Microprocessor and Assembly Language

Chapter-Six
Hardware specification, Memory and I/O Interfacing

Outline

- Introduction
- Hardware specification of 8086
- MPU Pin connections
- Memory Interface
- Memory Devices
- Memory Pin Connections
- I/O Interface
- Introduction to I/O Interface
- I/O Port Address Decoding

Introduction

- In this chapter we will examine the 8088 and 8086 microprocessor from the hardware point of view.
- We cover the 8088/8086 signal interfaces, memory interfaces, input output interfaces, and bus cycles.
- The 8086 was the first 16-bit microprocessor introduced by Intel Corporation.
- A second member of the 8086 family, the 8088 microprocessor followed it.
- The 8088 is fully software compatible with its predecessor the 8086.
- The difference between these two devices is in their hardware architecture.

The Intel Family

Addressable Memory, bytes $= 2^A$

TABLE 1–6 The Intel family of microprocessor bus and memory sizes.

Microprocessor	*Data Bus Width*	*Address Bus Width (A)*	*Memory Size*
8086	16	20	1M
8088	8	20	1M
80186	16	20	1M
80188	8	20	1M
80286	16	24	16M
80386SX	16	24	16M
80386DX	32	32	4G
80386EX	16	26	64M
80486	32	32	4G
Pentium	64	32	4G
Pentium Pro–Pentium 4	64	32	4G
Pentium Pro–Pentium 4 (if extended addressing is enabled)	64	36	64G

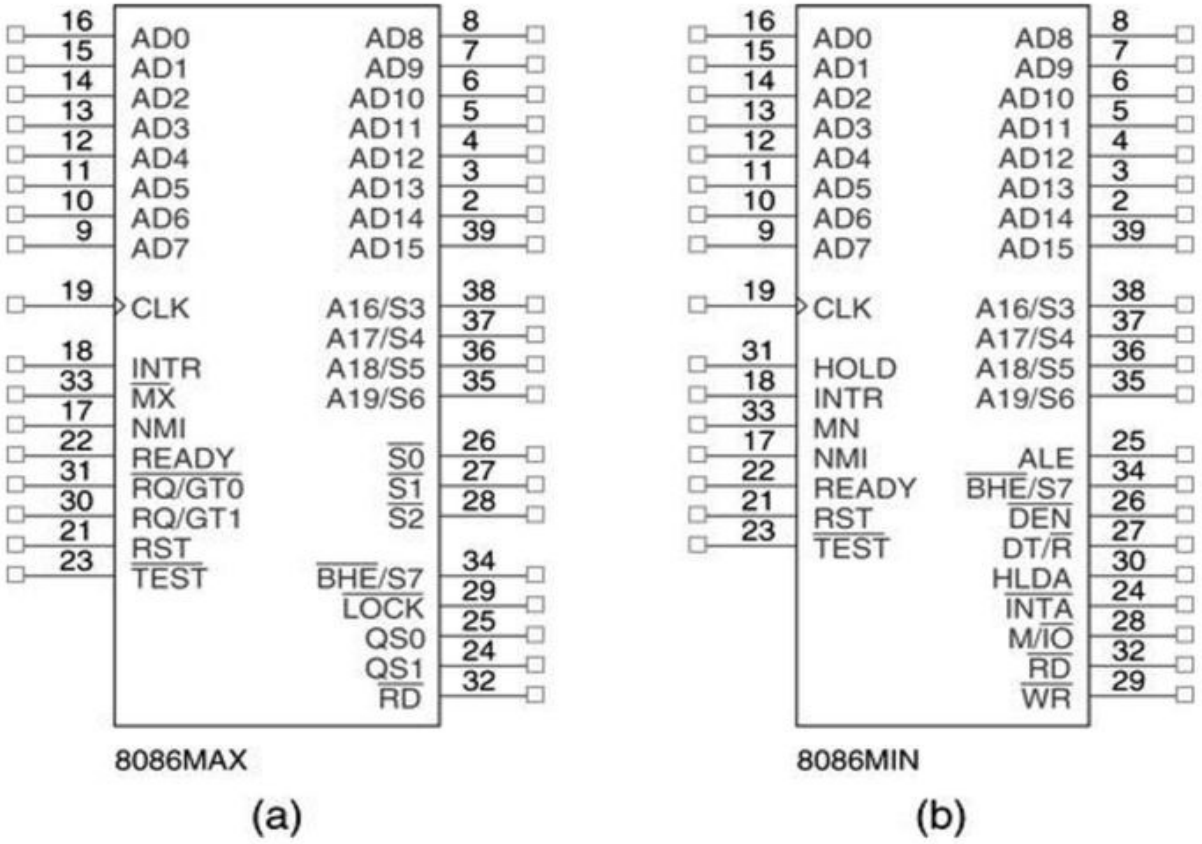
Increase

Hardware specification

- Just like the **8086**, the **8088** is internally a l6bit microprocessors.
- However, externally **8086** has a 16-bit data bus, but 8088 has an 8-bit data bus. **This is the key hardware difference**.
- Both devices have the ability to address up to lMbyte of nemory via their 20-bit address buses.
- Figure on the next slide illustrates the pin-outs of Min/Max 8086 microprocessors. It is packaged in 40-pin dual in-line packages (DIPs).
- In microelectronics, a dual in-line package (DIP or DIL), is a package with a rectangular housing and two parallel rows of pins.
- The 8086 Microprocessor Pin diagram has been designed to work in two operating modes: 1. Minimum Mode Configuration of 8086
  2. Maximum Mode Configuration of 8086

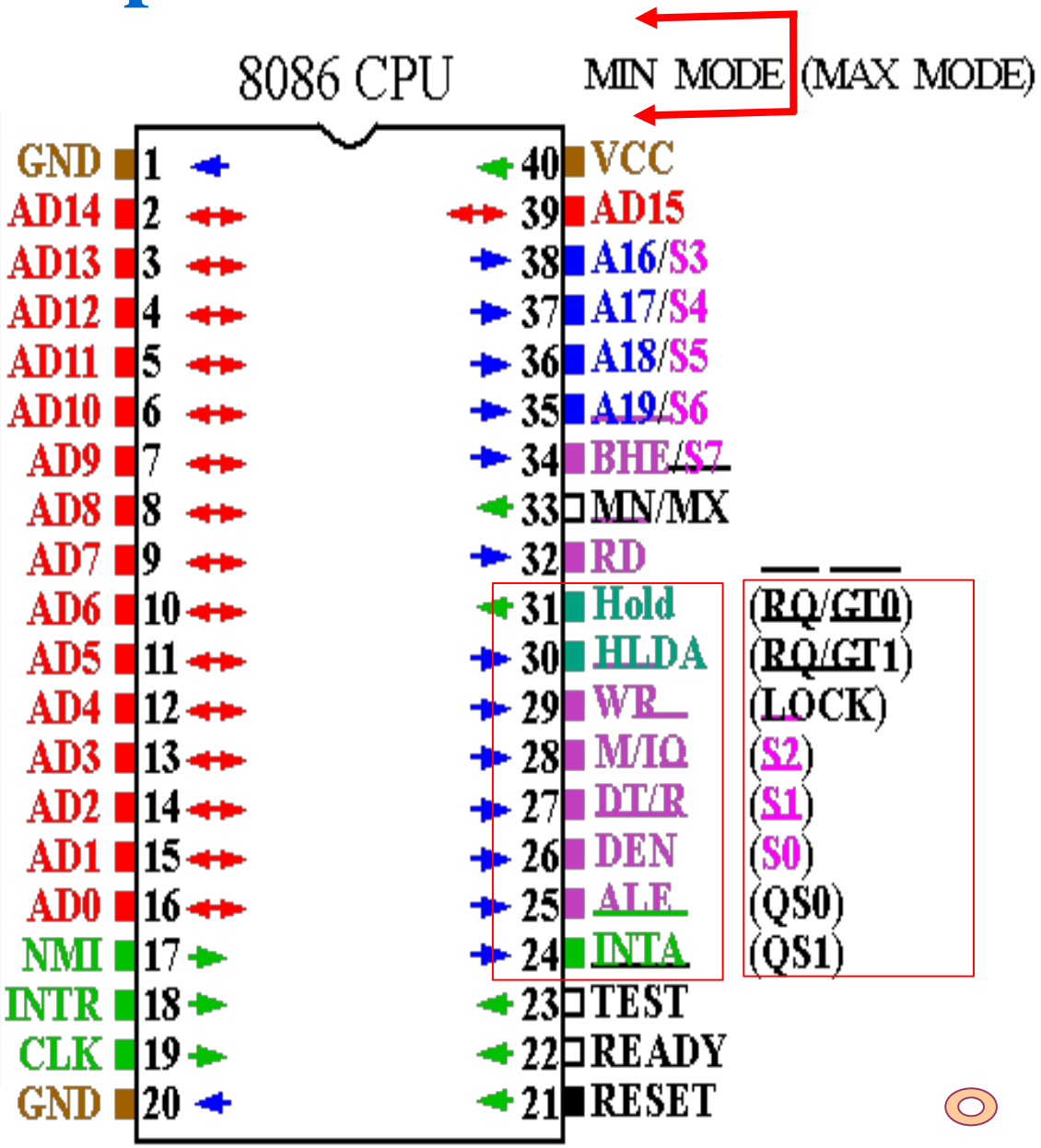
Hardware specification…

Figure 9–1 (a) The pin-out of the 8086 in maximum mode; (b) the pin-out of the 8086 in minimum mode.



Hardware specification...

- Note the pin differences when operated in Max and Min mode here.
- As shown in the next diagram Pin #24 – Pin #31 holds different informations while operating in Min mode and Max modes.
- This is similar to the previous one except pin ordering changes.



Hardware specification...

- Why do we need two operating modes(Max and Min modes)?
- The Minimum Mode Configuration of 8086 is used for a **small systems** that support only a single processor.
- Maximum Mode Configuration of 8086 is used for **medium size to large systems** which supports two or more processors.
- This minimum and maximum mode is controlled by MN/MX pin of microprocessor.(MN/MX=1, minimum and MN/MX=0, maximum).
- In any system there will be data bus, address bus and control buses which are all controlled by the processor.
- All the peripheral devices read or write into the bus under the control of the processor. i.e all control signals are generated by the processor.
- Now what happens if there are more than one processor sitting on the system?

Hardware specification...

- Those all processors will have to share all the resources and of-course the bus too.
- Maximum mode of 8086 is designed to facilitate this, so that 8086 can handle bus request and provide bus control, allowing the second (other) processors to take over.
- As we said earlier, 8088 microprocessor has the same pin-outs with 8086 except variation in few pins.

  i.e , the 8086 has pin connections AD0–AD15, but the 8088 has pin connections AD0–AD7.
- **Address/Data bus**(AD) width is therefore the only major difference between these microprocessors.
- The 16bit data bus allows the 8086 to transfer 16-bit data more efficiently.

Hardware specification...

- Each pins on the microprocessor send control signals, address or data between the processor and other components through the **bus**.
- What is bus?
- **Bus** is a group of conducting wires which carries information, all the other components are connected to microprocessor through this Bus.
- **Three** types of bus are used.

1. **Address bus** - carries memory addresses from the processor to other components such as primary storage and input/output devices. It is used to identify particular locations where data is stored.
2. **Data bus** - carries the data between the processor and other components.
3. **Control bus** - carries control signals from the processor to other components. Such as Read, write....

Pin Connections

- **AD7–AD0:** address/data bus lines are the multiplexed address/data bus of the 8086/88 and contain the rightmost 8 bits of the memory address or I/O port number whenever **ALE is active (logic 1)** or data whenever **ALE is inactive (logic 0)**.
- **AD15–AD8:** address/data bus lines compose the upper multiplexed address/data bus on the 8086. These lines contain address bits A15–A8 whenever **ALE is a logic 1**, and data bus connections D15–D8 when **ALE is a logic 0**.
- **ALE: Address latch enable:** a pin that determines whether the bus AD0 – AD15 carries address or data(if 1= address, 0 = data).
- **A19/S6–A16/S3:** address/status bus bits are multiplexed to provide address signals A19–A16 or status bits S6–S3.
  - Multiplexed mean: is when a single line is carrying multiple informations (≥2) as in AD0 - AD15.

Pin Connections

- Status bit S6 is always a logic 0, bit S5 indicates the condition of the interrupt flag (IF) flag bit.
- S4 and S3 show which segment is accessed during the current bus cycle.

S_4	S_3	Register
0	0	ES
0	1	SS
1	0	CS or none
1	1	DS

- **GND(Ground):** this pin is used to sink the large amount of current flowing through the circuits.
- **CLK:** is used to carry the clock pulse signal from the microprocessor to the externally connected hardware(helps for the synchronization between the microprocessor and external hardware).
- **BHE/S7:** bus high enable pin, is used in the 8086 to enable the most-significant data bus bits during read or write operation.

Pin Connections

- **S2**, **S1** and **S0** :the status bits indicate the function of the current bus cycle.

$\overline{S2}$	$\overline{S1}$	$\overline{S0}$	*Function*
0	0	0	Interrupt acknowledge
0	0	1	I/O read
0	1	0	I/O write
0	1	1	Halt
1	0	0	Opcode fetch
1	0	1	Memory read
1	1	0	Memory write
1	1	1	Passive

- **LOCK:** The lock output is used to lock peripherals off the system.
- **RQ0/GT0** and **RQ1/GT1:** request/grant pins are used in a multi processor system. Local bus masters of other processors force the processor to release the local bus at the end of the processors current bus cycle.

Read the other pin connection...

DeMultiplexing Buses

- The address/data bus on the 8086/8088 is multiplexed (shared) to reduce the number of pins required for the 8086/8088 microprocessor integrated circuit.
- Unfortunately, this burdens the hardware designer with the task of extracting or demultiplexing information from these multiplexed pins.
- Ofcourse there are also some other buses that carry 2 different signals and hence they have to be Dimultiplexed to produce a single signal.
- Multiplexing and dimultiplexing are two opposite terms.
- **Multiplexing** is when a single pin/bus carries several information to reduce the number of pins/bus required by microprocessor.
- **Dimultiplexing** is a process reconverting a bus containing multiple information back into separate.
  - ✓ It is the process of separating address and data from pins AD0–AD15.

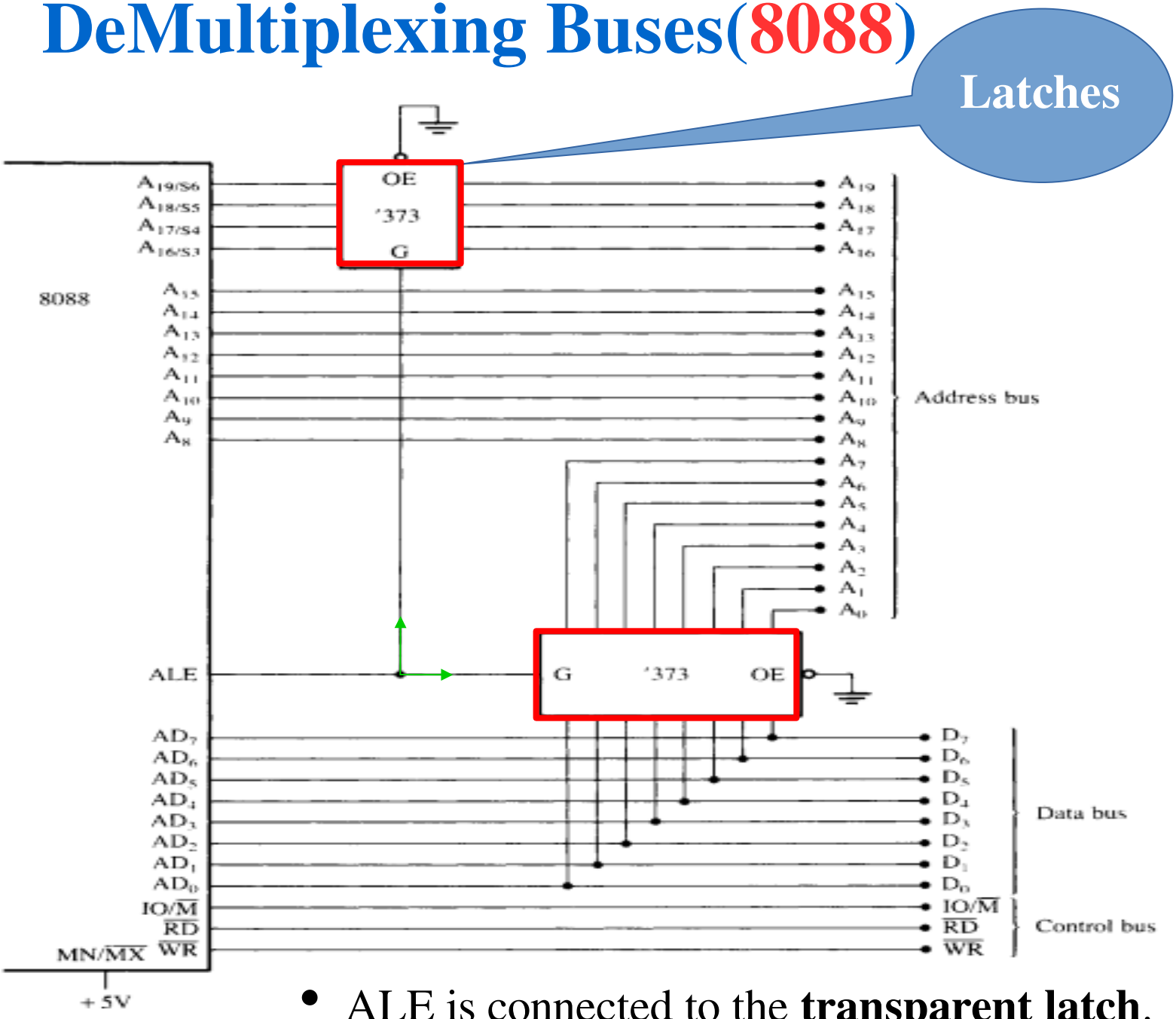
DeMultiplexing Buses(Cont...)

- As we stated previously it is the ALE that determines what information does AD0-AD15 pins carry.
- When 8088 sends out an **address**, it activates (sets high) the ALE, to indicate the information on pins AD0–AD7 is the address (A0–A7).

– When **data** is to be sent out or in, ALE is low, which indicates that AD0 – AD7 will be used as data buses (D0–D7).

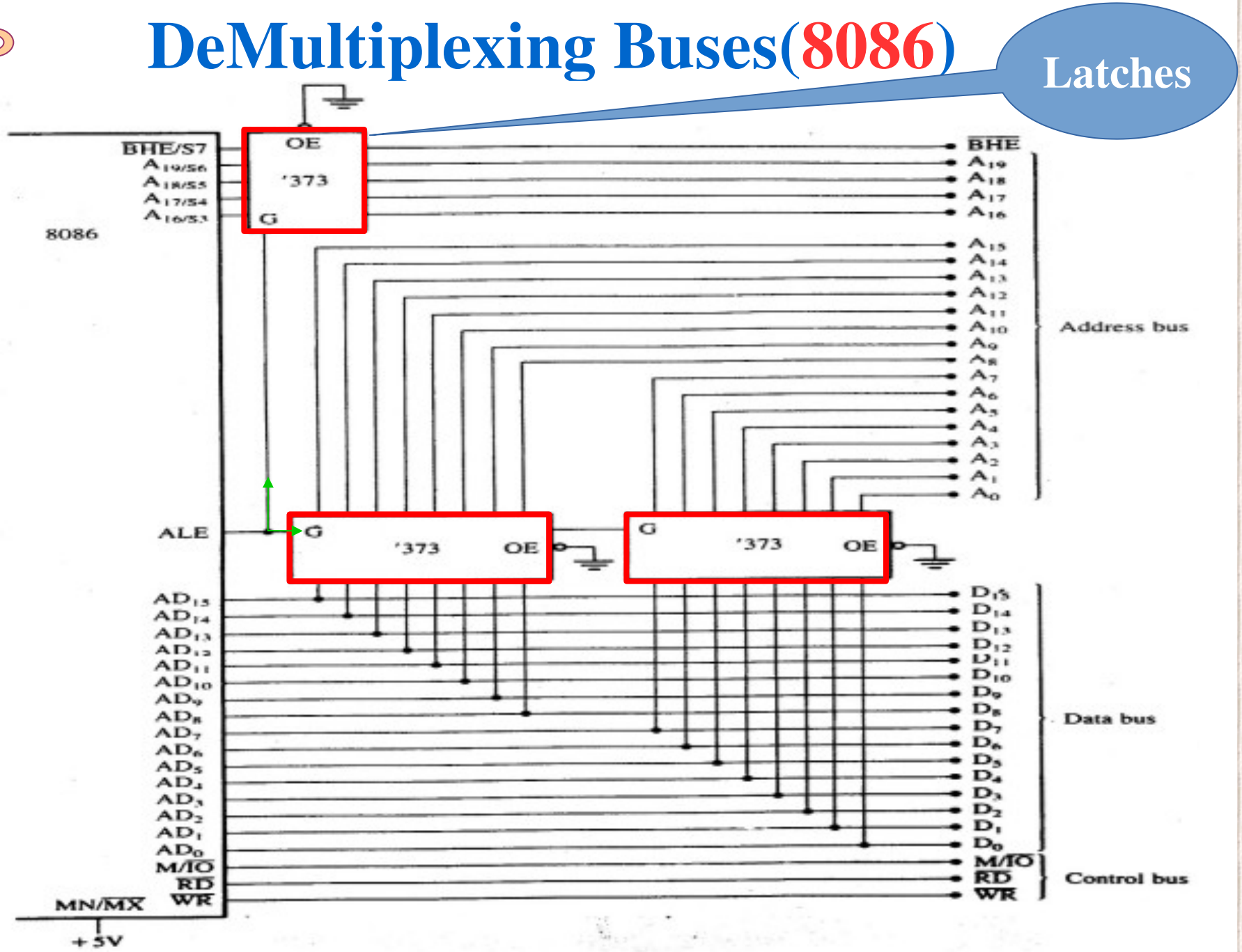
- The ALE will perform this task of dimultiplexing buses through the transparent latches. For instance, next slide shows the 8088 microprocessor in which the ALE is connected to transparent latches.
- In this case, two transparent latches(**74LS373** or **74LS573**) are used to demultiplex the address/data bus connections AD0–AD7 and address/status connections A16/S3–A19/S6.
- A **transparent latch** is a digital logic device that can store two stable states with a level sensitive control signal called enable.

DeMultiplexing Buses(8088)



- ALE is connected to the **transparent latch**.

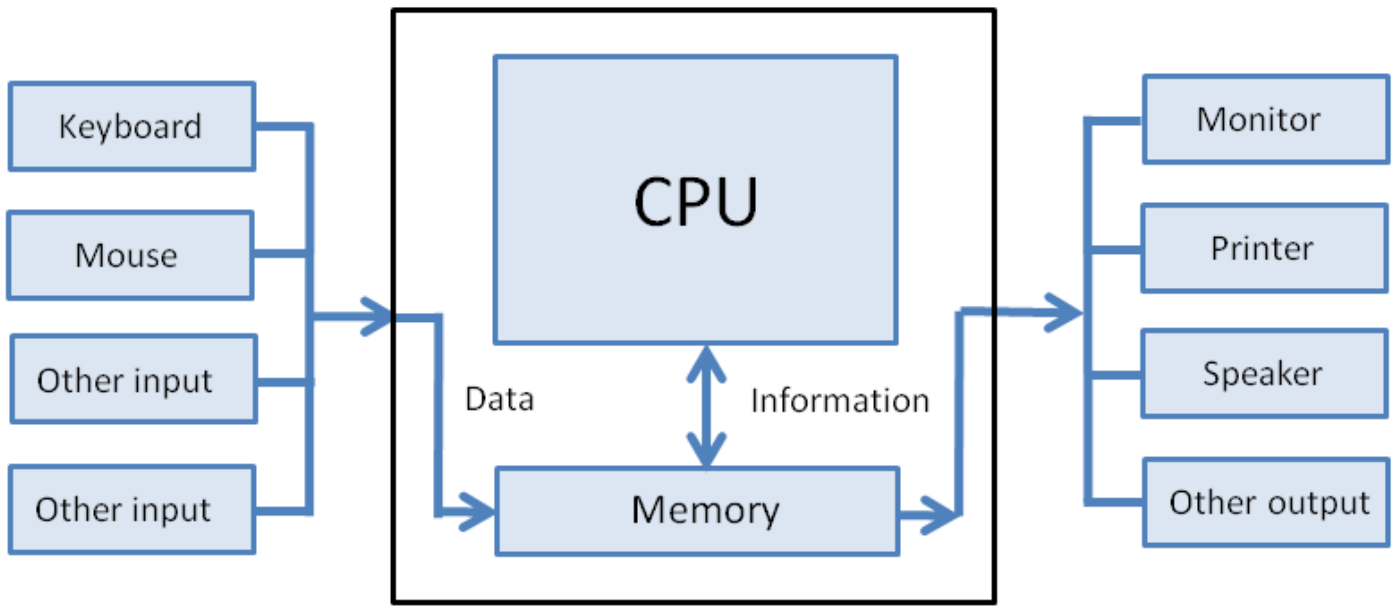
DeMultiplexing Buses(8086)



What is an Interface?

- An **interface** is a concept that refers to a point of interaction between components, and is applicable at the level of both hardware and software.
  - ✓ Its an interconnection point between microprocessor and other components.
- This allows a component, (such as a **graphics card** or an **Internet browser**), to function independently while using interfaces to communicate with other components via an **input/output system** and an **associated protocol**.
- Interfacing is of two types,
  1. Memory interfacing and
  2. I/O interfacing.

Interface….



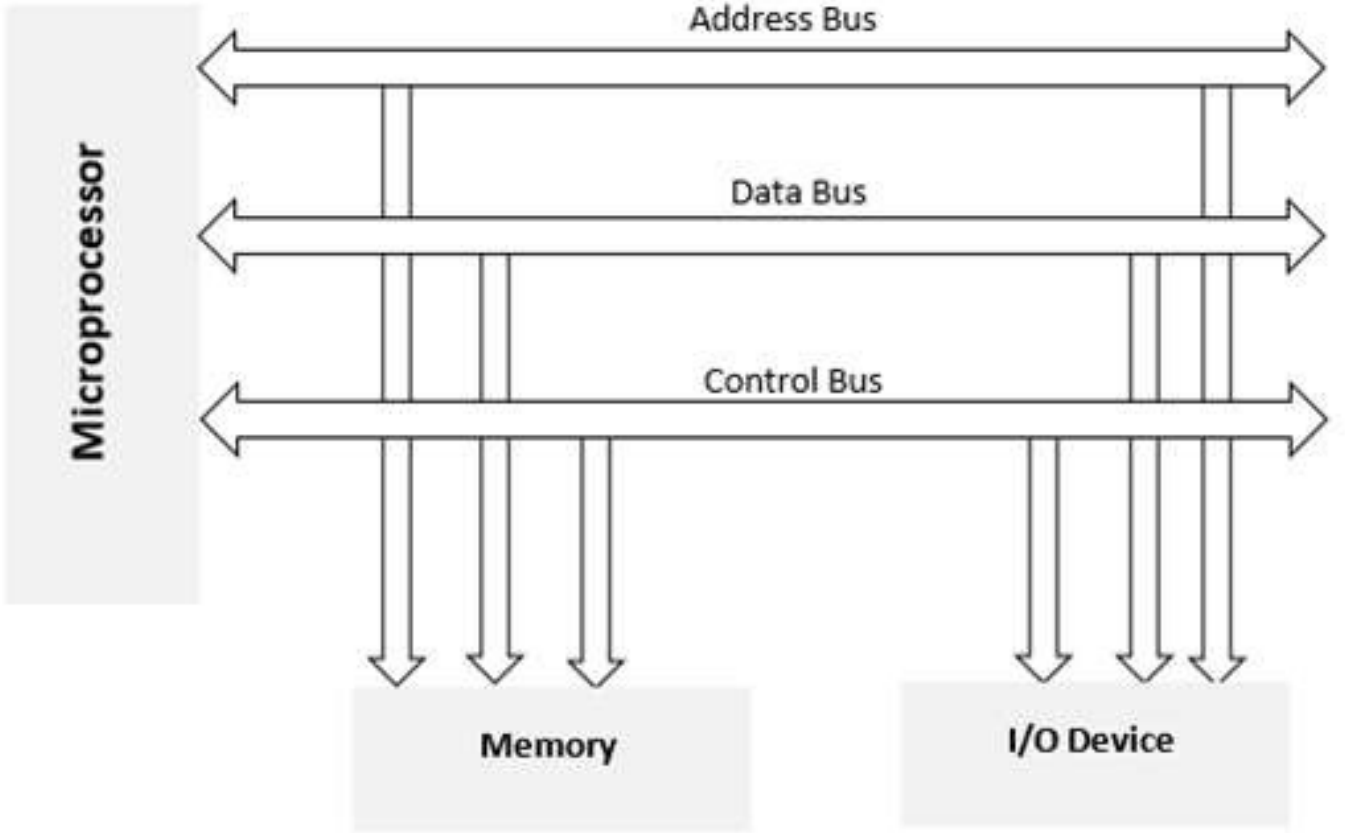
- **Memory Interfacing**: when we are executing any instruction, we need the microprocessor to access the memory for reading instruction codes and the data stored in the memory.
- Thus, interfacing enables that access.
  - i.e, both memory and the microprocessor requires some signals/interface to read from and write to registers.

Interface…..

- The interfacing therefore should be designed in such a way that it **matches** the memory signal requirements and the signals of the microprocessor.
- **IO Interfacing:** it's also called Input-Output interface.
  - It is the communication between various IO devices like keyboard, mouse, printer, etc and CPU.
  - It is where CPU communicates and controls the I/O device with the help of reading and writing.
  - There are also registers that are connected to the CPU using buses.
  - **Note that:** it's pin #28, M/IO pin, that determines the type of interfacing(Memory/IO) that the microprocessor is currently handling.

Interface...

- And weather to write or read from IO or memory depends on the RD and WR pins.
- Block Diagram of Memory and I/O Interfacing



Memory pin connections

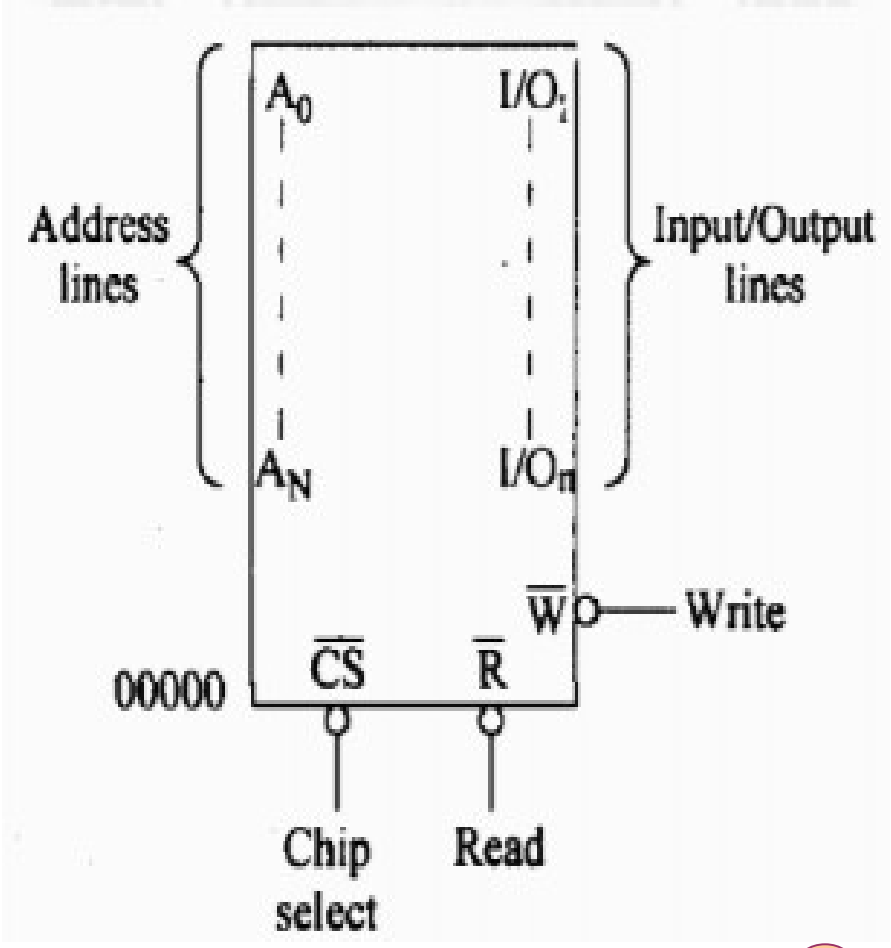
Address Connections: Memory device contain address lines, Input/output lines, selection and control input to perform read or write operation.

- The address inputs are used to select a memory location within the memory device.
- Address inputs are always labeled from $\mathbf{A_0}$ to $\mathbf{A_n}$ where **n**'s value is one less than the total number of address pins.
- A **1K memory** device has **10 address pins** (A0–A9); therefore, 10 address inputs are required to select any of its **1024 memory locations**.
- If a memory device has **11 address connections** (A0–A11), it has **2048 (2K) memory locations**. A **4K memory** device has **12 address** connections. **8K device** has **13**, and so forth.

Memory pin connections

Data Connections: memory devices have a set of data outputs or I/O's. This device may have separate I/O lines or common set of bidirectional I/O lines.

- This pins are points at which data are entered for writing or extracted for reading.
- Using these lines data can be transferred in either direction.
- Whenever output line is activated, the data is **read** and
- Whenever input line are activated the data is **written**.
- These lines are labeled as I/O ... I/On or DO .....Dn.



Memory pin connections

- The **size of each memory location** is dependent upon the number of data bits.
- If the numbers of data lines are **eight**, then 8 bits or 1 byte of data can be stored in each location. Thus, the data pins on this memory devices are labeled as D0 through D7.
  - That is, the memory is 8-bit wide.
- Note: memory device with 1K memory locations and 8 bits in each location is often listed as a **1K × 8** by the manufacturer.
- Similarly if numbers of data bits are **16**, then the memory size(size of each location) is 2 bytes.
- For example **2K x 8** indicates there are 2048 memory locations and each memory location can store 8 bits of data.

Memory pin connections

Selection Connections: Each memory device has one or more inputs that selects or enables the memory device.

- Named as chip select (CS) or chip enable (CE) or simply select (S) input.
  - If this pin is logic 0 - memory device performs read/write.
    logic 1 - the memory chip is disabled.
- **Control Connections:** memory devices have some form of control input like read, write etc.
  - A **ROM** usually has only one control input(Read), while **RAM** often has Read or Write control inputs.
- ROM Memory: read-only memory permanently stores programs and data that are resident to the system and must not change when power supply is disconnected.
  - They are called nonvolatile memory.

Memory pin connections

- When **ROM** is used , OE (output enable) pin allows data to flow out of the output data pins. I.e **R pin**, Read
- To perform this task both CS and OE must be active.
- A RAM contains one or two control inputs. They are R/W or RD/WR.

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.
- Without an address decoder, only one memory device can be connected to a microprocessor, which would make it virtually useless.

Address Decoding(Example...)

- When 8086/88 mp is compared to 2716 EPROM with 11 address pin, there is a difference in the number of address connections.
- B/c this EPROM has 11 address connections and the 8086/888 microprocessor has 20 address pins.
- This means that the microprocessor sends out a 20-bit memory address whenever it reads or writes data.
  - ✓ Meaning, there is a mismatch that must be corrected.
- If only 11 of the 8086/88's address pins are connected to the memory, the 8086/88 will see only **2^{11} = 2K bytes** of memory location instead of the **2^{20} =1M bytes** that it "expects" the memory to contain.
- The decoder corrects the mismatch by decoding the address pins that do not connect to the memory component.

Address Decoding…

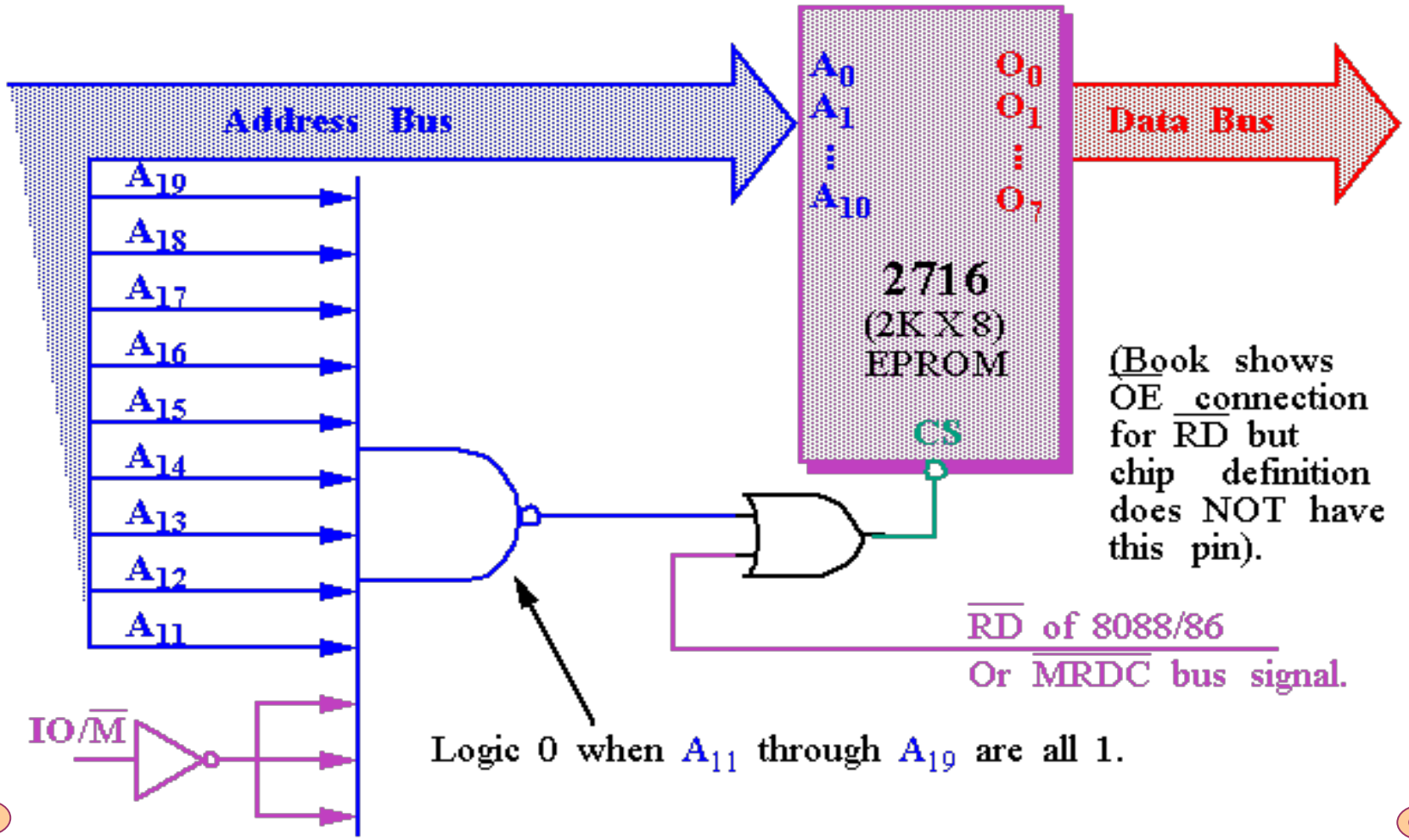
- EPROM with **11 pin connection** can also be represented as **2K × 8** EPROM(Assuming each location store 8bit data).
- Thus, when 2K × 8 EPROM is used, address connections A10–A0 of the 8086/88 are connected to address inputs A10–A0 of the EPROM.
- The remaining nine address pins (A19–A11) are connected to the inputs of a NAND gate decoder (see Figure on the next slide).
- The decoder selects the EPROM from one of the 2K-byte sections of the 1M-byte memory system in the 8086/88 microprocessor.
- What will be the last address when starting address of this 2K × 8 EPROM is FF800H?

Soln: since it's 11 bit address, the last address will be:

1111 1111 1000 0000 0000 - 1111 1111 1**111 1111 1111**

Address Decoding…

- Thus, the remaining nine address pins (A19–A11) allow the decoder to select the EPROM.



Address Decoding…

- In this circuit, a single NAND gate decodes the memory address.
- The output of the NAND gate is a logic 0 whenever the 8086/88 address pins attached to its inputs (A19–A11) are all logic 1s.
- The active low, logic 0 output of the NAND gate decoder is connected to the **CS** input pin that selects (enables) the EPROM.
- But it is also the **RD** pin that determines weather EPROM is selected.

- Although this example serves to illustrate decoding, **NAND** gates are rarely used to decode memory because each memory device requires its own NAND gate decoder.
- Because of the excessive cost of the NAND gate decoder and inverters that are often required, this option requires that an alternate be found.

The End…