gate decoder that selects a 2716 EPROM for memory location FF800H–FFFFFH.



- If the 20-bit binary address, decoded by the NAND gate, is written so that the leftmost nine bits are 1s and the rightmost 11 bits are don't cares (X), the actual address range of the EPROM can be determined.
  - a *don't care* is a logic 1 or a logic 0, whichever is appropriate
- Because of the excessive cost of the NAND gate decoder and inverters often required, this option requires an alternate be found.

The 3-to-8 Line Decoder (74LS138)



Inputs						Outputs							
Enable			Select										
$\overline{G2A}$	$\overline{G2B}$	G1	C	B	A	$\overline{0}$	$\overline{1}$	$\overline{2}$	$\overline{3}$	$\overline{4}$	$\overline{5}$	$\overline{6}$	$\overline{7}$
1	X	X	X	X	X	1	1	1	1	1	1	1	1
X	1	X	X	X	X	1	1	1	1	1	1	1	1
X	X	0	X	X	X	1	1	1	1	1	1	1	1
0	0	1	0	0	0	0	1	1	1	1	1	1	1
0	0	1	0	0	1	1	0	1	1	1	1	1	1
0	0	1	0	1	0	1	1	0	1	1	1	1	1
0	0	1	0	1	1	1	1	1	0	1	1	1	1
0	0	1	1	0	0	1	1	1	1	0	1	1	1
0	0	1	1	0	1	1	1	1	1	1	0	1	1
0	0	1	1	1	0	1	1	1	1	1	1	0	1
0	0	1	1	1	1	1	1	1	1	1	1	1	0

- a common integrated circuit decoder found in many systems is the 74LS138 3-to-8 line decoder.

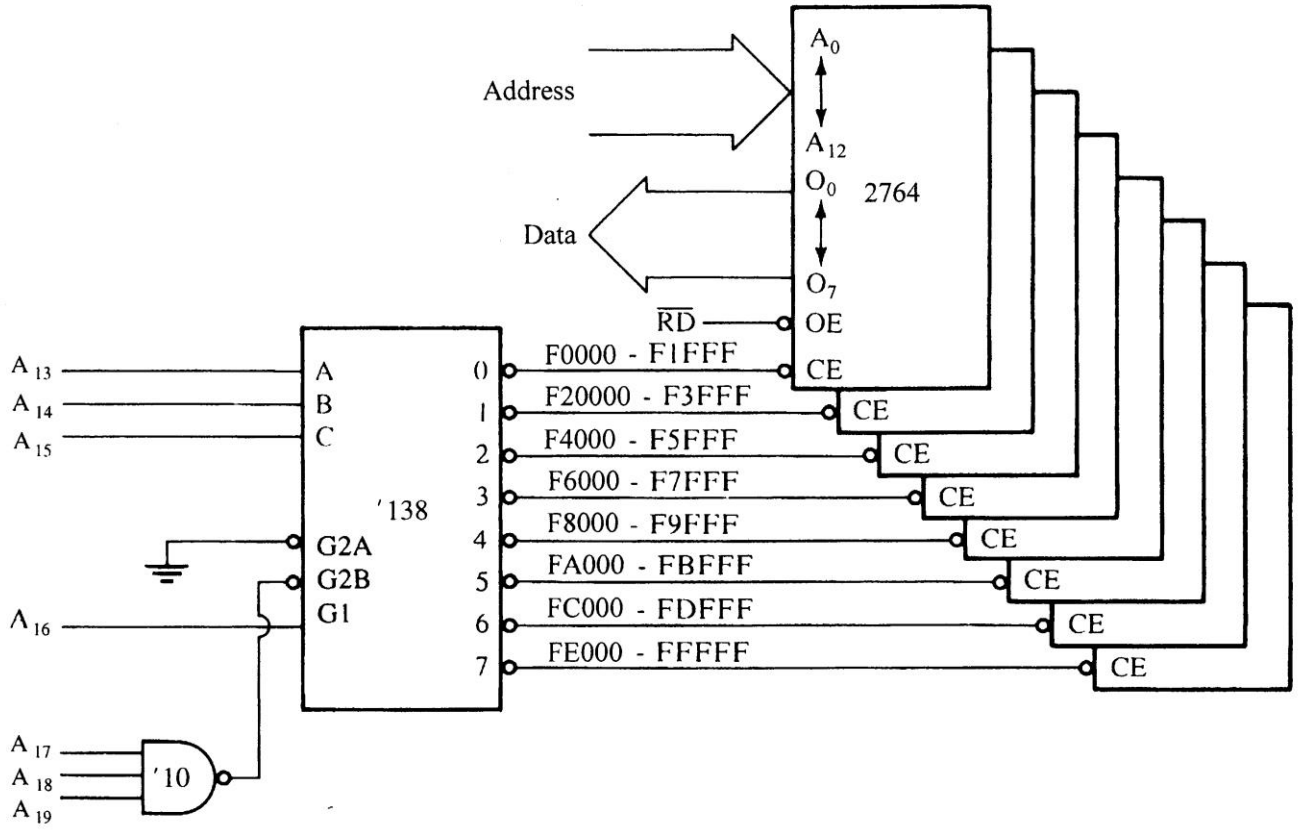
Figure 10–14 The 74LS138 3-to-8 line decoder and function table.

Sample Decoder Circuit

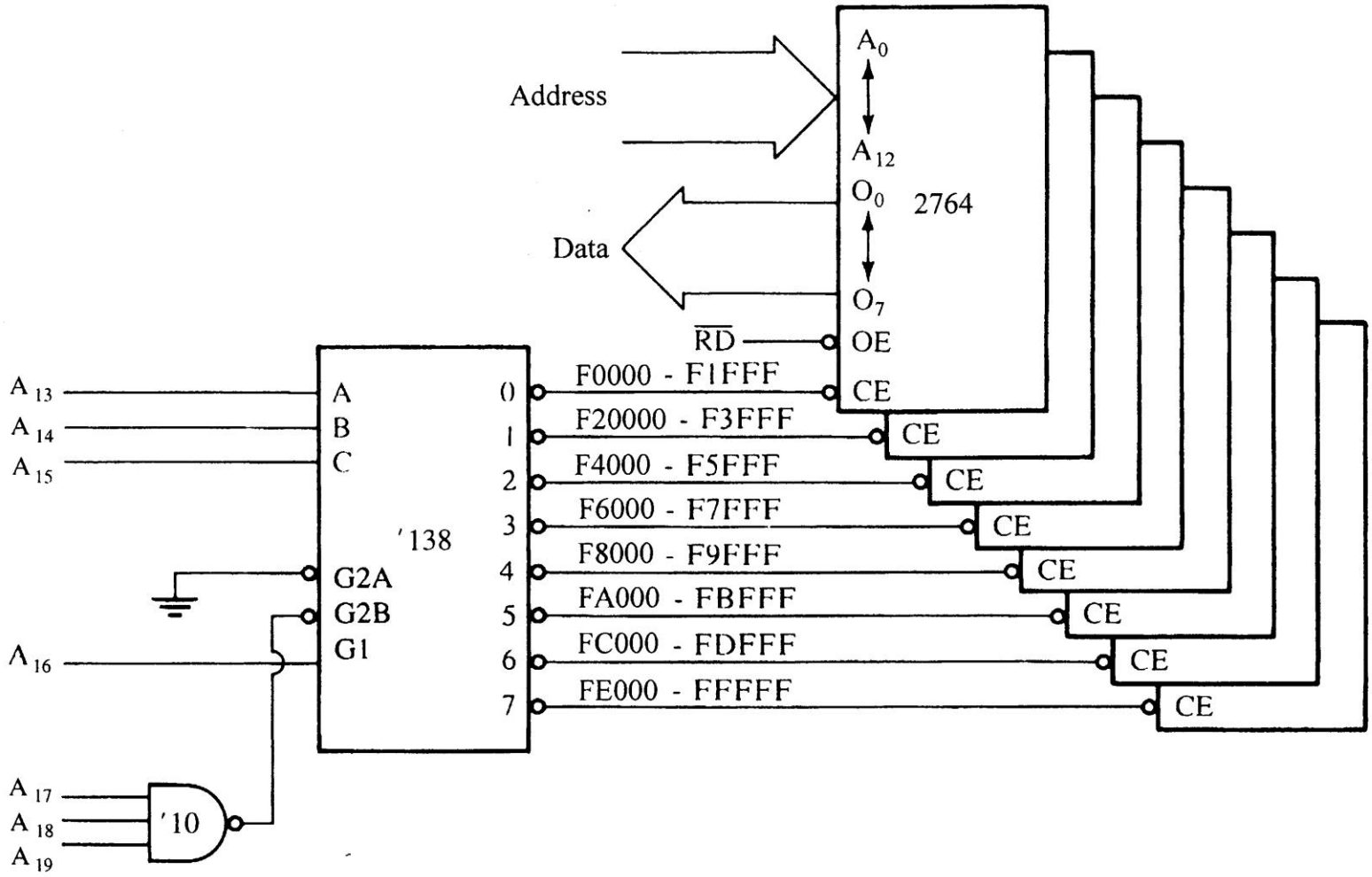
- The outputs of the decoder in Figure 10–15, are connected to eight different 2764 EPROM memory devices.
- The decoder selects eight 8K-byte blocks of memory for a total capacity of 64K bytes.
- This figure also illustrates the address range of each memory device and the common connections to the memory devices.

Figure 10–15 A circuit that uses eight 2764 EPROMs for a 64K × 8 section of memory in an 8088 microprocessor-based system. The addresses selected in this circuit are F0000H–FFFFFH.

- decoder's outputs are connected to the $\overline{CE}$ of the EPROMs,
- $\overline{RD}$ signal from 8088 is connected to the OE of the EPROMs



MEMORY INTERFACE



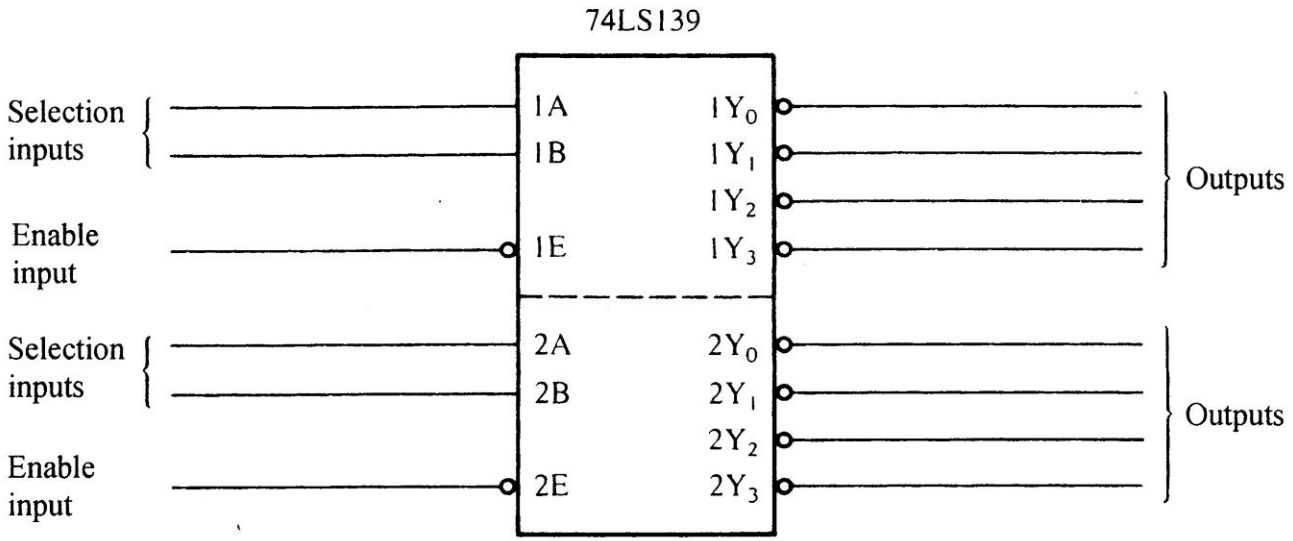
- In this circuit, a three-input NAND gate is connected to address bits A_{19}–A_{17}.
- When all three address inputs are high, the output of this NAND gate goes low and enables input $\overline{G2B}$ of the 74LS138.
- Input G1 is connected directly to A_{16}.
- In order to enable this decoder, the first four address connections (A_{19}–A_{16}) must all be high.

- Address inputs C, B, and A connect to microprocessor address pins A_{15}–A_{13}.
- These three address inputs determine which output pin goes low and which EPROM is selected whenever 8088 outputs a memory address within this range to the memory system.

The Dual 2-to-4 Line Decoder (74LS139)

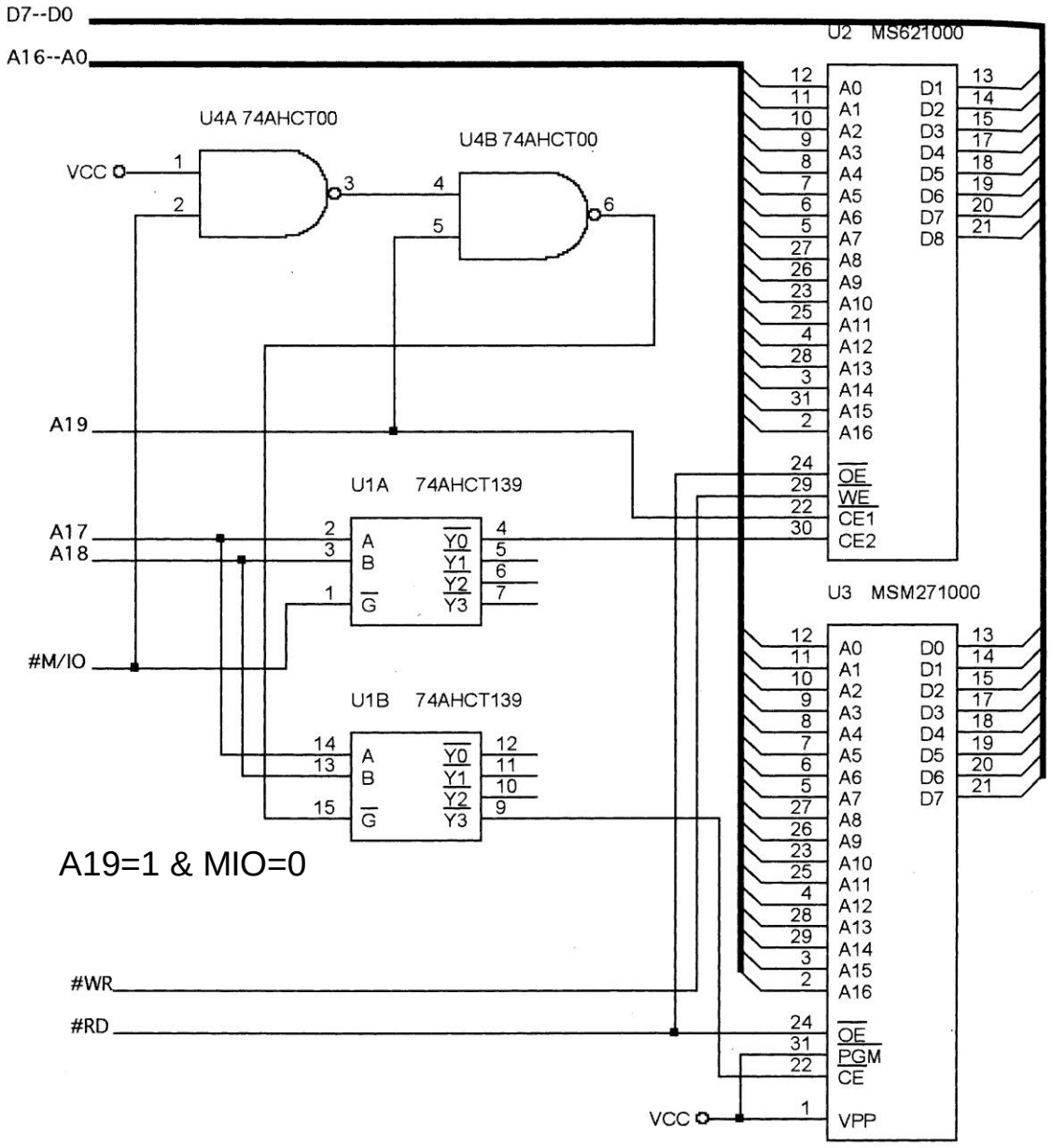
- Figure 10–16 illustrates both the pin-out and the truth table for the 74LS139 dual 2-to-4 line decoder.
- 74LS139 contains two separate 2-to-4 line decoders—each with its own address, enable, and output connections.
- A more complicated decoder using the 74LS139 decoder appears in Figure 10–17.

Figure 10–16 The pin-out and truth table of the 74LS139, dual 2-to-4 line decoder.



Inputs			Outputs			
$\overline{E}$	A	B	$\overline{Y_0}$	$\overline{Y_1}$	$\overline{Y_2}$	$\overline{Y_3}$
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0
1	X	X	1	1	1	1

Figure 10–17 A sample memory system constructed with a 74HCT139.



PLD Programmable Decoders

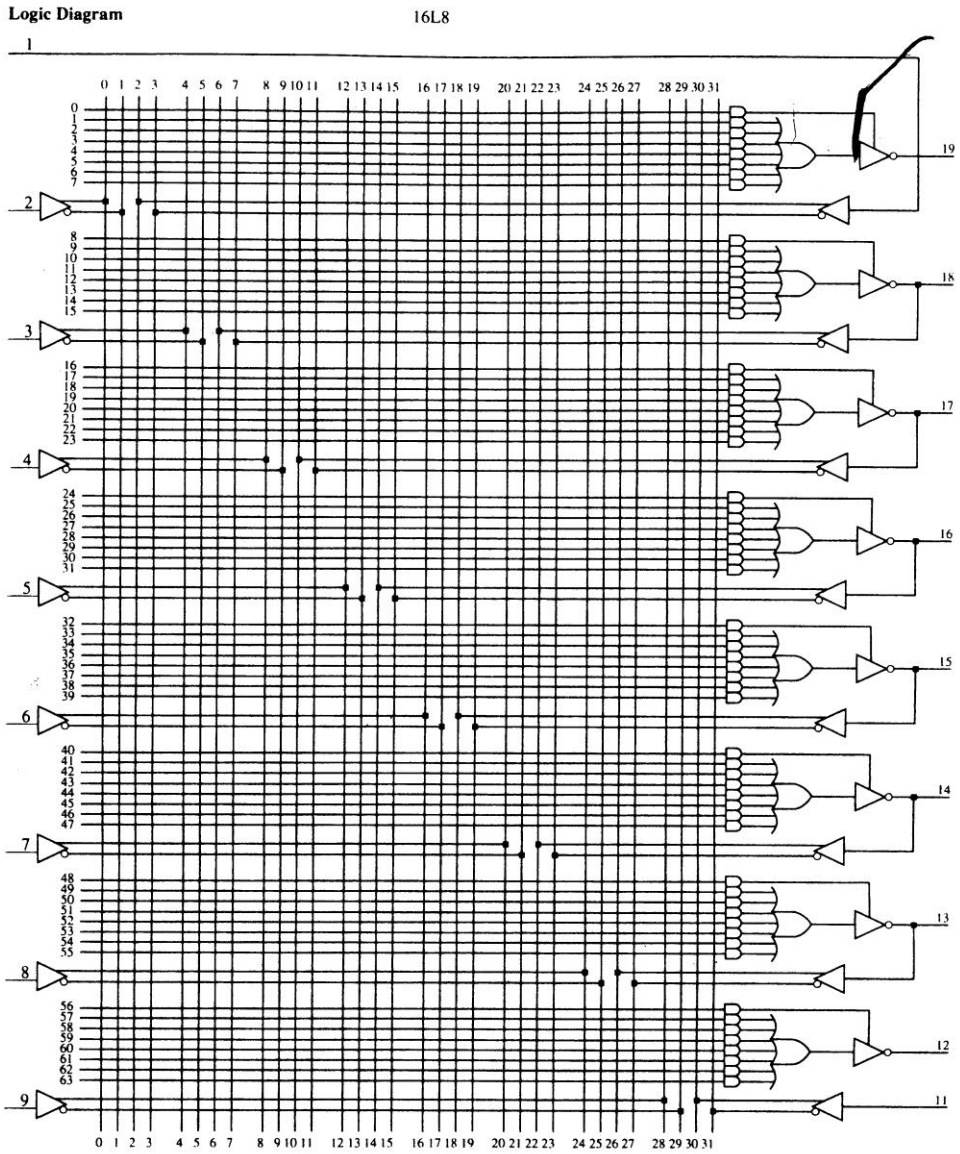
- Three SPLD (**simple PLD**) devices function in the same manner but have different names:
  - PLA (**programmable logic array**)
  - PAL (**programmable array logic**)
  - GAL (**gated array logic**)
- In existence since the mid-70s, they have appeared in memory system and digital designs since the early 1990s.

- PAL and PLA are fuse-programmed, and some PLD devices are erasable devices.
  - all are arrays of programmable logic elements
- Other PLDs available:
  - CPLDs (**complex programmable logic devices**)
  - FPGAs (**field programmable gate arrays**)
  - FPICs (**field programmable interconnect**)
- These PLDs are more complex than the SPLDs used more commonly in designing a complete system.

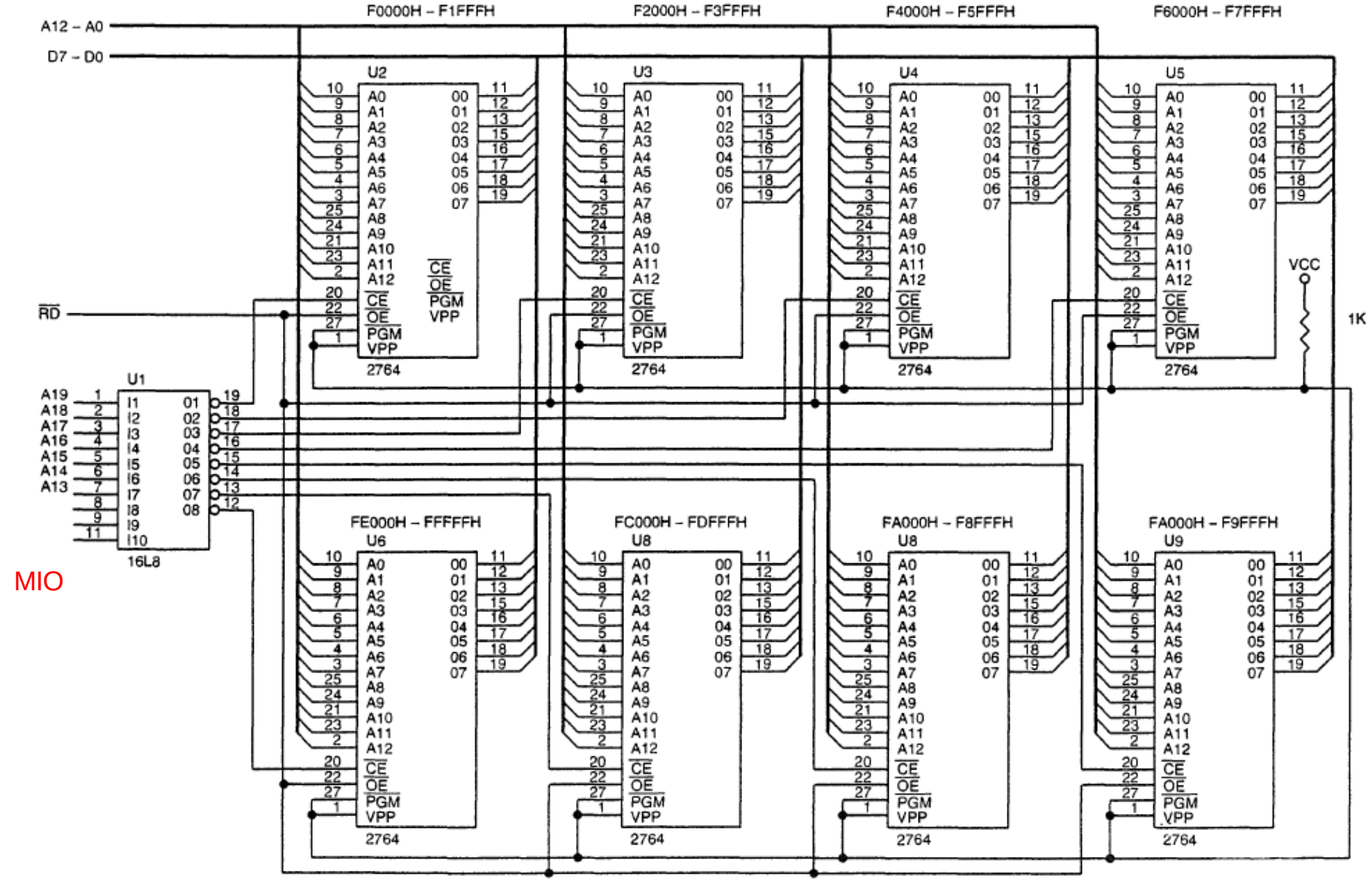
Combinatorial Programmable Logic Arrays

- Fig 10–18 shows the internal structure of a PAL16L8 constructed with AND/OR gate logic.
- It has 10 fixed inputs, two fixed outputs, and six pins programmable as inputs or outputs.
- Programming is accomplished by blowing fuses to connect inputs to the OR gate array.
- It is ideal as a decoder because of its structure, also because outputs are active low.

Figure 10–18 The PAL16L8. (Copyright Advanced Micro Devices, Inc., 1988. Reprinted with permission of copyright owner. All rights reserved.)



- A PAL is programmed with software such as PALASM, the PAL assembler program.
- PLD design is accomplished using HDL (**hardware description language**) or VHDL (**verilog HDL**).
  - VHDL and its syntax are currently the industry standard for programming PLD devices
- Various editors attempt to ease the task of defining the pins.
  - the authors believe it is easier to use NotePad



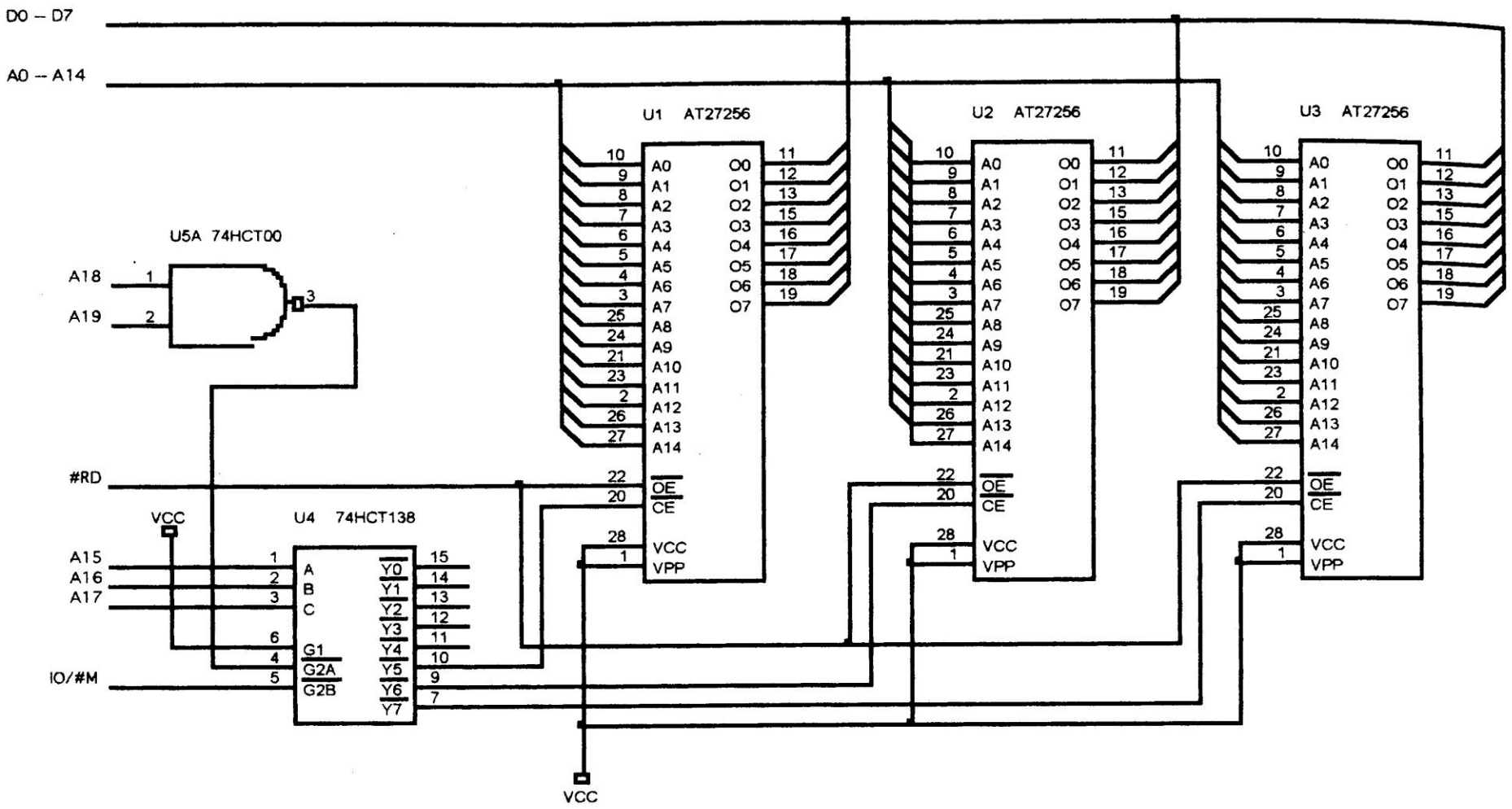
```
CHIP         DECODER1 PAL16L8

;pins 1    2    3    4    5    6    7    8  9  10
      A19 A18 A17 A16 A15 A14 A13 NC NC GND

;pins 11 12 13 14 15 16 17 18 19  20
       NC O8 O7 O6 O5 O4 O3 O2 O1 VCC

EQUATIONS

/O1 = A19 * A18 * A17 * A16 * /A15 * /A14 * /A13
/O2 = A19 * A18 * A17 * A16 * /A15 * /A14 * A13
/O3 = A19 * A18 * A17 * A16 * /A15 * A14 * /A13
/O4 = A19 * A18 * A17 * A16 * /A15 * /A14 * A13
/O5 = A19 * A18 * A17 * A16 * A15 * /A14 * /A13
/O6 = A19 * A18 * A17 * A16 * A15 * A14 * A13
/O7 = A19 * A18 * A17 * A16 * A15 * /A14 * /A13
/O8 = A19 * A18 * A17 * A16 * A15 * /A14 * A13
```

/A15 *A14*A13

Interfacing EPROM to the 8088

- Fig 10–20 shows an 8088/80188 connected to three 27256 EPROMs, 32K × 8 devices.
  - 74HCT138 decoder in this illustration decodes three 32K × 8 blocks of memory for a total of 96K × 8 bits of physical address space for 8088/80188
- The decoder is selected for an address range that begins at E8000H and continues through location FFFFFH—the upper 96K bytes of memory.

Figure 10–20 Three 27256 EPROMs interfaced to the 8088 microprocessor.

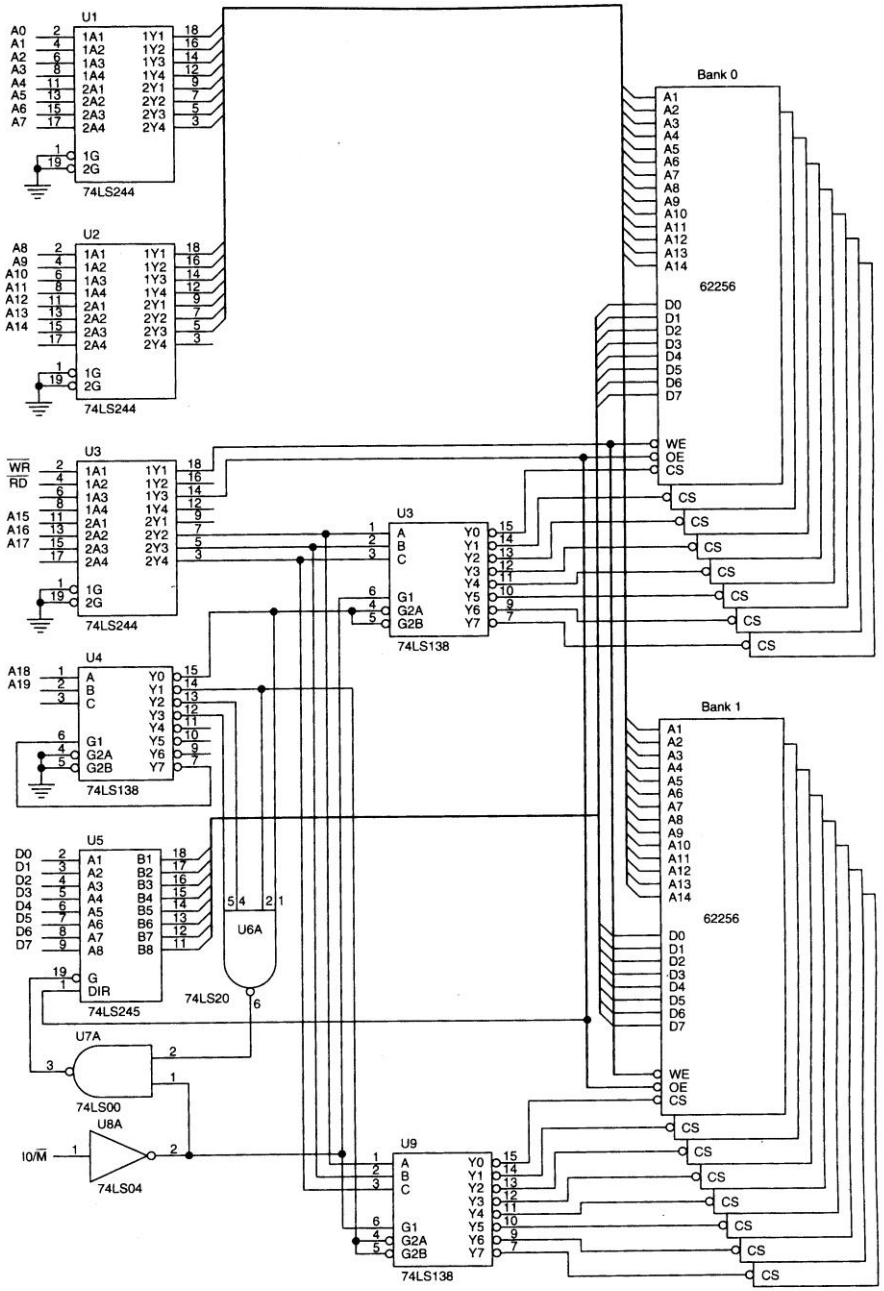


- This section of memory is an EPROM because FFFF0H is where the 8088 starts to execute instructions after a hardware reset.
  - often called the **cold-start location**
- The software stored in this section of memory would contain a JMP instruction at location FFFF0H that jumps to location E8000H
  - so the remainder of the program can execute
- In this circuit, U_1 is decoded at addresses E8000H–EFFFFH, U_2 is decoded at F0000H–F7FFFH, and U_3 at F8000H–FFFFFH.

Interfacing RAM to the 8088

- RAM is easier to interface than EPROM as most RAM does not require wait states.
  - ideal section of memory for RAM is the very bottom, which contains vectors for interrupts
- Interrupt vectors are often modified by software packages,
  - it is important to encode this section of the memory with RAM
- Fig 10–21 shows sixteen 62256 static RAMs interfaced to the 8088, beginning at 00000H.

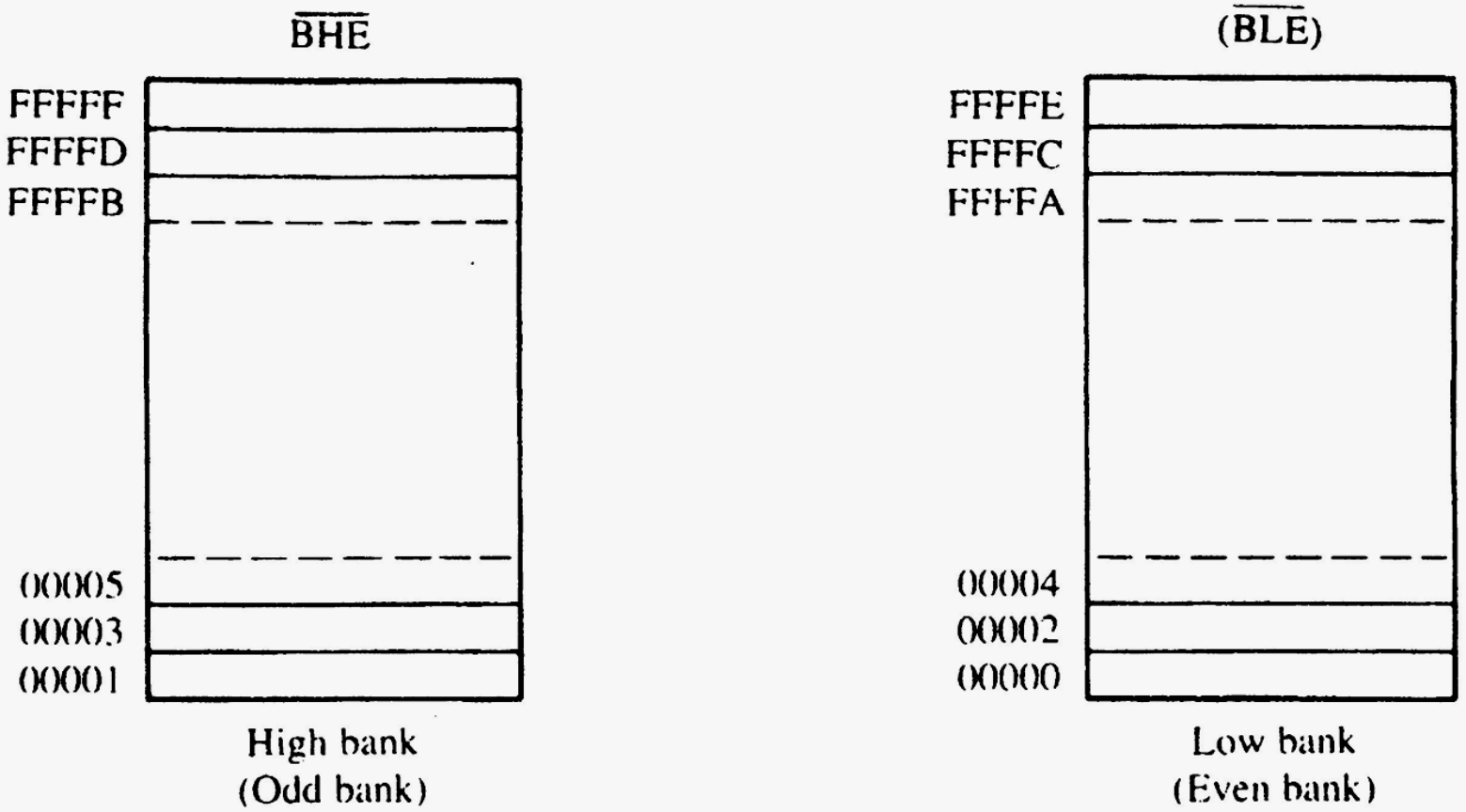
Figure 10–21 A 512K-byte static memory system using 16 62256 SRAMs.



- this board uses two decoders to select 16 RAM components, and a third to select other decoders for the appropriate memory sections
- sixteen 32K RAMs fill memory from 00000H through 7FFFFH, for 512K bytes memory.

16-Bit Bus Control

- The data bus of 8086, 80186, 80286, and 80386SX is twice as the 8088/80188.
  - processors must be able to write data to any 16-bit location—or any 8-bit location
- The 16-bit data bus is divided into 2 sections (or **banks**) 8 bits wide so the processor can write to either half (8-bit) or both halves (16-bit).
  - one bank (**low bank**) holds all even-numbered memory locations
  - the other bank (**high bank**) holds all the odd-numbered memory locations



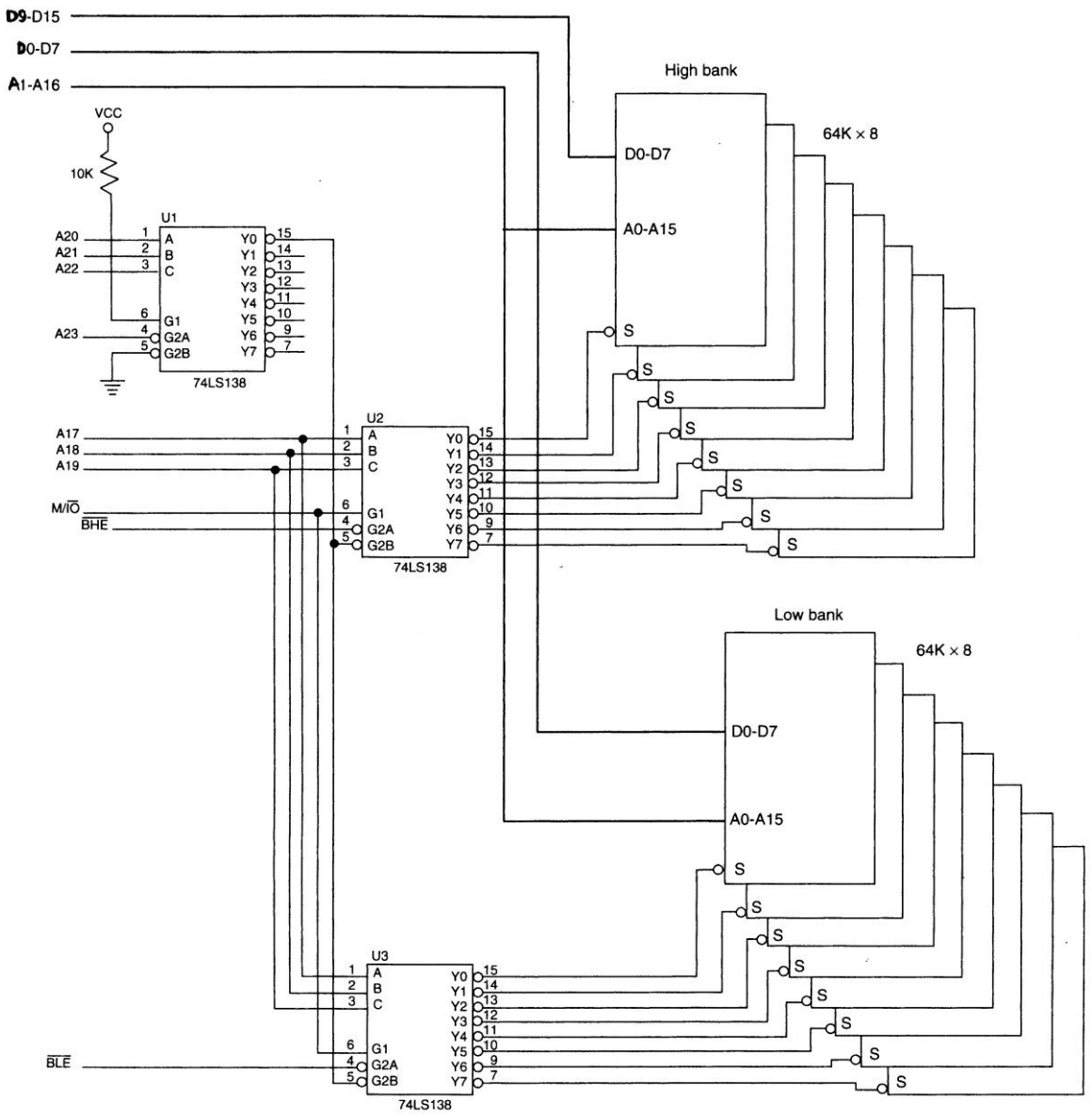
Note: A_0 is labeled $\overline{BLE}$ (Bus low enable) on the 80386SX.

- Bank selection is accomplished in two ways:
  1. separate write signal is developed to select a write to each bank of the memory
  2. separate decoders are used for each bank
- The first technique is by far the least costly approach to memory interface.
- The second technique is only used in a system that must achieve the most efficient use of the power supply.

Separate Bank Decoders

- Separate bank decoders is often the least effective way to decode memory addresses for the 8086, 80186, 80286, and 80386SX.
- This method is sometimes used, but it is difficult to understand why in most cases.
  - one reason may be to conserve energy, as only banks selected are enabled
- Fig 10–28 shows two 74LS138s used to select 64K RAM memory components for the 80386SX microprocessor (24-bit address).

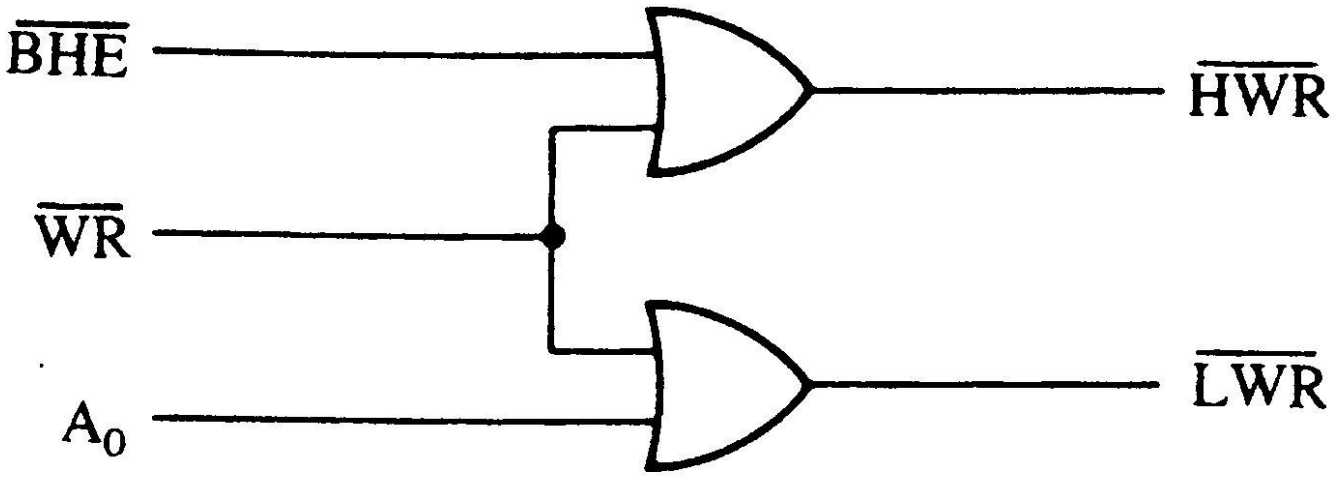
Figure 10–28 Separate bank decoders.



Separate Bank Write Strobes

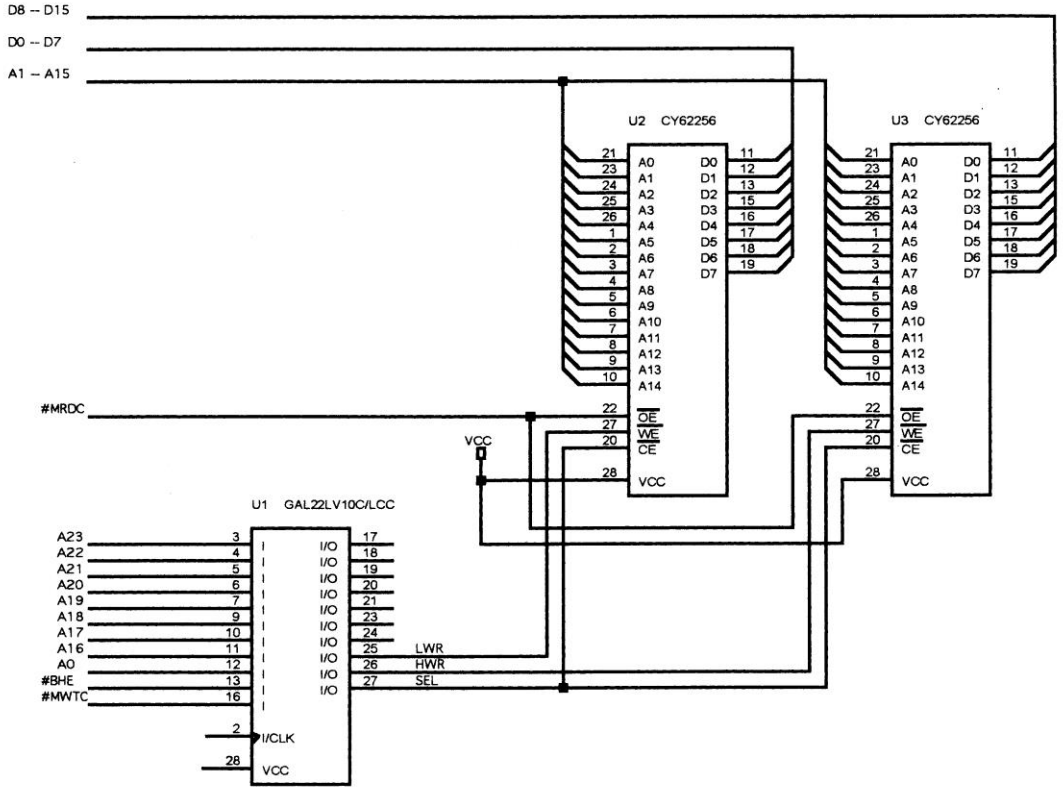
- The effective way to handle bank selection is a separate write strobe for each bank.
  - this requires only one decoder to select a 16-bit-wide memory, which saves money
- Fig 10–29 depicts generation of separate 8086 write strobes.
- Separate read strobes for each bank are usually unnecessary because 8086 through 80386SX read only the byte of data they need at any given time from half of the data bus.

Figure 10–29 The memory bank write selection input signals: $\overline{\text{HWR}}$ (high bank write) and $\overline{\text{LWR}}$ (low bank write).



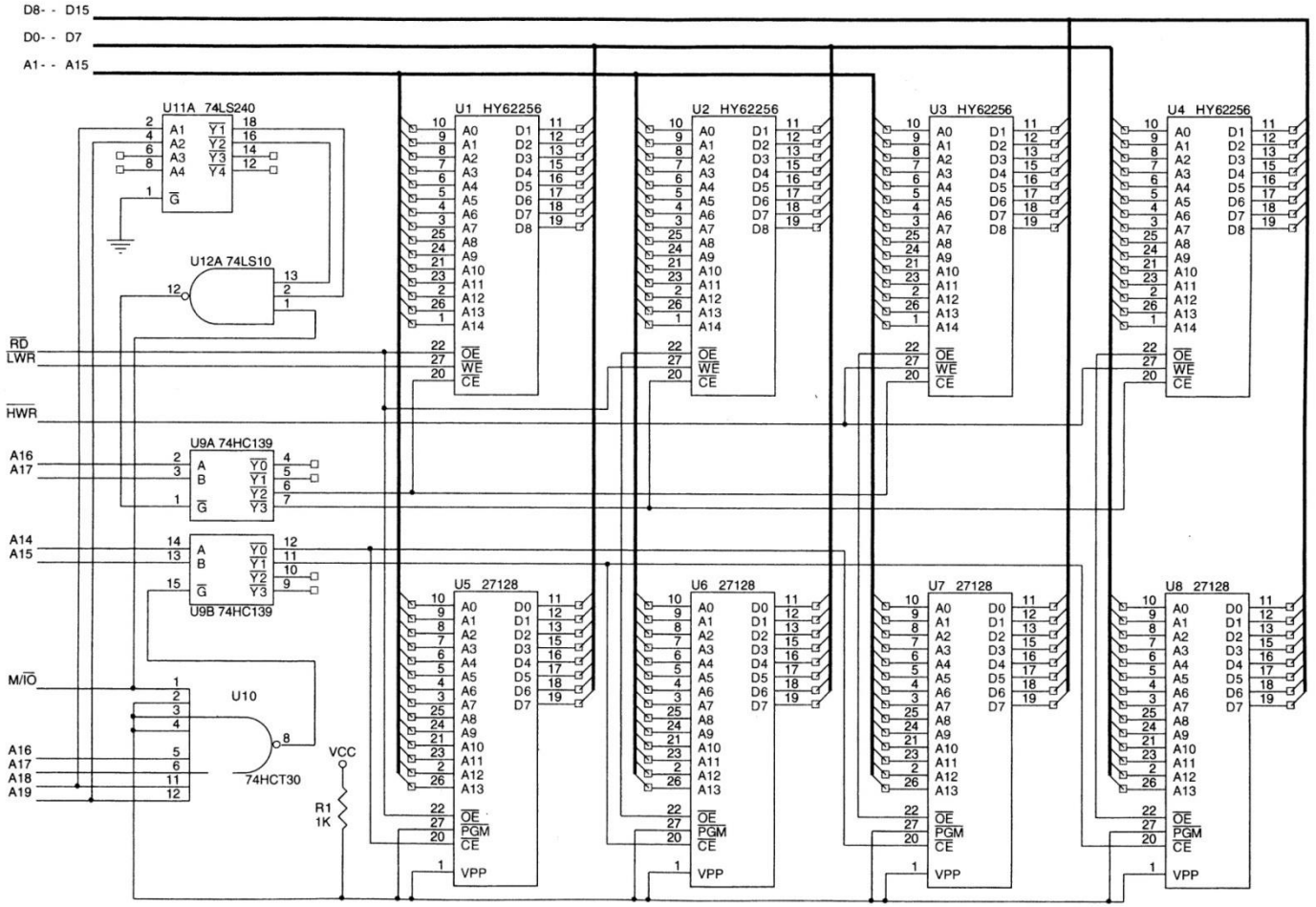
- a memory system that uses separate write strobes is constructed differently from the 8088 or the system using separate memory banks
- memory in a system using separate write strobes is decoded as 16-bit-wide

Figure 10–30 A 16-bit-wide memory interfaced at memory locations 06000H–06FFFH.

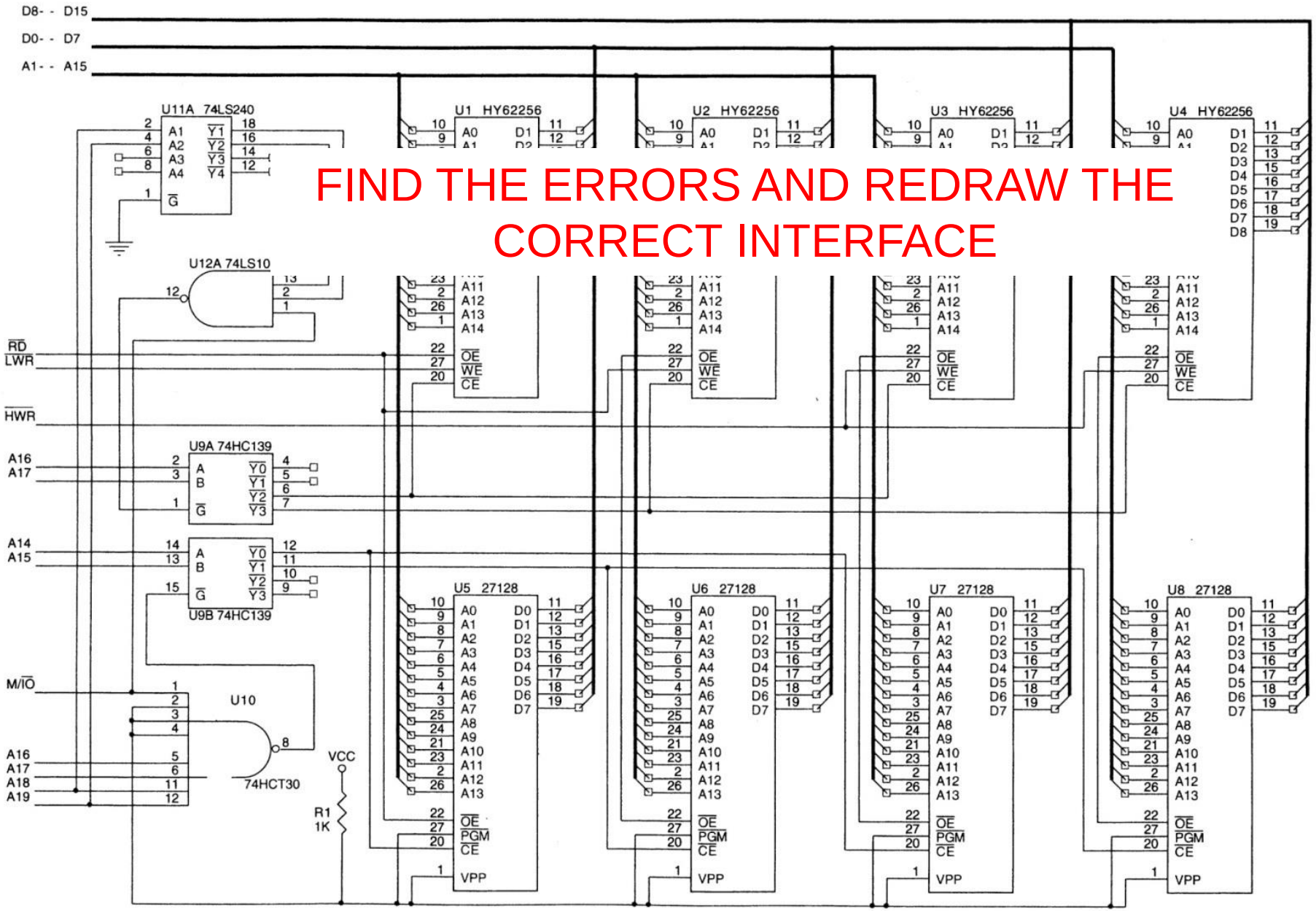


- the program for the GAL22V10 decoder is illustrated in Example 10–7
- notice that not only is the memory selected, but both the lower and upper write strobes are also generated by the PLD
- a GAL22V10 to both decode memory and generate the separate write strobe

Figure 10–31 A memory system for the 8086 that contains a 64K-byte EPROM and a 128K-byte SRAM.

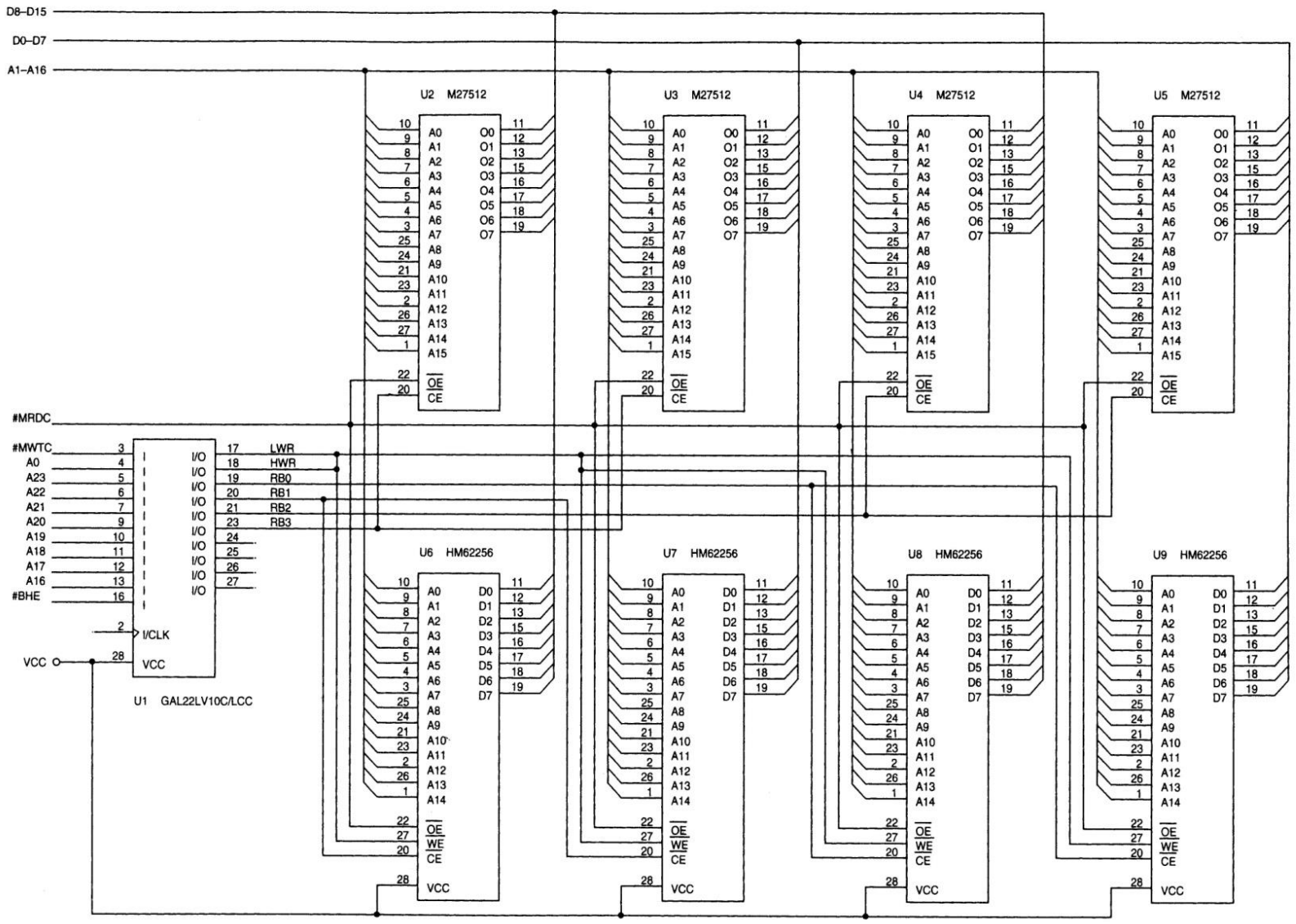


- four 27128 EPROMs compose a 32K × 16-bit memory at F0000–FFFFFH and four 62256 that compose 64K × 16-bit at 00000H–1FFFFH
- though 16 bits wide, it is still numbered in bytes



- Fig 10–32 shows a memory connected to an 80386SX by using a GAL22V10 as a decoder.
- This interface contains 256K bytes of EPROM in the form of four 27512 (64K × 8) EPROMs and 128K bytes of SRAM memory found in four 62256 (32K × 8) SRAMs.
- The PLD decodes the 16-bit-wide memory addresses at locations 000000H–01FFFFH for the SRAM and locations FC0000H–FFFFFFH for the EPROM.

Figure 10–32 An 80386SX memory system containing 256K of EPROM and 128K of SRAM.



SUMMARY

- All memory devices have address inputs; data inputs and outputs, or just outputs; a pin for selection; and one or more pins that control the operation of the memory.
- Address connections on a memory component are used to select one of the memory locations within the device.
- Ten address pins have 1024 combinations and therefore are able to address 1024 different memory locations.

SUMMARY (*cont.*)

- Data connections on a memory are used to enter information to be stored in a memory location and also to retrieve information read from a memory location.
- Manufacturers list their memory as, for example, 4K x 4, which means that the device has 4K memory locations (4096) and that four bits are stored in each location.

SUMMARY (*cont.*)

- An EPROM memory is programmed by an EPROM programmer and can be erased if exposed to ultraviolet light.
- Today, EPROMs are available in sizes from 1K x 8 up to 512K x 8 and larger.
- The flash memory (EEPROM) is programmed in the system by using a 12 V or 5.0 V programming pulse.

SUMMARY (*cont.*)

- Static RAM (SRAM) retains data for as long as the system power supply is attached.
- These memory types are available in sizes up to 128K x 8.

SUMMARY (*cont.*)

- Dynamic RAM (DRAM) retains data for only a short period, usually 2-4 ms.
- This creates problems for the memory system designer because the DRAM must be refreshed periodically.
- DRAMs also have multiplexed address inputs that require an external multiplexer to provide each half of the address at the appropriate time.

SUMMARY (*cont.*)

- Memory address decoders select an EPROM or RAM at a particular area of the memory.
- Commonly found address decoders include the 74LS138 3-to-8 line decoder, the 74LS139 2-to-4 line decoder, and programmed selection logic in a PLD.

SUMMARY (*cont.*)

- The access speed of the EPROM must be compatible with the microprocessor to which it is interfaced.
- Many EPROMs available today have an access time of 450 ns, which is too slow for the 5 MHz 8088.
- In order to circumvent this problem, a wait state is inserted to increase memory access time to 660 ns.

SUMMARY (*cont.*)

- Error-correction features are also available for memory systems, but these require the storage of many more bits.
- If an 8-bit number is stored with an error-correction circuit, it actually takes 13 bits of memory: five for an error checking code and eight for the data.
- Most error-correction integrated circuits are able to correct only a single-bit error.

SUMMARY

- Dynamic RAM controllers are designed to control DRAM memory components.
- Many DRAM controllers today are built into the chip set and contain address multiplexers, refresh counters, and the circuitry required to do a periodic DRAM memory refresh.