Registers of 8086: Pointers and Index Group

Pointers and Index Registers in 8086

> **Overview**
>
> The 8086 microprocessor contains a special group of registers called the **Pointers and Index Registers**. These are **16-bit registers** used to store **offset addresses of memory locations**. They play a key role in addressing memory, stack operations, and string manipulations.

Stack Pointer (SP)

- Holds the **offset address of the top of the stack**.

- The actual physical address is calculated as:

  $$\text{Stack Address} = (SS \times 10H) + SP$$

- Automatically updated (incremented/decremented) during `PUSH` and `POP`.

- Used exclusively for stack operations.

Source Index (SI)

- Used as a pointer to the **source string**.

- Physical address calculation:

  $$\text{Source Address} = (DS \times 10H) + SI$$

- Example: `MOV AH, [SI]` If $SI = 2000H$, then `AH` $\leftarrow$ `[2000H]`.

- Automatically incremented or decremented in string instructions depending on DF (Direction Flag).

Base Pointer (BP)

- Also points to the **stack segment**.

- Provides an alternative way to access data in the stack.

- Physical address calculation:

  $$\text{Address} = (SS \times 10H) + BP$$

- Often used in high-level languages to access function parameters and local variables.

Destination Index (DI)

- Used as a pointer to the **destination string**.

- Physical address calculation:

$$\text{Destination Address} = (ES \times 10H) + DI$$

- Works together with SI in string manipulation instructions such as `MOVSB`, `CMPS`, and `SCAS`.

- Like SI, it auto-increments/decrements based on DF.

Instruction Pointer (IP)

- Contains the **offset address of the next instruction** to be fetched by the CPU.

- The physical address is calculated as:

$$\text{Code Address} = (CS \times 10H) + IP$$

- Unlike other registers, **IP cannot be directly modified by the programmer**.

- Automatically updated during sequential execution and branch/jump instructions.
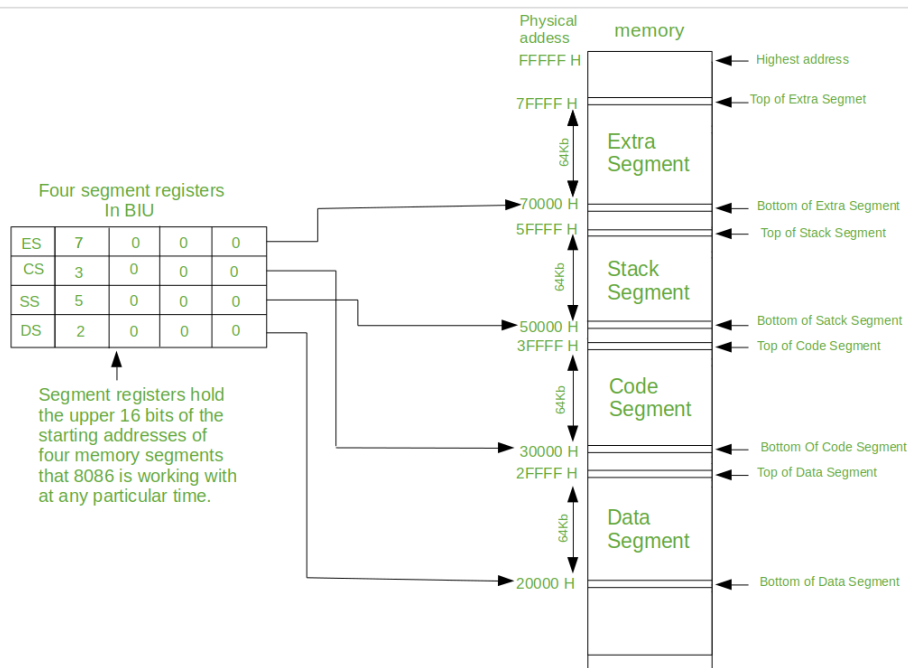
Memory Segments and Pointer Registers

Figure: Pointer and Index Registers pointing to their respective memory segments in 8086

> **Additional Notes**
>
> - **SP, BP, SI, DI** are general-purpose registers but optimized for addressing.
>
> - **IP** is dedicated to instruction sequencing and cannot be modified directly.
>
> - These registers make **stack handling, string operations, and instruction execution** efficient.