Explicați care sunt și care este rolul steagurilor în funcționarea microprocesorului 80x86. Prezentați steaguri, clasificarea lor și exemplificați influența celor mai importante steaguri oferind secvențe adecvate de instrucțiuni scrise în limbaj de asamblare. Care sunt instrucțiunile pentru accesarea directă a steagurilor? Care sunt instrucțiunile care iau în considerare valorile steagurilor? Cum putem modifica configurația registrului eflags dacă este necesar? Care sunt steagurile care reacționează la o situație de depășire, care este rolul lor exact și care sunt regulile care se stabilesc?

FLAGS

Un **flag** este un indicator reprezentat pe un bit. O **configuratie a registrului de flaguri** indica un rezumat sintetic a executiei fiecarei instructiuni. **Registrul EFLAGS** are 32 de biti dintre care doar 9 sunt folositi uzual: **OF** **DF** IF TF **SF** **ZF** AF PF **CF** . Acesta poate fi accesat si mia apoi modificat prin instructiunea **PUSHF,** respectiv datele pot fi transferate prin instructiunile LAHF (Load register AH from Flags) si SAHF (Store register AH into Flags).

**Rolul principal al flagurilor** este de *a exprima conditii / statusul unor conditii*. Cand trebuie luata o decizie, instructiunea curenta verifica statusul unor flaguri si in functie de valorile lor, ALU ( Aritmetic and Logic Unit) va decide asupra valorii conditie respective (T/F).

Flagurile se pot incadra in *doua categorii*:

* cele ce sunt o consecinta a rezultatului uoe, precum **CF**, PF, AF, ZF, SF, OF *(adica indica ce s-a intamplat)*
* cele ce sunt setate de catre programator anterior executiei unei instructiuni, precum TF, DF, IF, **CF** *(indica ce urmeaza sa se intample).*

**CF** este flagul de transport. Are valoarea 1 in cazul in care in cadrul uoe s-a efectuat transport in afara domeniului de reprezentare a rezultatului si valoarea 0 in caz contrar. CF semnaleaza depasirea in cazul interpretarii FARA SEMN.

Setarea CF: CLC =0 , STC =1 , CMC = complementarea

Instructiuni care iau in considerare valoarea CF: ADC (aduna doua valori + cf) , SBB (scade doua valori – cf),( RCL, RCR ) + salturile conditionate

*De exemplu, in cazul unei scaderi*:

0110- mov ax, 6

1001 sub ax, 8

= 0100

**PF** (Parity Flag) – verifica daca nu s-au pierdut biti din interpretare-valoarea lui se seteaza astfel incat impreuna cu numarul de biti 1 din octetul cel mai putin semnificativ al reprezentarii rezultatului uoe sa rezulte un numar impar de cifre de 1

**AF** (Auxiliary Flag) indica valoarea transportului de la bitul 3 la bitul 4 al rezultatului uoe.

**ZF** (Zero Flag) primeste valoarea 1 daca rezultatul ultimei operatii efectuate este egal cu 0, valoarea 0 contrar

**SF** (Sign Flag) primeste valoarea 1 daca rezultatul uoe este un numar strict negativ si valoarea 0 contrar.

**TF** (Trap Flag) este un flag de depanare, folosit pentru scrierea debugerului. Daca are valoarea 1, atunci masina se opreste dupa fiecare instructiune. (prea periculos pentru a exista instructiuni de setare a acestui flag !)

**IF** (Interrupt Flag) – ajuta la definirea sectiunilor critice in cadrul programului, insa este controlabil doar sub 16b

Setarea IF: CLI =0 , STI =1

**DF** (Direction Flag) este folosit in operarea asupra sirurilor de octeti sau de cuvinte. Valoarea lui se seteaza la 0 pentru parcurgerea de la incept – sfarsit, respectiv 1 pentru parcurgere inversa.

Setarea DF: CLD =0, STD =1

**OF** (Overflow Flag) semnaleaza depasirea in cadrul interpretarii CU SEMN

De ce există mai mult de un steag care se ocupă de o situație de depășire?

In ceea ce priveste CF si OF, ambele exista pentru a semnala depasirea insa in interpretari diferite ( CF – fara semn, OF – cu semn)