

## 2007 01 04

1. Užrašyti dešimtainį skaičių 31,47 slankaus kablelio formatu keturiuose baituose šešioliktaine sistema.
2. Kokia bus registro BX reikšmė, įvykdžius nurodytą komandą, kai AX = 00FA, DS = 3E21, SS = A634, CS = C13B, ES = 3EE1, BP = B2A2, BX = 367C, SI = A145, DI = 2AFC:  
8D 5A 94A2 (lea BX...)
3. Apskaičiuoti absoliutų adresą, kai AX = 0003, BX = 0002, CX = 0001, DX = 0000:  
FFFE E2 81 (loop... FFFE yra poslinkis)
4. Registrai yra: DS = 21FE, SS = 5634, CS = 0ADF, ES = 41E3, BP = 5033, BX = 7100, SI = 0074, DI = 2141. Apskaičiuoti absoliutų adresą, kai duota tokia komanda:  
71EA 2E FF 98 7700 (call CS: ... 71EA yra poslinkis)
5. DS = 21FE, SS = 5634, CS = 31CC, ES = 41E3, BP = 9A32, BX = 7536, SI = 45FA, DI = 22F1. Apskaičiuoti efektyvų adresą, pagal adresavimo baitą 6E:  
Po adresavimo seka baitai 89B0
6. DS = 21FE, SS = 5634, CS = 31CC, ES = 41E3, BP = 9A32, BX = 7536, SI = 45FA, DI = 22F1. Apskaičiuoti absoliutų adresą, pagal adresavimo baitą 6E:  
Po adresavimo seka baitai 89B0
7. Parašyti 2 mikrokomandas MPL kalba, kuriomis skaičius -44 nusiunčiamas į registrą MBR.
8. Suskaičiuot SF, kai baitų sudėtim sudedami du skaičiai.
9. SS reikšmė 1234, SP reikšmė 0002. vykdoma komanda RET (kitai grupei IRET). Apskaičiuoti absoliutų adresą.
10. CX = 0023, DI ir SI = 0002, SF = FFFF. Komanda rep stosw. SI + DI? (skaičiai netikslūs)

Ats.:

1. 41 FB C2 8F
2. 537B
3. AA := CS\*10h + 0000
4. 0A06F
5. 99BB
6. 5FCFB
7. X = 15; MBR = LEFT\_SHIFT(COM(1) + (-1));  
MBR = LEFT\_SHIFT(COM(X) + MBR)
8. -

9. 12344 arba 12346  
10.FFBE

## Sprendimai:

1. 1 bitas skirtas ženklui, 8 bitai charakteristikai (7F + eilė), 23 charakteristikai.
2. 8D – komandos baitas. 5A – adresavimo baitas.  $5A = 0101\ 1010$  (mod = 01, reg = 011, r/m = 010). Pagal reg lauką (taip pat ir komandos mnemoniką) suprantame, jog adresas bus įkeltas į registrą BX. Pagal r/m lauką suprantame, jog įkeliamas adresas bus formuojamas BP + SI + poslinkis. Pagal mod lauką suprantame, jog poslinkis 1 baito (pridedami plėšime jį iki 2 baitų pagal ženklo bitą).  $BX := B2A2 + A145 + FF94 = 1537B$ . Sudėję gauname atsakymą, didesnę nei FFFF, tad pirmąjį skaitmenį pašaliname. Ats.: 537B
3. Loop komanda iš pradžių sumažina registrą CX ( $CX := CX - 1$ ), tuomet tikrina ar  $CX <> 0$  ir jei ši sąlyga tenkinama prie dabartinio IP pridedamas poslinkis saugomas baite po E2. Šiuo atveju  $CX = 0001$ , taiga sumažinus tampa lygi 0. Komanda nevykdoma, bet IP vis tiek padidėja komandos ilgiu (buvo nuskaitytas E2 ir 81 bitai).  $IP := FFFE + 2 = 0000$ . IP yra šiuo atveju yra efektyvus adresas (EA). Kodo segmento (CS) nenurodyta, tad absoliutaus adreso apskaičiuoti negalime.  $AA := CS * 10_{16} + EA = CS * 10_{16} + 0000$ .
4. 2E – prefiksinis segmento keitimo baitas,  $2E = 0010\ 1110$ . Pagal vidurinius bitus 01 nustatome, jog segmentas keičiamas į CS. FF – komandos call baitas. 98 – adresavimo baitas,  $98 = 1001\ 1000$  (mod = 10, 011, r/m = 000). Pagal adreso lauką 011 suprantame, kad tai išorinis call ( $IP := [adresas]$ ,  $CS := [adresas + 2]$ ). Pagal r/m lauką suprantame, jog  $IP := [BX + SI + poslinkis]$ . Iš lauko mod suprantame, jog poslinkis 2 baitų.  $IP := [7100 + 0074 + 0077] = [71EB]$ ,  
  
 $CS := [71EB + 2] = [71ED]$ . Pagal duotą poslinkį 71EA nusistatome IP ir CS reikšmes,  $IP = 98FF$ ,  $CS = 0077$ . Absoliutus adresas  $AA := CS * 10_{16} + EA = 00770 + 98FF = 0A06F$
5.  $6E = 0110\ 1110\ 73$  (mod = 01, r/m = 110). Pagal r/m nusistatome, jog EA formuosime BP + poslinkis. Pagal mod lauką nustatome, jog poslinkis bus vieno baito. (pridedami jo reikšmę išplėšime iki 2 baitų pagal ženklo bitą).  
 $EA = 9A32 + FF89 = 199BB$ . Sudėję gauname atsakymą, didesnę nei FFFF, tad pirmąjį skaitmenį pašaliname.  
Ats.: 99BB
6. Absoliutus adresas formuojamas  $AA = seg.\ reg. * 10h + EA$ . Kadangi sąlyga tokia pat kaip 5 užduotyje, tai EA reikšmę jau esame apskaičiavę. Kadangi EA formavimui buvo panaudotas BP registras, tai segmento registras bus steko segmentas (SS).  $AA = 5634 * 10_{16} + 99BB = 5FCFB$ .
7.  $COM(1) = -2$ ,  $-2 + (-1) = -3$ . Komanda LEFT\_SHIFT padaugina atsakymą iš dviejų. Suformuojame reikšmę -6 ir ją išsaugome MBR.  $COM(X) = -16$ ,  $-16 + (-6) = -22$ . Komanda LEFT\_SHIFT padaugina atsakymą iš dviejų. Suformuojame reikšmę -44 ir ją išsaugome MBR. Norėdami panaudoti registrą X su reikšme 15, pirma turime šią reikšmę jam priskirti. Pilna komanda:  
 $X = 15$ ;  $MBR = LEFT\_SHIFT(COM(1) + (-1))$ ;  
 $MBR = LEFT\_SHIFT(COM(X) + MBR)$ ;
8. -

9. Sąlygoje nėra pateikta, ar komanda RET vidinė ar išorinė. Jei vidinė, tai  $SP := SP + 4 = 0002 + 0004 = 0006$ , jei vidinė, tai  $SP := SP + 6 = 0002 + 0006 = 0008$ . Kadangi dirbama su steku, absoliutus adresas bus apskaičiuojamas pagal taisyklę  $AA := SS * 10_{16} + SP$ .  $AA := 12340 + 0004 = 12344$  arba  $AA := 12340 + 0006 = 12346$ .
10. Komanda stos paveiks tik DI registrą. Kadangi SF registro DF bitas lygus 1, tai DI reikšmė bus mažinama.  
 $CX = 0023$ , tad komanda bus pakartota 23 kartus. Komandoje stosw raidė w parodo, jog bus operuojama žodžiais. DI bus sumažintas  $CX * 2 = 0023 * 2 = 46$  ( $DI = 0002 - 46 = FFBC$ ).  $SI + DI = 0002 + FFBC = FFFE$ .