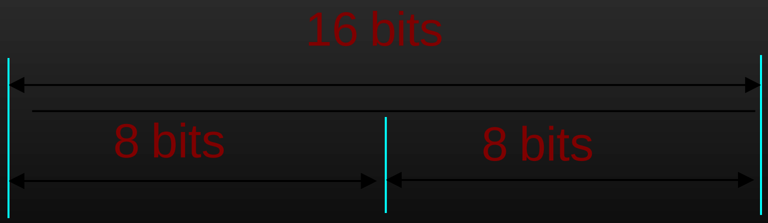
INTEL 8086 REGISTER SET

Course Teacher-
Shaila Rahman

GENERAL PURPOSE REGISTERS



AX	AH		AL	Accumulator
BX	BH		BL	Base
CX	CH		CL	Count
DX	DH		DL	Data
Pointer		SP		Stack Pointer
		BP		Base Pointer
Index		SI		Source Index
		DI		Destination Index

DATA REGISTERS

- AX- Data register, Word multiply, word divide, word I /O
- AL- Byte multiply, byte divide, byte I/O, decimal arithmetic
- AH- Data register, Byte multiply, byte divide
- BX- Store data and address information
- CX- Data register, String operation, loops, repeated shift and rotate
- CL- Same function as CX but for byte Variable shift and rotate
- DX- Word multiply, word divide, indirect I/O

(Used to hold I/O address during I/O instructions. If the result is more than 16-bits, the lower order 16-bits are stored in accumulator and higher order 16-bits are stored in DX register)

POINTER AND INDEX REGISTERS

- used to keep offset addresses.
- Used in various forms of memory addressing.
- In the case of SP and BP the default reference to form a physical address is the Stack Segment (SS- will be discussed under the BIU)
- The index registers (SI & DI) and the BX generally default to the Data segment register (DS).

- SP: Stack pointer

– Used with SS to access the stack segment

- BP: Base Pointer

Primarily used to access data on the stack

Can be used to access data in other segments

•

SI: SOURCE INDEX REGISTER

– CAN BE USED FOR STRING OR ARRAY OPERATION

• – When string operations are performed, the SI register points to memory locations in the data segment which is addressed by the DS register. Thus, SI is associated with the DS in string operations.

•

DI: DESTINATION INDEX REGISTER

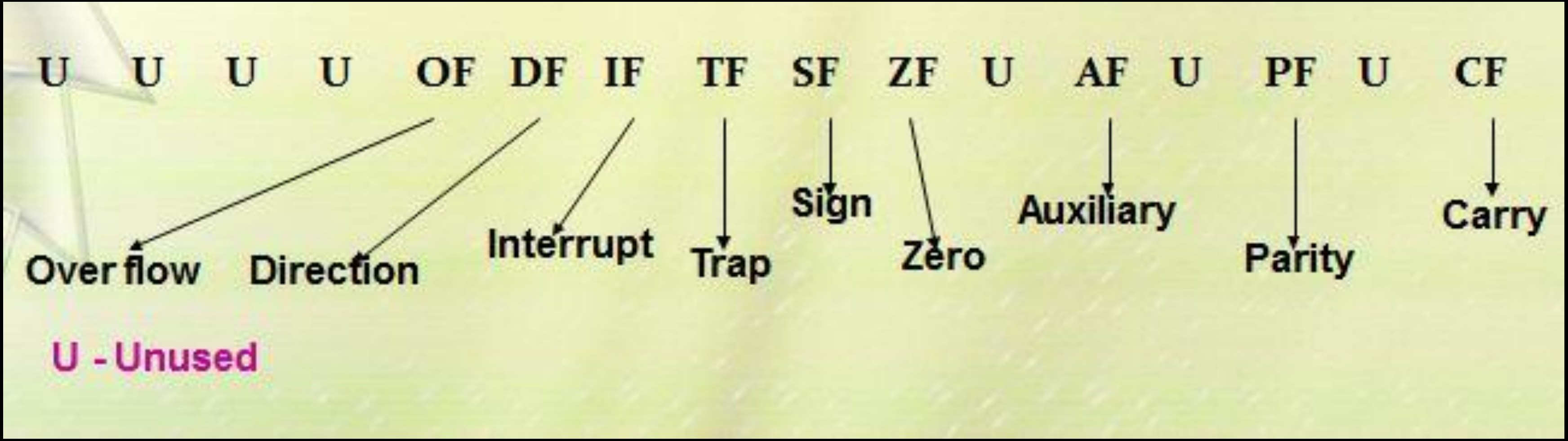
- is also required for some string operations.

- When string operations are performed, the DI register points to memory locations in the data segment which is addressed by the ES register. Thus, DI is associated with the ES in string operations.
- The SI and the DI registers may also be used to access data
- stored in arrays

FLAG REGISTER

- A flag is a flip flop which indicates some conditions produced by the execution of an instruction or controls certain operations of the EU .
- In 8086 The EU contains
- □ a 16 bit flag register
- □9 of the 16 are active flags and remaining 7 are undefined.
  - □ 6 flags indicates some conditions- status flags
  - □3 flags –control Flags

FLAG REGISTER



CONDITIONAL FLAGS

CF –Carry Flag

Holds the carry after addition or the borrow after subtraction. Also indicates some error conditions, as dictated by some programs and procedures .

PF- Parity Flag

The no. of one’s count in result

PF=0;odd parity, PF=1;even parity

AF- Auxiliary Flag

Holds the carry (half – carry) after addition or borrow after subtraction between bit positions 3 and 4 of the result (for example, in BCD addition or subtraction.)

ZF-Zero Flag

Shows the result of the arithmetic or logic operation. ZF=1; result is zero. ZF=0; The result is 0

SF-Sign Flag

Holds the sign of the result after an arithmetic/logic instruction execution. SF=1; negative, SF=0

OF- Overflow Flag

Overflow occurs when signed numbers are added or subtracted. An overflow indicates the result has exceeded the capacity of the Machine

CONTROL FLAG

TF-Trap Flag

Enables the trapping through an on-chip debugging feature.

IF-Interrupt Flag

Controls the operation of the INTR (interrupt request) I=0; INTR pin disabled. I=1; INTR pin enabled.

DF- Direction Flag

It selects either the increment or decrement mode for DI and /or SI registers during the string instructions.

These are programmable.

EXECUTION UNIT – FLAG REGISTER

- Six of the flags are status indicators reflecting properties of the last arithmetic or logical instruction.
- For example, if register AL = 7Fh and the instruction ADD AL,1 is executed then the following happen

AL = 80h

CF = 0; there is no carry out of bit 7

PF = 0; 80h has an odd number of ones

AF = 1; there is a carry out of bit 3 into bit 4

ZF = 0; the result is not zero

SF = 1; bit seven is one

OF = 1; the sign bit has changed

EXAMPLE

10011010

10001001

____

100100011

$\nwarrow$

CF

Thus CF=1,ZF=0 ,PF=0,SF=0,OF=1,AF=1

THE QUEUE (Q)

- The BIU uses a mechanism known as an **instruction stream queue** to implement a ***pipeline architecture.***

- This queue permits pre-fetch of up to **6 bytes** of instruction code. Whenever the queue of the BIU is not full, it has room for at least two more bytes and at the same time the EU is not requesting it to read or write operands from memory, the BIU is free to look ahead in the program by pre-fetching the next sequential instruction.

SEGMENT REGISTERS

- In 8086/88 the processors have 4 segments registers
- Code Segment register (CS), Data Segment register (DS), Extra Segment register (ES) and Stack Segment (SS) register.
- All are 16 bit registers.
- Each of the Segment registers store the upper 16 bit address of the starting address of the corresponding segments.

Segment and Address register combination

CS:IP

SS:SP SS:BP

DS:BX DS:SI

DS:DI (for other than string operations)

ES:DI (for string operations)