

COA

8086 MP | Flag Register

Prof. Dr. Mostafa Elhosseini

<https://youtube.com/drmelhosseini>

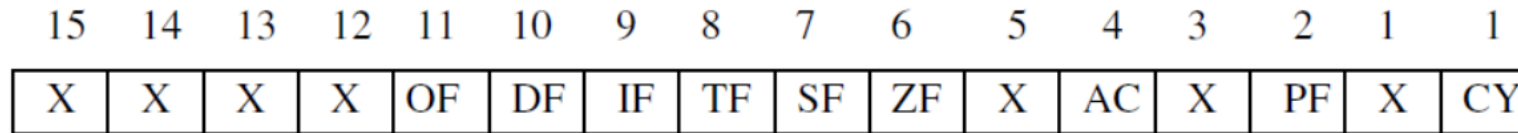
| AGENDA

- Flag Register

| Flag Register

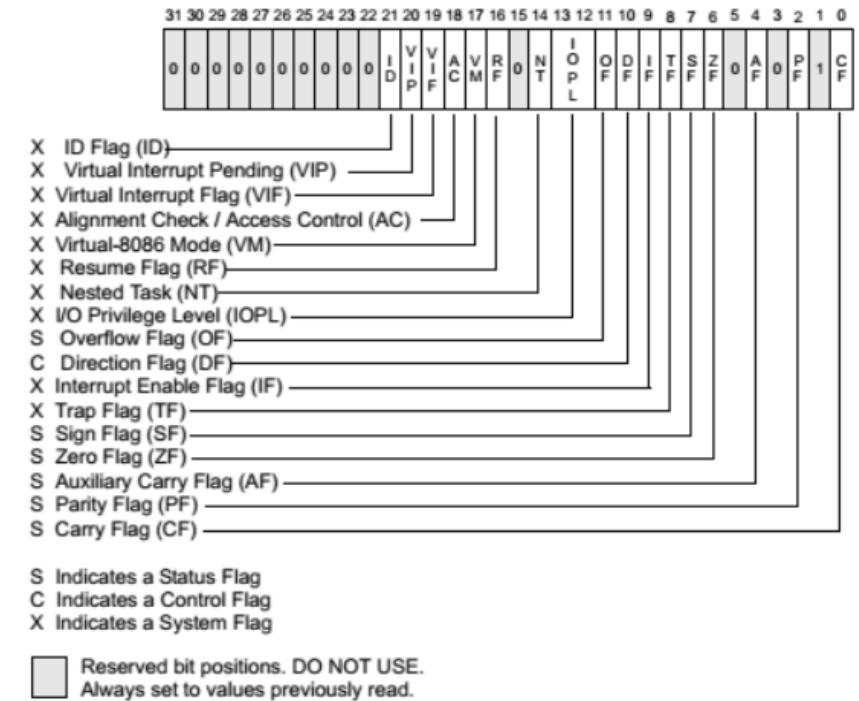
- The flag register is a **16-bit** register sometimes referred to as the **status register (Program Status Word PSW)**.
- **Although** the register is **16 bits wide**, only **some** of the bits are **used**. The **rest** are either **undefined** or **reserved** by Intel.

| Flag Register

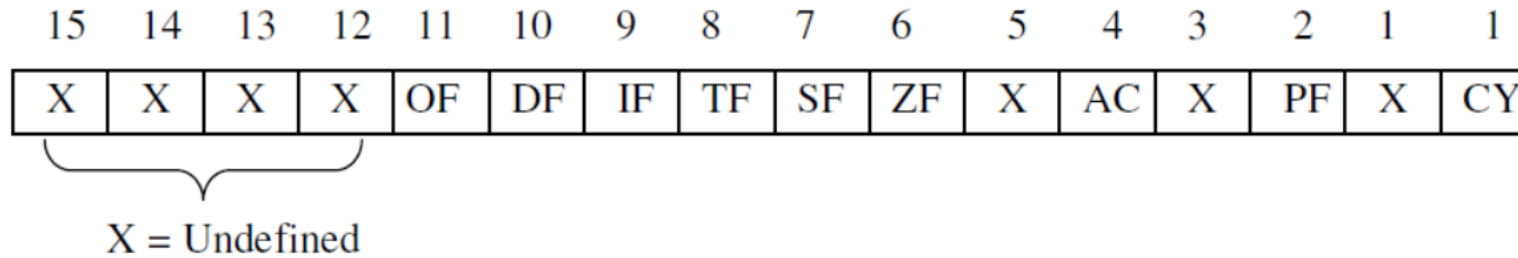


X = Undefined

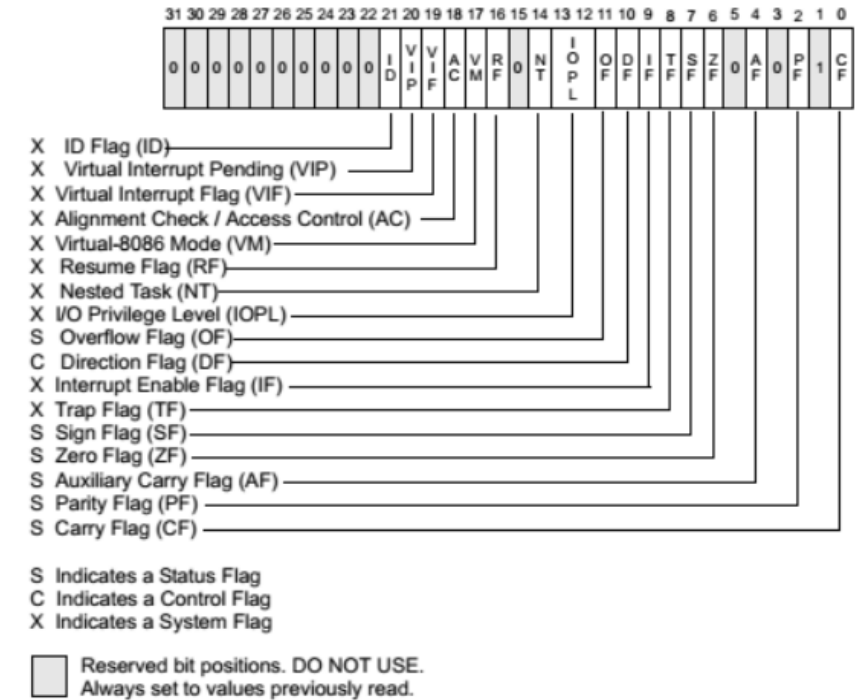
- Six of the flags are called **conditional** flags, meaning that they indicate some condition that resulted after an instruction was executed. These six are CF, PF, AF, ZF, SF, and OF



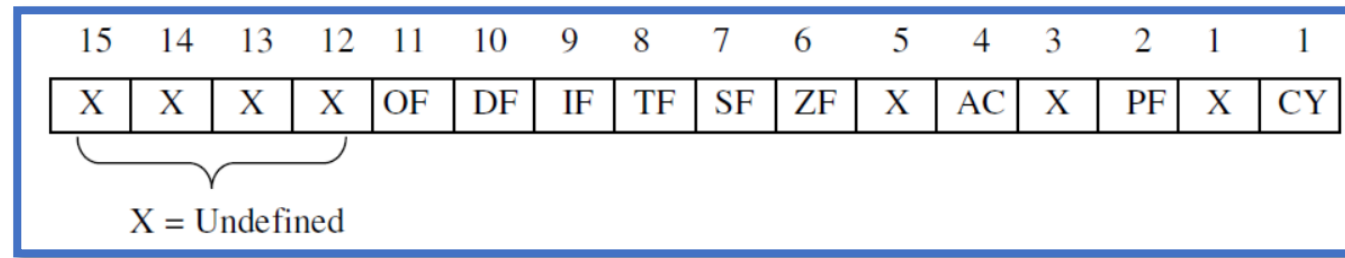
| Flag Register



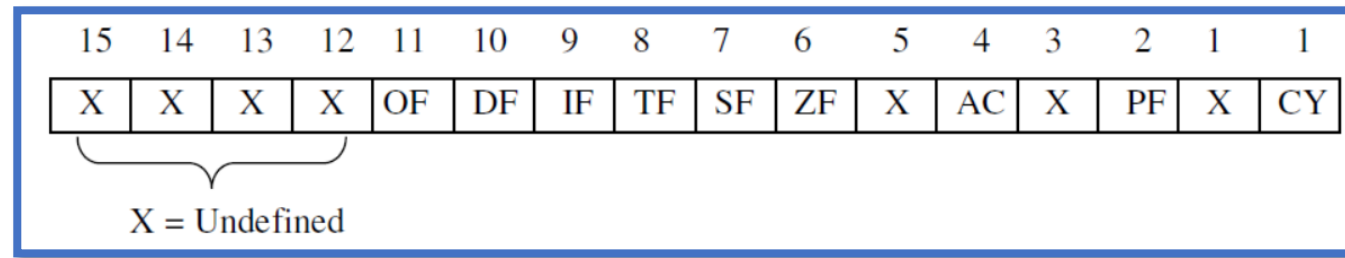
- The **three** remaining flags are sometimes called **control flags** since they are used to control the operation of the instructions before they are executed



| Carry Flag CF



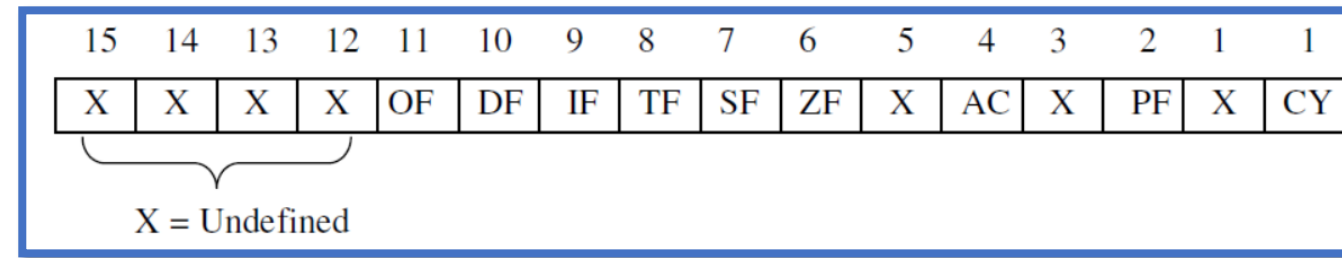
- This flag is **set** whenever there is a **carry output**, either from **d7** after an **8-bit operation** or from **d15** after a 16-bit operation
 - CF = 1 CY
 - CF = 0 NC



| Parity Flag PF

- After **certain operations**, the **parity** of the result's **low order byte** is checked. If the byte has an **even** number of 1s, the parity flag is set to **1**; otherwise it is cleared.
 - PF = 1 PE
 - PF = 0 PO

| Auxiliary Flag AF

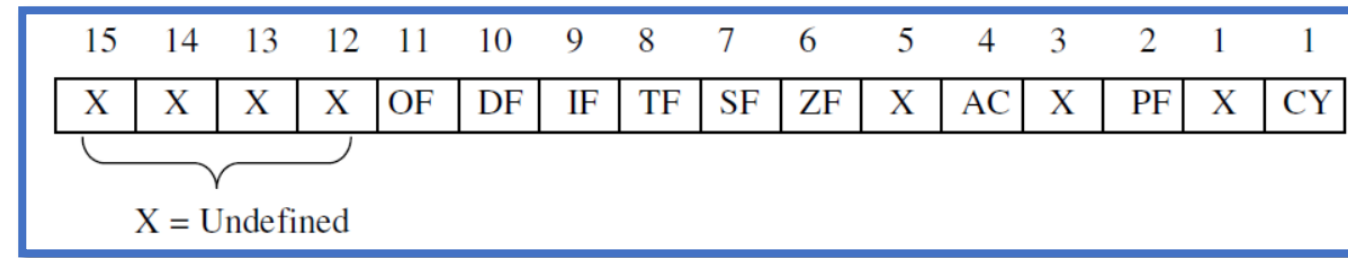


- If there is a **carry** from **d3 to d4** of an operation, this bit is set; otherwise, it is cleared (set equal to zero). This flag is used by the instructions that perform **BCD arithmetic**.

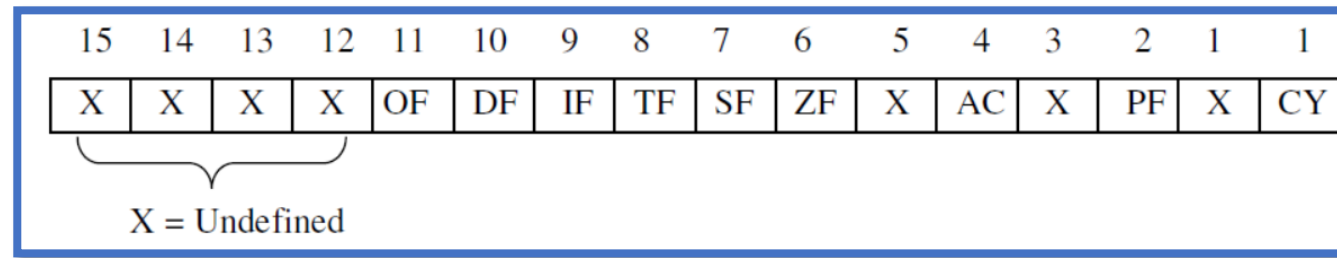
— AF = 1 AC

— AF = 0 NA

| Zero Flag ZF

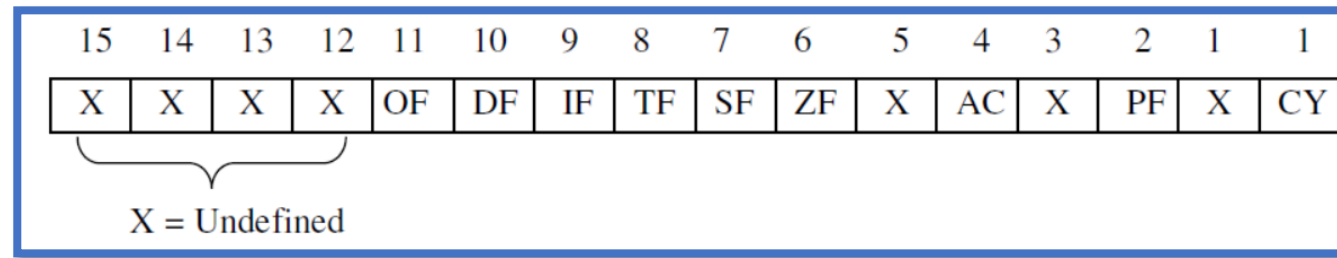


- The zero flag is set to 1 if the result of an arithmetic or logical operation is zero; otherwise, it is cleared
 - ZF = 1 ZR
 - ZF = 0 NZ



| Sign Flag SF

- Binary representation of **signed numbers** uses the **most significant** bit as the sign bit, after arithmetic or logic operations. The status of this sign bit is copied into the SF, thereby indicating the sign of the result
- This flag is **set**, when the result of any computation is **negative**
 - SF = 1 NG
 - SF = 0 PL



| Overflow Flag OF

- This flag is **set** whenever the **result of a signed** number operation is **too large**, causing the high-order bit to overflow into the sign bit. In general, the **carry flag** is used to detect errors in **unsigned** arithmetic operations. The **overflow** flag is only to **detect** errors in **signed** arithmetic operations
 - OF = 1 OV
 - OF = 0 NV

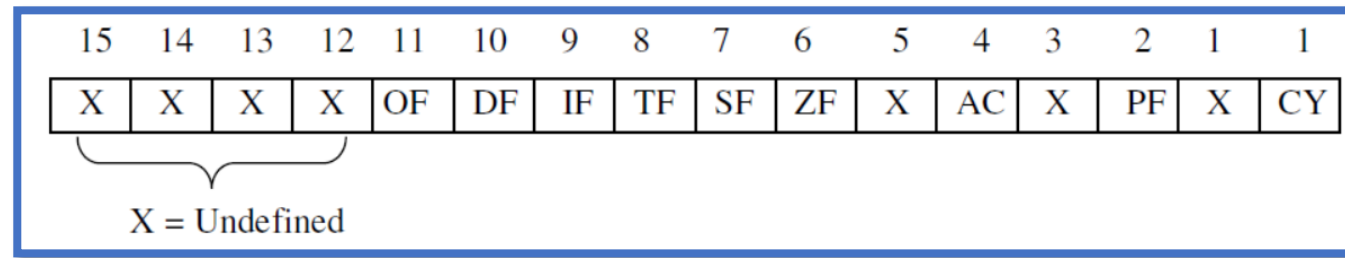
| Example

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	1
X	X	X	X	OF	DF	IF	TF	SF	ZF	X	AC	X	PF	X	CY

X = Undefined

- Show how the flag register is affected by the addition of 38H and 2FH.

Example



- Show how the flag register is affected by the addition of 38H and 2FH.



```
MOV BH, 38H
```

```
ADD BH, 2FH
```