



[MICROPROCESSORS ARCITECTURE]

[Chapter2: 8086 Registers]

Dr. DARIN Shafek Salim

Chapter Content

- **Introduction**
- **Features**
- **Internal Block Diagram**
 - **The Bus Interface Unit (BIU)**
 - **The Execution Unit (EU)**

Introduction

The 8086 is a 16-bit microprocessor intended to be used as the CPU in a microcomputer. The term “16-bit” means that its arithmetic logic unit, internal registers, and most of its instructions are designed to work 16-bit binary words. It has 16-bit data bus and 20-bit address bus. Words will be stored in two consecutive memory locations. If the first byte of a word is at an even address, the 8086 can read the entire word in one operation. If the first byte of the word is at an odd address, the 8086 will read the first byte in one operation, and the second byte in another operation.

Features

- ✚ 8086 is a 40 pin IC.
- ✚ It is a 16-bit processor.
- ✚ Its operating voltage is 5 volts.
- ✚ Its operating frequency is 5 MHz
- ✚ The total memory addressing capacity is 1MB (external).
- ✚ It has 16-bit data bus and 20-bit address bus.
- ✚ It has fourteen 16-bit registers.
- ✚ It has around 20000 transistors in its circuitry and it is made in HMOS technology.
- ✚ Pipelining improves the performance of the processor so that operation is faster. 8086 uses two stage of pipelining.

- ✚ First is Fetch Stage and the second is Execute Stage.
- ✚ Fetch stage that prefetch up to 6 bytes of instructions stores them in the queue.
- ✚ Execute stage that executes these instructions.
- ✚ Operates in two modes: 8086 operates in two modes: Minimum Mode: A system with only one microprocessor. Maximum Mode: A system with multiprocessor.
- ✚ 8086 uses memory banks: The 8086 uses a memory banking system. It means entire data is not stored sequentially in a single memory of 1 MB but memory is divided into two banks of 512KB.
- ✚ **Interrupts:**8086 has 256 vectored interrupts.
- ✚ **Multiplication and Division:**8086 has a powerful instruction set. So that it supports Multiply and Divide operation.

Internal Block Diagram

The 8086 CPU is divided into two independent functional parts, the bus interface unit or BIU, and the execution unit or EU.

❖ The Bus Interface Unit

The BIU handles all data and addresses on the buses for the execution unit such as it sends out addresses, fetches instructions from memory, reads data from ports and memory as well as writes data to ports and memory. In BIU there are so many functional groups or parts these are as follows.

○ Instruction Queue

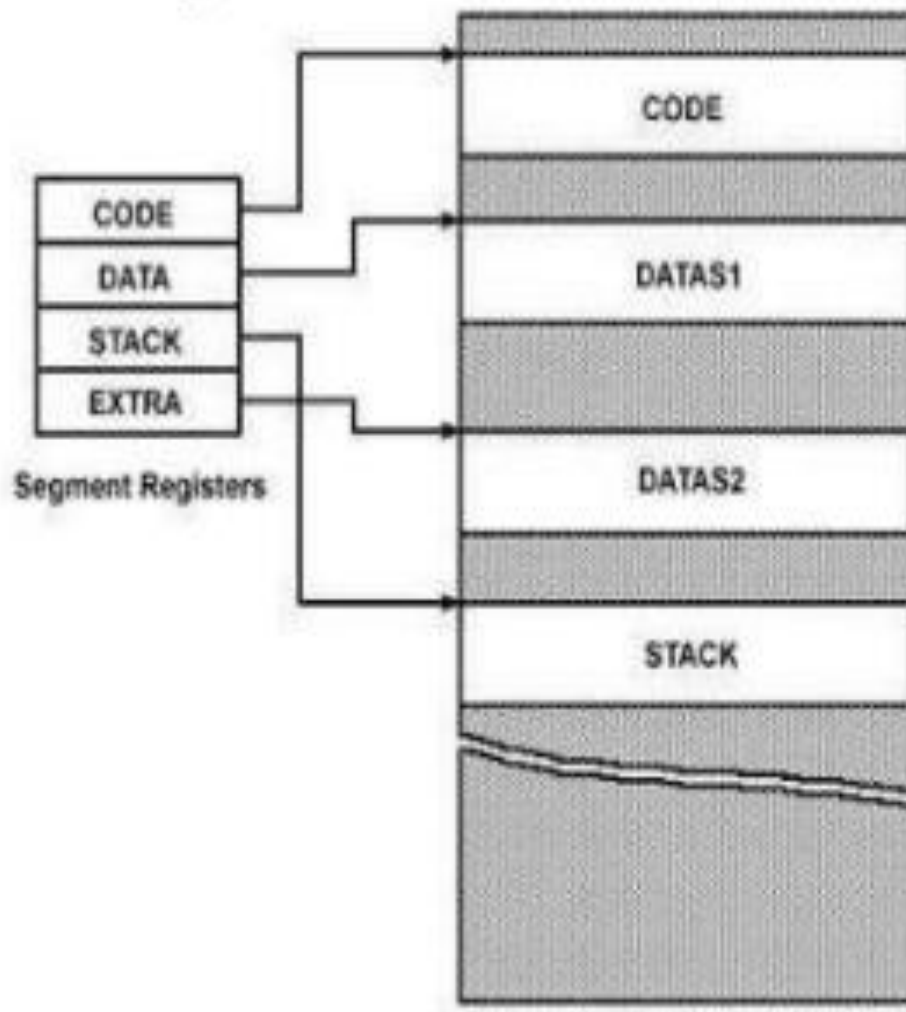
To increase the execution speed, BIU fetches as many as six instruction bytes ahead to time from memory.

The prefetched instruction bytes are held for the EU in a first in first out group of registers called an instruction queue. When the EU is ready for its next instruction, it simply reads the instruction from this instruction queue. This is much faster than sending out an address to the system memory and to send back the next instruction byte. Fetching the next instruction while the current instruction executes is called pipelining.

o Segment Registers

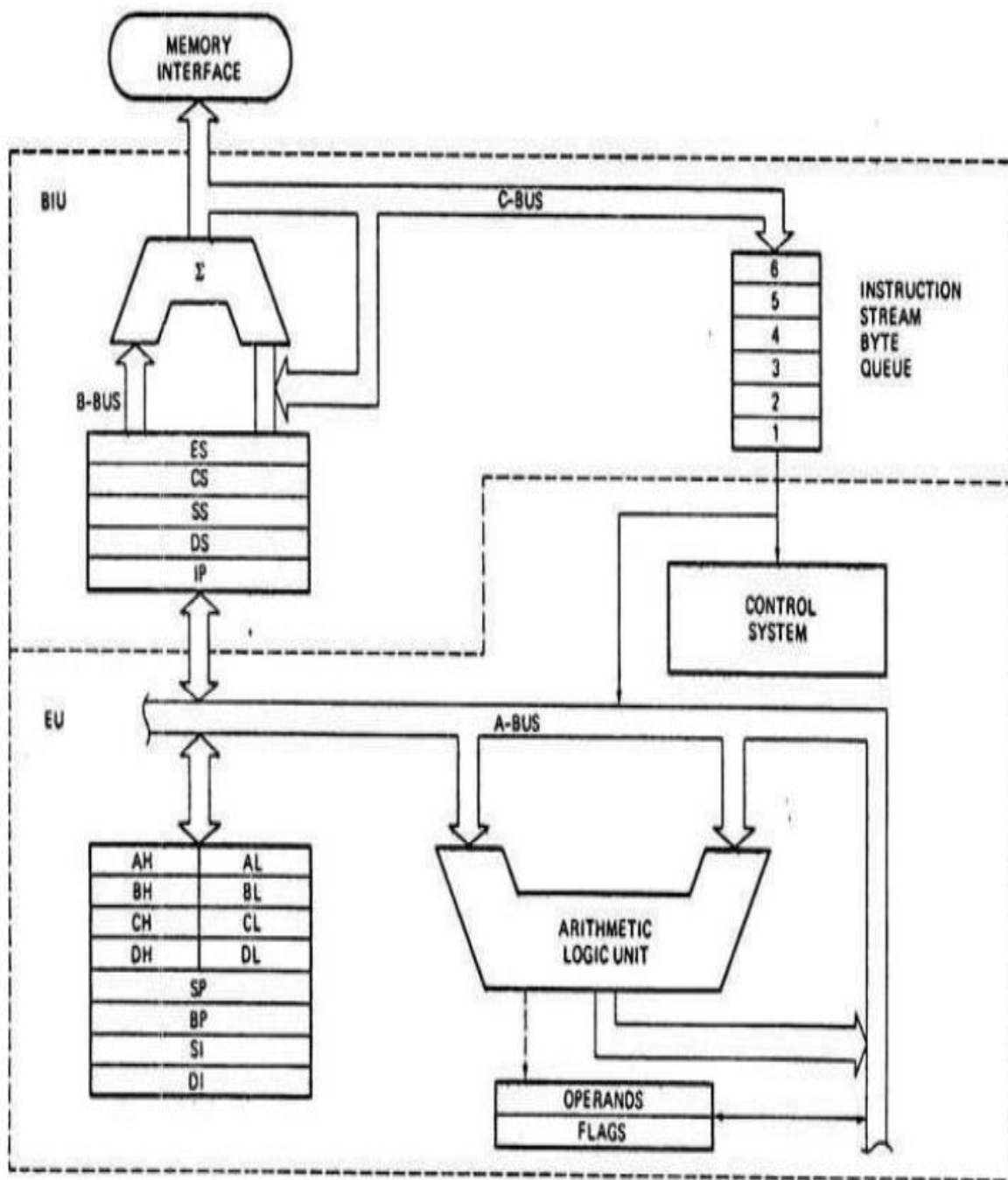
The BIU contains four 16-bit segment registers. They are: the extra segment (ES) register, the code segment (CS) registers, the data segment (DS) registers, and the stack segment (SS) registers. These segment registers are used to hold the upper 16 bits of the starting address for each of the segments. The part of a segment starting address stored in a segment register is often called the segment base.

1. **Code Segment (CS):** The CS register is used for addressing a memory location in the Code Segment of the memory, where the executable program is stored.
2. **Data Segment (DS):** The DS contains most data used by program. Data are accessed in the Data Segment by an offset address or the content of other register that holds the offset address.
3. **Stack Segment (SS):** SS defined a section of memory to store addresses and data while a subprogram executes.
4. **Extra Segment (ES):** ES is additional data segment that is used by some of the string to hold the extra destination data.



- **INSTRUCTION POINTER (IP)**

The CS contains the start of the current code segment. The IP contains the distance or offset from this address to the next instruction byte to be fetched. Code segment address plus an offset value in the IP indicates the address of an instruction to be fetched for execution.



❖ The Execution Unit

The EU decodes and executes the instructions. The EU contains arithmetic and logic (ALU), a control unit, and a number of registers. These features provide for execution of instructions and arithmetic and logical operations. It has nine

16 bit registers which are AX, BX, CX, DX, SP, BP, SI, DI and flag. First four can be used as 8 bit register (AH, AL, BH, BL, CH, DH, DL)

General Purpose Registers

The EU has eight general purpose registers labeled AH, AL, BH, BL, CH, CL, DH, and DL. These registers can be used individually for temporary storage of 8-bit data. The AL register is also called the accumulator. Certain pairs of these general-purpose registers can be used together to store 16-bit data. The valid register pairs are AH and AL, BH and BL, CH and CL and DH

1. AX Register: For 16-bit operations, AX is called the accumulator register that stores operands for arithmetic operations.

2. BX Register: This register is mainly used as a base register. It holds the starting base location of a memory region within a data segment.

3. CX Register: It is defined as a counter. It is primarily used in loop instruction to store loop counter.

4. DX Register: DX register is used to contain I/O port address for I/O instruction and DL. These register pairs are referred to the AX, BX, CX, and DX resp.

❖ Stack Pointer Register

The stack pointer (SP) register contains the 16-bit offset from the start of the segment to the memory location where a word was most recently stored on the stack.

❖ Flag Register

A 16-bit flag register is a flip-flop which indicates some condition produced by the execution of an instruction or controls certain operations of the EU. They are modified automatically by CPU after mathematical operations. It has 9 flags and they are divided into two categories:

1. Conditional Flags

2. Control Flags

1. Conditional Flags

Conditional flags represent result of last arithmetic or logical instructions.

- **Carry Flag (CF)**: This flag will be set to one if the arithmetic operation produces the carry in MSB position. It is also used in multiple-precision arithmetic.

- **Auxiliary Flag (AF)**: If an operation performed in ALU generates a carry/borrow from lower nibble (i.e. D0 – D3) to upper nibble (i.e. D4 – D7), the AF flag is set i.e. carry given by D3 bit to D4 is AF flag. This is not a general-purpose flag; it is used internally by the processor to perform Binary to BCD conversion.

- **Parity Flag (PF)**: This flag is used to indicate the parity of result. If lower order 8-bits of the result contains even number of 1's, the Parity Flag is set to one and for odd number of 1's, the Parity Flag is reset i.e. zero.

- **Zero Flag (ZF)**: It is set to one; if the result of arithmetic or logical operation is zero else it is reset.

- **Sign Flag (SF)**: In sign magnitude format the sign of number is indicated by MSB bit. If the result of operation is negative, sign flag is set to one.

Overflow Flag (OF): It occurs when signed numbers are added or subtracted. An OF indicates that the result has exceeded the capacity of machine.

2. Control Flags

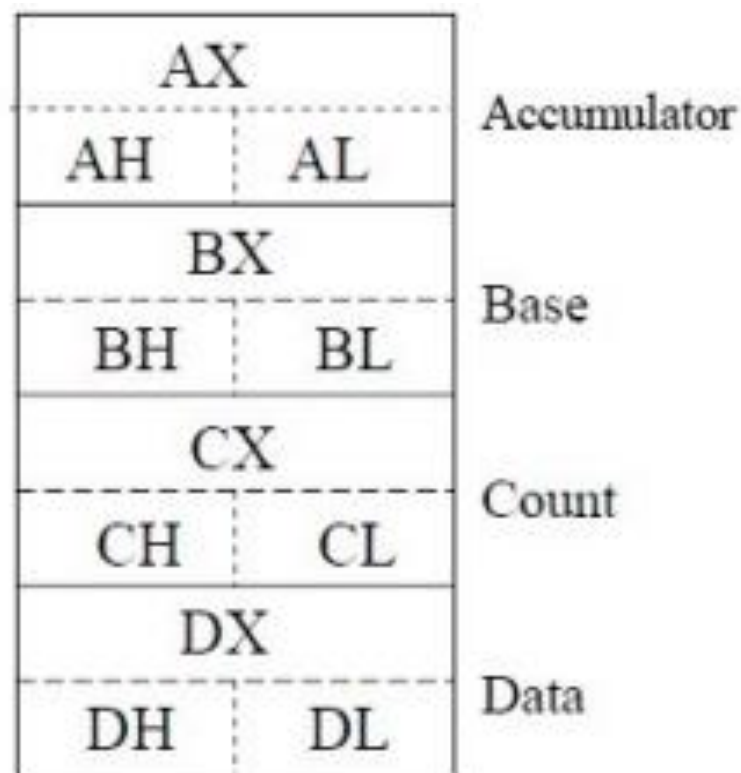
Control flags are intentionally set or reset to control certain operations of the processor with specific instructions put in the program from the user. Control flags are as follows:

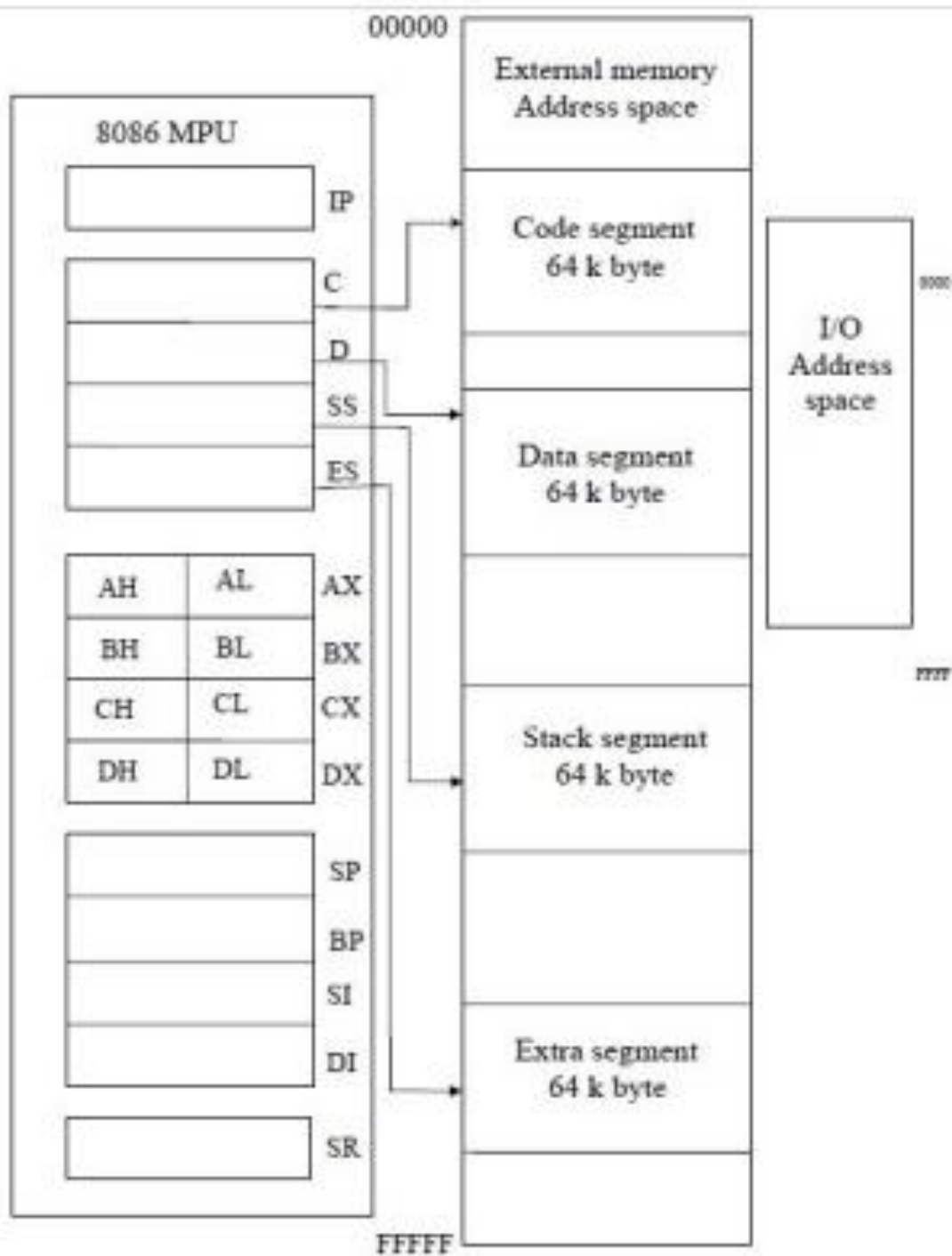
- **Trap Flag (TF)**: It is used for single step control. It allows user to execute one instruction of a program at a time for debugging. When trap flag is set, program can be run in single step mode.
- **Interrupt Flag (IF)**: It is an interrupt enable/disable flag, i.e. used to allow/prohibit the interruption of a program. If it is set, the maskable interrupt is enabled and if it is reset, the interrupt is disabled.

3. **Direction Flag (DF):** It is used in string operation. If it is set, string bytes are accessed from higher memory address to lower memory address. When it is reset, the string bytes are accessed from lower memory address to higher memory address.

❖ Other Pointer and Index Registers

The EU also contains a 16-bit source index (SI) register, base pointer (BP) registers, and Destination Index (DI) registers. These three registers can be mainly used for temporary storage of 16-bit data just like a general-purpose register.





Operation	Word
AX	multiply, word divide, word I/O
AL	Byte multiply, byte divide, byte I/O, translate, decimal arithmetic.
AH	Byte multiply, byte divide.
BX	Translate
CX	String operation, loops.
CL	Variable shift and rotate.
DX	Word multiply, word divide, indirect I/O.

(L)

