MEMORY SEGMENTATION OF INTEL 8086

Memory Segmentation

- The total memory size is divided into segments of various sizes.
- A segment is just an area in memory.
- The process of dividing memory this way is called **Segmentation**.

Memory Segmentation

- In memory, data is stored as bytes.
- Each byte has a specific address.
- Intel 8086 has 20 lines address bus.
- With 20 address lines, the memory that can be addressed is 2^{20} bytes.
- 2^{20} = 1,048,576 bytes (1 MB).
- 8086 can access memory with address ranging from 00000 H to FFFFF H.

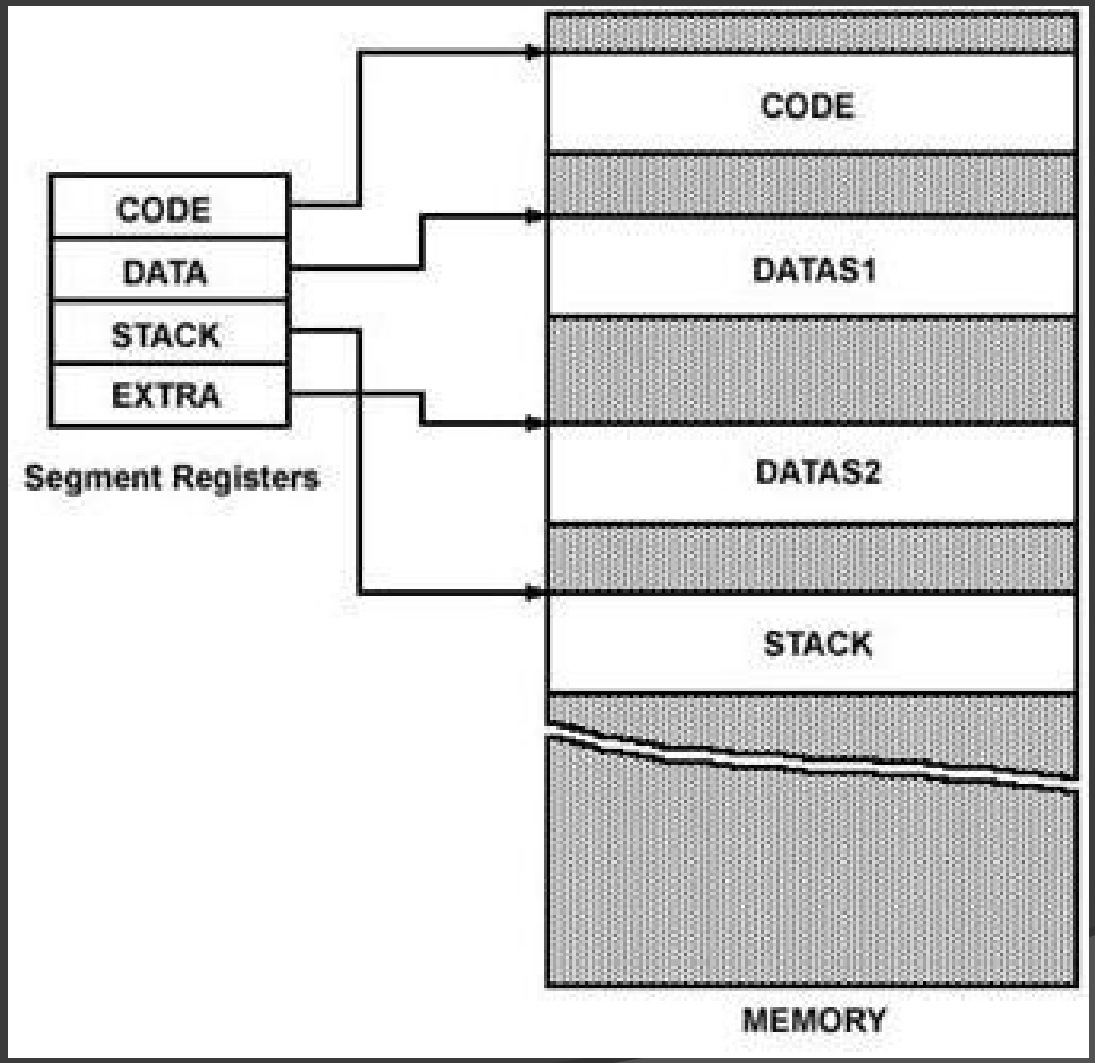
Memory Segmentation

- In 8086, memory has four different types of segments.
- These are:
  - Code Segment
  - Data Segment
  - Stack Segment
  - Extra Segment

Segment Registers

- Each of these segments are addressed by an address stored in corresponding segment register.
- These registers are 16-bit in size.
- Each register stores the base address (starting address) of the corresponding segment.
- Because the segment registers cannot store 20 bits, they only store the upper 16 bits.

Segment Registers



Segment Registers

- How is a 20-bit address obtained if there are only 16-bit registers?
- The answer lies in the next few slides.
- The 20-bit address of a byte is called its **Physical Address**.
- But, it is specified as a **Logical Address**.
- Logical address is in the form of:

Base Address : Offset

- Offset is the displacement of the memory location from the starting location of the segment.

Example

- The value of Data Segment Register (DS) is 2222 H.
- To convert this 16-bit address into 20-bit, the BIU appends 0H to the LSBs of the address.
- After appending, the starting address of the Data Segment becomes 22220H.

Example (Contd.)

- If the data at any location has a logical address specified as:

  2222 H : 0016 H

- Then, the number 0016 H is the offset.
- 2222 H is the value of DS.

Example (Contd.)

- To calculate the effective address of the memory, BIU uses the following formula:
  - Effective Address = Starting Address of Segment + Offset
- To find the starting address of the segment, BIU appends the contents of Segment Register with 0H.
- Then, it adds offset to it.

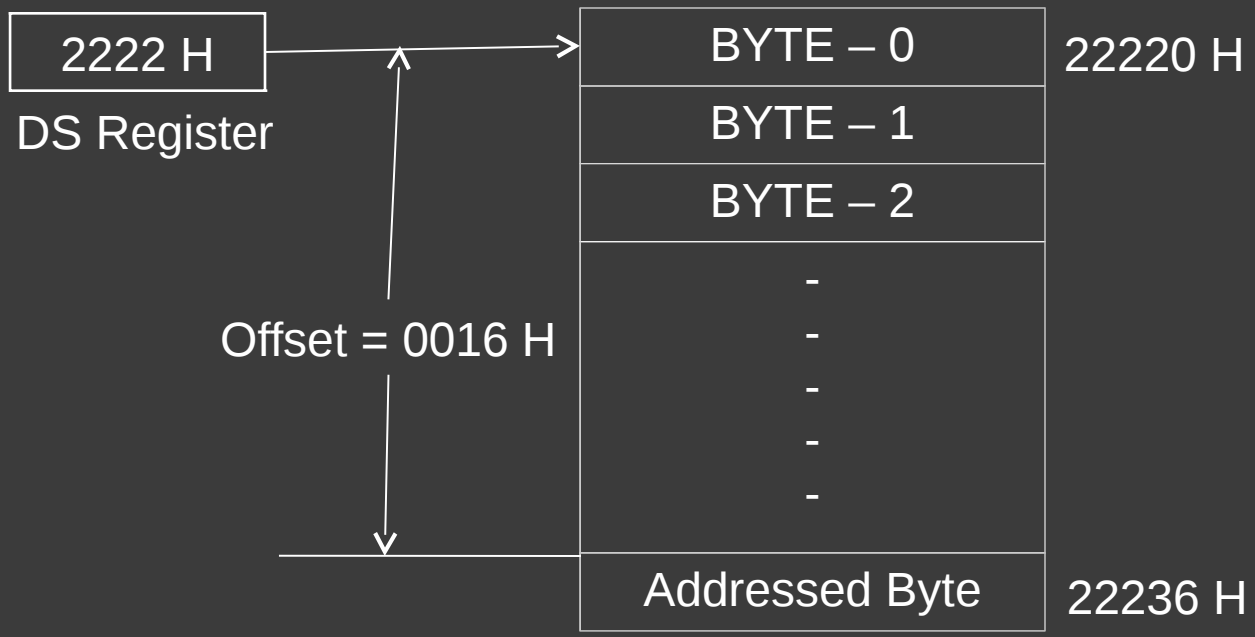
Example (Contd.)

- Therefore:
- EA = 22220 H

  + 0016 H

  22236 H

Example (Contd.)



Max. Size of Segment

- All offsets are limited to 16-bits.
- It means that the maximum size possible for segment is 2^{16} = 65,535 bytes (64 KB).
- The offset of the first location within the segment is 0000 H.
- The offset of the last location in the segment is FFFF H.

Where to Look for the Offset

Segment	Offset Registers	Function
CS	IP	Address of the next instruction
DS	BX, DI, SI	Address of data
SS	SP, BP	Address in the stack
ES	BX, DI, SI	Address of destination data (for string operations)

Question

- The contents of the following registers are:
  - CS = 1111 H
  - DS = 3333 H
  - SS = 2526 H
  - IP = 1232 H
  - SP = 1100 H
  - DI = 0020 H
- Calculate the corresponding physical addresses for the address bytes in CS, DS and SS.

Solution

1. CS = 1111 H

- The base address of the code segment is 11110 H.
- Effective address of memory is given by 11110H + 1232H = 12342H.

2. DS = 3333 H

- The base address of the data segment is 33330 H.
- Effective address of memory is given by 33330H + 0020H = 33350H.

3. SS = 2526 H

- The base address of the stack segment is 25260 H.
- Effective address of memory is given by 25260H + 1100H = 26350H.