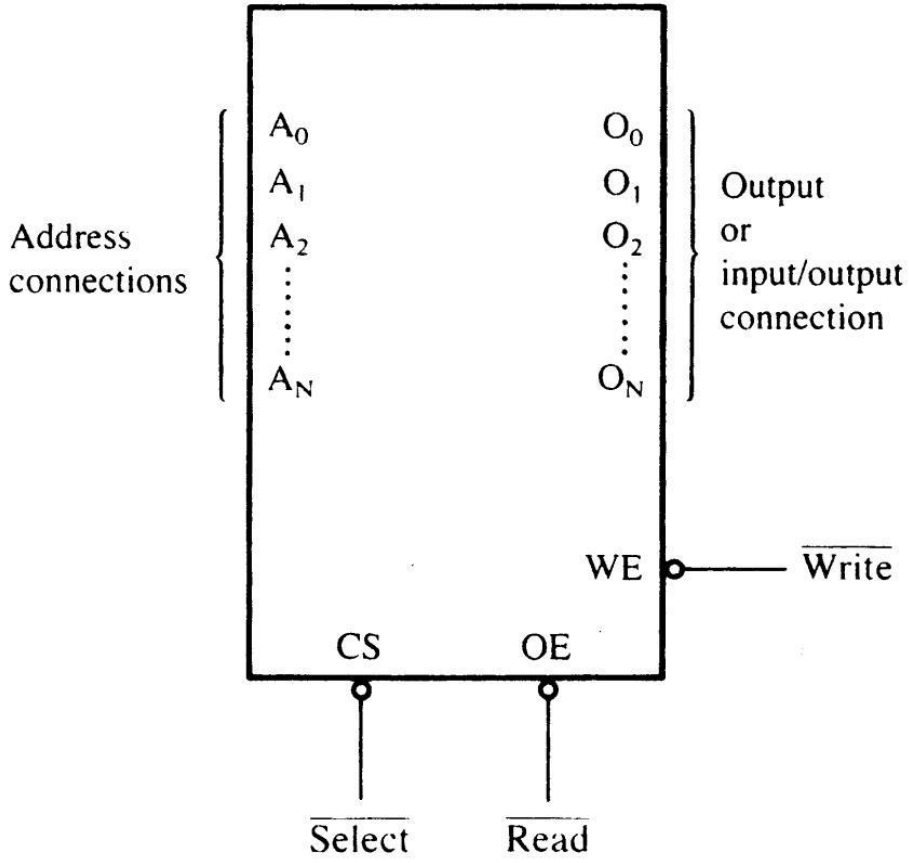
# Memory Interface

# Introduction

- Whether simple or complex, every microprocessor-based system has a memory system.
- Almost all systems contain two main types of memory:
  - read-only memory (ROM)
  - random Access memory (RAM) or read/write memory.
- Explains how to interface both memory types to the Intel 8085 microprocessors.
- 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.

# Memory Pin Connection

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



# Memory Pin Connection

- **Address Connections.** All memory devices have address inputs that select a memory location within the memory device.
- Address inputs are almost always labeled from $\mathbf{A_0}$, the least significant address input, to $\mathbf{A_n}$
  - where subscript **n** can be any value
  - but is always labeled as one less than the total number of address pins.
- For example, a memory device with 10 address pins has its address pins labeled from $A_0$ to $A_9$.
- The number of address pins found 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 ($2^{10}$) memory locations

# Memory Pin Connections – Address Space

- 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

# Memory Pin Connection

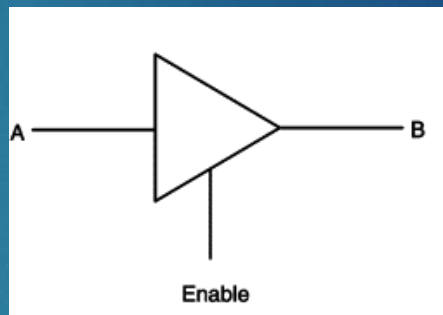
- **Data Connection:** All memory devices have a set of data outputs or input/outputs.
  - today, many devices have **bidirectional** common I/O pins
  - data connections are points at which data are entered for storage or extracted for reading
- Data pins on memory devices are labeled $D_0$ through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

# Memory Pin Connection

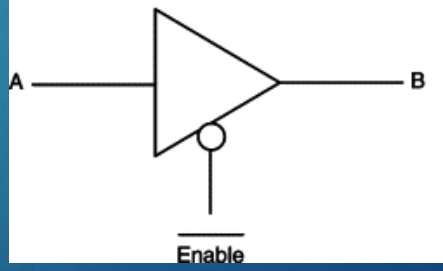
- **Selection Connections**: Each memory device has an input that selects or enables the memory device.
  - sometimes more than one
- This type of input is most often called a **chip select** ($\overline{CS}$) **chip enable** ($\overline{CE}$) or simply **select** ($\overline{S}$) input.

# Memory Pin Connection

- **Control Connection**: 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** ~~fr~~om output data pins.
- The OE connection enables and disables a set of **three-state buffers** located in the device and must be active to read data.
-




| Enable | A | B |
|---|---|---|
| 0 | 0 | Z |
| 0 | 1 | Z |
| 1 | 0 | 0 |
| 1 | 1 | 1 |




| $\overline{\text{Enable}}$ | A | B |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | Z |
| 1 | 1 | Z |

**three-state buffers truth table**

# Memory Pin Connection

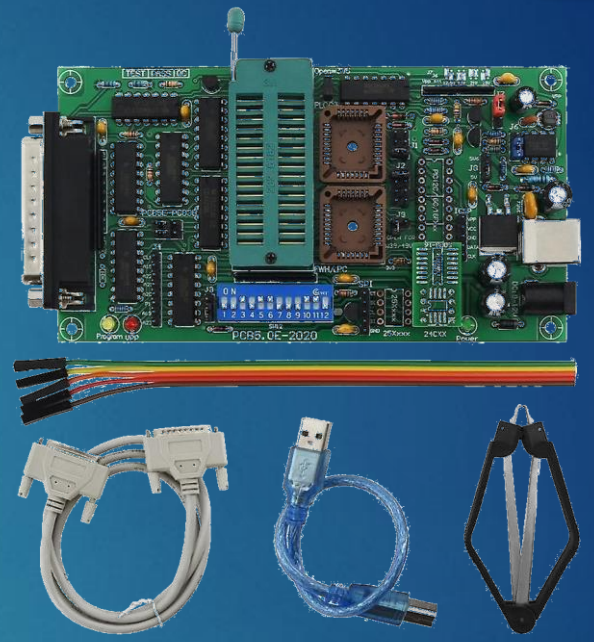
- **Control Connection**: RAM has either one or two control inputs.
  - if one control input, it is often called R/$\overline{W}$
- If the RAM has two control inputs, they are usually labeled $\overline{WE}$ (or $\overline{W}$ ), and $\overline{OE}$ (or $\overline{G}$ ).
  - **write enable** must be 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**
-

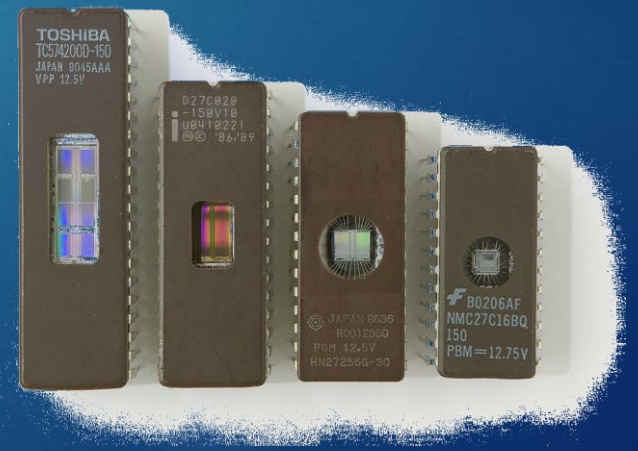
# Characteristics of Memory Devices

- Read-only memory (ROM)
- Flash memory (EEPROM)
- Static random access memory (SRAM)
- Dynamic random access memory (DRAM)

# Read Only Memory (ROM)

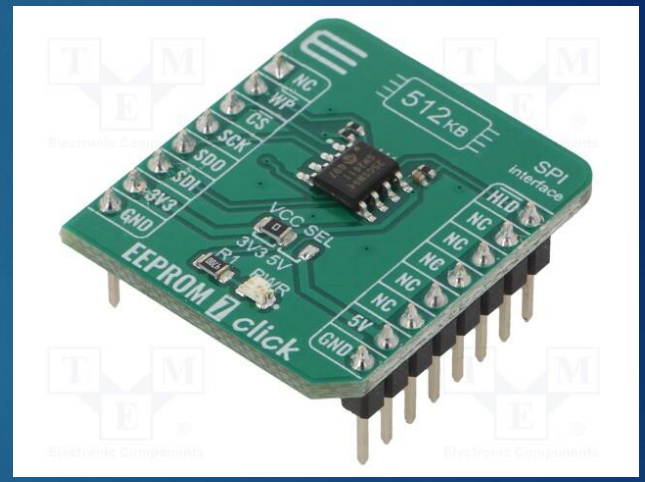
- The read-only memory (ROM) permanently stores programs and data that are resident to the system.
- Must not change when power supply is disconnected.
- The ROM is permanently programmed so that data are always present, even when power is disconnected.
  - This type of memory is often called **nonvolatile memory**.
- ROM is available in many forms.
- The EPROM (erasable programmable read-only memory).
- An EPROM is programmed in the field on a device called an EPROM programmer.
- The EPROM is also erasable if exposed to high-intensity ultraviolet light.
- PROM memory devices are also available, although they are not as common today. Once it is programmed, it cannot be erased.





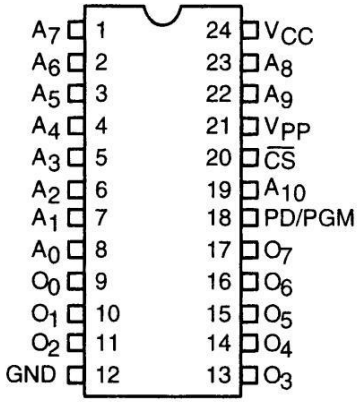
# EEPROM

- Newer type of read-mostly memory (RMM) is called the **flash memory**.
- The flash memory1 is also often called an EEPROM (electrically erasable programmable ROM),
- EAROM (electrically alterable ROM), or a NOVRAM (nonvolatile RAM). These memory devices are electrically erasable in the system, but they require more time to erase than a normal RAM.
- The flash memory device is used to store setup information for systems such as the video card in the computer.
- It has all but replaced the EPROM in most computer systems for the BIOS memory.
- Flash memory has its biggest impact in memory cards for digital cameras and memory in MP3 audio players.



# The Pinout of the Intel 2716 2K x 8 EPROM

PIN CONFIGURATION



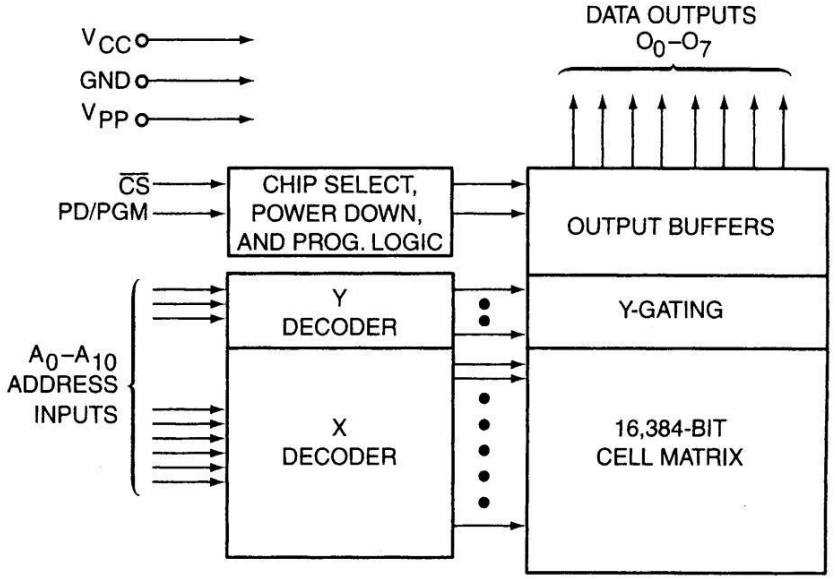

PIN NAMES

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

MODE SELECTION

| MODE \ PINS | 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

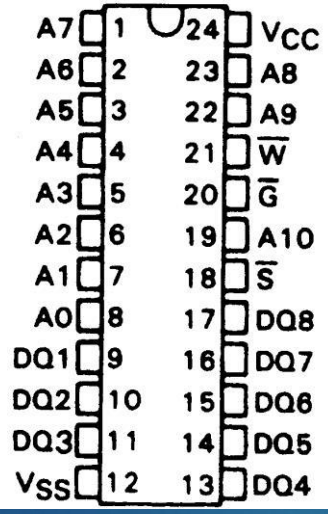


# Static RAM (SRAM) Devices

- 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**.

# Static RAM (SRAM) Devices

**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? (2009), a small memory is less than 1M byte

# Dynamic RAM (DRAM) Memory

- Available up to 256M x 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

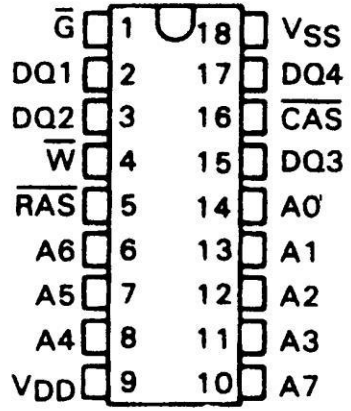
# Dynamic RAM (DRAM) Memory

- 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 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

# Dynamic RAM (DRAM) Memory

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

**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)

# 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.

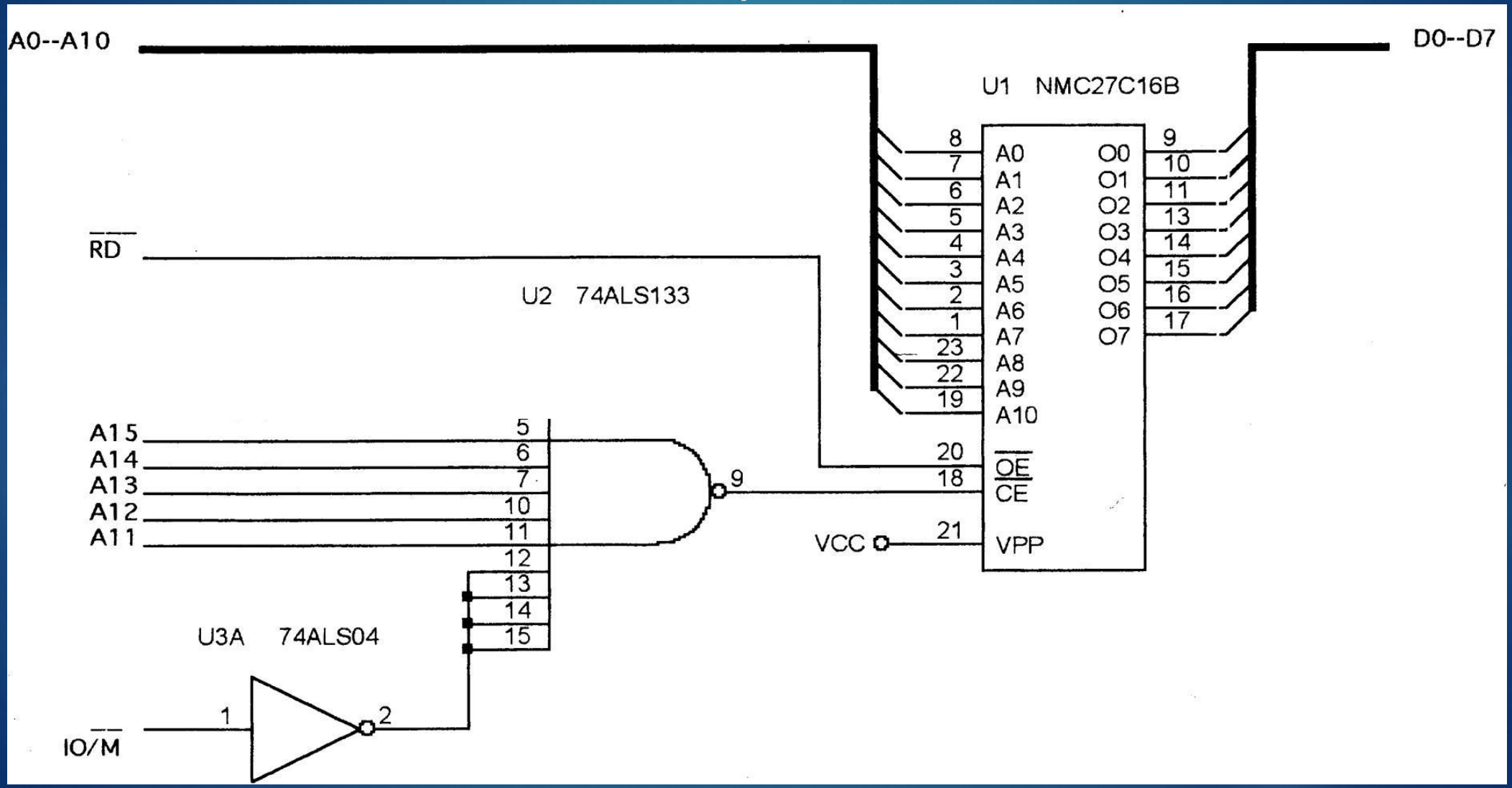
# Why Decode Memory

- The 8085 has 16 address connections and the 2716 EPROM has 11 connections.
- The 8085 sends out a 16-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 x 8 EPROM is used, address connections A10–A0 of 8088 are connected to address inputs A10–A0 of the EPROM.
  - the remaining nine address pins (A19–A11) 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.

# A simple NAND gate decoder that selects a 2716 EPROM for memory location F800H–FFFFH



# A simple NAND gate decoder that selects a 2716 EPROM for memory location F800H–FFFFH

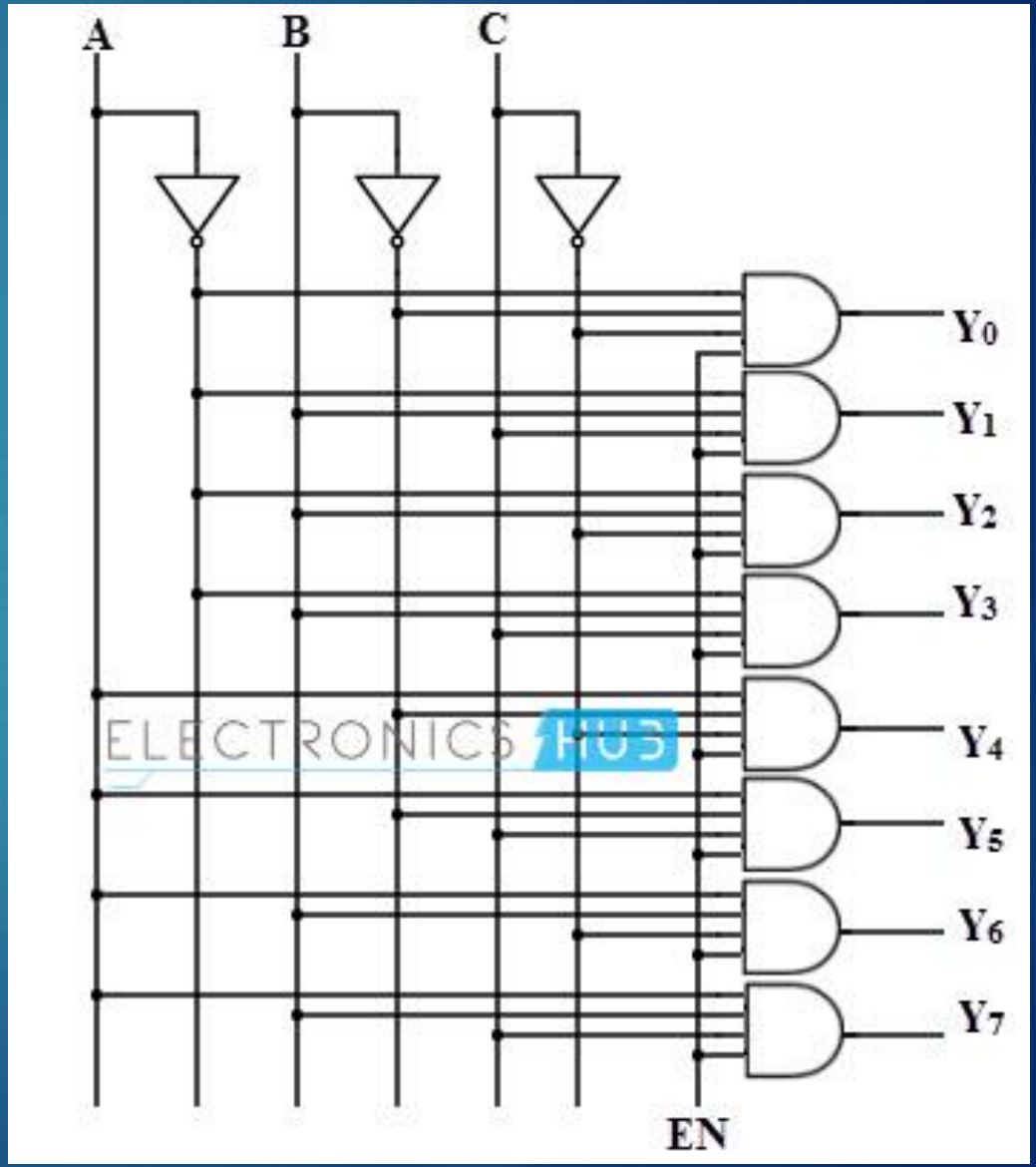
- If the 16-bit binary address, decoded by the NAND gate, is written so that the leftmost nine bits are 1s and the rightmost 11 bits are **don't cares (X)**, the actual address range of the EPROM can be determined.
  - a don't care is a logic 1 or a logic 0, whichever is appropriate
- Because of the 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)

3 to 8 Line Decoder

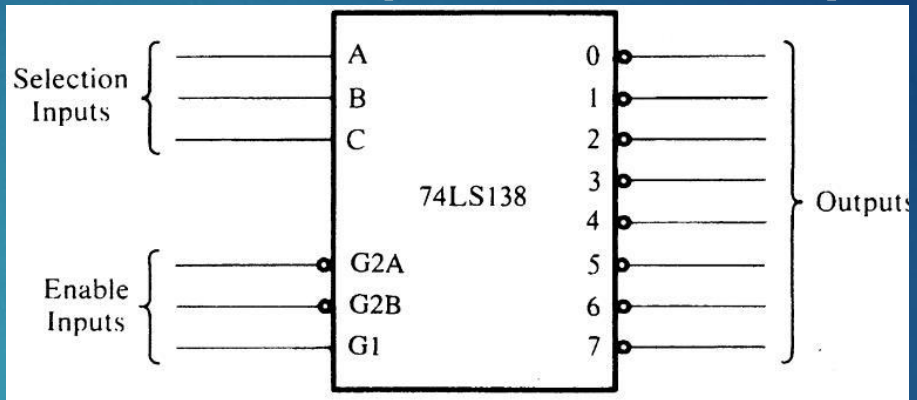
| Inputs | | | Outputs | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| x | y | z | D0 | D1 | D2 | D3 | D4 | D5 | D6 | D7 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

electroniclinic.com



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

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




| 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 |