

# Asemblersko programiranje

# Asemblersko programiranje

- Zasniva se na korišćenju **asemblerских naredbi**
- Programski jezici visokog nivoa i niskog nivoa
- Mane i prednosti asemblerskog programiranja
- Primene asemblerskog programiranja
- **Svaki asemblerski jezik** vezan je za neki **konkretan procesor**
- **Hipotetički procesor KONCEPT** – ovladavanje principima rada računara

# Procesor KONCEPT

- **16-bitne lokacije**, ukupno  $2^{16}$  adresa za lokacije
- **Registri opšte namene** %0, %1, ..., %15
- **Status registar** – flegovi N, M, P, V
- **Neposredno, direktno, registarsko, posredno** (indirektno) i **indeksno adresiranje**
- **Dvoadresne naredbe** (0, 1, ili 2 operanda)
- **Prvi operand je ulazni, drugi je ulazno-izlazni**

# Celobrojna aritmetika

- Sabiranje: **SABERI**, **SABERI\_P** – posmatra se i logička promenljiva P, za višestruku preciznost
- Oduzimanje: **ODUZMI**, **ODUZMI\_P**
- Inkrement, dekrement: **DODAJ\_I**, **ODBIJ\_I**
- Poređenje: **UPOREDI** – oduzimanje i postavljanje vrednosti logičkih promenljivih

# SABERI

$PO_i$	$DO_i$	$P_i$	$Z_i$	$P_{i+1}$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$Z_i = PO_i \wedge DO_i \wedge P_i$$

$$P_{i+1} = ((PO_i | DO_i) \& P_i) | (PO_i \& DO_i)$$

$$P_0 = 0$$

**SABERI\_P:**  $P_0 = P$

**DODAJ\_1**

$$N = \sim(Z_{15} | Z_{14} | Z_{13} | Z_{12} | Z_{11} | Z_{10} | Z_9 | Z_8 | Z_7 | Z_6 | Z_5 | Z_4 | Z_3 | Z_2 | Z_1 | Z_0)$$

$$M = Z_{15}$$

$$P = P_{16}$$

$$V = ((\sim PO_{15}) \& (\sim DO_{15}) \& Z_{15}) | (PO_{15} \& DO_{15} \& (\sim Z_{15}))$$

# ODUZMI

$DO_i$	$PO_i$	$P_i$	$R_i$	$P_{i+1}$
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

$$R_i = PO_i \wedge DO_i \wedge P_i$$

$$P_{i+1} = ((DO_i | P_i) \& (\sim PO_i)) | (DO_i \& P_i)$$

$$P_0 = 0$$

**ODUZMI\_P:  $P_0 = P$**

**ODBIJ\_1, UPOREDI**

$$N = \sim(R_{15} | R_{14} | R_{13} | R_{12} | R_{11} | R_{10} | R_9 | R_8 | R_7 | R_6 | R_5 | R_4 | R_3 | R_2 | R_1 | R_0)$$

$$M = R_{15}$$

$$P = P_{16}$$

$$V = ((\sim PO_{15}) \& DO_{15} \& R_{15}) | (PO_{15} \& (\sim DO_{15}) \& (\sim R_{15}))$$

# Rukovanje bitovima

- Logičke operacije na nivou bitova:
  - logičko i – naredba **I**
  - logičko ili – naredba **ILI**
  - logičko ne – naredba **NE**
- Pomeranje na nivou bitova:
  - za jedno mesto u levo, množenje sa 2 – **LEVO**
  - za jedno mesto u desno, deljenje sa 2 – **DESNO**

## I, ILI, NE



$$N = \sim(R_{15}|R_{14}| R_{13}|R_{12}| R_{11}|R_{10}| R_9|R_8| R_7|R_6| R_5|R_4| R_3|R_2| R_1|R_0)$$

$$M = R_{15}$$

$$P = 0$$

$$R_i = PO_i \& DO_i \quad V = 0$$



$$N = \sim(R_{15}|R_{14}| R_{13}|R_{12}| R_{11}|R_{10}| R_9|R_8| R_7|R_6| R_5|R_4| R_3|R_2| R_1|R_0)$$

$$M = R_{15}$$

$$P = 0$$

$$R_i = PO_i | DO_i \quad V = 0$$



$$N = \sim(R_{15}|R_{14}| R_{13}|R_{12}| R_{11}|R_{10}| R_9|R_8| R_7|R_6| R_5|R_4| R_3|R_2| R_1|R_0)$$

$$M = R_{15}$$

$$P = 0$$

$$R_i = \sim PO_i \quad V = 0$$



# LEVO, DESNO

## LEVO

$$R_{i+1} = PO_i$$

$$R_0 = 0$$

$$N = \sim(R_{15}|R_{14}| R_{13}|R_{12}| R_{11}|R_{10}| R_9|R_8| R_7|R_6| R_5|R_4| R_3|R_2| R_1|R_0)$$

$$M = R_{15}$$

$$P = PO_{15}$$

$$V = PO_{15} \wedge PO_{14}$$

## DESNO

$$R_i = PO_{i+1}$$

$$R_{15} = PO_{15}$$

$$N = \sim(R_{15}|R_{14}| R_{13}|R_{12}| R_{11}|R_{10}| R_9|R_8| R_7|R_6| R_5|R_4| R_3|R_2| R_1|R_0)$$

$$M = R_{15}$$

$$P = PO_0$$

$$V = 0$$

# Prebacivanje

- Naredba za prebacivanje sadržaja između registara procesora i memorijskih lokacija: **PREBACI**
  - u lokaciju određenu prvim operandom se smešta vrednost određena drugim operandom
  - ne utiče na vrednosti logičkih promenljivih
  - kombinacije sufiksa: **RR, NR, DR, PR, IR, RD, RP, RI**

# Upravljačke naredbe

- **Mašinske naredbe se smeštaju u redosledu izvršavanja u memorijske lokacije sa uzastopnim rastućim adresama**
- Nije podesno u slučajevima kada obrada podataka zavisi od vrednosti obrađivanih podataka
- Redosled izvršavanja naredbi mora se određivati dinamički u toku izvršavanja programa
- **Uslovne i bezuslovne upravljačke naredbe**
- **Uslovne:** jedini operand je adresa ciljne naredbe ako je ispunjen uslov

# Uslovne upravljačke naredbe

Oznaka uslovne upravljačke naredbe	uslov uslovne upravljačke naredbe
SKOČI_ZA_== ili SKOČI_ZA_N	N
SKOČI_ZA_!= ili SKOČI_ZA_NE_N	$\sim N$
SKOČI_ZA_< ili SKOČI_ZA_P	P
SKOČI_ZA_>= ili SKOČI_ZA_NE_P	$\sim P$
SKOČI_ZA_>	$(\sim P) \& (\sim N)$
SKOČI_ZA_<=	$P \vee N$
SKOČI_ZA_±_<	$M \wedge V$
SKOČI_ZA_±_>=	$\sim (M \wedge V)$
SKOČI_ZA_±_>	$(\sim (M \wedge V)) \& (\sim N)$
SKOČI_ZA_±_<=	$(M \wedge V) \vee N$
SKOČI_ZA_M	M
SKOČI_ZA_NE_M	$\sim M$
SKOČI_ZA_V	V
SKOČI_ZA_NE_V	$\sim V$

# Bezuslovne upravljačke naredbe

- **SKOČI** – jedan operand (ciljna adresa), jedino izmena redosleda izvršavanja
- **POZОВI** – jedan operand (ciljna adresa)
  - Izmjena redosleda izvršavanja i smeštanje povratne adrese (engl. *return address* – adresa naredbe neposredno iza naredbe POZОВI) u registar %15
- **NATRAG** – nema operanada, jedino izmena redosleda izvršavanja
  - skok na adresu iz %15

# Asemblerski jezik Koncept

# EBNF notacija

## Proširena Bekus-Naurova forma

(engl. *Extended Backus-Naur Form* – EBNF):

- ime\_pravila → način\_formiranja
- razmak – razdvajanje delova pravila
- | – alternative
- ( ) – grupisanje delova pravila
- [ ] – pojavljivanje nijednom ili jednom
- { } – pojavljivanje nijednom, jednom ili više puta
- “ ” – poništavanje značenja specijalnih znakova

# EBNF definicije

- malo\_slovo  $\rightarrow$  a|b|c|č|ć|d|đ|e|f|g|h|i|j|k|l|m|n|o|p|r|s|š|t|u|v|z|ž
- cifra  $\rightarrow$  0|1|2|3|4|5|6|7|8|9
- decimalni\_broj  $\rightarrow$  cifra{cifra}
- heksa\_cifra  $\rightarrow$  cifra|A|B|C|D|E|F
- heksadecimalni\_broj  $\rightarrow$  0x(heksa\_cifra){heksa\_cifra}
- broj  $\rightarrow$  decimalni\_broj|heksadecimalni\_broj
- labela  $\rightarrow$  malo\_slovo{malo\_slovo|cifra|\_}



# EBNF definicije

- neposredni\_operand  $\rightarrow$  (\$broj|\$labela)
- direktni\_operand  $\rightarrow$  labela
- registar  $\rightarrow$  %(0|1|2|3|4|5|6|7|8|9|10|11|12|13|14|15)
- registarski\_operand  $\rightarrow$  registar
- posredni\_operand  $\rightarrow$  “(“registar“)”
- indeksni\_operand  $\rightarrow$  (broj|labela)“(“registar“)”
- razmak  $\rightarrow$  “ “ {“ “}
- nova\_linija  $\rightarrow$  “početak linije“ {“ “}

# EBNF definicije

- Obrazovanje osnovnih asemblerskih naredbi:

```
osnovna_naredba -> nova_linija [labela:] razmak
    ( (SABERI|SABERI_P) razmak registarski_operand,registarski_operand
    | (ODUZMI|ODUZMI_P) razmak registarski_operand,registarski_operand
    | UPOREDI          razmak registarski_operand,registarski_operand
    | (I|ILI)          razmak registarski_operand,registarski_operand
    | (DODAJ_1|ODBIJ_1) razmak registarski_operand
    | (NE|LEVO|DESNO)  razmak registarski_operand)
```

- Obrazovanje asemblerskih naredbi prebacivanja:

```
naredba_prebacivanja -> nova_linija [labela:] razmak
    (PREBACI_RR razmak registarski_operand,registarski_operand
    | PREBACI_NR razmak registarski_operand,neposredni_operand
    | PREBACI_DR razmak registarski_operand,direktni_operand
    | PREBACI_PR razmak registarski_operand,posredni_operand
    | PREBACI_IR razmak registarski_operand,indeksni_operand
    | PREBACI_RD razmak direktni_operand,registarski_operand
    | PREBACI_RP razmak posredni_operand,registarski_operand
    | PREBACI_RI razmak indeksni_operand,registarski_operand)
```

# EBNF definicije

- Obrazovanje **asemblerskih upravljačkih naredbi**:

```

upravljačka_naredba -> nova_linija [labela:] razmak
                        ( (SKOČI
                          | SKOČI_ZA_==
                          | SKOČI_ZA_!=
                          | SKOČI_ZA_<
                          | SKOČI_ZA_>=
                          | SKOČI_ZA_>
                          | SKOČI_ZA_<=
                          | SKOČI_ZA_±_<
                          | SKOČI_ZA_±_>=
                          | SKOČI_ZA_±_>
                          | SKOČI_ZA_±_<=
                          | SKOČI_ZA_N
                          | SKOČI_ZA_NE_N
                          | SKOČI_ZA_P
                          | SKOČI_ZA_NE_P
                          | SKOČI_ZA_M
                          | SKOČI_ZA_NE_M
                          | SKOČI_ZA_V
                          | SKOČI_ZA_NE_V
                          | POZOVI) razmak labela)
                          | NATRAG)

```

# EBNF za asemblerske direktive

- Direktive upućene assembleru za **zauzimanje i inicijalizaciju memorijskog prostora – ZAUZMI i NAPUNI**
- Direktive **nisu izvršne**, već su **uputstva assembleru**, prethode naredbama

direktiva → nova\_linija [labela:] razmak

(ZAUZMI|NAPUNI) razmak broj

# EBNF za asemblerski program

telo → {direktiva  
|osnovna\_naredba  
|naredba\_prebacivanja  
|upravljачka\_naredba}

program → POČETAK  
razmak labela telo nova\_linija  
KRAJ