Microprocessor & Interfacing
Lecture 13
EU and BIU

PARUL BANSAL
ASST PROFESSOR
ECS DEPARTMENT
DRONACHARYA COLLEGE OF ENGINEERING

Contents

Introduction

- 8086 microprocessor architecture divided in two parts first is execution unit and second is bus interface unit. Execution unit works all the calculation and manipulation work and bus interface unit work as data transfer from memory to microprocessor or ports and vice versa.

Architecture of 8086

- The architecture of 8086 includes

– Arithmetic Logic Unit (ALU)

– Flags

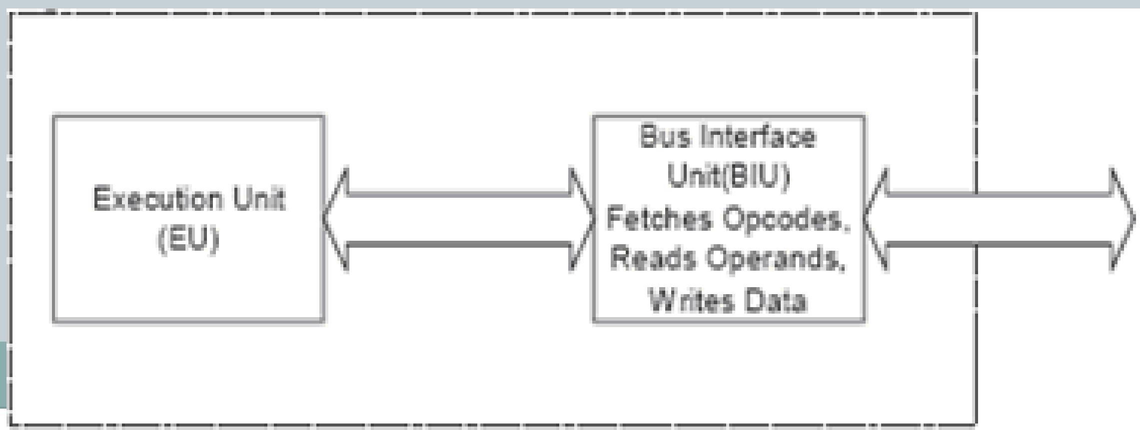
– General registers

– Instruction byte queue

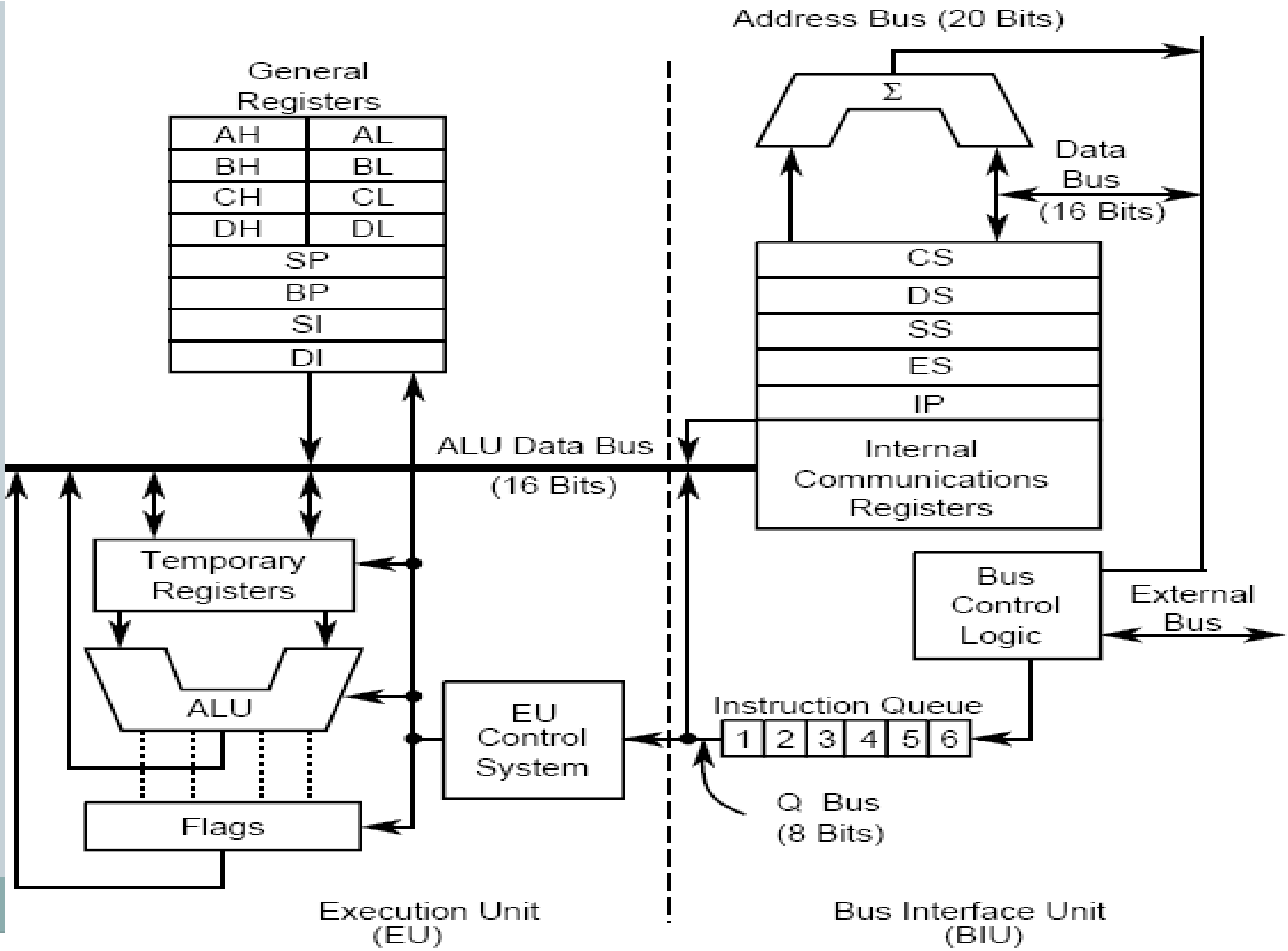
– Segment registers

EU & BIU

- The 8086 CPU logic has been partitioned into two functional units namely Bus Interface Unit (BIU) and Execution Unit (EU)
- The major reason for this separation is to increase the processing speed of the processor
- The BIU has to interact with memory and input and output devices in fetching the instructions and data required by the EU
- EU is responsible for executing the instructions of the programs and to carry out the required processing



Architecture Diagram



Execution Unit

- The Execution Unit (EU) has

– Control unit

– Instruction decoder

– Arithmetic and Logical Unit (ALU)

– General registers

– Flag register

– Pointers

– Index registers

Cont..

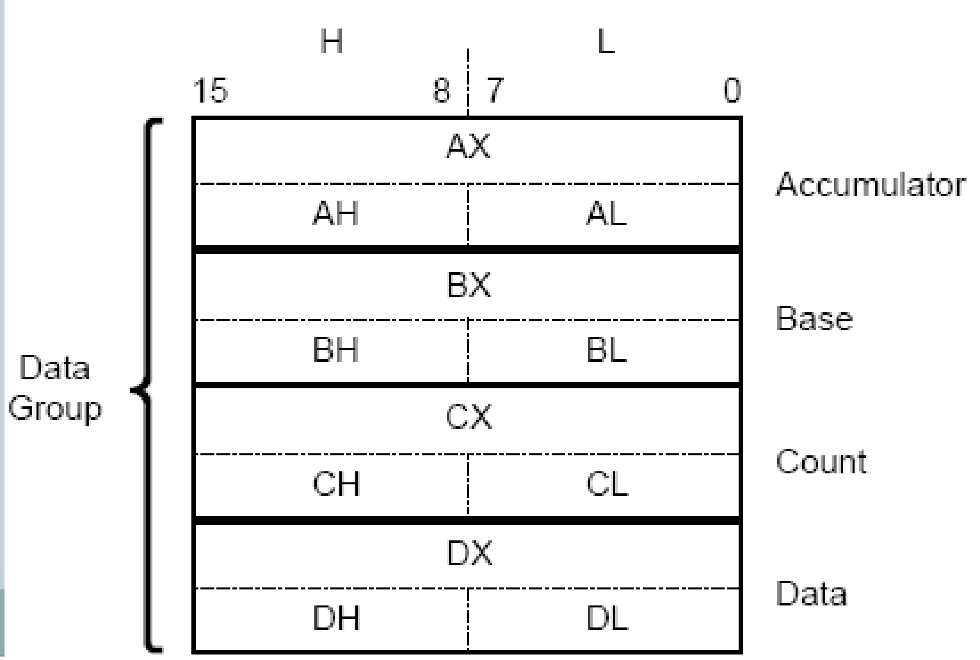
- Control unit is responsible for the coordination of all other units of the processor
- ALU performs various arithmetic and logical operations over the data
- The instruction decoder translates the instructions fetched from the memory into a series of actions that are carried out by the EU

EU Registers

- General registers are used for temporary storage and manipulation of data and instructions
- Accumulator register consists of two 8-bit registers AL and AH, which can be combined together and used as a 16-bit register AX
- Accumulator can be used for I/O operations and string manipulation
- Base register consists of two 8-bit registers BL and BH, which can be combined together and used as a 16-bit register BX
- BX register usually contains a data pointer used for based, based indexed or register indirect addressing
- Count register consists of two 8-bit registers CL and CH, which can be combined together and used as a 16-bit register CX
- Count register can be used as a counter in string manipulation and shift/rotate instructions

Cont..

- Data register consists of two 8-bit registers DL and DH, which can be combined together and used as a 16-bit register DX
- Data register can be used as a port number in I/O operations
- In integer 32-bit multiply and divide instruction the DX register contains highorder word of the initial or resulting number

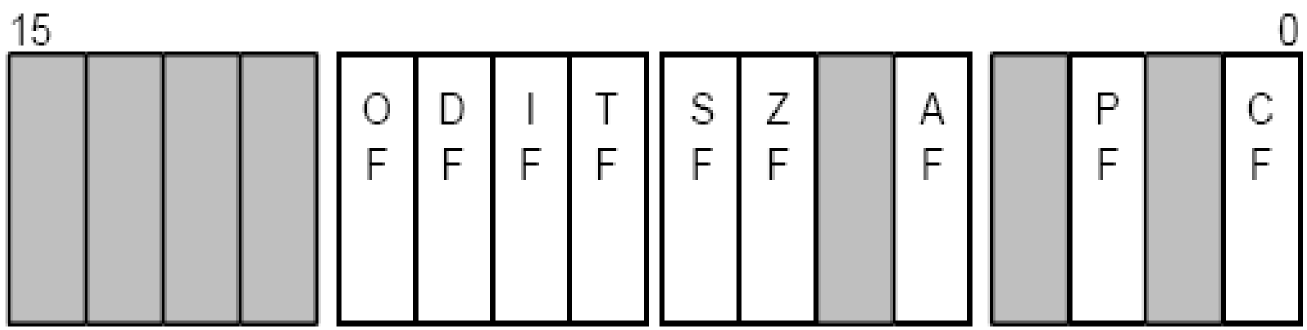


Execution Unit - Flags

Register Name: Processor Status Word

Register Mnemonic: PSW (FLAGS)

Register Function: Posts CPU status information.



Cont..

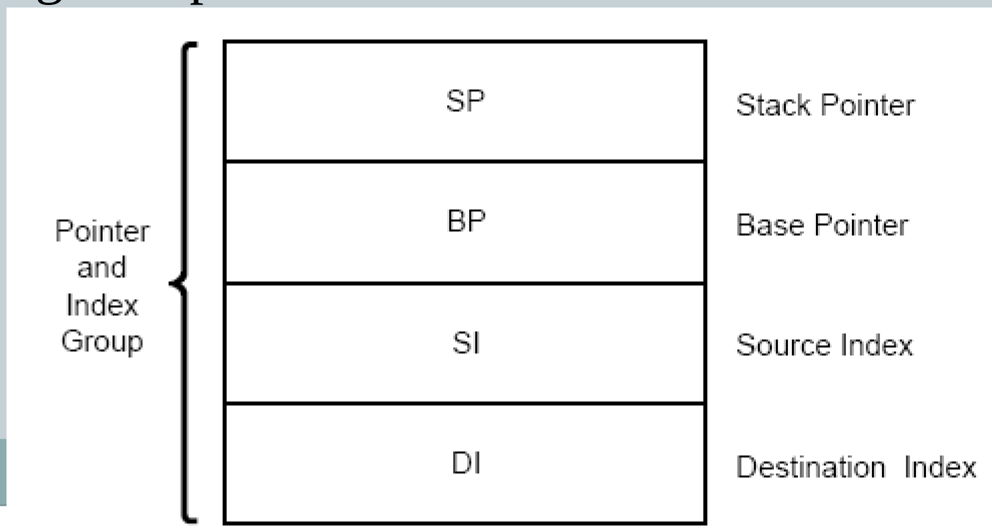
- Overflow Flag (OF) - set if the result is too large positive number, or is too small negative number to fit into destination operand
- Direction Flag (DF) - if set then string manipulation instructions will auto-decrement index registers. If cleared then the index registers will be auto-incremented
- Interrupt-enable Flag (IF) - setting this bit enables maskable interrupts
- Single-step Flag (TF) - if set then single-step interrupt will occur after the next instruction
- Sign Flag (SF) - set if the most significant bit of the result is set.
- Zero Flag (ZF) - set if the result is zero.
- Auxiliary carry Flag (AF) - set if there was a carry from or borrow to bits 0-3 in the AL register.
- Parity Flag (PF) - set if parity (the number of "1" bits) in the low-order byte of the result is even.
- Carry Flag (CF) - set if there was a carry from or borrow to the most significant bit during last result calculation

Cont..

Bit Mnemonic	Bit Name	Reset State	Function
OF	Overflow Flag	0	If OF is set, an arithmetic overflow has occurred.
DF	Direction Flag	0	If DF is set, string instructions are processed high address to low address. If DF is clear, strings are processed low address to high address.
IF	Interrupt Enable Flag	0	If IF is set, the CPU recognizes maskable interrupt requests. If IF is clear, maskable interrupts are ignored.
TF	Trap Flag	0	If TF is set, the processor enters single-step mode.
SF	Sign Flag	0	If SF is set, the high-order bit of the result of an operation is 1, indicating it is negative.
ZF	Zero Flag	0	If ZF is set, the result of an operation is zero.
AF	Auxiliary Flag	0	If AF is set, there has been a carry from the low nibble to the high or a borrow from the high nibble to the low nibble of an 8-bit quantity. Used in BCD operations.
PF	Parity Flag	0	If PF is set, the result of an operation has even parity.
CF	Carry Flag	0	If CF is set, there has been a carry out of, or a borrow into, the high-order bit of the result of an instruction.

Execution Unit - Pointers

- Stack Pointer (SP) is a 16-bit register pointing to program stack
- Base Pointer (BP) is a 16-bit register pointing to data in stack segment. BP register is usually used for based, based indexed or register indirect addressing.
- Source Index (SI) is a 16-bit register. SI is used for indexed, based indexed and register indirect addressing, as well as a source data addresses in string manipulation instructions.
- Destination Index (DI) is a 16-bit register. DI is used for indexed, based indexed and register indirect addressing, as well as a destination data addresses in string manipulation instructions.



Bus Interface Unit

- The BIU has

– Instruction stream byte queue

– A set of segment registers

– Instruction pointer

Instruction Byte Queue

- 8086 instructions vary from 1 to 6 bytes
- Therefore fetch and execution are taking place concurrently in order to improve the performance of the microprocessor
- The BIU feeds the instruction stream to the execution unit through a 6 byte prefetch queue.
- This prefetch queue can be considered as a form of loosely coupled pipe
- Execution and decoding of certain instructions do not require the use of buses.
- While such instructions are executed, the BIU fetches up to six instruction bytes for the following instructions (the subsequent instructions)
  - The BIU store these prefetched bytes in a first-in-first out register by name instruction byte queue
  - When the EU is ready for its next instruction, it simply reads the instruction byte(s) for the instruction from the queue in BIU lining

Segment: Offset Notation

- The total addressable memory size is 1MB.
- Most of the processor instructions use 16-bit pointers the processor can effectively address only 64 KB of memory.
- To access memory outside of 64 KB the CPU uses special segment registers to specify where the code, stack and data 64 KB segments are positioned within 1 MB of memory

Cont..

- A simple scheme would be to order the bytes in a serial fashion and number them from 0 (or 1) to the end of memory.
  - The scheme used in the 8086 is called segmentation
  - Every address has two parts, a SEGMENT and an OFFSET (Segmnet:Offset).
  - The segment indicates the starting of a 64 kilobyte portion of memory, in multiples of 16.
  - The offset indicates the position within the 64k portion
  - Absolute address = (segment * 16) + offset

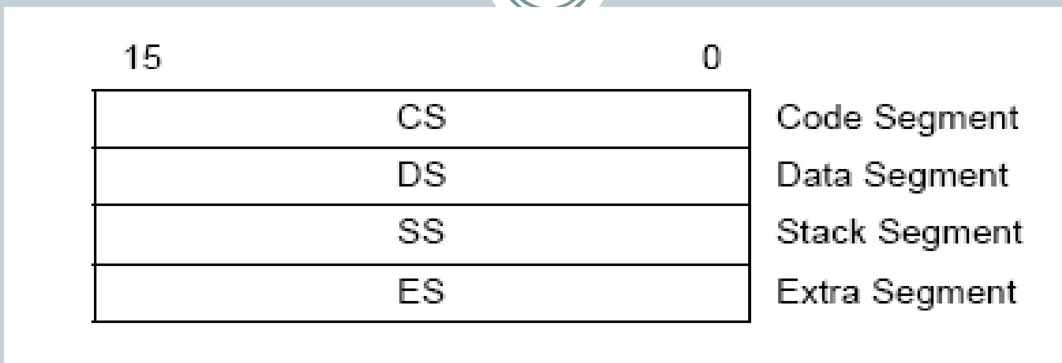
Segment Registers

- The memory of 8086 is divided into 4 segments namely
  - – Code segment (program memory)
  - – Data segment (data memory)
  - – Stack memory (stack segment)
  - – Extra memory (extra segment)

Different Areas in Memory

- Program memory – Program can be located anywhere in memory.
- Data memory – The processor can access data in any one out of 4 available segments.
- Stack memory – A stack is a section of the memory set aside to store addresses and data while a subprogram executes
- Extra segment – This segment is also similar to data memory where additional data may be stored and maintained

Segment Registers



- Code Segment (CS) register
  - a 16-bit register containing address of 64 KB segment with processor instructions
  - The processor uses CS segment for all accesses to instructions referenced by instruction pointer (IP) register
- Stack Segment (SS) register
  - a 16-bit register containing address of 64KB segment with program stack
  - By default, the processor assumes that all data referenced by the stack pointer (SP) and base pointer (BP) registers is located in the stack segment

Cont..

- Data Segment (DS) register:
  - a 16-bit register containing address of 64KB segment with program data
  - By default, the processor assumes that all data referenced by general registers (AX, BX, CX, DX) and index register (SI, DI) is located in the data segment
- Extra Segment (ES) register:
  - a 16-bit register containing address of 64KB segment, usually with program data
  - By default, the processor assumes that the DI register references the ES segment in string manipulation instructions

Scope of research

We can design such microprocessor which can reduce the fetching time of the instruction so that bus interface unit works faster or that can pre-fetch more instructions before the execution.