# Unit 1
# Lecture 5

# The steps of writing into Memory

▸ What happens when the programmer issues the STA instruction?

  ◦ The microprocessor would turn **on** the WR control (WR = 0) and turn **off** the RD control (RD = 1).

  ◦ The address is applied to the address decoder which generates a **single** Enable signal to turn on **only one** of the memory registers.

  ◦ The data is then applied on the data lines and it is stored into the enabled register.

# Dimensions of Memory

- Memory is usually measured by two numbers: its length and its width (Length X Width).
  - The length is the total number of locations.
  - The width is the number of bits in each location.

  - The length (total number of locations) is a function of the number of address lines.

  $$\text{\# of memory locations} = 2^{(\text{\# of address lines})}$$

    - So, a memory chip with 10 address lines would have
    $$2^{10} = 1024 \text{ locations (1K)}$$

    - Looking at it from the other side, a memory chip with 4K locations would need
    $$\text{Log}_2\ 4096 = 12 \text{ address lines}$$

# The 8085 and Memory

- The 8085 has 16 address lines. That means it can address

$$2^{16} = 64K \text{ memory locations.}$$

  ◦ Then it will need 1 memory chip with 64 k locations, or 2 chips with 32 K in each, or 4 with 16 K each or 16 of the 4 K chips, etc.

- how would we use these address lines to control the multiple chips?

# Chip Select

▸ Usually, each memory chip has a CS (Chip Select) input. The chip will only work if an active signal is applied on that input.

▸ To allow the use of multiple chips in the make up of memory, we need to use a number of the address lines for the purpose of "chip selection".

  ◦ These address lines are decoded to generate the $2^n$ necessary CS inputs for the memory chips to be used.

# Chip Selection Example

◦ Assume that we need to build a memory system made up of 4 of the 4 X 4 memory chips we designed earlier.

◦ We will need to use 2 inputs and a decoder to identify which chip will be used at what time.

◦ The resulting design would now look like the one on the following slide.

# Memory Map and Addresses

The memory map is a picture representation of the address range and shows where the different memory chips are located within the address range.
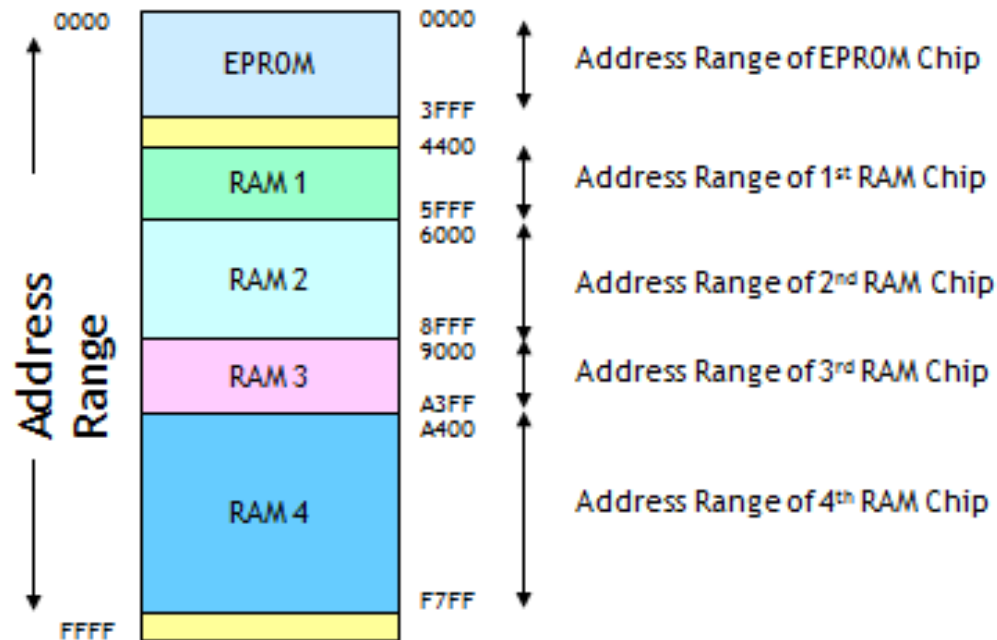
# Address Range of a Memory Chip

- The address range of a particular chip is the list of all addresses that are mapped to the chip.

  - An example for the address range and its relationship to the memory chips would be the Post Office Boxes in the post office.
    - Each box has its unique number that is assigned sequentially. (memory locations)
    - The boxes are grouped into groups. (memory chips)
    - The first box in a group has the number immediately after the last box in the previous group.

# Input and output devices

Microprocessor need to Identify I/O devices with binary number.

IO devices can be interfaced:
●**Memory-Mapped I/O** (using addresses from memory space).Device is identified by 16-bit address (Space ranges from 0000H –FFFFH

●**Standard I/O mapped or isolated I/O mapping /Peripheral Mapped I/O** has separate numbering scheme for I/O devices. Instructions IN/OUT are used for data transfer. Device is identified by 8-bit address (Space ranges from 00H –FFH)

| Memory Mapping of I/O device | I/O Mapping of I/O device |
| --- | --- |
| 1. 16-bit addresses are provided for I/O devices. | 1. 8-bit addresses are provided for I/O devices. |
| 2. The devices are accessed by memory read or memory write cycles. | 2. The devices are accessed by I/O read or I/O write cycle. During these cycles the 8-bit address is available on both low order address lines and high order address lines. |
| 3. The I/O ports or peripherals can be treated like memory locations and so all instructions related to memory can be used for data transfer between I/O device and the processor. | 3. Only IN and OUT instructions can be used for data transfer between I/O device and the processor. |
| 4. In memory mapped ports the data can be moved from any register to ports and vice-versa. | 4. In I/O mapped ports the data transfer can take place only between the accumulator and ports. |
| 5. When memory mapping is used for I/O devices, the full memory address space cannot be used for addressing memory. Hence memory mapping is useful only for small systems, where the memory requirement is less. | 5. When I/O mapping is used for I/O devices then the full memory address space can be used for addressing memory. Hence it is suitable for systems which requires large memory capacity. |
| 6. In memory mapped I/O devices, a large number of I/O ports can be interfaced. | 6. In I/O mapping only 256 ports $(2^8 = 256)$ can be interfaced. |
| 7. For accessing the memory mapped devices, the processor executes memory read or write cycle. During this cycle IO/$\overline{\text{M}}$ is asserted **low** (IO/$\overline{\text{M}}$ = 0). | 7. For accessing the I/O mapped devices, the processor executes I/O read or write cycle. During this cycle IO/$\overline{\text{M}}$ is asserted **high** (IO/$\overline{\text{M}}$ = 1). |