

Laborator 4

DEBUG

Programul Debug, lansat dintr-o fereastră de comandă ("Command Prompt") conține o serie de comenzi cu parametri valori în hexazecimal care permit accesul la resursele primare ale calculatorului: locații de memorie, registrele procesorului, porturi de intrare / ieșire. Printre cele mai utile comenzi sunt:

d<adr1>,<adr2><ENTER>

("dump") afișează conținutul unei zone de memorie cuprinsă între adresele *adr1* și *adr2*.

f<adr1>,<adr2>,<lista_constante><ENTER>

("fill") umple zona de memorie cuprinsă între adresele *adr1* și *adr2* cu lista de constante.

e<adr><ENTER>

("enter") permite introducerea unor valori de octeți începând de la adresa *adr*. Trecerea la octetul următor se face cu spațiu. Comanda se încheie cu <ENTER>.

q<ENTER>

("quit") încheie programul Debug.

c<adr1>,<adr2>,<adr3><ENTER>

("compare") compară zona de memorie cuprinsă între adresele *adr1* și *adr2* cu zona începând de la adresa *adr3*.

i<port><ENTER>

("input") citește conținutul portului specificat.

o<port><octet><ENTER>

("output") înscrie octet la portul specificat.

a<adr><ENTER>
. instructiuni
<ENTER>

("assemble") permite introducerea și asamblarea unei secvențe de instrucțiuni în limbaj de asamblare începând cu adresa *adr*. Comanda se încheie cu instrucțiune vidă <ENTER>.

u<adr><ENTER>

("unassemble") dezassembleaza continutul unei zone de memorie incepand de la adresa *adr* si afiseaza secventa de instructiuni sursa in limbaj de asamblare.

r<ENTER>

("registers") afiseaza continuturile registrelor utilizator, accesibile prin program.

r<reg><ENTER>

afiseaza si permite modificarea registrului utilizator *reg*.

t<nr_instr><ENTER>

("trace") executa numarul de instructiuni *nr_instr* de la valoarea curenta a contorului de instructiuni IP.

m<adr1>,<adr2>,<adr3><ENTER>

("move") muta zona de memorie cuprinsa intre adresele *adr1* si *adr2* in noua zona incepand de la adresa *adr3*.

s<adr1>,<adr2>,<lista_oct><ENTER>

("search") cauta lista de octeti in zona cuprinsa intre adresele *adr1* si *adr2*.

g=<adr1>,<adr2><ENTER>

("go") lanseaza in executie codul aflat in memorie incepand de la adresa *adr1* pana la adresa *adr2* ("breakpoint" - punct de suspendare a executiei programului). Daca *adr1* lipseste (lipseste si caracterul '=') executia este lansata de la adresa curenta a contorului de program IP, pana la adresa *adr2*.

Tema

- 1) Sa se afiseze continutul zonei de memorie cuprinsa intre adresele 100-13f.
- 2) Sa se umple zona de memorie 100-13f cu valoarea constanta 55 si sa se verifice operatia.
- 3) Sa se introduca in fiecare locatie din zona 100-13f o valoare reprezentata de ultimii 8 biti de adresa a locatiei respective (00, 01, 02, . . .) si sa se verifice.
- 4) Sa se mute zona de memorie precedenta in zona cuprinsa intre adresele 300-33f.
- 5) Sa se compare zona de memorie 100-13f cu zona 200-23f si apoi cu zona 300-33f.
- 6) Sa se caute octetii 0f, 1f, 2f si 3f in zona de memorie 100-33f.
- 7) Sa se afiseze si sa se modifice continuturile registrelor utilizator.

8) Sa se introduca o secventa de program in limbaj de asamblare incepand de la adresa 300 si sa se execute secventa respectiva urmarind modificarea registrelor utilizator.

9) Sa se execute secventa precedenta de program instructiune cu instructiune, urmarind modificarea registrelor.

10) Sa se dezassembleze continutul zonei de memorie de la punctul precedent, precum si continutul zonei cuprinse intre adresele 200-23f.