



PRIMERI UKAZOV

Stran 6-5

je zadnji podatek v vrsti.

6.4 POSEBNI UKAZI

Od ostalih ukazov si oslejšmo še ukaz PUSHx, ki zapiše podatek ali več podatkov na sklad in ukaz INDEX, ki izračuna naslov podatka v enodimenzionalnem polju, če podamo indeks tega podatka.

Imamo dve obliki ukaza PUSH. Z ukazom PUSHL zapišemo na sklad dolgo besedo, z ukazom PUSHAX pa naslov podatka, ki je lahko byte, beseda, dolga beseda itd. V vseh teh primerih se zapiše na sklad naslov, ki je dolga beseda. Kako velik je podatek je važno le pri indeksnem naslavljanju, ker se s pomočjo dolžine izračuna naslov podatka. Če je vrednost v registru R7 10, zapiše prvi ukaz na sklad vrednost TABELA+4*10, drugi pa vrednost TABELA+2*10:

```
PUSHAL TABELA R7
PUSHAW TABELA R7
```

Druga vrsta ukaza PUSH je PUSHR. Z njim zapišemo na sklad vsebine registrov, ki jih določimo z masko. Ukaz ima en parameter in to je beseda, v kateri vsak bit pomeni en register. Če je v bitu N zapisana enica, to pomeni, da želimo shraniti na sklad vrednost registra RN.

```
MASKA: .WORD 12 ; Maska s postavljenima bitoma 2
; in 3.
```

```
PUSHR MASKA ; Zapiše na sklad vsebino
; registrov R2 in R3.
```

Z ukazom INDEX izračunamo naslov podatka. Ukaz ima šest parametrov. Prvi je indeks, naslednja dva sta spodna in zgornja meja za indeks, četrti je velikost podatka, peti je vrednost indeksa, od katerega štejeemo podatke, zadnji pa je odmik od začetka polja.

```
INDEX: .LONG 5
VELIKOST: .LONG 14 ; Velikost podatka je 14 bytov.
```

```
INDEX INDEX, #4, #8, VELIKOST, #-4, R4
10$: TSTB PODATKI(R4)+
BNEQ 10$
```

Zgornji ukaz najprej testira, če je vrednost indeksa med 4 in 8 in ker je, prišteje vrednosti indeksa -4, kar pomeni, da je indeks prvega podatka 4, nato pa rezultat pomnoži z velikostjo podatka. Ilobljeno