

Инструкција MOV

Општи облик инструкције: MOV destination, source. И циљ и извор су операнди. Правила о врсти операнда:

MOV REG, memory	MOV SREG, memory	REG су сви регистри опште намене
MOV memory, REG	MOV memory, SREG	SREG су сви сегментни регистри
MOV REG, REG	MOV REG, SREG	memory: [BX], [BX+SI], промењиве (меморијска локација)
MOV memory, immediate	MOV SREG, REG	immediate: 5, -24, 3Fh, 11000111b (вредности)
MOV REG, immediate		

Ова инструкција чини да се вредност извора пребаца у вредност циља и оба морају бити исте величине (бајт или реч)
Инструкција не може сетовати вредности CS и IP регистра; не може пребацивати вредности из сегментних регистра у сегменте регистре; не може пребацивати вредност директно у сегментни регистар. Сви флегови су непромењени.

Инструкција INT

Општи облик инструкције: INT immediate. Врши релизацију прекида пребацавањем бајта података на стек. Ово утиче флег, CS, IP. Ресетује се вредност I флага.

Пристап меморији

За пристап меморији се користе регистри BX, SI, DI, BP. Стављањем вредности у угласту заграду [] приступа се меморијским локацијама на адресама једнаким тим вредностима. Могуће су следеће врсте адресирања: [BX+SI], [SI], [BX+DI+d8]. d8 је 8 битна вредност (22, 55h, -1).

Често се вредност у сегментним регистрима (CS, DS, SS, ES) назива **сегмент** а вредност у регистрима опште намене (BX, SI, DI, BP) се назива **офсет**.

Пример: DS = 100, BX = 30, SI = 70

Ако је мод адресирања: [BX+SI] + 25

Прорачуном процесора се добија физичка адреса: $100 * 16 + 30 + 170 + 25 = 1725$

По дифолту, DS се користи за све адресне модове (осим са BP где се користи SS сегмент).

Када DS садржи вредност 1234h а SI садржи вредност 7890h, може се записати као 1234:7890. Физичка адреса ће бити $1234h * 10h + 7890h = 19BD0h$. Емулатор користи и следеће скраћенице: b. [BX] за бајт, w.[BX] за реч.

Пример01:

```

ORG 100h           ; startovanje .com programa
MOV AX, 0B800h    ; postavlja AX na vrednost B800h (VGA memorija)
MOV DS, AX        ; vrednost iz AX se prebacuje u DS
MOV CL, 'A'       ; u CL se prebacuje ASCII kod od 'A', a to je 41h
MOV CH, 11011111b ; postavlja CH na vrednost boje
MOV BX, 15Eh      ; postavlja BX na 15Eh (pozicija na ekranu)
MOV [BX], CX      ; kopira sadrzaj CX u memorijsku lokaciju na B800:015E (w.[0B800h:015Eh] = CX)
RET               ; povratak u operativni sistem

```

