Registers of 8086 Microprocessor

Overview

The Intel 8086 microprocessor has a rich set of registers used for data storage, address calculation, and control purposes. Registers are **fast storage units** inside the CPU that allow quick access to frequently used data and instructions.
Registers are broadly categorized into:

- **General Purpose Registers (GPRs)**
- **Segment Registers**
- **Pointer and Index Registers**
- **Instruction Pointer (IP)**
- **Flag Register**

1. General Purpose Registers (16-bit)

General Purpose Registers

- AX (Accumulator) – Used for arithmetic, logic, and I/O operations.
- BX (Base) – Used as a base pointer for memory access.
- CX (Count) – Used as loop counter and for string operations.
- DX (Data) – Used in I/O operations and multiply/divide instructions.

Each 16-bit GPR can be split into 2 8-bit registers:

- AX → AH (high byte), AL (low byte)
- BX → BH, BL
- CX → CH, CL
- DX → DH, DL

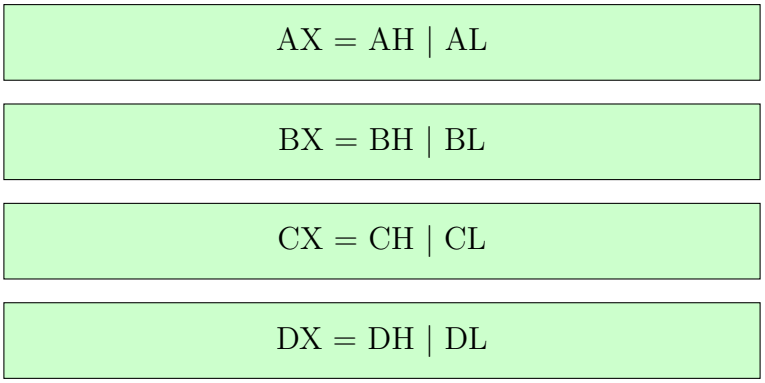

Figure: General Purpose Registers of 8086

2. Segment Registers

Segment Registers

Segment registers are used to calculate **physical memory addresses** in the segmented memory model.

- **CS (Code Segment)** – Points to the segment containing the current program code.
- **DS (Data Segment)** – Points to the segment containing data variables.
- **SS (Stack Segment)** – Points to the segment containing the stack.
- **ES (Extra Segment)** – Extra data segment for string operations.

Physical Address Calculation:

$$\text{Physical Address} = (\text{Segment} \times 16) + \text{Offset}$$

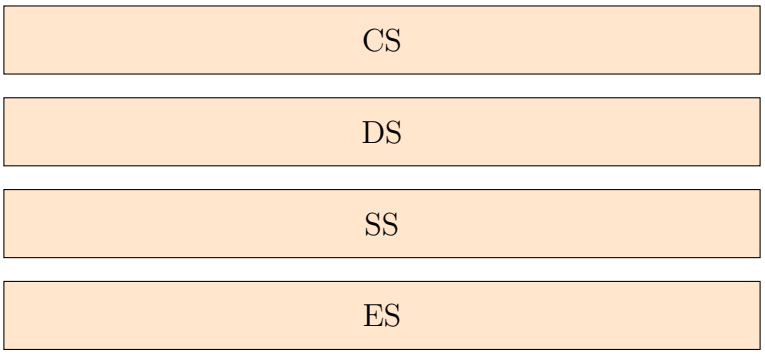

Figure: Segment Registers of 8086

3. Pointer and Index Registers

Pointer and Index Registers

- **SP (Stack Pointer)** – Points to the top of the stack (relative to SS).
- **BP (Base Pointer)** – Used for accessing parameters in the stack.
- **SI (Source Index)** – Source pointer for string operations.
- **DI (Destination Index)** – Destination pointer for string operations.

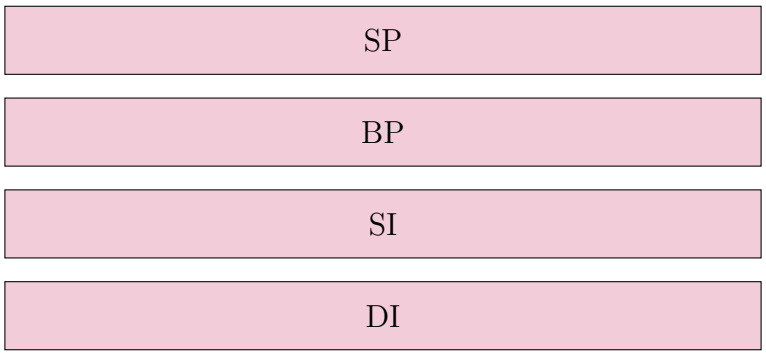

Figure: Pointer and Index Registers of 8086

4. Instruction Pointer (IP)

Instruction Pointer

- 16-bit register pointing to the next instruction in the code segment.
- Works with CS to generate 20-bit physical addresses:

$$\text{Physical Address} = (\text{CS} \times 16) + \text{IP}$$

- Automatically increments as instructions are fetched.

5. Flag Register (Status Register)

Flag Register

- A 16-bit register that indicates the **status of the processor** after arithmetic or logical operations.
- Contains **condition flags**:
  - **Carry (CF)** – Set if arithmetic carry/borrow occurs.
  - **Parity (PF)** – Set if number of 1s in result is even.
  - **Auxiliary Carry (AF)** – Carry from lower nibble.
  - **Zero (ZF)** – Set if result is zero.
  - **Sign (SF)** – Set if result is negative.
  - **Overflow (OF)** – Set if signed overflow occurs.
- Contains **control flags**:
  - **Trap (TF)**, **Interrupt Enable (IF)**, **Direction (DF)**, etc.

6. Visual Diagram of Registers

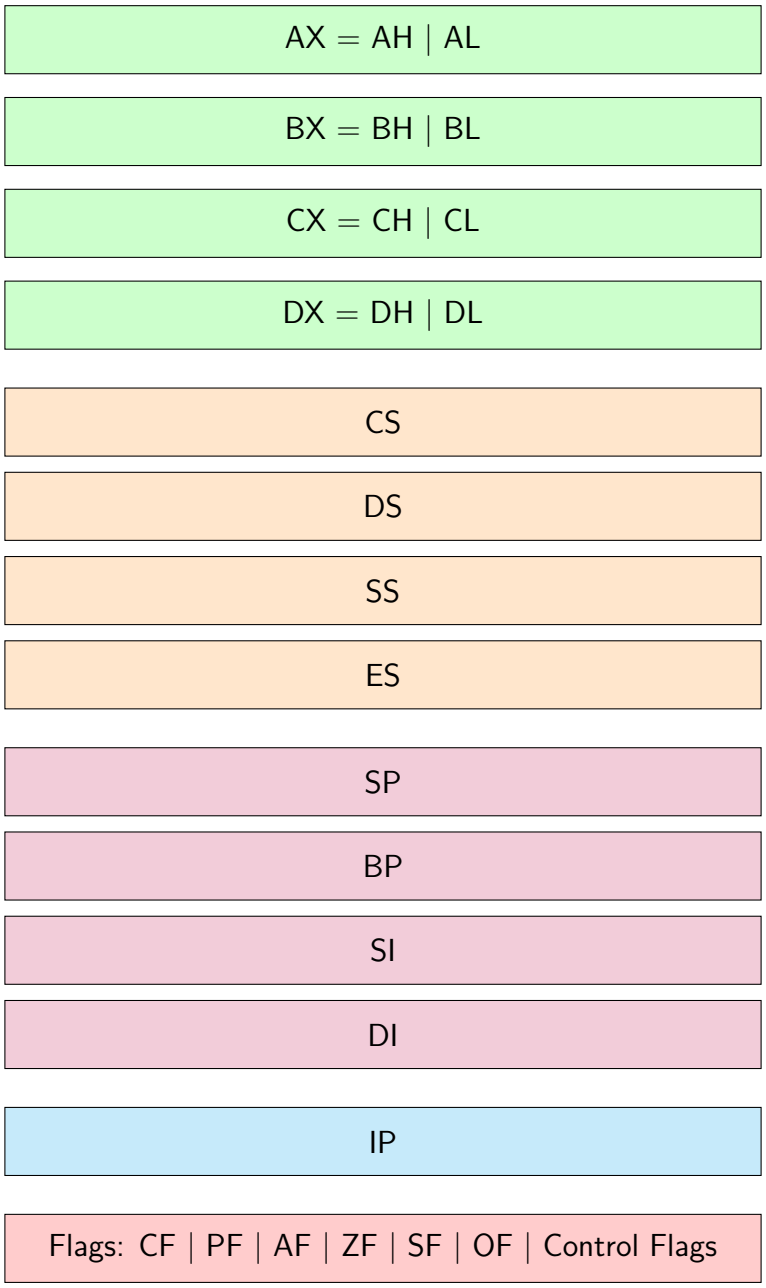

Figure: 8086 Registers Layout with CS:IP for Physical Address

Summary

- 8086 has a structured register set for data storage, address calculation, and control.
- **General Purpose Registers** – For arithmetic, logic, and temporary data.
- **Segment Registers** – Generate 20-bit physical addresses with IP.
- **Pointer and Index Registers** – For stack and string operations.
- **Instruction Pointer (IP)** – Points to the next instruction and works with CS.
- **Flag Register** – Tracks CPU status and controls operations.