

Ⓔ PECOM 64

---

Kučni računar



Ⓔ PECOM 64

---

Kućni računar

---

---

**UPUTSTVO ZA  
RUKOVANJE**

---

# S A D R Ž A J:

UVOD .....	5
DODELA MEMORIJE U PECOM u 64 .....	6
RAD NA PECOM-u .....	7
KORIŠĆENJE ĆIRILICE .....	7
UPUTSTVO ZA UPOTREBU MONITOR a .....	7
Pozivanje MONITOR-a+ .....	8
Naredbe MONITOR a+ .....	8
D, I, E, S, F .....	8
M, W, CTRL R, C, I, O, B, K, L .....	9
CREDIT - EKRANSKI EDITOR .....	9
POZIVANJE EDITORA .....	10
1. NAREDBE ZA OBRADU TEKSTA .....	10
CTRL , CTRL , CTRL , CTRL , CTRL E, CTRL S, CTRL K,..	10
CTRL L .....	10
CTRL blanko, CTRL A, DEL, CTRL V, LINE FEED, ESC, CTRL Q,..	11
CTRL T, CTRL RETURN, BREAK, .....	11
Naredbe za rad sa blokovima .....	11
CTRL Y, CTRL U, CTRL SHIFT DEL, CTRL C, CTRL I, CTRL R .....	11
CTRL P, CTRL O .....	12
2. KOMANDNI NAĆIN RADA .....	12
2.1. Ostale naredbe .....	12
LOAD xxxx, LOAD, SAVE xxxx, BSAVL xxxx, ESC, ASS, TAB, PRINT,	12
LINE FEED .....	12
CTRL C .....	13
ASEMBLER ZA CDP 1802 .....	13
Pozivanje ASEMBLER-a .....	13
Naćin rada ASEMBLER-a .....	13
Naredbe ASEMBLER-a .....	13
KONSTANTE .....	14
LABELA .....	14
MATEMATIĆKI IZRAZ .....	14
Naredbe za rad sa registrima .....	14
GHI, GLO, PHI, PLO, INC, .....	14
DEC, LDI, LD, PUSH, POP .....	15
Naredbe kratkog granjanja .....	15
BR .....	15

BZ, BNZ, BDF, BNF, (BQ, BNQ, B1, BN1, B2, BN2, B3, BN3, B4, BN4)	16
Naredbe dugog granjanja	16
LBR, LBZ,	16
LBNZ, LBDF, LBNF, LBQ, LBNQ	17
Naredbe preskoka	17
SKP, LSKP, LSZ, LSNZ, LSDF, LSQ, LSNQ, LSIF	17
Naredbe za rad sa memorijom	17
LDN	17
LDA, STR, LDX, LDXA, STXD	18
Naredbe za logičke operacije	18
OR, ORI, XOR, XRI, AND, ANI, SHR, SHRC, SHL, SHLC	18
Naredbe za aritmetičke operacije	18
ADD, ADI, ADC, ADCI, SD, SDI, SDB, SDBI	18
SM, SMI, SMB, SMBI	19
Upravljačke naredbe	19
SFP, SEX, IDLE, SLO, REQ, SAV, MARK, RET, DIS, NOP	19
Specijalne naredbe	19
CALL, RETS, BYTE	19
WORD, TEXT, TEXT0	20
Pseudo naredbe ASEMBLER-a	20
EQU, ORG, OPT	20
<b>PRILOG:</b>	
NAREDBE MONITOR-a	21
NAREDBE EDITOR-a	22
NAREDBE KOMANDNOG NAČINA RADA EDITOR-a	23
GREŠKE KOMANDNOG NAČINA RADA EDITOR a	23
NAREDBE ASEMBLER-a	24
GREŠKE PRI ASEMBLIRANJU	29
ASCII TABELA ZA UGRAĐENE ZNAKE I SIMBOLE	30
PRIMERI PROGRAMIRANJA	32

# EI PECOM 64 KUĆNI RAČUNAR

## U V O D

Kućni računar PECOM 64 predstavlja poboljšanu verziju kućnog računara PECOM-a (odnosno PECOM-a 48). Poboljšanje se sastoji u tome što novi PECOM sada ima još jedan EPROM koji sadrži sledeće sistemske programe:

- MONITOR + (programi za rad sa memorijom i programi za rad u mašinskom kodu)
- ekranski EDITOR-CREDIT (služi za unošenje programa u računar, a omogućuje i prevođenje programa iz ASSEMBLER-a u mašinski jezik).
- programi za podršku rada PECOM-a u računarskoj učionici.

Prema tome, da biste od postojećeg PECOM-a 32 (48) dobili PECOM 64 za rad u računarskoj učionici, potrebno je da zamenite oba EPROM-a.

Dodela ukupnog memorijskog prostora u PECOM-u 64 prikazana je na sl.1

EI PECOM 64

Kućni računar

UPUTSTVO ZA  
RUKOVANJE

### Nepravilno

strana

- 3 — Sadržaj - CTRL ,CTRL ,CTRL ,CTRL ,CTRL E,CTRL S,CTRL K, ..10
- 13 — Način rada ASSEMBLER-a - zadnja rečenica  
Pri tom se kursor postavlja na mesto na kome je dekovana greška.
- 16 — Naredbe kratkog grananja - nedostaje BZ parametar
- 28 — Pseudo naredbe ASSEMBLER-a - drugi red  
ERG par 1, (par 2) - Startna adresa (relocirana adresa)

### Pravilno

strana

- 3 — Sadržaj - CTRL→,CTRL←,CTRL↓,CTRL↑,CTRL E,CTRL S,CTRL K, ..10
- 13 — Pri tom se kursor postavlja na mesto na kome je detektovana greška.
- 16 — BZ parametar  
Kratki skok ako je sadržaj akumulatora D jednak nuli.
- 28 — ORG par 1, (par 2) - Startna adresa,(relocirana adresa)

**ISPRAVKE**

0000		RADNA OBLAST BASIC-a
-		
-		
-		KORISNIČKA
-		OBLAST
-		
7000		
7CA0		
7CB0		STEK
8000		BAFER TASTATURE / UPIS - ČITANJE SA TRAKE
		USLUŽNI PROGRAM
8800		
8BC0		INICIJALNI KARAKTER GENERATOR
9000		NESTANDARDNE NAREDBE ZA BASIC
-		
-		BASIC
		INTERPRETATOR
C000		
		EDITOR
D000		MONITOR +
E000	-	+
		ASEMBLER
E400		KARAKTER GENERATOR ZA ĆIRILICU
E7C0		PROGRAMI ZA KONVERZIJU LATINICA - ĆIRILICA
E920		
F000	-	
F400		
F7C0		KARAKTER GENERATOR U RAM-u
F800		
FBC0		MEMORIJA DISPLEJA
FFFF		

Slika 1.

Prilikom uključanja PECOM-a na ekranu se pojavi poruka:

```
EI PECOM 64  
READY  
:
```

Ova poruka znači da je računar spreman za prihvatanje naredbi u BASIC-u.

Da bi se prešlo u režim rada MONITOR+ potrebno je uneti

```
: PROB
```

a zatim pozvati program MONITOR+ koji počinje na adresi D000

```
> PD000
```

U EDITOR se dalje ulazi unošenjem naredbe

```
> E
```

Ekran se briše. Trepćući kursor pojavljuje se u gornjem levom uglu ekrana.

Prelazak u komandni način rada EDITOR-a vrši se pritiskom na taster

```
BREAK.
```

Na ekranu se pojavljuje horizontalna crta sa trepćućim kursorom u gornjem levom uglu.

Iz komandnog načina rada EDITOR-a može se pozvati ASSEMBLER naredbom

```
ASS
```

Povratak iz komandnog načina rada EDITOR-a u datoteku vrši se pritiskom na taster

```
LINE FEED.
```

Prelazak iz komandnog načina rada EDITOR-a u MONITOR+ vrši se pritiskom na taster

```
CTRL C
```

Prelazak iz režima rada MONITOR-a+ u BASIC vrši se naredbom

```
> B
```

### KORIŠĆENJE ĆIRILICE

Ako se želi ispisivanje teksta na ekranu ćirilicom potrebno je da prva naredba programa napisanog u BASIC-u bude:

```
: 1 CALL (&E7C0)
```

Normalno, broj linije može biti bilo koji ali je bitno da naredba CALL bude upotrebljena pre prve PRINT naredbe u programu.

Posle startovanja programa sa RUN, tekst koji se prikazuje na ekranu biće ispisan ćirilicom. Ako se dalje želi neka izmena u programu (npr. dodavanje nekih naredbi ili slično) potrebno je preći u režim rada latinicom na sledeći način:

```
ЦАЛЛ (&E818)
```

Primećujete da ovde nije upotrebljen broj linije, što znači da se sada naredba CALL koristi u neposrednom režimu. Na ovaj način vaš BASIC program ostaje nepromenjen, a računar se (posle RET) nalazi ponovo u režimu rada latinicom.

### UPUTSTVO ZA UPOTREBU MONITORA+

MONITOR+ je naziv za poboljšanu verziju originalnog MONITOR-a računara PECOM 32. Obuhvata sve naredbe za rad sa memorijom i to:

- startovanje mašinskih programa,
- prikazivanje sadržaja memorijskih lokacija,
- unošenje sadržaja memorijskih lokacija,
- prenošenje blokova podataka iz memorije PECOM-a na kasetu
- učitavanje blokova podataka sa kasete u memoriju PECOM-a

### Pozivanje MONITOR-a+

MONITOR+ poziva se iz PECOM-ovog originalnog MONITOR-a naredbom PD000. MONITOR se javlja znakom ' > ', što znači da je spreman za prihvatanje naredbi. Prekid izvršenja bilo koje naredbe vrši se pomoću tastera BREAK.

### Naredbe MONITOR-a+

D

Ova naredba služi za štampanje sadržaja memorijskih lokacija u heksadecimalnom kodu na ekranu ili štampaču. Na pitanje FROM korisnik unosi početnu adresu bloka koji se štampa. Potrebno je uneti najmanje jedan, a najviše četiri znaka. Ukoliko se unese više od 4 znaka, u obzir se uzimaju poslednja 4 otkucana znaka. Prikazivanje teče toliko dugo, koliko se drži pritisnut taster CTRL. Prekidanje prikaza obavlja se pritiskom na dirku BREAK.

Npr.

```
> D
FROM 5500
```

daje

```
5500 00 00 00 00 00 00 00 00
```

što znači da je 8 memorijskih lokacija počev od adrese 5500 prazno.

Prikazivanje teče toliko dugo, koliko se drži pritisnut taster CTRL. Prekidanje prikaza obavlja se pritiskom na taster BREAK.

I

Ovom naredbom upisuje se sadržaj u memorijske lokacije.

Na pitanje FROM unosi se adresa na kojoj se započinje upis. Vrednost sadržaja unosi se u heksadecimalnom kodu.

Npr.

```
>I
FROM 5500
5500 F8 01 B8 F8 00 A8
```

znači da se u memorijske lokacije sa heksadecimalnom adresom počev od 5500 do 5505 upisuje sadržaj F801B8F800A8. šest bajtova na mašinskom jeziku upisuje konstantu 0100 u registar R8.

E

Ovom naredbom poziva se editor CREDIT.

S

Ovo je naredba za pretraživanje. Njome se u delu definisanom početnom i krajnjom adresom traže mesta na kojima se nalazi specifikirani niz ASCII znakova. Početna adresa memorijskog bloka u kome se obavlja pretraživanje unosi se na pitanje FROM, a njegova konačna adresa na pitanje TO. Na pitanje WHAT unosi se sadržaj niza u heksadecimalnom kodu. MONITOR+ javlja adrese na kojima se nalazi definisani niz.

Naredno prikazivanje reda vrši se pritiskom na dirku RETURN.

F

Ovom naredbom se sve lokacije nekog memorijskog bloka pune vrednošću datom u heksadecimalnom kodu. Na pitanje FROM unosi se početna, a na pitanje TO krajnja adresa bloka. Na pitanje What unosi se jednobajtna vrednost kojom se pune lokacije bloka.

Npr.

```
>F
FROM 3000
TO 300C
WHAT 8
>D
FROM 3000
```



```
3000 08 08 08 08 08 08 08 08
3008 08 08 08 08 08 00 00 00
```

M

Ovo je naredba kojom se vrši pomeranje memorijskog bloka u memoriji. Na pitanje FROM unosi se početna, na pitanje TO krajnja adresa bloka, a na pitanje WHERE adresa na koju se pomera blok.

Npr.

```
> M
FROM 3000
TO 300C
WHERE 3500
> D
FROM 3500
3500 08 08 08 08 08 08 08 08
3508 08 08 08 08 08 00 00 00
```

W

Memorijski blok se snima na kasetu. Potrebno je uneti samo početnu i krajnju adresu bloka. Snimanje se može prekinuti pritiskom na taster BREAK. Po završetku snimanja (posle prekida) kontrolu preuzima originalni MONITOR PECOM-a. Povratak u MONITOR+ vrši se naredbom PD000.

CTRL R

Ovom naredbom se datoteka sa kasete učitava u memoriju, i to na iste adrese sa kojih je bilo izvršeno snimanje. Prekidanje naredbe moguće je izvršiti jedino pritiskom na taster za RESET.

C

Ovo je naredba za startovanje programa u mašinskom kodu. Na pitanje FROM unosi se početna adresa programa. Zavisno od toga kako je napisan, završetak programa može biti sledeći:

- po izvršenju, program se vraća u MONITOR+
- program skače u originalni MONITOR (povratak u MONITOR+ vrši se sa PD000)
- program skače u BASIC
- program se 'ukopa' (za nastavak rada pokušati pritisak na RESET dirku)

1

Ovo je naredba kojom se PECOM priprema za štampanje na štampaču. Sve što bi se (nakon ove naredbe) prikazalo na ekranu odštampa se na štampaču.

```
> 1
> D
> FROM (adresa) i štampanje se obavlja sve dok je pritisnut taster CTRL.
```

0

Ovom naredbom PECOM se priprema za štampanje na ekranu monitora ili televizora.

B

Skok u BASIC.

K

Ovom naredbom vrši se konverzija teksta u ĆIRILICU.

L

Ova naredba vrši prebacivanje teksta iz ĆIRILICE u LATINICU.

## CREDIT - EKRANSKI EDITOR

Za unošenje nekog programa (nekog teksta) u računar potrebno je da se znakovi otkucani na tastaturi prikladno protumače i smeste u memoriju računara. Poželjno je koris-

niku obezbediti preglednost u toku rada i mogućnost unošenja ispravki i izmena u tekstu, brisanje i oblikovanje teksta.

EDITOR-CREDIT pruža čak i mnogo više od toga.

Ime je nastalo kao skraćenica izraza Cathod Ray tube EDITOR što u prevodu znači i elektronski EDITOR. On omogućava:

- kreiranje programa, odnosno proizvoljnih datoteka, podataka i tekstova,
- pozivanje ASEMBLER-a za mikroprocesor CDP 1802.

U tu svrhu CREDIT koristi dve grupe naredbi:

1. Naredbe za obradu teksta (pomeranje kursora, brisanje i dodavanje znakova, definisanje, prebacivanje, brisanje i pretraživanje blokova i itd.)
2. Naredbe za komandni način rada (zapis i čitanje na kaseti, štampanje, pozivanje ASEMBLER-a i prekidanje rada).

Ukoliko neka od naredbi ne može da se izvrši, EDITOR javlja grešku.

## POZIVANJE EDITORA

CREDIT se poziva naredbom iz MONITOR-a+

> E

Ukoliko je pre pozivanja EDITOR-a u memorijskom delu PECOM-a korišćenog za smeštanje teksta već ranije postojao neki tekst kreiran EDITOR-om, onda će se na ekranu pojaviti ona strana teksta sa koje je izvršen izlaz iz EDITOR-a. Ako se EDITOR poziva prvi put, pojavi se prazan ekran sa trepćućim kursorom u gornjem levom uglu. Tada se može otpočeti i sa unošenjem teksta. Na raspolaganju su svi znakovi abecede, numerički znaci, razne oznake i neki posebni znaci.

### 1. NAREDBE ZA OBRADU TEKSTA

Ovaj skup naredbi služi za pomeranje kursora po datoteci, brisanje znakova i operacije sa blokovima.

CTRL →

Pomeranje kursora udesno. Ukoliko se nalazi na kraju reda kursor se pomera na početak sledećeg reda.

CTRL ←

Pomeranje kursora u levo. Ukoliko se nalazi na početku reda kursor se pomera na kraj prethodnog reda.

CTRL ↓

Pomeranje kursora na kraj tekućeg reda. Ukoliko se već nalazi na kraju reda kursor se pomera na kraj sledećeg reda.

CTRL ↑

Pomeranje kursora na kraj prethodnog reda.

CTRL E

Skok kursora na kraj datoteke.

CTRL S

Skok kursora na početak datoteke.

CTRL K

Skok kursora na početak prethodne reči.

CTRL L

Skok kursora na početak sledeće reči.

**CTRL blanko**  
Skok kursora na kraj reči.

**CTRL A**  
Na mestu na kome se nalazi kursor ubaci se blanko znak.

**DEL**  
Brisanje znaka levo od kursora.

**CTRL V**  
Ova naredba se koristi kada se na mestu na kome se nalazi kursor želi umetnuti neki tekst. Pritiskom na tastere CTRL i V ulogu kursora preuzima trepćući znak \*. Sve što se od tog trenutka otkuca na tastaturi, ostane umetnuto u tekst. Upotrebom neke od naredbi za pomeranje (npr. CTRL → , ← ) umetanje se završava, vraća se stari kursor, izvrši se pomeranje i rad dalje teče na uobičajeni način.

**LINE FEED**  
Skok kursora na sledeću stranu.  
Sadržaj tekuće strane na ekranu zameni se sadržajem naredne strane. Strana broji 24 reda. Naredna strana započinje sa poslednjim redom tekuće strane.

**ESC**  
Naredba poput LINE FEED, s tom razlikom što se skok vrši za jednu stranu unatrag.

**CTRL Q**  
Za početak strane postavi se red u kome se nalazi kursor.

**CTRL T**  
Skok kursora do najbliže tabulacione tačke.

**CTRL RETURN**  
Na mestu na kome se nalazi kursor ubaci se znak za novi red (CR) i izvrši skok kursora do prve tabulacione tačke.

**BREAK**  
Prelazak u komandni način rada.

Naredbe za rad sa blokovima

Blok može biti bilo koji deo datoteke, može da sadrži od jednog znaka pa do nekoliko strana. Svrha bloka je da omogući jednostavno prebacivanje, brisanje ili kopiranje dela programa, čime se štedi na nepotrebnom kucanju. Za ove operacije EDITOR obezbeđuje dodatni deo memorije za rad (buffer). Dužina buffera iznosi 1kB (Byte), koliko iznosi ograničenje za veličinu bloka.

Blok je definisan svojim početkom i krajem. EDITOR uzima za početak bloka mesto na kome se nalazi (nevidljivi) znak CTRL Y koji se naziva marker. Kraj bloka definisan je trenutnim položajem kursora (naravno, marker mora prethoditi kursoru). Početak bloka predefiniše se tako da se na željeno mesto otkuca CTRL Y.

**CTRL Y**  
Postavljanje početka bloka.

**CTRL U**  
Sadržaj bloka od markera do kursora se ubacuje u buffer.

**CTRL SHIFT DEL**  
Sadržaj bloka briše se iz datoteke.

**CTRL C**  
Zamena pozicije kursora i markera bloka.

**CTRL I**  
Sadržaj buffera ubaci se na mesto na kome se trenutno nalazi kursor.

**CTRL R**  
Rolovanje kursora na liniji : trenutni položaj kursora, početak datoteke i početak bloka.

## CTRL P

Ako se u datoteci od kursora nadalje nalazi niz znakova koji je u potpunosti identičan nizu znakova smeštenom u bufferu, onda kursor skoči na početak prvog takvog niza. Ako ovakvog niza u bufferu nema, kursor ostaje nepomeren.

## CTRL O

Obavlja istu funkciju kao i CTRL P, jedino što se prilikom pretraživanja ne pravi razlika između velikih i malih slova.

## 2. KOMANDNI NAČIN RADA

### 2.1. Ostale naredbe

#### LOAD xxxx

Ovom naredbom se program (podaci) učitaju sa kasete i upišu u memoriju PECOM-a 64 (tj. u editiranu datoteku na mestu kursora).

Može se upotrebiti na dva načina:

#### LOAD xxxx

Gde je xxxx ime datoteke. U tom slučaju sa kasete se učitava prva po redu datoteka koja je bila zapisana pod datim imenom.

#### LOAD

Na ovaj način se, bez obzira na ime, učitava prva po redu datoteka koja se nađe na kaseti.

Nakon pozivanja ove naredbe potrebno je startovati kasetofon.

#### SAVE xxxx

Ovom naredbom se sadržaj editovane datoteke snimi na kasetu pod imenom definisanim ASCII nizom xxxx. Ime datoteke je neophodno navesti. Pre pozivanja naredbe potrebno je startovati kasetofon.

#### BSAVE xxxx

Pomoću ove naredbe se sadržaj bloka snimi na kasetu pod imenom određenog ASCII nizom xxxx.

#### ESC

Pritiskom na dirku ESC ponovo se otkuca poslednja upotrebljena naredba u komandnom načinu rada.

#### ASS

Pozivanje ASEMBLER-a

#### TAB

Podešavanje tabulacije.

Prilikom kucanja nekih tekstova, kao npr. formulara asemblerskih programa dolazimo u situaciju da nam treba mogućnost skoka kursora od njegovog trenutnog položaja do najbliže unapred utvrđene pozicije. Raspored unapred fiksiranih pozicija naziva se tabulacija, a odgovarajuće tačke tabulacijske tačke.

Postavljanje tabulacijskih tačaka obavlja se na sledeći način:

- kada se otkuca naredba TAB na ekranu se pojavi 'tabulaciona osa' tj. niz oznaka od 0 do 9 kojima se označavaju kolone.
- dirkom CAPS LOCK namestimo da se na ekranu mogu štampati samo velika slova.
- pomeranje kursora do tabulacione tačke vrši se dirkama → i ←, a postavljanje ili brisanje tabulacione tačke ostvaruje se dirkom ↑. Skakanje na postavljene tabulacione tačke vrši se dirkom ↓.
- završetak postavljanja tabulacije ostvaruje se pritiskom na dirku BREAK.

#### PRINT

Ovom naredbom se blok definisan markerom i kursorom odštampa na štampaču.

#### LINE FEED

Povratak u datoteku iz komandnog načina rada.

## CTRL C

Skok iz EDITOR-a (tj. iz komandnog načina rada) u MONITOR+.

## ASSEMBLER ZA CDP 1802

ASSEMBLER služi za prevođenje programa iz mnemoničkog u mašinski kod. Pored originalnih naredbi za mikroprocesor CDP 1802 ovaj assembler na računaru sadrži i neke dodatne (korisne) naredbe, koje se inače pojavljuju kod nekih drugih mikroprocesora napredne arhitekture. To se pre svega odnosi na operacije punjenja registara i memorijskih lokacija. Korišćenjem ovakvih naredbi korisnik izbegava da za jednu istu operaciju stalno ponavlja niz originalnih naredbi CDP-a 1802.

### Pozivanje ASSEMBLER-a

Asembliranje izvornog programa koji je trenutno pod kontrolom EDITOR-a vrši se nakon pozivanja ASSEMBLER-a naredbom ASS u komandnom načinu rada.

### Način rada ASSEMBLER-a

ASSEMBLER prelazi preko izvornog programa dvaput. U toku prvog prolaska kreira se tabela sa imenima labela i tabela sa vrednostima labela. U toku drugog prolaza vrši se dekodiranje naredbi i upis mašinskog koda u memoriju. U slučaju neke sintaksne greške u izvornom programu ASSEMBLER prekida sa daljim radom i vraća se u EDITOR. Pri tom se kursor postavlja na mesto na kome je dekovana greška.

### Naredbe ASSEMBLER-a

Sve naredbe ASSEMBLER-a možemo svrstati u nekoliko grupa:

- naredbe za rad na registrima,
- naredbe za rad sa memorijom,
- naredbe kratkog grananja,
- naredbe dugog grananja,
- naredbe preskoka,
- naredbe za logičke operacije,
- naredbe za aritmetičke operacije,
- upravljačke naredbe,
- posebne naredbe

Izuzev specijalnih naredbi opšti oblik naredbe u ASSEMBLER-u je:

MNEMONIC (operand 1), (operand 2)

pri čemu operandi mogu biti:

- registar (ili deo registra) i
- parametar

Za obeležavanje registara koriste se sledeći simboli:

R oznaka za registar

n, m heksadecimalni brojevi koji mogu imati vrednost 0,1,...,E ili F (npr. oznaka RA govori da je operand deseti registar iz grupe registara opšte namene)

q, p promenljive koje mogu imati vrednosti H (high) ili vrednost L (low). Vrednost H se uzima onda kada se želi označiti byte veće težine, a L za byte manje težine (npr. R7.H znači da je operand viši byte registra 7).

Razlikujemo tri vrste parametara:

- konstante
- labela i
- matematički izraz

## KONSTANTE

Pod pojmom konstante podrazumeva se neka jednobitna ili dvobitna reč. Može se izraziti u binarnom, heksadecimalnom i decimalnom kodu pomoću ASCII znaka. Način označavanja je sledeći:

a) binarni izraz: %xxxx

Nakon znaka % sledi najviše 8 (za jednobitni) ili 16 (za dvobitni operand) znakova 1 ili 0.

Primer:

%10110101 : vrednost jednobitnog operanda

%1100011011101000 : vrednost dvobitnog operanda

b) heksadecimalni broj: &xxxx

Nakon znaka & slede heksadecimalni brojevi 0, 1, ..., E ili F (1 ili 2 znaka jednobitne i 1 do 4 znaka za dvobitne operande)

c) decimalni broj: xxxx

Vrednost operanda može se izraziti pozitivnim ili negativnim celim brojem.

d) ASCII znak: "x"

Vrednost jednobitnog operanda jednaka je kodu ASCII znaka navedenog između navodnika.

## LABELA

Labela je niz alfanumeričkih znakova kojima se u programu označava neka memorijska lokacija. Svakoј labeli pridružena je dvobitna vrednost koju joj ASSEMBLER dodeli prilikom prvog prolaska kroz program.

## MATEMATIČKI IZRAZ

Matematički izraz predstavljaju dva ili više operanda (labela ili konstante) koji su međusobno povezani nekom aritmetičkom ili logičkom operacijom (+, -, AND, OR, XOR). Koначna vrednost matematičkog izraza izračunava se prilikom drugog prolaska ASSEMBLERA kroz izvorni program. Operacije se izvode redom kojim se navode. Upotreba zagrada nije moguća.

## Naredbe za rad sa registrima:

Programi na mašinskom jeziku slobodno koriste registre RB, RA, RC, RD, i RE.

Ako treba da se upotrebe neki drugi registri, sadržaj tih registara treba prvo da se smesti u magacin i obnovi pre vraćanja u BASIC 3.

GHI Rn ili parametar

Sadržaj višeg byta registra n smešta se u akumulator.

Primer:

Naredbe GHI RE ili GHI 14 ili GHI &E su ekvivalentne.

GLO Rn ili parametar

Sadržaj nižeg byta registra n smešta se u akumulator.

PHI Rn ili parametar

Sadržaj akumulatora smešta se u viši byte registra n.

PLO Rn ili parametar

Sadržaj akumulatora smešta se u niži byte registra n.

INC Rn ili parametar

Sadržaj registra n uveća se za 1.

DEC Rn ili parametar

Sadržaj registra n umanji se za 1.

LDI parametar

Akumulator se puni vrednošću parametra.

LD Rn, Rm

Sadržaj registra m prepíše se u registar n. Prvobitni sadržaj akumulatora se menja.

LD Rn.q, Rm.p

Viši (niži) byte registra n puni se vrednošću višeg (nižeg) byta registra m.

LD Rn.q, A

Viši (niži) byte registra n puni se vrednošću akumulatora.

LD Rn.q, parametar

Viši (niži) byte registra n puni se vrednošću parametra. Nakon operacije prvobitni sadržaj akumulatora se menja.

LD Rn, parametar

Registar n puni se vrednošću parametra. Po završetku operacije sadržaj akumulatora se menja.

LD A, parametar

Akumulator A puni se vrednošću parametra. Ova naredba je ekvivalentna naredbi LDI.

LD A, Rn.q

Akumulator A puni se sadržajem višeg (nižeg) byta registra n. Ova naredba ekvivalentna je naredbama PHI i PLO.

PUSH Rn

Sadržaj registra n smešta se u stek.

PUSH Rn.q

Sadržaj višeg (nižeg) byta registra n smešta se u stek.

PUSH A

Sadržaj akumulatora smešta se u stek.

POP Rn

Sadržaj registra n obnavlja se iz steka.

POP Rn.q

Sadržaj višeg (nižeg) byta registra n obnavlja se iz steka.

POP A

Sadržaj akumulatora A obnavlja se iz steka.

### Naredbe kratkog grana nja

Ovim naredbama vrši se grananje programa na memorijske lokacije unutar tekuće strane

BR parametar

Vrši se безусловni skok na adresuna tekućoj strani pri čemu je sadržaj nižeg byta adrese jednak vrednosti parametra.

**BNZ parametar**

Kratki skok ako sadržaj akumulatora D nije jednak nuli.

**BDF parametar**

Kratki skok ako je DF registar na logičkoj 1.

**BNF parametar**

Kratki skok ako je DF resetovan.

**BQ parametar**

Kratki skok ako je Q setovan.

**BNQ parametar**

Kratki skok ako je Q resetovan.

**B1 parametar**

Kratki skok ako je EF1 setovan.

**BN1 parametar**

Kratki skok ako je EF1 resetovan.

**B2 parametar**

Kratki skok ako je EF2 setovan.

**BN2 parametar**

Kratki skok ako je EF2 resetovan.

**B3 parametar**

Kratki skok ako je EF3 setovan.

**BN3 parametar**

Kratki skok ako je EF3 resetovan.

**B4 parametar**

Kratki skok ako je EF4 setovan.

**BN4 parametar**

Kratki skok ako je EF4 resetovan.

### Naredbe dugog grananja

Kod naredbi dugog grananja operand je dvo-bytan i predstavlja adresu lokacije u memoriji na koju se eventualno vrši skok.

**LBR parametar**

Bezuslovni skok na lokaciji čija je adresa jednaka vrednosti parametra.

**LBZ parametar**

Dugi skok ukoliko je vrednost sadržaja akumulatora D jednaka nuli.



**LBNZ parametar**

Dugi skok ukoliko sadržaja akumulatora nije nula.

**LBDF parametar**

Dugi skok ukoliko je DF setovan

**LBNF parametar**

Dugi skok ukoliko je DF na nuli.

**LBQ parametar**

Dugi skok ukoliko je Q setovan.

**LBNQ parametar**

Dugi skok ukoliko je Q resetovan.

### Naredbe preskoka

**SKP**

Naredba kratkog preskoka.

**LSKP**

Naredba dugog preskoka.

**LSZ**

Dugi preskok ukoliko je vrednost u akumulatoru D nula.

**LSNZ**

Dugi preskok ukoliko vrednost u akumulatoru nije nula.

**LSDF**

Dugi preskok ukoliko je DF na logičkoj jedinici.

**LSNF**

Dugi preskok ukoliko je DF na logičkoj nuli.

**LSQ**

Dugi preskok ukoliko je Q setovan.

**LSNQ**

Dugi preskok ukoliko je Q resetovan.

**LSIE**

Dugi preskok ukoliko je IE na logičkoj jedinici.

### Naredbe za rad sa memorijom

**LDN Rn ili parametar**

Memorijski byte adresiran sadržajem registra n prenosi se u akumulator.

#### LDA Rn ili parametar

Memorijski byte adresiran sadržajem registra n prenosi se u akumulator, a sadržaj registra n uvećava se za jedan.

#### STR Rn ili parametar

Sadržaj akumulatora se prebacuje u memorijsku lokaciju adresiranu sadržajem registra n.

#### LDX

Sadržaj memorijske lokacije adresirane sadržajem ukazatelja steka prebacuje se u akumulator.

#### LDXA

Sadržaj memorijske lokacije adresirane sadržajem ukazatelja steka prebacuje se u akumulator, a ukazatelj steka se uveća za jedan.

#### STXD

Sadržaj akumulatora prebacuje se u memorijsku lokaciju adresiranu sadržajem ukazatelja steka, a ukazatelj steka se smanji za jedan.

### Naredbe za logičke operacije

Kod logičkih operacija jedan operand se nalazi u akumulatoru, a drugi se eksplicitno navodi uz kod operacije ili se nalazi u memorijskoj lokaciji koja je adresirana sadržajem ukazatelja steka. Rezultat se smešta u akumulator.

OR ORI parametar	Logička "ili" operacija
XOR XRI parametar	Operacija "ekskluzivno ili"
AND ANI parametar	Logička "i" operacija
SHR SHRC	Sadržaj akumulatora se pomera za jedno mesto udesno (bez ili uz pomoć BIT-a za prenos)
SHL SHLC	Sadržaj akumulatora se pomera za jedno mesto ulevo (bez ili uz pomoć BIT-a za prenos)

### Naredbe za aritmetičke operacije

Kao i kod logičkih operacija tako i ovde jedan operand stoji u akumulatoru, a drugi je kao parametar naveden uz kod operacije ili se nalazi u memorijskoj lokaciji na koju ukazuje ukazatelj steka.

ADD ADI parametar ADC ADCI parametar	sabiranje (bez i sa prenosom)
SD SDI parametar SDB SDBI parametar	oduzimanje od akumulatora (bez i sa pozajmljivanjem)

SM  
SMI parametar  
SMB  
SMBI parametar

oduzimanje od memorije  
(bez i sa pozajmljivanjem)

### Upravljačke naredbe

Omogućuju upravljanje prekidom, grananje i povezivanje potprograma i postavljanje stanja kontrolnog bita Q.

SEP Rn ili parametar

Registar n postavlja se kao programski brojač.

SEX Rn ili parametar

Registar n postavlja se kao ukazatelj memorijske lokacije  $M(R(x))$ .

IDLE

Čeka se DMA ili prekid.

SEQ

Postavlja se Q flip-flop.

REQ

Resetuje se Q flip-flop

SAV

Sadržaj registra T upisuje se u memorijsku lokaciju adresiranu sa  $R(X)$ .

MARK

Sadržaj registra X i P upisuje se u registar T i u stek.

RET

Memorijski byte  $M(R(X))$  prenosi se u registre X i P. Flip-flop za dozvolu prekida IE se resetuje.

DIS

Memorijski byte  $M(R(x))$  prenosi se u registre X i P. Flip-flop za dozvolu prekida IE se setuje.

NOP

Naredba bez dejstva.

### Specijalne naredbe

CALL parametar

Ovom naredbom poziva se potprogram na adresi jednakoj vrednosti parametra.

RETS

Naredba za povratak iz potprograma u glavni program.

BYTE parametar 1, parametar 2, ..., parametar n

Kada ASEMBLER naiđe na ovu naredbu, onda vrednosti navedenih parametara ubacuje u memorijske lokacije koje su u programu. Pri tome broj parametara nije ograničen

WORD parametar 1, parametar 2, ..., parametar n

Ova naredba je slična naredbi BYTE, s tom razlikom što parametri predstavljaju dvobytnne vrednosti koje se pakuju unutar programa.

TEXT "niz znakova"

Ova naredba služi za kreiranje tabela. Prilikom asembliranja u memoriji se pakuju ASCII kodovi znakova navedenih u nizu.

TEXTO "niz znakova"

Prilikom asembliranja na kraju niza postavi se znak END OF FILE (kod 00). Ova naredba se koristi sa rutinom (programom) za štampanje ASCII nizova.

### Pseudo naredbe ASEMBLER-a

#### EQU

To je specijalna naredba koja služi za definisanje vrednosti labela. Format naredbe je: Labela: EQU parametar.

Ova naredba koristi se na početku programa.

Npr.

```
A9000 : EQU &4800
```

Ograničenje za ime labela je da ima najviše šest znakova, s tim što prvi znak mora da bude alfa znak.

#### ORG

Ova naredba obavezno dolazi na početak svakog programa. Ovom naredbom definiše se adresa memorijske lokacije na kojoj se asemblira program. Format naredbe je sledeći:

```
ORG parametar 1, (parametar 2)
```

gde je parametar 1 adresa na koju se asemblira, a parametar 2 relocirana adresa na koju može ASEMBLER da asemblira program.

Npr.

```
ORG &80BF, &5000
```

Znači da se program koji sledi asemblira na adresu &80BF. Adresa &80BF kod PECOM-a 64 je u stvari adresa EPROM-a. Na ovaj način korisnik može da pripremi sadržaj za eventualnu promenu sadržaja ERPOM-a.

Program se istovremeno smešta počev od adrese &5000 u korisničkoj oblasti (RAM-u).

#### OPT n

Ovom naredbom bira se jedan od načina rada ASEMBLER-a. Zato je to obavezno prva naredba programa. Broj n može da uzima vrednost od 0 do 15. Predstavimo ga u binarnom kodu: n - (D3) (D2) (D1) (D0). Način rada zavisi od vrednosti pojedinih bitova:

D0 = 1 prevedeni kod se upisuje u memoriju

D1 = 1 listing datoteke štampa se na ekranu

D2 = 1 na kraju listinga datoteke napravi se tabela upotrebljenih labela

D3 = 1 listing datoteke štampa se na štampaču.

Npr.

Ako se upotrebi naredba OPT 1 nakon poziva ASEMBLER-a, prevedeni kod će se upisati u memoriju, a ukoliko ima grešaka u programu one će se pojaviti na ekranu. Ako se koristi naredba OPT 15, prevedeni kod se upisuje u memoriju, listing datoteke prikazuje se na ekranu sa tabelom upotrebljenih labela, a listing datoteke štampa se na štampaču.

PRILOG:

NAREDBE MONITOR-a+

D	štampanje sadržaja memorijskih lokacija u heksadecimalnom kodu na ekranu ili štampaču.
I	upisivanje sadržaja u memorijske lokacije
E	pozivanje CREDIT EDITOR-a
S	naredba za pretraživanje
F	memorijski blok puni se vrednošću datom u heksadecimalnom kodu
M	pomeranje memorijskog bloka u memoriji
W	memorijski blok se snima na kasetu
CTRL R	učitavanje datoteke sa kasete u memoriju
C	startovanje programa u mašinskom kodu
I	štampanje na štampaču
O	prikazivanje na ekranu monitora ili televizora
B	skok u BASIC
K	konverzija teksta u ĆIRILICU
L	prebacivanje teksta iz ĆIRILICE u LATINICU

## PRILOG:

### NAREDBE EDITOR-a

CTRL ← , → , ↓ , ↑	kretanje kursora u određenom smeru
ESC	strana nazad
LINE FEED	strana napred
CTRL E	kraj datoteke
CTRL S	početak datoteke
CTRL V	režim umetanja (izlazi sa strelicama ili bilo kojom drugom kontrolnom funkcijom)
CTRL Y	postavljanje markera bloka
CTRL C	zamena pozicija kursora i markera bloka
CTRL R	rolovanje markera
CTRL U	ubacivanje sadržaja od markera do kursora u bafer (maksimalna dužina je 1KByte)
CTRL I	izbacivanje sadržaja bafera na poziciju kursora
CTRL P	pronalaženje sadržaja bafera od kursora do kraja datoteke
CTRL O	kao CTRL P (ne pravi razliku između malih i velikih slova)
CTRL L	skok na početak sledeće reči
CTRL K	skok na početak prethodne reči
CTRL blanko	skok kursora na kraj reči
CTRL T	skok kursora do najbliže tabulacione tačke
CTRL RETURN	CR + tabulacija
CTRL Q	postavljanje početka strane na tekuću liniju
CTRL SHIFT DEL	brisanje bloka
CTRL A	ubacivanje jednog blanko znaka na poziciju kursora
DEL	brisanje znaka levo od kursora
PRINT	blok definisan markerom i kursorom štampa se na štampaču
BREAK	prelazak u komandni način rada (horizontalna crta nam pokazuje da se nalazimo u komandnom načinu rada)

PRILOG:

**NAREDBE KOMANDNOG NAČINA RADA EDITOR-a**

<b>TAB</b>	setovanje i resetovanje tabulacionih tačaka
<b>ASS</b>	asembliranje izvorne datoteke
<b>PRINT</b>	štampanje bloka na štampaču
<b>SAVE IME</b>	snimanje datoteke
<b>LOAD (IME)</b>	učitavanje datoteke sa kasete
<b>CTRL C</b>	izlazak iz EDITOR-a u MONITOR+
<b>BSAVE</b>	snimanje bloka
<b>LINE FEED</b>	povratak iz komandnog načina rada u tekst

**GREŠKE KOMANDNOG NAČINA RADA EDITOR-a**

<b>1</b>	nepostojeća funkcija
<b>2</b>	neppravilna upotreba funkcija
<b>3</b>	kursor se nalazi ispred markera bloka
<b>5</b>	greška prilikom učitavanja

NAREDBE ASEMBLER-a

NAREDBE ZA RAD SA REGISTRIMA

Opis simbola i njihova upotreba:

R - registri  
n,m - redni broj registra (n,m(-/0,1,...,F))  
q,p - oznaka težine bajta (q,p = L, niži; q,p = H, viši)  
par - parametar

LD Rn, Rm  
LD Rn.q, Rm.p  
LD Rn.q, A  
LD Rn.q, parametar  
LD RN, parametar

Punjenje registra n

LD A parametar  
LD A, Rn.q

Punjenje akumulatora

PUSH Rn  
PUSH Rn.q  
PUSH A

Smeštanje sadržaja u stek

POP Rn  
POP Rn.q  
POP A

Obnavljanje sadržaja iz steka

GHI Rn ili parametar  
GLO Rn ili parametar

Smeštanje sadržaja u akumulator

PHI Rn ili parametar  
PLO Rn ili parametar

Smeštanje sadržaja akumulatora u registre

INC Rn ili parametar -

Povećanje sadržaja registra n za 1

DEC Rn ili parametar -

Smanjenje sadržaja registra n za 1

LDI parametar -

Punjenje akumulatora vrednošću parametra



**PRILOG:****NAREDBE KRÁTKOG GRANANJA**

BR	parametar	-	Bezuslovni skok
BZ	parametar	-	Skok ako je $D = 0$
BNZ	parametar	-	Skok ako je $D \neq 0$
BDF	parametar	-	Skok ako je $DF = 1$
BNF	parametar	-	Skok ako je $DF = 0$
BQ	parametar	-	Skok ako je $Q = 1$
BNQ	parametar	-	Skok ako je $Q = 0$
B1	parametar	-	Skok ako je $EF1 = 1$
BN1	parametar	-	Skok ako je $EF1 = 0$
B2	parametar	-	Skok ako je $EF2 = 1$
BN2	parametar	-	Skok ako je $EF2 = 0$
B3	parametar	-	Skok ako je $EF3 = 1$
BN3	parametar	-	Skok ako je $EF3 = 0$
B4	parametar	-	Skok ako je $EF4 = 1$
BN4	parametar	-	Skok ako je $EF4 = 0$

**NAREDBE DUGOG GRANANJA**

LBR	parametar	-	Bezuslovni skok
LBZ	parametar	-	Skok ako je $D = 0$
LBNZ	parametar	-	Skok ako je $D \neq 0$
LBDF	parametar	-	Skok ako je $DF = 1$
LBNF	parametar	-	Skok ako je $DF = 0$
LBQ	parametar	-	Skok ako je $Q = 1$
LBNQ	parametar	-	Skok ako je $Q = 0$

**NAREDBE PRESKOKA**

SKP	-	Kratak preskok
LSKP	-	Dugi preskok
LSZ	-	Dugi preskok ako je $D = 0$
LSNZ	-	Dugi preskok ako je $D \neq 0$
LSDF	-	Dugi preskok ako je $DF = 1$
LSNF	-	Dugi preskok ako je $DF = 0$
LSQ	-	Dugi preskok ako je $Q = 1$
LSNQ	-	Dugi preskok ako je $Q = 0$
LSIE	-	Dugi preskok ako je $IE = 1$

## PRILOG:

### NAREDBE ZA RAD SA MEMORIJOM

LDN Rn LDN par	Sadržaj registra n prenosi se u akumulator
LDA Rn LDA par	Sadržaj registra n prenosi se u akumulator i sadržaj registra n uvećava se za 1
STR Rn STR par	Sadržaj akumulatora se prebacuje u memorijsku lokaciju adresiranu sadržajem registra n
LDX -	Sadržaj memorijske lokacije adresirane sadržajem steka pre- bacuje se u akumulator
LDX A -	isto kao LDX, samo što se ukazatelj steka uvećava za 1
STXD -	isto kao LDX, samo što se ukazatelj steka smanjuje za 1

### NAREDBE ZA LOGIČKE OPERACIJE

OR ORI par	Logička "ili" operacija
XOR XOR par	Operacija "ekskluzivno ili"
AND ANI par	Logička "i" operacija
SHR SHRC	Pomeranje sadržaja akumulatora udesno (bez i sa prenosom)
SHL SHLC	Pomeranje sadržaja akumulatora ulevo (bez i sa prenosom)

### NAREDBE ZA ARITMETIČKE OPERACIJE

ADD ADI par ADC ADCI par	Sabiranje (bez i sa prenosom)
SD SDI par SDB SDBI par	Oduzimanje od akumulatora (bez i sa pozajmljivanjem)

## PRILOG:

SM  
SMI par  
SMB  
SMBI par

Oduzimanje od memorije (bez i sa pozajmljivanjem)

### UPRAVLJAČKE NAREDBE

SEP Rn  
SEP par

Postavljanje programskog brojača

SEX Rn  
SEX par

Postavljanje ukazatelja steka

IDLE —

čeka se DMA ili prekid

SEQ —

Q = 1

REQ —

Q = 0

SAV —

Sadržaj registra T upisuje se u memorijsku lokaciju adresiranu sa R(X)

MARK —

Sadržaj registra X i P upisuje se u registar T i u stek.

RET —

$M(R(X)) \rightarrow (X, P) \quad 1 \rightarrow IE$

DIS —

$M(R(X)) \rightarrow (X, P) \quad 0 \rightarrow IE$

NOP —

Bez dejstva

### SPECIJALNE NAREDBE

CALL par

— Poziv potprograma

RETS

— Povratak iz potprograma u glavni program

BYTE par 1, par 2,  
..., par n

— Upis jednobajtnih parametara u memoriju

WORD par 1, par 2,  
..., par n

— Upis dvobajtnih parametara u memoriju

TEXT "niz znakova"

— Kreiranje tabela

TEXTO "niz znakova"

— Štampanje ASCII nizova

## PRILOG:

### PSEUDO NAREDBE ASEMBLER-a

- EQU par — Definisanje vrednosti labela
- ERG par 1, (par 2) — Startna adresa (relocirana adresa)
- OPT n — Opcija rada ASEMBLER-a

Postavljanjem bitova : n = (D3) (D2) (D1) (D0), biramo jednu od šesnaest opcija

- D0 = 1 prevedeni kod upisuje se u memoriju  
D1 = 1 prikazuje se listing datoteke na ekranu  
D2 = 1 na kraju listinga datoteke daje se tabela upotrebljenih labela  
D3 = 1 listing datoteke štampa se na štampaču

### LABELA

Koriste se labela definisane sa naredbom EQU i labela iz programa

Primer:

```
MIKA : EQU &0D55
BROJAČ: EQU 7
      :
      :
PERA  LD A,1
      :
      :
      LDN BROJAČ      .. vrednost iz memorijske lokacije čija
                    .. adresa je u R7 se smešta u akumulator
      :
      :
      BR PERA        .. bezuslovni skok
      :
      :
      ADI MIKA.H     .. vrednost akumulatora sabere se sa
                    .. bajtom veće težine labela MIKA (&0D)
      :
      :
      INC BROJAČ     .. sadržaj registra 7 uvećava se za 1
```

MATEMATIČKI IZRAZ

Opšti oblik izraza je:

$$\text{izraz} = \text{op1} \oplus \text{op2} \oplus \dots \oplus \text{opn}$$

gde opl može biti labela ili konstanta, a znak  $\oplus$  je jedan od operatora +, -, AND, OR i XOR.

Primeri :

a) BR - 123 + &4AD AND &10011

b) CALL \$ + "A"

.. poziva se potprogram sa lokacije jednake zbiru tekuće linije i 65

c) LD R5, TAB OR -44

d) LD A, PERA XOR 44

e) LD RF.H, TAB.L

viši bajt registra RF puni se nižim delom adrese TAB.

f) LD RC.L, TAB.H

.. niži bajt registra RF puni se višim delom adrese TAB.

GREŠKE PRI ASEMBLIRANJU

- |    |                                    |
|----|------------------------------------|
| 11 | nepostojeći mnemonik               |
| 12 | dve labela sa istim imenom         |
| 13 | nepravilna upotreba operanda       |
| 14 | nedozvoljen završetak rada         |
| 15 | nepostojeći registar               |
| 16 | ] Neppravilna upotreba mnemonika   |
| 17 |                                    |
| 18 |                                    |
| 22 | labela i BR nisu na istoj stranici |
| 99 | greška u matematičkom izrazu       |

ASCII TABELA ZA UGRADENE ZNAKE I SIMBOLE

Decimalni kod	Heksa kod	Znak i simbol	Pritisnuta dirka	Decimalni kod	Heksa kod	Znak i simbol	Pritisnuta dirka
0	00		NUL	64	40	Ⓐ	
1	01	CA	CTRL A	65	41	A	
2	02	□	CTRL B	66	42	B	В
3	03	Cc	CTRL C	67	43	C	С
4	04			68	44	D	Д
5	05	┘	CTRL E	69	45	E	
6	06	—	CTRL F	70	46	F	Ф
7	07		CTRL G	71	47	G	Г
8	08		CTRL H	72	48	H	Х
9	09	—	CTRL I	73	49	I	И
10	0A		LF	74	4A	J	
11	0B	⌘	CTRL K	75	4B	K	
12	0C		CTRL L	76	4C	L	Л
13	0D		CR	77	4D	M	
14	0E	=	CTRL N	78	4E	N	Н
15	0F		CTRL O	79	4F	O	
16	10	Π	CTRL P	80	50	P	П
17	11	┘	CTRL Q	81	51	Q	Љ
18	12	/	CTRL R	82	52	R	Р
19	13	Cs	CTRL S	83	53	S	С
20	14	┘	CTRL T	84	54	T	
21	15	Σ	CTRL U	85	55	U	У
22	16		CTRL V	86	56	V	В
23	17	↘	CTRL W	87	57	W	Ъ
24	18	⌘*	CTRL X	88	58	X	
25	19	┘*	CTRL Y	89	59	Y	Ц
26	1A	ž*	CTRL Z	90	5A	Z	З
27	1B	Es	CTRL	91	5B	↓	Ш
28	1C	←	CTRL	92	5C	←	Ъ
29	1D	→	CTRL	93	5D	→	Ѓ
30	1E	↑	CTRL	94	5E	↑	Ч
31	1F			95	5F		
32	20		BLANKO	96	60		
33	21	!	SHIFT 1	97	61	a	CAPS A
34	22	"	SHIFT 2	98	62	b	Б CAPS B
35	23	#	SHIFT 3	99	63	c	С CAPS C
36	24	\$	SHIFT 4	100	64	d	Д CAPS D
37	25	%	SHIFT 5	101	65	e	Е CAPS E
38	26	&	SHIFT 6	102	66	f	Ф CAPS F
39	27	'	SHIFT 7	103	67	g	Г CAPS G
40	28	(	SHIFT 8	104	68	h	Х CAPS H
41	29	)	SHIFT 9	105	69	i	
42	2A	*	SHIFT :	106	6A	j	
43	2B	+	SHIFT ;	107	6B	k	К
44	2C	,		108	6C	l	Л
45	2D	-	SHIFT =	109	6D	m	М
46	2E	.		110	6E	n	Н
47	2F	/		111	6F	o	
48	30	0		112	70	p	П
49	31	1		113	71	q	Љ
50	32	2		114	72	r	Р

Decimalni kod	Heksa kod	Znak i simbol	Pritisnuta dirka	Decimalni kod	Heksa kod	Znak i simbol	Pritisnuta dirka
51	33	3		115	73	s c	
52	34	4		116	74	t T	
53	35	5		117	75	u y	
54	36	6		118	76	v B	
55	37	7		119	77	w Ё	
56	38	8		120	78	x	
57	39	9		121	79	y u	
58	3A	:		122	7A	z	
59	3B	;		123	7B	š m	SHIFT ↓
60	3C	<	SHIFT,	124	7C	d b	SHIFT ←
61	3D	=		125	7D	č h	SHIFT →
62	3E	>	SHIFT.	126	7E	č 4	SHIFT ↑
63	3F	?	SHIFT/	127	7F		

## PRIMERI PROGRAMIRANJA

Primer 1

Program za smeštanje konstante &ABCD u registar RF. Sadržaj registra RF preneti u memorijske lokacije sa adresama 5000 i 5001.

Pošto se program unosi u režimu rada EDITOR-a na PECOM-u 64 potrebno je prethodno ući u taj režim rada.

```
READY
:PROB
>PD000
>E
```

Sada može da otpočne unošenje programa.

```
OPT 7
ORG &4000
START:
LD RF, &ABCD      ..RF = ABCD
LD R8, &5000      ..R8 = 5000
GHI RF            viši bajt RF
STR R8            ..se prenosi u M(5000)
INC R8
GLO RF            ..niži bajt RF
STR R8            ..se prenosi u M(5001)
RETS
```

Zatim je potrebno program asemblirati.

To ćemo učiniti tako što prvo pređemo u komandni način rada EDITOR-a pritiskom na taster BREAK.

Kao znak da smo prešli u komandni način rada EDITOR-a javlja se horizontalna crta u gornjem delu ekrana sa trepćućim kursorom.

Naredbom ASS vrši se pozivanje ASEMBLER-a.

Program koji smo uneli sada će izgledati ovako:

```
ASS
0000                                OPT 7
4000                                ORG &4000
4000 F8ABBFF8CDAF                   LD RF, &ABCD
4006 F850B8F800A8                   LD R8, &5000
400C 9F                              GHI RF
400D 58                              STR R8
400E 18                              INC R8
400F 8F                              GLO RF
4010 58                              STR R8
4011 D5                              RETS
4012

START                                4000
```

Za izvršenje programa potrebno je da se vratimo u režim rada MONITOR-a+, a to ćemo izvesti pritiskom na tastere CTRL i C.



## PRILOG:

Pošto smo sada u režimu rada MONITOR-a+ važe sledeće naredbe:

Za listanje programa na mašinskom jeziku koristimo naredbu D. Listanje se vrši sve dok se drži pritisnut taster CTRL. Prekidanje prikaza obavlja se pritiskom na dirku BREAK.

Za startovanje programa u mašinskom kodu koristi se naredba C. Na pitanje FROM unosi se početna adresa programa.

```
>C
FROM 4000
>
Za proveru pravilnosti rada programa ponovo koristimo naredbu D.
>D
FROM 5000
5000 AB CD 00 00 00 00 00 00
```

Program koji se unosi u ASSEMBLER-u smešta se počev od adrese &0300. To treba imati na umu pri izboru početne izvršene adrese programa.

Proverom naredbom D dobija se:

```
>D
FROM 300
0300 4F 50 54 20 37 0D 4F 52
0308 47 20 26 34 30 30 30 0D
0310
0318
0320
0328
0330
0338
0340
0348
0350 38 0D 52 45 54 53 0D 00
0358
```

### Primer 2

Program za spašavanje sadržaja registra.

Da bi ste videli sadržaj pojedinih registra mikroprocesora CDP 1802 u različitim režimima rada PECOM-a 64, napravite program koji će vam to omogućiti.

Program neka počne na adresi &6000 a sadržaj registra se smešta počev od adrese &7000.

U navedenom primeru smešteni su sadržaji registra R8, R9, RA, RB, RD, RE i RF. Registar RC služi kao ukazatelj memorijskog prostora u RAM-u gde se vrši smeštanje.

Pokušajte za vezbu da spasite sadržaje svih registara uz korišćenje registra RC i naredbe STR. Program će biti daleko efikasniji.

```
0000 OPT 7
6000 ORG &6000
6000 F870BCF800AC LD RC, &7000
6006 98 CHI R8
6007 5C STR RC
6008 1C INC RC
```

## PRILOG:

6009 88	GLO R8
600A 5C	STR RC
600B 1C	INC RC
600C 99	GHI R9
600D 5C	STR RC
600E 1C	INC RC
600F 89	GLO R9
6010 5C	STR RC
6011 1C	INC RC
6012 9A	GHI RA
6013 5C	STR RC
6014 1C	INC RC
6015 8A	GLO RA
6016 5C	STR RC
6017 1C	INC RC
6018 9B	GHI RB
6019 5C	STR RC
601A 1C	INC RC
601B 8B	GLO RB
601C 5C	STR RC
601D 1C	INC RC
601E 9C	GHI RC
601F 5C	STR RC
6020 1C	INC RC
6021 8C	GLO RC
6022 5C	STR RC
6023 1C	INC RC
6024 9D	GHI RD
6025 5C	STR RC
6026 1C	INC RC
6027 8D	GLO RD
6028 5C	STR RC
6029 1C	INC RC
602A 9E	GHI RE
602B 5C	STR RC
602C 1C	INC RC
602D 8E	GLO RE
602E 5C	STR RC
602F 1C	INC RC
6030 9F	GHI RF
6031 5C	STR RC
6032 1C	INC RC
6033 8F	GLO RF
6034 5C	STR RC
6035 1C	INC RC
6036 D5	SEP R5
6037	

### Primer 3

Program za inicijalizaciju sistema

Po uključanju PECOM-a 64, vrši se inicijalizacija sistema. Program za inicijalizaciju počinje od adrese &8000

## PRILOG:

Ponovna inicijalizacija može da se izvrši iz MONITOR-a ili MONITOR-a+ jednostavno bezuslovnim skokom na adresu &8002.

Tako na primer, od adrese &0300 možemo da napišemo bezuslovni skok (na mašinskom jeziku) koristeći naredbu I na sledeći način:

```
>I
FROM 0300
0300 C0 8002
>C
FROM 300
```

Po završetku programa na ekranu se pojavljuje poruka:

```
Ei PECOM 64
READY
:
```

## PRIMER 4

Mikroračunar PECOM 64 nakon inicijalizacije sistema radi sa karakterima koji se upisuju u 24 linije ekrana sa 40 karaktera u liniji.

Navodimo jedan primer koji će vam ilustrovati kako valičina karaktera može softverski da se menja.

```
0000          A8002:EQU &8002
0000          OPT 3
1400          ORG &1400
1400 F813BCF800AC LD RC, &1300
1406 F863      LDI &63
1408 A8        PLO R8
1409 5C        STR RC
140A E3        SEX R3
140B E36102    OUT 1, &02          ; SELEKCIJA VIS-a
140E          ADR:
140E 340E      B1 ADR1
1410 EC        SEX RC
1411 63        OUT 3              ; &63 SE ŠALJE U VIS
1412 F800BAF800AA LD RA, &0000
1418 EA        SEX RA
1419 65        OUT 5              ; &0000 SE ŠALJE U VIS
141A E3        SEX R3
141B E36100    OUT 1, 800         ; RAZSELEKCIJA VIS-a
141E F8FFB8F8FFA8 LD R8, &FFFF   ; POSTAVLJANJE VREMENSKE KONSTAN-
                                     NTE
1424          A1600:
1424 28        DEC R8
1425 98        GHI R8
1426 3A24      BNZ A1600
1428          A1700:
1428 28        DEC R8
1429 88        GLO R8
142A 3A28      BNZ A1700
142C C08002    LBR A8002         ; SKOK NA PROGRAM ZA INICIJALIZA-
                                     CIJU SISTEMA
142F

A8002          8002
ADR1           140E
A1600          1424
A1700          1428
```

**ELEKTRONSKA INDUSTRIJA – NIŠ**  
**E-RO „E-RACUNARI“**  
OOUR Fabrika računarskih mašina  
Bul. Veljka Vlahovića 80-82  
18000 Niš  
Direktor (018) 325-461  
Direktor marketinga (018) 55-583  
Prodavac (018) 54-779, 51-568  
Softver (018) 52-782, 52-876  
Školski centar (018) 54-090  
Servis (018) 54-867  
Tlx. 16283 YU E1 FRM

**POSLOVNA JEDINICA BEOGRAD**  
Rude 2 – 11000 Beograd  
Direktor (011) 488-260, 483-266

**POSLOVNA JEDINICA TITOGRAD**  
Ul. Brace Bracanovica 38  
81000 Titograd  
Tel. (081) 34-739, 34-812