



$CM\bar{P} \quad A, B \quad \begin{matrix} B \\ -A \end{matrix} \quad \begin{matrix} 84 \\ 6A \end{matrix} \quad N = \phi$

OSNOVNI UKAZI

$TST \quad A, \phi$

$N \quad A < \phi \quad \text{Stran 4-4}$

#### 4.2.3 Primerjanje števil

Števila primerjamo z ukazoma  $CM\bar{P}x$  in  $TSTx$ . S  $CM\bar{P}x$  primerjamo dva podatka enakega tipa,  $TSTx$  pa primerja podatek s celoštevilčno ali realno ničlo. Oba ukaza ne spremenita stanja v pomnilniku ali registrih, ampak postavita le pogojne bite v statusu procesorja.

#### 4.2.4 Ukazi za delo s posameznimi biti

Testiramo ali postavljamo lahko tudi posamezne bite ali skupine bitov v nekem podatku. Z ukazom  $BITx$  testiramo izbrane bite. Ukaz ima dva podatka, prvi je maska, ki je enakega tipa kot testirani podatek. V podatku testiramo bite na mestih, kjer so v maski zapisane enice. Z drugimi besedami lahko opišemo ukaz tako, da v podatku brišemo vse bite, ki ne ustrezajo postavljenim bitom v maski, torej naredimo logično operacijo  $IN$  med masko in podatkom, nato pa testiramo dobljeno vrednost kot z ukazom  $TSTx$ . Tudi ta ukaz podatka ne spremeni, postavi pa pogojne bite v statusu procesorja.

Z ukazoma  $BISx$  in  $BICx$  zapišemo enice ali ničle v bite, ki jih določa maska. Z ukazom  $BISx$  (bit set) naredimo torej logični  $ALI$  po bitih med masko in podatkom, z  $BICx$  (bit clear) pa naredimo logični  $IN$  med komplementom maske in podatkom. Po bitih lahko izvedemo tudi operacijo ekskluzivni  $ALI$ , ki da rezultat 1 ali točno le, ko je natančno eden od operandov 1. Ime ukaza je  $XORx$ .

V podatku lahko premaknemo vse bite v levo ali desno, s tem da pri premiku v levo izubimo vodilne bite, če uporabimo ukaz  $ASHx$ , ali pa jih prenesemo na desni konec števila z ukazom  $ROTL$ , pri premiku v desno pa izubimo ali prenesemo najmanj pomembne bite.

#### 4.2.5 Razni ukazi

Ničlo lahko zapišemo na nek naslov z ukazom  $MOVx$ , hitreje in s krajšo kodo pa dosežemo isto z ukazom  $CLRx$  (clear). Podobno imamo za prištevanje in odštevanje enice ukaza  $INCx$  (increment) in  $DECx$  (decrement), ki povečata ali zmanjšata podatek za 1.

Pri klicu podprogramov pogosto prenašamo parametre na sklad. Tudi to lahko naredimo z ukazom  $MOVx$ , vendar imamo za zapis dolge besede na sklad poseben ukaz  $PUSHL$  (push longword). Za zapisovanje na sklad uporabljajo izraz "push", za jemanje podatkov s sklada pa "pop".

Zanimiv je ukaz  $POLYx$ , ki izračuna vrednost polinoma v neki točki. Koefficienti polinoma in argument so realna števila. Koefficiente polinoma podamo v tabeli, kot je natančneje opisano v Architecture Handbook na strani 213 in naslednjih.