

CHAPTER 9

Software Model of the 8088\8086

The 8086 Microprocessor:

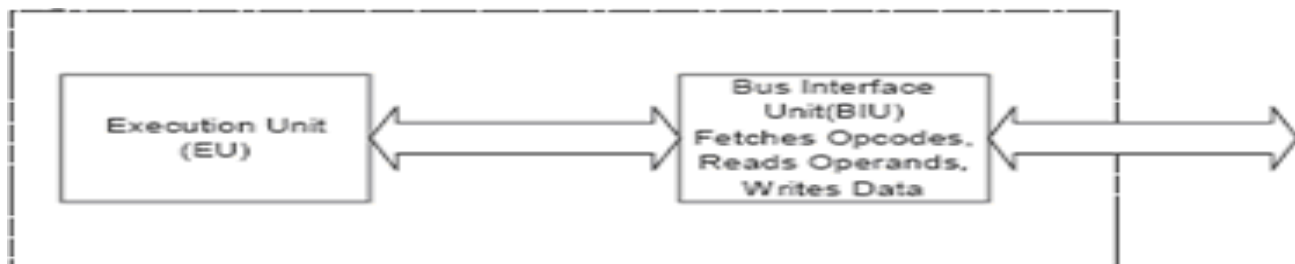
The 8086 have 16 bits internal and external data paths and a 20 bits address bus that can address up to 1 MB of memory.

The Internal Architecture of the 8086 MP:

The processor is partitioned into two logical units: an **Execution Unit (EU)** and **Bus Interface Unit (BIU)**. The role of the EU is to execute instruction, whereas the BIU delivers instruction and data to EU.

The EU contains **ALU**, **CU** and number of registers. This feature enables the EU to execute instructions and perform arithmetic and logical operations.

The most important function of BIU is to manage the bus control unit, segment registers, and instruction queue. The BIU controls the busses that transfer data to the EU, to memory and to external input/output devices, whereas the segment registers control the memory addressing.



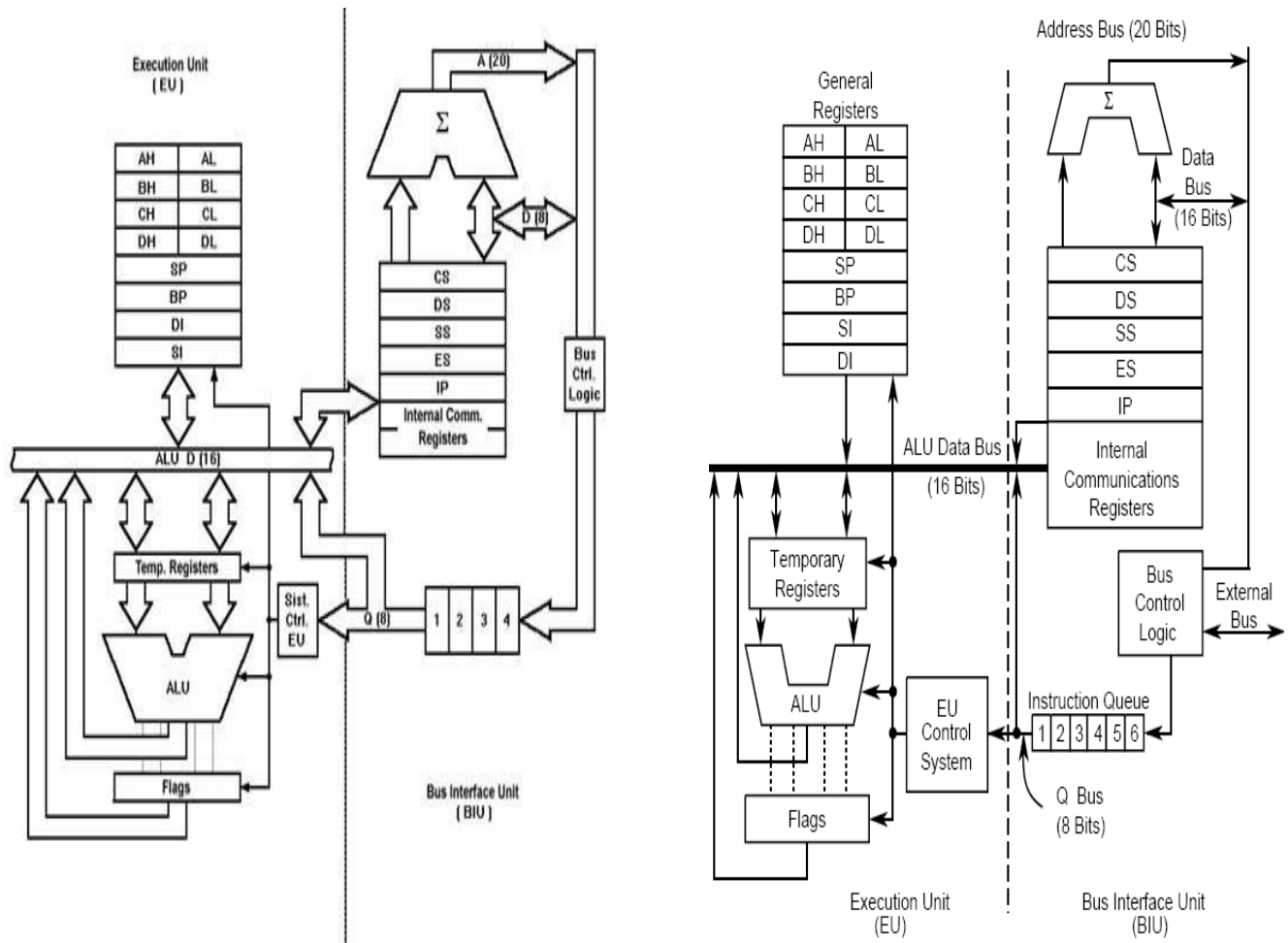
Another function of the BIU is to provide access to instructions. The instructions for a program which executing are kept in memory, the BIU must access it from memory and place them in an instruction queue, which varies in size depending on the processor (the size of instruction queue for 8086 MP is 6 bytes).

This feature enables the BIU to look ahead and pre-fetch instructions, so that there is always a queue of instructions ready to execute.

The EU and BIU work in parallel, with the BIU keeping one step ahead The EU notifies the BIU when it needs access to data in memory or I/O devices.

Also the EU request machine code instructions from the BIU instruction queue. The top instruction is the currently executable one, and while the EU is occupied executing an instruction, the BIU fetch another instruction from memory.

This fetching overlaps with execution and speeds up processing.



The only difference between an 8088 microprocessor and an 8086 microprocessor is the BIU. In the 8088, the BIU data bus path is 8 bits wide versus the 8086's 16-bit data bus. Another difference is that the 8088 instruction queue is four bytes long instead of six. The important point to note, however, is that because the EU is the same for each processor, the programming instructions are exactly the same for each. Programs written for the 8086 can be run on the 8088 without any changes.

Registers 8088\8086:

General Purpose Registers:

The 8086 microprocessor has a total of *fourteen* registers that are accessible to the programmer.

Eight of the registers are known as *general purpose registers* i.e. they can be used by the programmer for data manipulation.

Each of the registers is 16 bits long i.e. can contain a 16-bit binary number.

The first four registers are sometimes referred to as *data registers*. They are the AX, BX, CX, and DX registers.

The second four are referred to as *index/pointer registers* or *Offset Address Registers*. They are the SP, BP, SI and DI registers.

The data registers can be treated as 16-bit registers or they can each be treated as two 8-bit register.

As a 16-bit register, a general purpose registers name ends in an "X" e.g. AX. As an 8-bit register, its name ends in either an "H", if it is the "high order" half of the 16-bits, or an "L", if it is the "low order" half; e.g. AH or AL, each 8-bit register can be used independently.

AX: Accumulator are registers that can be used for arithmetic, logical, shift, rotate, or other similar operations.

BX: *Base register* the only general purpose register whose contents can be used by instructions as a 16-bit offset address. This offset is paired by default with segment register "DS:", i.e. the memory reference [BX] actually means[DS:BX].

CX: *Count register* used by several instructions to control the number of times a looping process is performed.

DX: *Data register* often used to hold single-byte character data and referenced as DH or DL; also combines with AX to form a 32-bit register for some operations (e.g. MULtiPLY).

SI: *Source index register:* Offset address relative to DS.

DI: *Destination index register:* Offset address relative to ES.

BP: *Base pointer:* Offset address relative to SS.

SP: *Stack pointer:* A very important register. Maintains the program stack, and so should be used carefully, Offset address relative to SS.

Special Purpose Registers:

There are two special purpose registers on the 8086 the *instruction pointer* (IP) and the *flag register*. The IP is sometimes referred to as the pc (*program counter*). These registers cannot be accessed directly; rather they are modified by the CPU during execution.

IP or PC - (Instruction Pointer Register) or (program counter):

This is a crucially important register which is used to control which instruction the CPU executes. The IP, or program counter, is used to store the memory location of the next instruction to be executed. The CPU checks the program counter to ascertain which instruction to carry out next. It then updates the program counter to point to the next instruction. Thus the program counter will always point to the next instruction to be executed. This address is the offset within the code segment. specified by the CS register.

Status (Flags) Register:

Flags register determines the current state of the processor. They are modified automatically by CPU after mathematical operations, this allows to determine the type of the result, and to determine conditions to transfer control to other parts of the program.

Nine individual bits of the status register are used as control flags (3 of them) and status flags (6 of them). The remaining 7 are not used. For most practical purposes, only 4 flags are used: zero, carry, sign and overflow.

A flag can only take on the values 0 and 1. We say a flag is set if it has the value 1.

1. **Carry Flag (CF)** - this flag is set to 1 when there is an unsigned overflow. For example when you add bytes $255+1$ (*result is not in range 0...255*). When there is no overflow this flag is set to 0.
2. **Parity Flag (PF)** - this flag is set to 1 when there is even number of one bits in result, and to 0 when there is odd number of one bits.
3. **Auxiliary Flag (AF)** - set to 1 when there is an unsigned overflow for low nibble (4 bits).
4. **Zero Flag (ZF)** - set to 1 when result is zero. For non-zero result this flag is set to 0.
5. **Sign Flag (SF)** - set to 1 when result is negative. When is positive it is set to 0. (This flag takes the value of the most significant bit.)
6. **Trap Flag (TF)**
7. **Interrupt enable Flag (IF)**

8. Direction Flag(DF)

9. Overflow Flag (OF)

Registers 8088\8086:

General Purpose Registers

AX	AH	AL	Accumulator
BX	BH	BL	Base
CX	CH	CL	Count
DX	DH	DL	Data

Pointer and Index Registers

SP		Stack Pointer
BP		Base Pointer
SI		Source Index
DI		Destination Index
IP		Instruction Pointer

Segment Registers

CS		Code Segment
DS		Data Segment
SS		Stack Segment
ES		Extra Segment

Registers 8088\8086:

