

**RADNO MESTO ZA RAZVOJ I ISPITIVANJE
UPRAVLJAČKIH ALGORITAMA ZASNOVANO NA
MIKROKONTROLERU 8098**

- UPUTSTVO ZA KORISNIKE -

Antić Stevan, Beograd, oktobar, 1997.

1	UVOD -----	1
2	OSOBI NE SISTEMA I NJIHOVA HARDVERSKA REALIZACIJA -----	2
2.1	Processor -----	3
2.2	Serijska veza -----	3
2.3	PWM Izlazi-----	3
2.4	Povezivanje inkrementalnog enkodera sa HSI uređajem -----	4
2.5	Analogni ulazi-----	5
2.6	Analogni izlaz-----	6
2.7	Digitalni ulazi i izlazi-----	6
2.8	Svetleća signalizacija na kartici-----	7
2.9	Adresni prostor-----	7
3	PROGRAMSKA PODRŠKA -----	13
3.1	Prilagodni program -----	13
3.1.1	Startovanje prilagodnog programa serial.exe-----	13
3.1.2	Lista komandi i njihovo objašnjenje-----	14
3.2	Nadgledni program -----	16
3.2.1	Rezervisane memorijske lokacije-----	16
3.2.2	Pravljenje korisničkog programa u kombinaciji sa nadglednim programom -----	17
3.2.3	Pravljenje korisničkog programa bez mogućnosti rada nadglednog programa-----	18
3.2.4	Signalizacija nadglednog programa korisniku pomoću svetlećih dioda -----	18
4	PRIMER UPOTREBE MIKROPROCESORSKOG SISTEMA -----	19
4.1	Primer izvršavanja korisničkog programa u kombinaciji sa nadglednim programom-----	20
4.1.1	Prevođenje korisničkog programa i formiranje binarne datoteke koja se učitava u RAM kartice -----	20
4.1.2	Redosled postupaka korisnika pri radu sa laboratorijskom stanicom u cilju ispitivanja svog programa -----	20
4.2	Primer izvršavanja korisničkog programa bez nadglednog programa -----	26
4.2.1	Prevođenje korisničkog programa i formiranje binarne datoteke koja se učitava u RAM kartice -----	26
4.2.2	Redosled postupaka korisnika pri radu sa laboratorijskom stanicom u cilju ispitivanja svog programa -----	27
DODATAK		
	Skica kartice-----	29
	Raspored konektora -----	30
	Opis konektora -----	31
	Spisak delova ugrađenih u karticu-----	32
	Šeme mikroprocesorskog sistema-----	34

UVOD

Ovo je uputstvo za upotrebu mikroprocesorskog sistema za ispitivanje algoritama upravljanja energetskim pretvaračima i pogonima, i namenjeno je studentima koji pohađaju kurseve: Mikroprocesorsko upravljanje elektromotornim pogonima (TE5MUE) i Mikroprocesorsko upravljanje energetskim pretvaračima, a koji žele da se praktično upoznaju sa pretvaračima i pogonima kroz semestralni rad. Takođe je namenjeno i studentima diplomcima koji su zainteresovani za praktičnu realizaciju algoritama. U ovom uputstvu su izložene potrebne informacije za upotrebu sistema, kao i primer upotrebe.

Svrha ovog uputstva je da korisniku objasni kako može ispitivati različite algoritme upravljanja pretvaračima korišćenjem ovog sistema. Cilj ovog uputstva je da korisnika osposobi za samostalno korišćenje sistema.

Mikroprocesorski sistem se koristi u okviru laboratorijska stanice* za vektorsku kontrolu koja je smeštena u prostoriji 40a, vreme za rad sa laboratorijskom stanicom se određuje u dogovoru sa asistentom stažistom Antić Stevanom, termin kada prima studente je istaknut pored vrata laboratorije 40.

Delovi ovog uputstva su:

- prvo poglavlje je uvod;
- u drugom poglavlju se nalazi opis osobina sistema i njegova hardverska realizacija;
- u trećem poglavlju je dato objašnjenje i listing programa koji podržavaju rad sistema;
- četvrto poglavlje je primer korišćenja sistema: njegovo upuštanje u rad, kratak opis algoritma koji će se izvršavati, način njegovog upisa i kontrole izvršenja;
- na kraju uputstva su priložene šeme sistema, spisak ugrađenih komponenti, opis konektora i skica kartice.

* Uputstvo za laboratorijsku stanicu se nalazi na serveru Kiklop, datoteka: X:\nastava\leg4ev\uputstvo.pdf.

OSOBINE SISTEMA I NJIHOVA HARDVERSKA REALIZACIJA

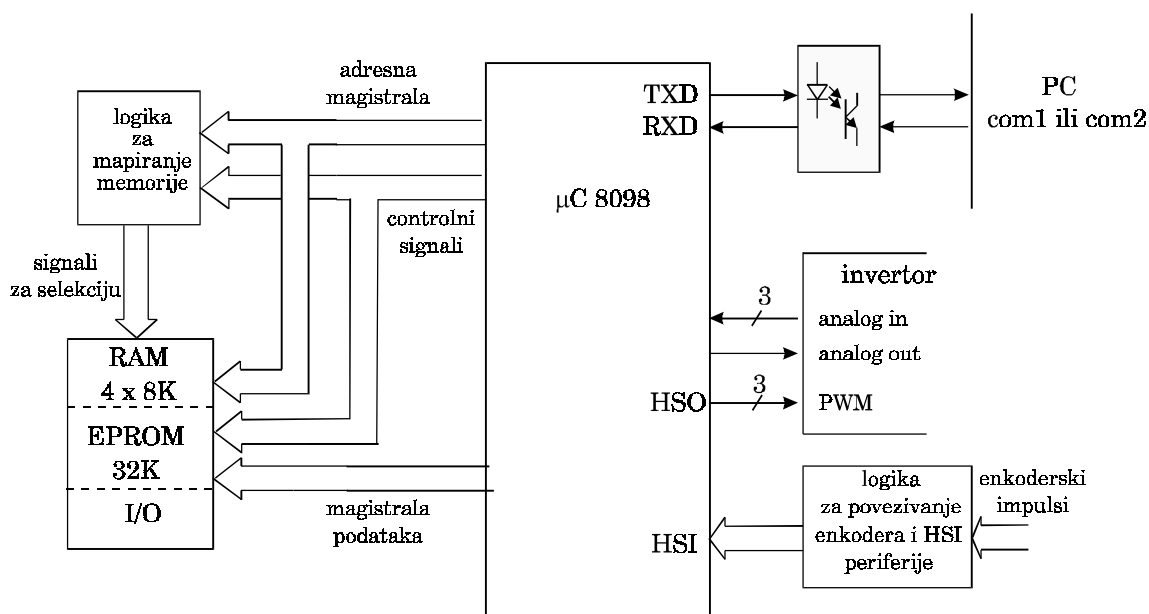
Ovo poglavlje opisuje osobine mikroračunarskog sistema zasnovanog na μC 8098.

Mikroračunarski sistem za ispitivanje algoritama upravljanja pretvaračima i elektromotornim pogonima omogućava upravljanje trofaznim inverterima i primenu savremenih metoda upravljanja motorima naizmenične struje, realizovan je na kartici formata 100x190 mm.

Njegove glavne osobine su:

- zasnovan na kontroleru *INTEL* 8098-12 MHz;
- optički izolovana serijska veza;
- 4xPWM izlaza formata +/- 15V (HSO);
- interfejs za inkrementalni optički enkoder prema HSI periferiji μC , dve faze plus ulaz za indeksni signal;
- tri analogna ulaza formata 0V do +5V;
- jedan analogni izlaz formata 0V do +5V;
- digitalni I/O;
- 64KB memorije, 32KB EPROM plus 32KB RAM memorije;
- promenljiv adresni prostor, u smislu zamene mesta EPROM, RAM* .

Na slici 1 je prikazan blok dijagram sistema koji ilustruje njegove glavne osobine.



Slika 1. Blok dijagram mikroračunarskog sistema.

*Obično je programski kod algoritma koji se izvršava smešten u EPROM memoriji, kako je njegov sadržaj nepromenljiv onda je svaka izmena algoritma moguća jedino promenom EPROM-a (fizičkom promenom čipa). Da bi se izbegla fizička promena EPROM-a radi promene ispitivanog algoritma, algoritam se prvo upisuje u RAM memoriju posle čega zadavanjem odgovarajuće programske naredbe EPROM i RAM razamene mesta u memorijskom adresnom prostoru kartice, efekat je isti kao da je EPROM promenjen. Detaljnije objašnjenje mehanizma promene adresnog prostora kartice je u poglavlju 2.9.

2.1 PROCESOR

Jezgro mikroračunarskog sistema predstavlja Intelov 16-to bitni mikrokontroler 8098 12MHz, on pripada familiji MCS-96 mikrokontrolera. Njegov adresni prostor je veličine 64KB, poseduje adresnu magistralu širine 16 bita i magistralu podataka širine 8 bita. Interno sadrži 256 8-bitnih registara, od kojih su većina akumulatori dok je ostatak konfiguracionog, statusnog ili baferskog tipa. Takođe sadrži više ugrađenih periferija (HSI, HSO, tajmere, serijski port, ...). Uputstvo za mikrokontroler se može naći u knjizi "*Intel Microcontroller Handbook, 1986*" kao i u brojnoj literaturi koju studenti mogu dobiti ako se obrate Stevanu Antiću ili sa Intelovog servera na Internetu čija je adresa: <http://www.intel.com>

2.2 SERIJSKA VEZA

Serijska veza između PC računara i mikrokontrolera 8098 je galvanski izolovana pomoću optokaplera TLP550 (na šemama komponente U13 i U14). Optički izolovana veza je potrebna da bi zaštitila PC računar od pojave prenapona na invertoru.

Standard za serijsku komunikaciju RS232 je naponski standard kome logičkoj 0 odgovara +12V a logičkoj 1 odgovara -12V. Naponski nivo na kome se detektuje promena logičke vrednosti signala je +2V. Ukoliko je naponski signal na RX ulazu serijskog porta PC računara veći od +2V onda je njegova logička vrednost 0, a ako je naponski signal manji od +2V onda je njegova logička vrednost 1. Za kraća rastojanja između PC računara i kartice (par metara) nije potrebno poštovati RS232 standard već je dovoljno obezbediti da izlazni signal sa kartice ka RX pinu serijskog porta PC-ja bude veći ili manji od +2V da bi se detektovala njegova logička vrednost. Serijska veza se sa strane PC računara napaja sa +12V preko DTR pina serijskog porta PC-ja (pin 4 devetopinskog konektora na šemama označenog sa CON2 DB9).

Da bi se obezbedilo napajanje serijske veze, pre početka komunikacije potrebno je upisati 1 instrukcijom OUT u lokaciju DTR serijskog porta da bi potencijal DTR pina postavio na +12V. To se može postići sledećim asemblerskim programom:

```
.model small
.stack
.code
start:
    MOV AX,1           ;Upisuje se 1 u DTR (+12_V)
    MOV DX,03FCH      ;Adresa DTR je bit 0 lokacije #3FCH
                     ;tzv. modem control register

    OUT DX,AX
    MOV AH,4CH        ;Priprema
    MOV AL,00H        ;za povratak u DOS
    INT 21H           ;Povratak u DOS
    END start         ;Kraj
```

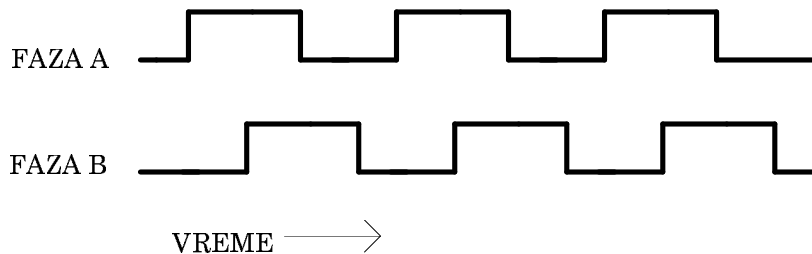
Serijska veza je asinhrona, odvija se preko TXD i RXD pinova serijskog porta sa maksimalnom brzinom prenosa podataka od 19200 b/s.

2.3 PWM IZLAZI

Na kartici se nalazi ukupno 4 PWM izlaza formata +/-15V. Tri izlaza se nalaze na glavnom konektoru (na šemama je označen sa CON1, pinovi 18, 20 i 22) i predstavljaju naponski uobličene signale sa **HSO.0**, **HSO.1** i **HSO.2** (*high speed output*) izlaza mikrokontrolera. Četvrti PWM izlaz je prisutan na konektoru CON4 pin 17, i predstavlja naponski uobličeni signal sa **HSO.3** izlaza mikrokontrolera.

2.4 POVEZIVANJE INKREMENTALNOG ENKODERA SA HSI UREĐAJEM

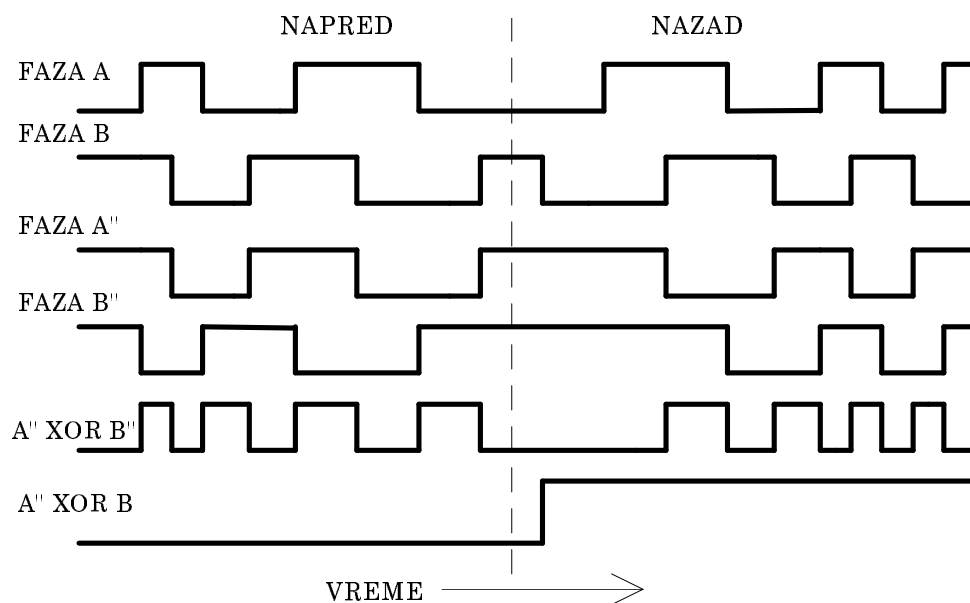
Kartica je povezana preko 9-to pinskog konektora DB-9 CON2 (pogledaj šeme) sa optičkim enkoderom. Optički enkoder dobija napajanje sa kartice +5 V. Idealizovan izgled signala faza A i B koji dolaze sa optičkog enkodera pre prolaska kroz RC filter i komponente U16A,B,C (Šmit inverter) a kada se motor obrće u istom smeru je prikazan na slici 2.



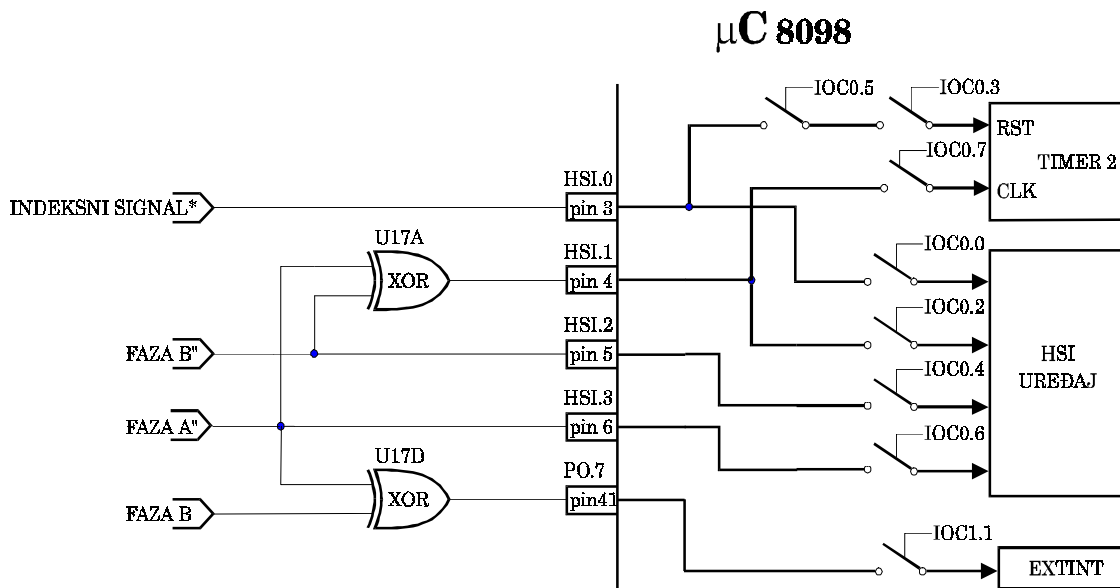
Slika 2. Signali sa optičkog enkodera.

Signali faza A i B se ne dovode direktno na tajmerske ulaze HSI uređaja mikrokontrolera jer kada je motor zaustavljen i osovina podrhtava može doći do pojave dve uzastopne ivice signala jedne faze bez pojave uzlaznih ili silaznih ivica signala druge faze i time pogrešnog detektovanja događaja od strane HSI uređaja. Da bi se ovo sprečilo signali faza A i B se filtriraju.

Na slici 3 su prikazani signali sa optičkog enkodera faza A i B pre i posle promene smera obrtanja motora. Posle logičkog filtriranja signala A i B dobijaju se signali A'' i B'' koji se dovode na ulaz u mikrokontroler. Ideja filtriranja je da uzlazne ili silazne ivice signala jedne faze određuje trenutak detekcije nivoa signala druge faze. Tako pojava ivice signala faze A (uzlazna ili silazna) određuje trenutak upisa vrednosti signala faze B u D flip flop (komponenta U19b), a pojava ivice signala B određuje trenutak upisa vrednosti signala faze A u drugi D flip flop (komponenta U19a).



Slika 3. Logičko filtriranje signala faza A i B.



Slika 4. Povezivanje optičkog enkodera sa HSI uređajem mikrokontrolera 8098.

Na slici 4 je prikazan način povezivanja optičkog enkodera sa HSI uređajem mikrokontrolera 8098. Signal faze A'' se dovodi na HSI uređaj preko tajmerskog ulaza HSI.3. Signal faze B'' se dovodi na HSI uređaj preko tajmerskog ulaza HSI.2. XOR signala faza A'' i B'' predstavlja digitalni signal koji svoju vrednost menja pri svakoj ivici faza A'' i B''. On se dovodi na CLK ulaz tajmera 2 preko ulaza HSI.1 koji broji ivice (uzlazne i silazne) obe faze. Indeksni signal se dovodi na RST ulaz tajmera 2. Na ovaj način se čitanjem sadržaja registra tajmera 2 može utvrditi apsolutni položaj osovine motora⁺.

Konstantnim vremenskim intervalom čitanja registra podataka uređaja TIMER 2 je moguće odrediti brzinu obrtanja enkoderskog diska. Ovakvo merenje brzine se primenjuje kada se FIFO bafer HSI uređaja puni brže od raspoložive brzine čitanja podataka iz njega.

XOR signala faza B' i A'' daje informaciju o smeru obrtanja motora. Ovaj signal se dovodi na ulaz P0.7 (pin 41 µC) odakle može da se pročita ili da posluži kao spoljni izvor prekida EXTINT.

2.5 ANALOGNI ULAZI

Na kartici postoje tri analogna ulaza, format ulaznog signala je od 0V do +5V. Ulazi se preko pasivnog filtra dovode na 10-to bitni A/D (analogno-digitalni) konvertor mikrokontrolera na kanale ACH4, ACH5 i ACH6 i to: preko konektora CON1, pinovi 24, 25 i 26 i preko konektora CON4, pinovi 15 i 16. Signal na ACH6 ulaz, se može dovesti sa trimmer potencijometra na šemama označenog sa C-POT. Izbor sa kog konektora će se dovesti signal na A/D konvertor se vrši uklanjanjem (tj. postavljanjem) odgovarajućeg 100Ω otpornika. Na primer uklanjanje otpornika R-79 ili R-42 određuje da li će se signal na ulaz ACH4 dovesti sa pina 24 konektora CON1 ili pina 16 konektora CON4.

* Indeksni signal dolazi sa enkodera preko pina 2 konektora CON2.

⁺ Pre prvog prolaska indeksnog otvora enkoderskog diska preko fotodetektora apsolutni položaj osovine nije poznat.

2.6 ANALOGNI IZLAZ

Kartica ima jedan analogni izlaz formata od 0V do 5V. Signal analognog izlaza pretstavlja filtrirani signal PWM izlaza sa procesora (pin 13, port 2.5) i prisutan je na pinu 23 konektora CON1.

2.7 DIGITALNI ULAZI I IZLAZI

Kartica poseduje 8 digitalnih ulaza formata 0V/5V, logičkoj nuli odgovara 0V a logičkoj jedinici odgovara +5V (pogledaj sliku 5), digitalni ulaz je prisutan na konektoru za proširenje kartice CON4 pinovi 3, 4, ...,9, deo je memorijskog adresnog prostora kartice i pristupa mu se kao osmobitnoj memorijskoj lokaciji. Adresa ulaza je #0FFXXh*. Pinovima 3, 4, ...,10 konektora CON4 respektivno odgovara 0, 2, 1, 4, 3, 6, 5 i 7 bit.

Kartica ima 2 digitalna izlaza formata 0V/5V, logičkoj nuli odgovara 0V a logičkoj jedinici odgovara +5V (pogledaj sliku 5), digitalni izlaz je prisutan na konektoru za proširenje kartice CON4 pinovi 1 i 2, deo je memorijskog adresnog prostora kartice i pristupa mu se kao osmobitnoj memorijskoj lokaciji. Adresa ulaza je #0FFXXh. Pinovima 1 i 2 konektora CON4 respektivno odgovara 6 i 7 bit.

ADRESA #FFXXh

KADA SE PIŠE

bit 0	Upisivanjem 1 u ovaj bit gasi se dioda L0
bit 1	Upisivanjem 1 u ovaj bit gasi se dioda L1
bit 2	Upisivanjem 1 u ovaj bit gasi se dioda L2
bit 3	Upisivanjem 1 u ovaj bit gasi se dioda L3
bit 4	Upisivanjem 1 u ovaj bit gasi se dioda L4
bit 5	Komandni bit R/E# za promenu mape memorijskog prostora
bit 6	Digitalni izlaz na konektor za proširenje CON4 pin 2
bit 7	Digitalni izlaz na konektor za proširenje CON4 pin 1

RESET STANJE: #0000 0000b

(posle hardverskog reseta, pritiskom na rst taster ili posle uključenja)

KADA SE ČITA

bit 0	Digitalni ulaz sa konektora za proširenje pin 3
bit 1	Digitalni ulaz sa konektora za proširenje pin 5
bit 2	Digitalni ulaz sa konektora za proširenje pin 4
bit 3	Digitalni ulaz sa konektora za proširenje pin 7
bit 4	Digitalni ulaz sa konektora za proširenje pin 6
bit 5	Digitalni ulaz sa konektora za proširenje pin 9
bit 6	Digitalni ulaz sa konektora za proširenje pin 8
bit 7	Digitalni ulaz sa konektora za proširenje pin 10

RESET STANJE: zavisi od priključenog uređaja na konektor za proširenje

Slika 5. *Ulazno izlazna jedinica.*

* X znači da nema uticaja na adresu, tj. na mestu x može da stoji proizvoljan četvorobitni broj.

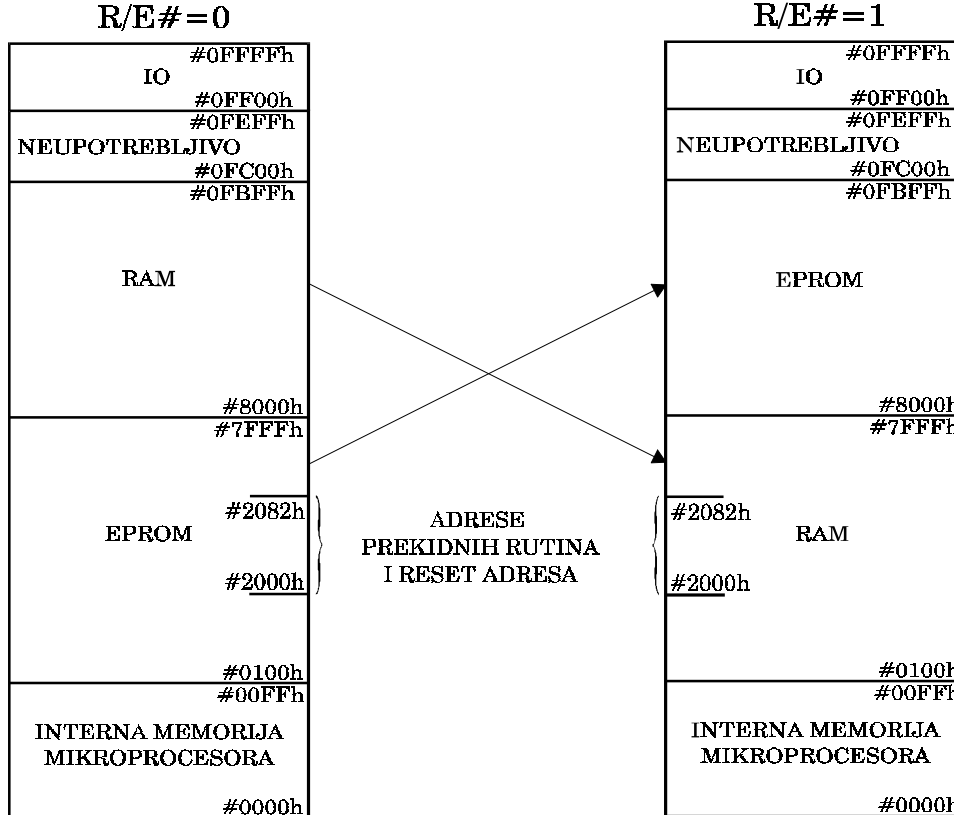
2.8 SVETLEĆA SIGNALIZACIJA NA KARTICI

Na kartici je ugrađeno 6 svetlećih LED dioda. Pet dioda svetle zelenom svetlošću i na šemama su obeležene oznakama L0, L1, L2, L3 i L4. Paljenje i gašenje dioda se obavlja upisivanjem 0 tj 1 u osmobitnu memorijsku lokaciju #0FFXXh u bitove 0, 1, ...,4 (pogledaj sliku 5), u reset stanju diode svetle. Šesta dioda svetli crvenom svetlošću, na šemama nosi oznaku L-Q. Njenim paljenjem i gašenjem se upravlja preko konektora za proširenje pinovi 11 i 12 koji dovode signale na CLK i D ulaz D flip-flopa, komponenta U16A. Reset stanje ove diode je nedefinisano.

2.9 ADRESNI PROSTOR

Na kartici se nalazi 32KB EPROM memorije i 32KB RAM memorije. Magistrala podataka mikrokontrolera je 8-bitna, adresna magistrala je 16-to bitna pa je adresni prostor kartice veličine 2^{16} bajtova ili 64 KB, sa adresama od #0h do #0FFFFh. Sve ulazne i izlazne jedinice su memorijski raspoređene (mapirane), tj obraćanje nekoj memorijskoj lokaciji je isto kao i obraćanje ulazno-izlaznoj jedinici. Deo adresa od #0h do #0FFh obuhvata registre u mikrokontroleru 8098 (njima se treba obraćati registarski direktnim adresiranjem, pogledaj uputstvo za procesor). Ostale adrese od #100h do #0FFFFh pripadaju memorijskim čipovima i ulaznim i izlaznim jedinicama, koji su fizički van mikrokontrolera 8098. Da bi mikrokontroler imao pristup celokupnoj spoljnoj memoriji pin 39 EA# je spojen na potencijal mase kartice VGND = 0V (pogledaj uputstvo za procesor).

Obično je programski kod algoritma koji se izvršava smešten u EPROM memoriji, kako je njegov sadržaj nepromenljiv onda je svaka izmena algoritma moguća jedino promenom EPROM-a (fizičkom promenom čipa). Da bi se izbegla fizička promena



Slika 6.: Mapa adreznog prostora kartice.

EPROM-a radi promene ispitivanog algoritma, algoritam se prvo upisuje u RAM memoriju posle čega upisom jedinice u konfiguracioni bit R/E# EPROM i RAM razamene mesta u memorijskom adresnom prostoru kartice (pogledaj sliku 6), efekat je isti kao da je EPROM promenjen. Bit R/E# (pogledaj sliku 5) je peti bit osmobicne memorijske lokacije na adresi #0FFXh, njegova vrednost ne može da se pročita. Posle hardverskog reseta (posle pritiska na reset prekidač ili posle uključenja napajanja) stanje konfiguracionog bita je R/E#=0. Programski reset (izvršenje rst instrukcije) nema uticaja na R/E# bit. Mapa memorijskog adresnog prostora u zavisnosti od stanja kontrolnog bita izgleda kao na slici 6.

EPROM je za R/E#=0 mapiran na bloku adresa od #100h do #7FFFh, njegov sadržaj na adresama od #0h do #0FFh je tada neupotrebljiv jer te adrese pripadaju registrima mikrokontrolera. RAM memorija je za R/E#=0 mapirana na bloku adresa od #8000h do #0FBFFh, deo memorijskog prostora sa adresama od #0FC00h do #0FEFFh je neupotrebljiv a deo od adrese #0FF00h do adrese #0FFFFh pripada ulazno izlaznom uređaju opisanom na slici 5.

RAM je za R/E#=1 mapiran na bloku adresa od #100h do #7FFFh, njegov sadržaj na adresama od #0h do #0FFh je tada neupotrebljiv jer te adrese pripadaju registrima mikrokontrolera. EPROM memorija je za R/E#=1 mapirana na bloku adresa od #8000h do #0FBFFh, deo memorijskog prostora sa adresama od #0FC00h do #0FEFFh je neupotrebljiv a deo od adrese #0FF00h do adrese #0FFFFh pripada ulazno izlaznom uređaju opisanom na slici 5.

Treba zapamtiti da stanje konfiguracionog R/E# bita nema uticaja na adrese od #0FC00h do #0FFFFh kao i na internu memoriju mikroprocesora.

Radi boljeg razumevanja mapiranje adresnog prostora priložen je program koji testira memoriju na kartici i pali zelene led diode na interesantan način.

```

;*****
;Program za testiranje memorije posle ukljucenja napajanja ili hardverskog reseta
;(pritiskom;na reset taster R/E#=0 zapocinje testiranje memorije tako sto prvo testira visih
;32k memorije izuzev lokacija koje su rezervisane za I/O ili neupotrebljive, zatim ukoliko
;je taj deo memorije ispravan kopira sadrzaj EPROMA (u kome se nalazi program za
;testiranje memorije) u sada iztestiran RAM tako da se sve lokacije iz EPROMA
;prebacuju sa ofsetom od 32k u RAM, zatim upisuje u jedinicu u konfiguracioni bit R/E# i
;resetuje karticu programskim resetom tj. rst instrukcijom posle cega se program za
;testiranje memorije izvrsava iz ram memorije i testira ostatak rama koji nije ;istestiran a
;koji je bio ;neupotrebljiv ili I/O pri R/E#=0, na kraju ukoliko je memorija OK skace na
;labelu memory_ok1 i ukljucuje svetlosne diode na karakteristican nacin. Ukoliko je u
;toku testiranja memorije otkrivena losa lokacija program skace na labelu
;display_bad_adres1 i pali svetlece diode po semi koja odgovara adresi lose lokacije tako
;da se moze vizuelno utvrditi adresa lose lokacije.
;*****
testmem module main ;Naziv rutine, atribut 'glavni'
#include(etfdek.inc) ;Deklaracije (imena internih registara)
;*****
;ovaj deo definise segment registara u fajlu registara interne memorije
;*****
RSEG AT 28H
TEMP: dsb 1 ;registar u kome se stavlja sadrzaj memorije
;da bi se poredio sa upisanim sadrzajem
TEMP0F: dsb 1 ;za testiranje memorije u njega se upisuje #0Fh
TEMPF0: dsb 1 ;za testiranje memorije u njega se upisuje #0F0h
flash: dsb 1 ;registar ciji se sadrzaj koristi za paljenje dioda L0 do L1
counter: dsw 1 ;brojac pomeraja za registarski indirektno adresiranje: [counter]
counter_hi equ (counter + 1):byte ;gornji bajt brojaca
RAM_EPROM_FLAG: dsb 1 ;u njega se upisuje sadrzaj lokacije na adresi #0A018h
;*****
;ucitavanje konfiguracionog bajta
;*****
CSEG AT 2018H

```

```

        dcw        11101101B        ;Chip configuration for 8098
                                        ;definis 8-bitnu data magistralu
                                        ;CCR se ucitava pri resetu iz lokacije 2018H
                                        ;three wait state, neophodno da bi kartica radila
;*****
;start programa
;*****
        cseg at (2080H)
;-----
reset_vector:                          ;START OF CODE
        ldb        TEMP0F,#0FH        ;inicijalizacija
        ldb        TEMPF0,#0F0H      ;inicijalizacija
;*****
;nazalost nije moguće pročitati sadržaj R/E# bita zato je potrebno poslužiti se sledecim
;trikom da bi se utvrdila koja je mapa aktivna: vrsi se provera da li je program iz eproma
;iskopiran u ram jer ako jeste onda je sadržaj lokacija #2018h i #0A018h isti, tj R/E#=1
;*****
        ldb        RAM_EPROM_FLAG,0A018H[zero]
        cmpb       RAM_EPROM_FLAG,#11101101b
                                        ;da li je ccb na adresi #2018H + #8000H
        jne        ram_je_gore        ;ako nije znaci da se program prvi put izvrsava
        jmp        ram_je_dole        ;znaci R/E=1, ram je na nizih 32k
;*****
;ovaj deo testira ram memoriju poredjenjem upisane i procitane vrednosti ukoliko nisu iste
;vrednosti onda se skace na sekvencu display_bad_adres koja izbacuje losu adresu na led
;diode, ovaj deo se izvrsava posle ukljucenja napajanja, u toku testiranja rama svetlece
;diode se ukljucuju: prvo L0, pa L0 i L1, pa L0,L1 i L2 i tako sve dok se sve ne ukljuce
;osim L4 posle cega se sve gasi i cela sema paljenja se ponavlja
;*****
ram_je_gore:
        ld         counter,#8000H      ;ram pocinje od #8000h za R/E#=0
Flash_LEDs:
        ldb        flash,#0FFH        ;pre load flash leds
start1:
;*****
;provera da li je counter stigao do neupotrebljivog dela rama
;*****
        cmp        counter,#0FC00H
;*****
;ako jeste sledeci zadatak je da se iskopira sadržaj eproma u ram koji je tada istestiran
;*****
        je         copy_eprom
        stb        TEMP0F,[counter]    ;upis u memoriju u adresu [counter]
        ldb        TEMP,[counter]     ;citanje iz memorije iz adrese [counter]
        cmpb       TEMP,#0FH          ;poredjenje sadržaja
        jne        display_bad_adres  ;skok ako je lokacija losa, adresa lokacije je u counter registru
        stb        TEMPF0,[counter]
        ldb        TEMP,[counter]
        cmpb       TEMP,#0F0H
        jne        display_bad_adres  ;skok ako je lokacija losa ,adresa lokacije je u counter registru
        inc        counter
;timer1:
        bbc        ios1,5,start1      ;if timer1 overflow do fl_left

fl_left:
        shlb       flash,#1           ;shift one bit now
        andb       flash,#0001111B    ;maska, dioda L4 je ugasena za R/E#=0
        orb        flash,#00010000B
;*****
;upisivanjem 1 dioda se gasi, ima 5 led dioda na lokaciji #0FFXXH na nizih 5 bita, bit 5
;R/E# je zaduzen za promenu mape adresnog prostora
;*****

```

```

        stb          flash,0FFFFH[zero] ;illuminate left, indeksno adresiranje,bazna adresa je nula
next:
        cmpb        flash,zero          ;is flash zero yet
        bne         start1              ;branch again if not zero
        br          Flash_LEDs         ;start pattern over again
;*****
;ovaj deo kopira eprom u ram i zato vreme diode trepere (ovo se i suvise brzo desava da bi
;se videlo!)
;*****
copy_eprom:
        xorb        flash,flash
        ld          counter,#2000H      ;u epromu ispod lokacije #2000h nije upisan nikakav podatak
prestart2:
        cmp         counter,#5000H     ;ovaj program nije duzi, zar ne ?
;*****
;kada se iskopira ide se na promenu adresnog prostora, reset i nop, nop
;*****
        je          do_reset
        ldb         TEMP,[counter]      ;kopiranje koda iz eproma u ram pre promene
;adresnog prostora tj upisivanja 1 u R/E#
        stb         TEMP,8000H[counter];offset je 32k
        inc         counter
pretimer1:
        bbc         ios1,5,prestart2   ;if timer1 overflow do blink blink
        notb        flash
        andb        flash,#00011111B   ;maska, dioda L4 je ugasena za R/E#=0
        orb         flash,#00010000B   ;ima 5 led dioda na lokaciji #0FFXXH
;na nizih 5 bita, bit 5 R/E# je zaduzen
;za promenu adresnog prostora
        stb         flash,0FF00H[zero] ;adresa registra je #0FFXX
        br          prestart2
;*****
;ovaj deo adresu lose memorijske lokacije sadrzanu u counter registru, ako postoji, salje na
;svetlece diode i to tako sto se prvo izbacuju bitovi 0 do 3 pa bitovi 4 do 7 pa bitovi 8 do
;11 i na kraju bitovi 12 do 15 ,counter registra, da bi imalo vremena da se procita losa
;adresa izmedju dva paljenja dioda prodje 4 prekoračenja tajmera1 (overflow)
;*****
display_bad_adres:
        bbc         ios1,5,$           ;
        andb        flash,counter,#00001111b
        stb         flash,0FFFFH[0]   ;adresa je #FFXXh
        bbc         ios1,5,$           ;
        ldb         flash, counter
        shrb        flash,#4
        stb         flash,0FFF0H[0]   ;adresa je #FFXXh
        bbc         ios1,5,$           ;
        andb        flash,counter_hi,#00001111b
        stb         flash,0FF0FH[zero] ;adresa je #FFXXh
        bbc         ios1,5,$           ;
        ldb         flash, counter_hi
        shrb        flash,#4
        stb         flash,0FF00H[zero] ;adresa je #FFXXh
        bbc         ios1,5,$           ;
        br          display_bad_adres  ;loop forever
;*****
;prvo R/E#=1 pa onda reset, red za cekanje sadrzi, reset, reset je potreban da bi se
;system ponovno inicijalizovao ali sada iz rama
;*****
do_reset:
        ldb         TEMP,#00101111b
        stb         TEMP,0FFFFH[zero] ;konacno R/E#=1, upaljena peta dioda
        rst         ;red za cekanje je cetiri bajta sto znaci da je rst u redu
;kada dodje do promene prostora
        nop         ;ipak malo nop nop, mada se sadrzaj reda resetuje rst instrukcijom

```

```

;*****
;ovaj deo se izvrsava iz ram memorije posle reset instrukcije sada je R/E#=1testira donjih
;32k memorije izuzev prvih 256 bajta koji pripadaju fajlu registara mikrokontrolera
;*****
ram_je_dole:
    ld            counter,#7000H
Flash_LEDs2:
    ldb          flash,#0FFH        ;pre load flash
start2:
    cmp          counter,#7FFFH     ;da li je counter stigao do kraja
                                        ;neupotrebljivog dela rama za R/E#=1

    je          mem_ok
    stb         TEMP0F,[counter]
    ldb         TEMP,[counter]
    cmpb        TEMP,#0FH
    jne         display_bad_adres1
    stb         TEMPF0,[counter]
    ldb         TEMP,[counter]
    cmpb        TEMP,#0F0H
    jne         display_bad_adres1
    inc         counter
;timer12:
    bbc         ios1,5,start2       ;if timer1 overflow do fl_right
fl_right:
    shrb        flash,#1           ;shift one bit now
    andb        flash,#0011111B
;*****
;maska, dioda L4 je upaljena za R/E#=1 ima 5 led dioda na lokaciji #0FFXXH na nizih 5
;bita, bit 5 R/E# je zaduzen za promenu adresnog prostora
;*****
    orb         flash,#00110000B
    stb         flash,0FFFFH[zero] ;illuminate right
next2:
    cmpb        flash,zero         ;is flash zero yet
    bne         start2             ;branch again if not zero
    br          Flash_LEDs2        ;start pattern over again
mem_ok:
    ljmp        mem_ok1
;*****
;ovaj deo adresu lose memorijske lokacije sadrzanu u counter registru, ako postoji, salje na
;svetlece diode i to tako sto se prvo izbacuju bitovi 0 do 3 pa bitovi 4 do 7 pa bitovi 8 do
;11 i na kraju bitovi 12 do 15 ,counter registra da bi imalo vremena da se procita losa
;adresa izmedju dva paljenja dioda prodje 4 prekoracenja tajmera1 (overflow)
;*****
display_bad_adres1:
    bbc         ios1,5,$
    andb        flash,counter,#00001111b
    orb         flash,#00110000b
    stb         flash,0FFFFH[0]
    bbc         ios1,5,$
    ldb         flash, counter
    shrb        flash,#4
    orb         flash,#00110000b
    stb         flash,0FFFFH[0]
    bbc         ios1,5,$
    andb        flash,counter_hi,#00001111b
    orb         flash,#00110000b
    stb         flash,0FFFFH[zero]
    bbc         ios1,5,$
    ldb         flash, counter_hi
    shrb        flash,#4
    orb         flash,#00110000b
    stb         flash,0FFFFH[zero]
    bbc         ios1,5,$

```

```

        br          display_bad_adres1 ;do it forever
;*****
;ovaj deo se izvrsava na kraju programa i signalizira da je memorija OK tako sto ukljucujU
;svetlece diode po semi 10101, 01010, ...
;*****
mem_ok1:
        ldb          flash,#00110101B
        stb          flash,0FFFFH[zero]
        bbc          ios1,5,$
        bbc          ios1,5,$
        ldb          flash,#00101010B
        stb          flash,0FFFFH[zero]
        bbc          ios1,5,$
        bbc          ios1,5,$
;*****
;da bi program posle hardverskog reseta poceo da se izvrsava iz pocetka potrebno je
;promeniti sadrzaj rama na adresi CCB bajta
;*****
        ldb          RAM_EPROM_FLAG,#0ffh
        stb          RAM_EPROM_FLAG,02018H[zero]
        br          mem_ok
end                                     ;kraj

```

PROGRAMSKA PODRŠKA

U ovom poglavlju je opisana programska podrška mikroprocesorskom sistemu koja se sastoji iz dva programa: nadglednog programa koji je upisan u EPROM kartice i prilagodnog programa koji se izvršava u PC računaru. Svrha nadglednog programa je da održava serijsku komunikaciju kartice sa PC računarem i izvršava naredbe koje prima serijskom vezom od PC računara. Svrha prilagodnog programa je: da održava serijsku komunikaciju PC računara sa karticom, komande zadate preko tastature PC-ja prosledi kartici, i pristigle poruke sa kartice obradi i prikaže na ekranu ili uskladišti na tvrdi disk u vidu ASCII datoteke.

3.1 PRILAGODNI PROGRAM

U ovom poglavlju je opisan prilagodni program koji podržava serijsku vezu između mikroprocesorskog sistema i PC računara i izvršava se u PC računaru. Svrha ovog programa je da komande zadate preko tastature serijskom vezom prenese mikroprocesorskom sistemu i da prispele podatke od strane mikroprocesorskog sistema obradi i prikaže na ekranu ili uskladišti na tvrdom disku u vidu ASCII datoteke.

Ime prilagodnog programa je serial.exe, pisan je u programskom jeziku C za DOS operativni sistem i pomoću njega korisnik može serijskom vezom da učita svoj kod u RAM memoriju mikroprocesorskog sistema, ispita ga, promeni i zada njegovo izvršenje. Podržava serijsku komunikaciju asinhronog tipa brzine 9600 bps ili 19200 bps preko COM1 ili COM2 porta PC računara. Obezbeđuje pristup svim lokacijama 16-bitnog adresnog prostora kartice u toku izvršavanja korisničkog programa. U okviru prilagodnog programa se može aktivirati grafički interfejs, radi prikaza 8-bitne ili 16-bitne memorijske lokacije (sa adrese koju zadaje korisnik) na ekranu PC računara u vidu vremenskih krivih čije amplitude odgovaraju sadržaju memorijske lokacije, kao i čuvanje tih krivih u vidu fajla radi dokumentovanja. Ima šesnaest komandi koje je moguće zadati mikroprocesorskom sistemu.

3.1.1 STARTOVANJE PRILAGODNOG PROGRAMA SERIAL.EXE

Rad sa mikroprocesorskim sistemom se obavlja u prostoriji 40a, termin za rad se dobija u dogovoru sa Stevanom Antićem. Mikroprocesorski sistem radi u okviru laboratorijske stanice za upravljanje asinhronim motorom, uputstvo za rad sa ovom stanicom se nalazi na serveru Kiklop datoteka x:\nastava\eg4ev\uputstvo.pdf.

Pre startovanja programa serial.exe potrebno je uključiti inverter u koji je ubodena mikroprocesorska kartica (u prisustvu Stevana Antića ili asistenta koga je odredio profesor Slobodan Vukosavić), spojiti kablom raspoloživi serijski port PC računara (na računaru koji je povezan sa laboratorijskom stanicom je prisutan samo port COM2, DB9) i konektor na mikroprocesorskoj kartici CON3 (pogledaj opis konektora i šemu kartice) i uključiti PC računar. Kabl treba da ima na svojim krajevima konektore DB9, ženski i muški.

Host program serial.exe se nalazi u direktorijumu d:\196demo\radni\, startuje se naredbom:

```
serial com2
```

posle čega se na ekranu pojavi sledeća slika:

```
Status:
COM 2
Baud rate 9600

Lista komandi:

read byte: rb
write byte: wb
reset board: rstb
write file: wf
quit: q
write word: ww
read word: rw
set board: setb
write double: wd
read double: rd
send byte: sb
draw location byte: dlb
draw speed: ds
set baud rate: sbr
copy serial to ram: csr
start user: su

Zadaj komandu malim slovima: _
```

Kada se program prvi put startuje serijska veza je podešena na 9600 bps.

3.1.2 LISTA KOMANDI I NJIHOVO OBJAŠNJENJE

Ima šesnaest komandi koje je moguće zadati mikroprocesorskom sistemu. Komande se zadaju malim slovima (najviše četiri slova). U listi komandi koja sledi, komande su odštampane u bold italic formatu, da bi se startovalo izvršenje komande posle kucanja imena komande potrebno je pritisnuti taster ENTER:

1. ***rb***<ENTER>

Ovom komandom se čita sadržaj memorijske lokacije kartice veličine bajt na ekranu PC računara. Adresa lokacije sa koje se čita se unosi u nastavku komande kao odgovor na pitanje koje se pojavljuje na ekranu posle pritiska na taster ENTER.

2. ***wb***<ENTER>

Ovom komandom se učitava sadržaj memorijske lokacije kartice veličine bajt. Adresa i sadržaj lokacije koja se učitava unose se u nastavku komande kao odgovor na pitanja koja se pojavljuju posle pritiska na taster ENTER..

3. ***rstb***<ENTER>

Ovom komandom se startuje izvršenje programa upisanog u RAM: nadgledni program setuje konfiguracioni bit R/E#=1, zatim resetuje procesor 8098 (programski reset nema uticaja na R/E# bit) reset instrukcijom, posle čega započinje izvršavanje programa upisanog u RAM kartice od reset adrese #2080h.

4. ***wf***<ENTER>

Ovom komandom se upisuje korisnički program u RAM kartice, korisnički program je u vidu binarne datoteke koja se formira kao što je opisano u 4 poglavlju. U nastavku komande posle pritiska na taster ENTER korisnik navodi ime datoteke, ****.bin*** koja mora da bude smeštena u istom direktorijumu kao i prilagodni program serial.exe.

5. **q<ENTER>**
Ovom komandom se prekida serijska komunikacija sa mikroprocesorskim sistemom i vraća se u MS-DOS operativni sistem. Posle ove komande mikroprocesorski sistem ostaje u stanju u kome je bio posle zadavanja komande koja joj je prethodila.
6. **ww<ENTER>**
Ovom komandom se učitava sadržaj memorijske lokacije kartice veličine reč (16 bita). Adresa i sadržaj lokacije koji se upisuje se unose u nastavku komande kao odgovor na pitanja koja se pojavljuju posle pritiska na taster ENTER..
7. **rw<ENTER>**
Ovom komandom se čita sadržaj memorijske lokacije kartice veličine reč (16 bita). Adresa lokacije koja se čita se unosi u nastavku komande kao odgovor na pitanje koje se pojavljuje na ekranu posle pritiska na taster ENTER.
8. **setb<ENTER>**
Ovom komandom se startuje izvršenje programa upisanog u EPROM: nadgledni program resetuje konfiguracioni bit R/E#=0, zatim resetuje procesor 8098 (programski reset nema uticaja na R/E# bit) reset instrukcijom, posle čega započinje izvršavanje programa upisanog u EPROM kartice od reset adrese #2080h.
9. **wd<ENTER>**
Ovom komandom se učitava sadržaj memorijske lokacije kartice veličine dupla reč (32 bita). Adresa i sadržaj lokacije koji se upisuje se unose u nastavku komande kao odgovor na pitanja koja se pojavljuju posle pritiska na taster ENTER.
10. **rd<ENTER>**
Ovom komandom se čita sadržaj memorijske lokacije kartice veličine dupla reč (32 bita). Adresa lokacije koja se čita se unosi u nastavku komande kao odgovor na pitanje koje se pojavljuje na ekranu posle pritiska na taster ENTER.
11. **sb<ENTER>**
Ovom komandom se serijskom vezom mikroprocesorskom sistemu šalje podatak veličine bajt. Vrednost podatka se unosi u nastavku komande kao odgovor na pitanje koje se pojavljuje na ekranu posle pritiska na taster ENTER. Kako će mikroprocesorski sistem tumačiti pristigli podatak zavisi od vrednosti poslatog bajta. Korisniku se preporučuje da ovu komandu ne koristi pre nego što ne upozna programe serial.c (ASCII izvorni kod prilagodnog programa serial.exe) i serial.a96 (ASCII izvorni kod nadglednog programa) koji se nalaze u istom direktorijumu kao i serial.exe.
12. **dlb<ENTER>**
Ovom komandom se aktivira grafički interfejs koji prikazuje proizvoljnu memorijsku lokaciju veličine bajt na ekranu PC računara u vidu vremenskih krivih čije amplitude odgovaraju sadržaju memorijske lokacije. Adresa lokacije čiji se sadržaj crta na ekranu PC računara se unosi u nastavku komande kao odgovor na pitanje koje se pojavljuje posle pritiska na taster ENTER. U okviru grafičkog interfejsa postoji opcija čuvanja ekrana na hard disk u vidu ASCII datoteke koja sadrži neponderisane numeričke vrednosti memorijske lokacije u toku deset sekundi, ukupno petsto tačaka. U toku rada grafičkog interfejsa moguće je zadati trenutak promene neke memorijske lokacije veličine bajt pomoću opcije stimulus koja se aktivira pritiskom na taster sa brojem 1.
13. **ds<ENTER>**
Ovom komandom se aktivira grafički interfejs koji crta proizvoljnu memorijsku lokaciju veličine reč na ekranu PC računara u vidu vremenskih krivih čije amplitude odgovaraju sadržaju memorijske lokacije ponderisane konstantom 1/8. Adresa lokacije koja se crta na ekranu PC računara se unosi u nastavku komande kao odgovor na pitanje koje se

pojavljuje na ekranu posle pritiska na taster ENTER. U okviru grafičkog interfejsa postoji opcija čuvanja ekrana na hard disk u vidu ASCII fajla koji sadrži ponderisane numeričke vrednosti memorijske lokacije u toku deset sekundi, ukupno petsto tačaka. U toku rada grafičkog interfejsa moguće je zadati trenutak promene neke memorijske lokacije veličine bajt pomoću opcije stimulus koja se aktivira pritiskom na taster sa brojem 1. Ova komanda je specijalno pravljen za prikaz brzine na ekranu PC računara koja se proračunava u okviru demonstracionog programa ukrozf.a96 koji se može naći u istom direktorijumu sa serial.exe programom.

14. *sbr*<ENTER>

Ovom komandom se menja brzina serijske komunikacije sa 9600bps na 19200bps. Zadavanjem ove komande mikroprocesorski sistem ulazi u reset stanje posle čega se serijski port inicijalizuje na 19200bps, R/E# konfiguracioni bit ostaje isti, nemenja se mapa adresnog prostora. Da bi se brzina serijske komunikacije vratila na 9600bps potrebno je izaći iz prilagodnog programa komandom 5 i isključiti napajanje mikroprocesorskom sistemu i ponovo uključiti.

15. *csr*<ENTER>

Ovom komandom se kopira monitor program iz EPROM-a u RAM memoriju. Ovu komandu je neophodno izvršiti posle upisa korisničkog programa u RAM komandom 4 ukoliko korisnik želi da njegov program radi sa monitor programom, takode korisnik treba da izvrši modifikacije u kodu svog programa kao što je objašnjeno u poglavlju 3.2.2.

16. *su*<ENTER>

Ovom komandom se startuje izvršenje korisničkog programa prethodno upisanog u RAM kartice. Ovom komandom se zadaje početak izvršenja korisničkog programa počev od adrese #2090h (pogledaj poglavlje 3.2.2).

3.2 NADGLEDNI PROGRAM

Ovaj program podržava serijsku komunikaciju mikroprocesorskog sistema i PC računara i upisan je u EPROM kartice. Po uključanju ili resetu kartice nadgledni program pali svetleće diode L0, L1, L2 i L3, paleći prvo L0 zatim L1 pa L2 i na kraju L3 posle čega isključuje sve diode i ponavlja šemu paljenja ispočetka, dioda L4 je ugašena kad je konfiguracioni bit R/E#=0 a upaljena za R/E#=1. Ovakva signalizacija znači da je kartica ispravna i spremna za uspostavljanje serijske veze. Nadgledni program izvršava komande pristigle serijskom vezom sa PC računara (u poglavlju 3.1 je dat spisak komandi njihov opis i način zadavanja).

3.2.1 REZERVISANE MEMORIJSKE LOKACIJE

Da bi nadgledni program radio, deo memorijskog adresnog prostora korisnik nesme da koristi pri pisanju svog programa koji namerava da upiše u RAM kartice, jer je rezervisan za nadgledni program. Rezervisane adrese su od #7000h do #7FFFh. Ukoliko korisnik locira svoj kod ili deo koda u rezervisani adresni prostor i ovakav kod učita u RAM i startuje njegovo izvršenje (komande koje to omogućuju su opisane u poglavlju 3.1) onda nadgledni program neće raditi kao ni korisnički program koji je upisan u RAM. Takođe je i deo registara internog fajla registara mikrokontrolera rezervisan za potrebe nadglednog programa, i obuhvata adrese od #0E0h do #0EFh. Ukoliko korisnik locira svoj segment registara ili deo segmenta u rezervisani adresni prostor i ovakav kod učita u RAM i startuje njegovo izvršenje (komande koje to omogućuju su opisane u poglavlju 3.1) onda nadgledni program neće raditi kao ni korisnički program koji je upisan u RAM. Maksimalna

velečina korisničkog programa koja može da se učita u karticu je određena veličinom RAM memorije i iznosi (ukoliko se izuzmu rezervisane lokacije) #7000h-#100h=#6f00h bajtova. Najveća adresa do koje korisnik sme da smesti svoj kod je #6FFFh. Treba zapamtiti da je upis u RAM memoriju korisničkog programa moguć samo kad je R/E#=0, dakle, RAM je u toku upisa programa smešten u gornjoj polovini memorijskog prostora. Treba zapamtiti da je aktiviranje korisničkog programa moguće samo kad je R/E#=1, dakle, RAM je u toku izvršavanja korisničkog programa smešten u donjoj polovini memorijskog prostora. Pošto se korisnički program uvek izvršava iz donjeg bloka adresa, iz RAM memorije, onda pri pisanju programa kod treba locirati u niži blok adresa.

3.2.2 PRAVLJENJE KORISNIČKOG PROGRAMA U KOMBINACIJI SA NADGLEDNIM PROGRAMOM

Da bi korisnički program kada se učita u RAM kartice i startuje njegovo izvršenje radio u kombinaciji sa nadglednim programom potrebno je uneti izvesne izmene u kodu korisničkog programa. Sledeći primer pokazuje na koji način korisnik treba da modifikuje svoj kod da bi radio u kombinaciji sa nadglednim programom.

```

;*****
;Sledeci deo korisnik ubacuje u svoj program pre reset adrese #2018h
;*****
CSEG AT 200Ch ;Definisane interapt vektora serijske
    dcw serial_isr ;veze serial_isr interrupt at #7000h
CSEG AT 7000h
serial_isr:
CSEG AT 7200h ;Locira se kod glavnog dela nadglednog
main_serial: ;programa (pozadinska petlja)
CSEG AT 2018h ;Da bi mikroprocesorski sistem radio
    dcw 11101101b ;potrebno je da CCB bude 11101101b
    ;cip konfiguracija definise:
    ;three wait state,8 bitnu data magistralu
    ;CCR se ucitava pri resetu iz lokacije
    ;2018H
;*****
;Sledeci deo korisnik ubacuje na pocetak instrukcijskog dela programa, to je deo
;koji se izvrsava odmah posle reseta (programskog reseta posto je konfiguracioni
;bit R/E#=1 setovan jer hardverski reset uvek upisuje nulu u konfiguracioni bit) i
;pocinje od adrese #2080h jer je to reset adresa ;procesora 8098.
;*****
CSEG AT 2080h ;2080H je reset adresa, prvo se izvrsava
    br main_serial ;pozadinska petlja nadglednog
    ;programa dok korisnik komandom ne
    ;zada pocetak izvrsavanja korisnickog
    ;programa
;*****
;ovde je pocetak izvrsenja korisnickog programa (misl se na instrukcijski
;kod) iz pozadinske petlje se skace na adresu #2090h od koje pocinje instrukcijski
;kod koji je napisao korisnik, trenutak skoka na tu adresu je odredjen zadavanjem
;komande preko tastature za pocetak izvrsenja korisnickog programa, pogledaj
;poglavlje 3.1
;*****
CSEG AT 2090h
korisnicki_program:

```

```

;*****
;sledeci deo korisnik stavlja na kraju inicijalizacionog dela svog programa
;*****
    ldb      ioc1,#0010000b    ;turn TXD pin on, port 2.0 je THD pin !!!!
    ldb      int_mask,#0100000b ;mora da se osposobi serijski interapt !!!!
                                           ;podrazumeva se da ce korisnik osposobiti i
                                           ;svoje interapte setujuci odgovarajuce bite u
                                           ;int_mask registru
    ei       ;enable interrupts !!!!!

```

3.2.3 PRAVLJENJE KORISNIČKOG PROGRAMA BEZ MOGUĆNOSTI RADA NADGLEDNOG PROGRAMA

Korisnički program koji se učita u RAM kartice može da radi i bez nadglednog programa. Pri pisanju ovakvog programa korisniku stoji na raspolaganju ceo adresni prostor RAM memorije od adrese #100h do #07BFFh. Da bi korisnički program radio potrebno je deklarirati CCB bajt koji se učitava sa adrese #2018h u CCR registar posle softverskog reseta iz RAM memorije za R/E#=1. Njegov sadržaj treba da bude deklarisan u kodu na isti način kao i kad se korisnički program piše u kombinaciji sa nadglednim programom jer je hardver kartice ostao isti:

```

;*****
;Da bi mikroprocesorski sistem ispravno radio potrebno je da CCB bude 11101101b
;*****
CSEG AT 2018H
    dcw      11101101b        ;cip konfiguracija mikroprocesorskog
                                           ;sistema, three wait state, definise 8
                                           ;bitnu data magistralu, CCR se učitava
                                           ;pri resetu iz lokacije 2018H

```

3.2.4 SIGNALIZACIJA NADGLEDNOG PROGRAMA KORISNIKU POMOĆU SVETLEĆIH DIODA

U toku reseta kartice diode L0 do L4 svetle.

Posle uključanja napajanja, posle reseta pritiskom na reset taster ili posle softverskog reseta, diode trepere na interesantan način. Redom se pale diode L0 do L3 posle čega se sve četiri diode gase i šema paljenja se iznova ponavlja. Dioda L4 je ugašena ako je R/E#=0 a upaljena ako je R/E#=1.

Posle zadavanja prve komande (osim komandi 3, 8 i 11, pogledaj poglavlje 3.1), diode prestaju da trepere na prethodno opisan način. Dioda L0 i L1 su uvek ugašene, dioda L2 se pali kada se startuje korisnički program komandom *su* (pogledaj poglavlje 3.1, spisak komandi) inače je ugašena. Dioda L5 je upaljena ako je u toku upis fajla u RAM kartice inače je ugašena. Dioda L4 je ugašena ako je R/E#=0 a upaljena ako je R/E#=1.

Odstupanje od ovakve signalizacije znači da je došlo do greške, preporučuje se reset kartice i PC računara, a ako i tada sistem neispravno funkcioniše onda treba isključiti i ponovno uključiti napajanje mikroprocesorskog sistema i PC računara*.

* Signalizacija neće funkcionisati ako je korisnik startovao svoj program bez nadglednog programa.

PRIMER UPOTREBE MIKROPROCESORSKOG SISTEMA

U ovom poglavlju je dat primer upotrebe mikroprocesorskog sistema. Primer obuhvata način formiranja binarne datoteke korisničkog programa koja se učitava u RAM memoriju kartice, i sekvencu radnji koje je potrebno uraditi da bi se startovalo izvršenje korisničkog programa iz RAM-a kartice. Sekvenca radnji podrazumeva postupke korisnika od uključanja PC računara i laboratorijske stanice u okviru koje mikroprocesorski sistem radi, do startovanja izvršenja korisničkog koda i njegovog ispitivanja.

Mikroprocesorski sistem se nalazi u prostoriji 40a u okviru laboratorijske stanice za kontrolu asinhronog motora. Studentima se preporučuje da pročitaju uputstvo za rad sa ovom laboratorijskom stanicom, uputstvo se nalazi u laboratoriji 40a u štampanom obliku, i u vidu datoteke na serveru Kiklop, datoteka x:\nastava\eg4ev\uputstvo.pdf.

U PC računaru koji je povezan sa laboratorijskom stanicom u direktorijumu d:\196demo\radni\ se nalazi programski paket za rad sa mikroprocesorskim sistemom i u njemu će korisnik smestiti kod koji namerava da ispita. Paket sadrži assembler, linker i kompajler za procesor Intel 8098 i prilagodni program serial.exe. U direktorijumu d:\196demo\primer\ se pored assemblera, linkera i kompajlera nalaze i programski primeri pisani u assembleru za mikrokontroler 8098, oni služe da se korisnik upozna sa osobinama mikroprocesorskog sistema i načinom rada sa sistemom u okviru laboratorijske stanice.

Datoteke sa primerima su:

- d:\196demo\primer\serial.a96, izvorna datoteka nadglednog programa opisanog u 3.2 poglavlju;
- d:\196demo\primer\ukrozf.a96, izvorna datoteka koja služi kao primer korisničkog programa koji se učitava u RAM kartice i radi u kombinaciji sa nadglednim programom, korisniku se preporučuje da upozna ovaj program pre nego što počne rad na svom kodu;
- d:\196demo\primer\ukrozfbz.a96, izvorna datoteka koja služi kao primer korisničkog programa koji se učitava u RAM kartice, to je verzija programa ukrozf.a96 koja radi bez nadglednog programa, korisniku se preporučuje da upozna ovaj program pre nego što počne rad na svom kodu;
- d:\196demo\primer\ukrozfsr.a96, izvorna datoteka programa koji je upisan u EPROM kartice i predstavlja kombinaciju programa serial.a96 i ukrozf.a96;
- d:\196demo\primer\serial.c, izvorna datoteka prilagodnog programa opisanog u 3.1 poglavlju, a koji se izvršava u PC računaru.

4.1 PRIMER IZVRŠAVANJA KORISNIČKOG PROGRAMA U KOMBINACIJI SA NADGLEDNIM PROGRAMOM

Program koji će poslužiti kao primer je d:\196demo\primer\ukrozf.a96. Pretpostavka je da je korisnik napisao ovaj program sa namerom da ga učita u RAM kartice, startuje njegovo izvršavanje i ispita ga koristeći programsku podršku opisanu u 3 poglavlju, u nastavku je opisana sekvenca radnji koje korisnik obavlja u cilju izvršenja prethodnog.

4.1.1 PREVOĐENJE KORISNIČKOG PROGRAMA I FORMIRANJE BINARNE DATOTEKE KOJA SE UČITAVA U RAM KARTICE

Prvi korak po završetku formiranja ASCII asemblerske datoteke korisničkog programa je prevođenje u binarnu datoteku koja se učitava u RAM kartice. Primer koje opcije treba koristiti pri prevođenju korisničkog programa i formiranju binarne datoteke čiji se sadržaj učitava u RAM kartice je "batch" program d:\196demo\primer\ukf.bat. Ovaj program prevodi korisnički program ukrozf.a96, formira datoteku ukrozf.lst u kojoj se nalaze poruke o greškama i tabela simbola (nazivi registara i labela) sa adresama simbola, formira datoteku ukrozf.m96 u kojoj se nalaze adrese upotrebljenih memorijskih lokacija u korisničkom programu (korisnik pregleda ovaj program da bi utvrdio da li njegov segment registara i segmenti koda i podataka ne zalaze u lokacije rezervisane za nadgledni program i da li se međusobno preklapaju), na kraju "batch" program formira datoteku ukrozf.bin koja se upisuje u RAM kartice. Sadržaj ovog "batch" programa je:

```
ASM96 ukrozf.a96 nodb  
RL96 ukrozf.OBJ TO ukrozf.ABS  
OH ukrozf.abs TO ukrozf.hex  
hexbin ukrozf.hex ukrozf.bin I
```

4.1.2 REDOSLED POSTUPAKA KORISNIKA PRI RADU SA LABORATORIJSKOM STANICOM U CILJU ISPITIVANJA SVOG PROGRAMA

- a) Korisnik uključi laboratorijsku stanicu pridržavajući se postupka opisanog u uputstvu na serveru Kiklop datoteka x:\nastava\eg4ev\uputstvo.pdf, i uključi PC računar koji je u sklopu stanice (korisniku koji prvi put radi sa laboratorijskom stanicom pomaže asistent određen od strane profesora Slobodana Vukosavića), po uključenju stanice, signalizacione diode na mikroprocesorskoj kartici počinju da svetle na način opisan u 3 poglavlju.
- b) Korisnik se pozicionira u direktorijum d:\196demo\primer\ i startuje program ukf.bat opisan u poglavlju 4.1.1 radi formiranja datoteke ukrozf.bin, koja se učitava u RAM kartice. Od ove tačke se zahteva samostalan rad korisnika.
- c) Po završetku rada programa ukf.bat, otvori datoteku ukrozf.lst komandom:

```
edit ukrozf.lst <ENTER>
```

pozicionira se na kraj datoteke i primeti poruku o greškama u toku prevođenja, u ovom slučaju pošto nema grešaka poruka će biti:

ASSEMBLY COMPLETED, NO ERROR(S) FOUND

Zatim se korisnik pozicionira na početku tabele simbola koja počinje od linije 1341 na kojoj piše:

SYMBOL TABLE LISTING

U okviru ove tabele se nalaze imena, veličine i adrese simbola koje se koriste u programu ukrozf.a96 (simboli to su imena promenljivih, labele tj. imena adresa, i imena prekidnih rutuna koje je korisnik upotrebio u toku pisanja svog koda).

Korisnik treba da zabeleži adrese memorijskih lokacija koje su od interesa pri ispitivanju korisničkog programa.

Registar BX se nalazi u tabeli simbola na liniji 1359, iz tabele se vidi da je on veličine 16 bita (**word**) i da se nalazi na adresi #20h. Sadržaj višeg bajta BX registra (adresa #21h) određuje učestanost fundamentala PWM izlaznog signala invertora koji napaja asinhroni motor, u smislu $256 \Leftrightarrow 100\text{Hz}$, $0 \Leftrightarrow 0\text{Hz}$.

Registar TMP1, linija 1438, je veličine reč (16 bita) i nalazi se na adresi #5Eh, njegov sadržaj je u svakom trenutku srazmeran trenutnoj vrednosti indeksa modulacije jedne od invertorskih faza.

Registar U_F_STAT, linija 1439, je veličine bajt (8 bita) i nalazi se na adresi #5Dh, to je konfiguracioni registar čija prva dva bita određuju mod izvršavanja programa za U/F kontrolu asinhronog motora. Bit 0 ovog registra pod imenom SORS_FLAG linija 1426, određuje kako će se zadavati učestanost fundamentala PWM-a, ako je SORS_FLAG=0 onda se učestanost zadaje pomoću PC-računara (sadržaj gornjeg bajta BX registra, na adresi #21h) a ako je SORS_FLAG=1 onda se učestanost zadaje pomoću potencijometra (na šemama označen kao T-POT) u smislu $0V \Leftrightarrow 0\text{Hz}$, $5V \Leftrightarrow 100\text{Hz}$. Reset stanje ovog bita je 0. Bit 1 ovog registra pod imenom GRAPH_MODE_FLAG, linija 1368, određuje da li će se meriti brzina obrtanja motora pomoću impulsa koji dolaze u HSI uređaj sa inkrementalnog enkodera. Merenje je urađeno tako da se sve uzlazne i silazne ivice obe faze tretiraju kao isti događaj koji inkrementira stanje brojača timer2, enkoder daje ukupno 5000 takvih događaja po obrtaju, pogledaj 2 poglavlje, periodičnim iščitavanjem brojača timer2 izračunava se inkrement brojača između dva čitanja i smešta u registar INCR. Ako je GRAPH_MODE_FLAG=1 merenje je omogućeno. Ako je GRAPH_MODE_FLAG=0 merenje je onemogućeno. Reset stanje ovog bita je 1. Ostali biti ovog registra su neiskorišćeni.

Registar INCR, linija 1377, je veličine reč (16 bita) nalazi se na adresi #5Ah i sadrži podatak o brzini obrtanja asinhronog motora.

- d) Korisnik otvori datoteku ukrozf.m96 u kojoj se nalaze adrese upotrebljenih memorijskih lokacija od strane korisničkog programa (korisnik pregleda ovu datoteku da bi utvrdio da li se njegov segment registara i segmenti koda i podataka ne preklapaju međusobno i da li ne zalaze u lokacije rezervisane za nadgledni program) komandom:

edit ukrozf.m96 <ENTER>

Pozicionira se na kraj datoteke i primeti poruku o greškama u toku povezivanja programa, u ovom slučaju pošto nema grešaka poruka će biti:

RL96 COMPLETED, 0 WARNING(S), 0 ERROR(S)

Pozicionira se na liniju 28 na kojoj piše:

*** GAP *** 006CH 1F94H

što znači da su te memorijske lokacije neiskorišćene od strane korisničkog programa, pošto u tom bloku adresa leži i segment registara rezervisanih za nadgledni program (pogledaj 3 poglavlje i datoteku serial.a96), znači da neće doći do preklapanja segmenta registara između korisničkog programa i nadglednog programa.

Pozicionira se na liniju 44 na kojoj piše:

*** GAP *** 4C00H B400H

što znači da su te memorijske lokacije neiskorišćene od strane korisničkog programa, pošto u tom bloku adresa leži i segment adresa rezervisanih za kod nadglednog programa (pogledaj 3 poglavlje), znači da neće doći do preklapanja segmenta koda između korisničkog programa i nadglednog programa.

- e) Primeti postojanje datoteke ukrozf.bin, i zada komandu:

wpview ukrozf.bin <ENTER>

pritisne funkcijski taster F4 da bi video sadržaj koji se učitava u RAM kartice. Sadržaj datoteke ukrozf.bin koji korisnik vidi na ekranu PC računara je u heksadecimalnom formatu, krajnja leva kolona na ekranu prikazuje adrese 8 bitnih memorijskih lokacija, a srednja kolona sadržaj tih adresa, jedna vrsta se sastoji iz 16 bajtova pri čemu je prva kolona vrste adresa prvog bajta čiji je sadržaj ispisan u drugoj koloni vrste. Na primer, pozicionira se na deo koji prikazuje vrstu:

02010 00 00 00 00 | 00 00 00 00 | ED 00 00 00 | 00 00 00 00

ona pokazuje da je adresa prvog bajta #2010h a da je sadržaj svih lokacija od adrese #2010h do #201Fh nula, osim sadržaja lokacije #2018h, sadržaj lokacije #2018h predstavlja CCB, konfiguracioni bajt čija je vrednost #0EDh = #11101101b (pogledaj 3 poglavlje).

- f) Aktivira prilagodni program komandom:

serial com2 <ENTER>

posle čega se na ekranu prikaže lista od 16 komandi i status serijske veze. Početna brzina serijske komunikacije je 9600 bps.

- g) Zada komandu za učitavanje korisničkog programa u RAM memoriju kartice, to je komanda broj 4 sa liste komandi:

wf<ENTER>

posle čega se na ekranu pojavi poruka:

Navedi ime fajla u *.bin formatu, početna adresa upisa je #8000h:

u nastavku poruke se dopiše ime binarne datoteke koja se upisuje u RAM:

ukrozf.bin<ENTER>

u toku upisa datoteke u RAM, nadgledni program vrši proveru ispravnosti upisa i na ekran šalje niz poruka koje obavestavaju korisnika o stanju serijske komunikacije, i pali svetleću diodu L3 dok su ostale ugašene što znači da je transfer u toku. Po završetku transfera pojavljuje se poruka da je transfer završen i da je sistem spreman za zadavanje sledeće komande.

- h) Da bi se omogućio rad nadglednog programa u toku izvršavanja korisničkog programa potrebno je iskopirati nadgledni program iz EPROM memorije u RAM memoriju, oblast RAM-a u koju se kopira nadgledni program pripada rezervisanim lokacijama (pogledaj 3 poglavlje), komanda koja kopira nadgledni program je komanda broj 15, treba je zadati:

csr<ENTER>

proces kopiranja kratko traje, deo sekunde.

- i) Sada je potrebno promeniti mapu adresnog prostora, to se postiže komandom broj 3:

rstb<ENTER>

primetite da je posle zadavanja ove komande svetleća dioda L4 upaljena, to znači da je stanje konfiguracionog bita R/E#=1.

- j) Podesi autotransformator koji je u sklopu laboratorijske stanice na 200 podeoka, vodite računa autotransformator nesmete da podesite preko 220 podeoka, zbog reagovanja prenaponske zaštite!!!

Resetuje prekostrujnu zaštitu invertora pritiskom na crveni taster koji se nalazi na ivici kartice prekostrujne i prenaponske zaštite K7, pri tome pazi da ne dodirne kabl za serijsku komunikaciju.

- k) Startuje korisnički program komandom broj 16:

su<ENTER>

primeti da se upalila dioda L2 čime je označeno da je korisnički program počeo sa radom.

- l) Kako je stanje konfiguracionog registra U_F_STAT=#0000 0010b (pogledaj tačku c)) zadavanje učestanosi fundamentalna PWM je kontrolisano sadržajem višeg bajta BX registra čija je adresa #21h. Zadavanje učestanosti od 50 Hz se ostvaruje upisom broja #80h u lokaciju #21h komandom broj 1:

wb<ENTER>

posle čega se sadržaj i adresa lokacije veličine bajt u koju se vrši upis unose kao odgovori na pitanja koja se pojavljuju u nastavki izvršenja komande:

80<ENTER>

21<ENTER>

posledica ove komande je početak okretanja motora i njegovo ubrzavanje do brzine nešto manje od 1500 o/min.

m) Da bi korisnik aktivirao grafički interfejs koji radi prikaza brzine na ekranu PC računara potrebno je:

- poveže inkrementalni optički enkoder sa konektorom na kartici označenim sa CON2 (kabl sa odgovarajućim konektorima se uzima od asistenta, preporučljivo je da se povezivanje obavi pre uključanja mikroprocesorskog sistema!),
- zada komandu broj 14

sbr<ENTER>

- posle ove komande sistem se softverski resetuje (konfiguracioni bit R/E#=1 ostaje nepromenjen kao i sadržaj RAM-a) i serijski port se inicijalizuje na 19200 bps koliko je potrebno da bi grafički interfejs ispravno radio,
- ponoviti tačku k),
- zada komandu broj 13 koja aktivira grafički interfejs za prikaz brzine na ekranu:

ds<ENTER>

u nastavku komande se kao odgovor na pitanje:

unesi adresu lokacije sa koje crtaš brzinu (HEX):

ukucava adresu registra INCR (pogledaj tačku c)):

5a<ENTER>

Time je aktivirao grafički interfejs za prikaz brzine na ekranu.

n) Aktivira opciju STIMULUS pritiskom na broj 1 i odgovara na poruke u skladu sa tačkom l), na poslednje pitanje koje traži unos trenutka dejstva stimulusa, zadaje npr. 1000ms. Dakle, korisnik ukucava:

50<ENTER>

21<ENTER>

1000<ENTER>

- o) Aktivira opciju FREEZ pritiskom na broj 3 i sačeka da se brzina iscrta preko celog ekrana, posle čega ekran ostaje neizmenjen sve do aktiviranja neke druge raspoložive opcije. Aktivira opciju SAVE SCREEN pritiskom na broj 5 i navede ime datoteke u koju se upisuje sadržaj koji odgovara talasnom obliku brzine koja je prikazana na ekranu. Datoteka će imati dve kolone po 500 brojeva, u drugoj koloni će se nalaziti brojevi koji su dobijeni periodičnim iščitavanjem registra INCR u toku 10 sekundi prebačeni u o/min. Aktivira opciju QUIT pritiskom na broj 2.
- p) Aktivira grafički prikaz lokacije veličine bajt komandom broj 12:

dlb<ENTER>

u nastavku komande kao odgovor na pitanje koja je adresa lokacije čiji se sadržaj crta, korisnik navede adresu #5fh, to je viši bajt registra TMP1, pogledaj tačku c):

5f<ENTER>

zbog velike učestanosti fundamentala koja je zadana sadržajem višeg bajta BX registra na adresi #21h nije moguće razaznati talasni oblik koji je srazmeran indeksu modulacije PWM signala jedne od invertorskih faza, zato treba smanjiti zadatu učestanost upisivanjem nekog manjeg broja u BX registar, recimo broja 5, to se može postići pomoću opcije STIMULUS kao u tački n). Sekvenca korisnikovog upisa preko tastature prati postavljena pitanja posle pritiska na broj 1:

5<ENTER>

21<ENTER>

1000<ENTER>

Talasni oblik indeksa modulacije se može sačuvati na hard disku u vidu ASCII datoteke, koristeći opcije FREEZ i SAVE SCREEN isto kao u tački o) prva kolona datoteke će imati 500 tačaka koje su dobijene periodičnim iščitavanjem višeg bajta registra TMP1 u toku 10 sekundi. Aktivira opciju QUIT pritiskom na broj 2.

- q) Upiše u statusni registar U_F_STATUS (pogledaj tačku c)) broj #3h, time se menja stanje drugog bita (ime drugog bita je SORS_FLAG) kontrolnog registra, sekvenca upisa preko tastature je:

wb<ENTER>

3<ENTER>

5d<ENTER>

sada je brzinu obrtanja motora moguće kontrolisati pomoću potencijometra na kartici označenog sa T-POT.

- v) Deaktivira korisnički program naredbom broj 8:

setb<ENTER>

ova naredba programski resetuje karticu i menja mapu adresnog prostora upisom nule u konfiguracioni bit R/E#. Korisnički program ostaje upisan u RAM kartice sve dok se neisključi napajanje laboratorijske stanice.

- w) Vraća se u DOS naredbom broj 5:

q<ENTER>

- x) Isključuje laboratorijsku stanicu pridržavajući se postupka opisanog u datoteci uputstvo.pdf
- y) Pregleda sadržaj datoteka koje su snimljene u toku rada grafičkog interfejsa.
- z) Obrisuje sve datoteke koje je napravio u toku rada sa laboratorijskom stanicom.

4.2 PRIMER IZVRŠAVANJA KORISNIČKOG PROGRAMA BEZ NADGLEADNOG PROGRAMA

Program koji će poslužiti kao primer je d:\196demo\primer\ukrozfbz.a96. Pretpostavka je da je korisnik napisao ovaj program sa namerom da ga učita u RAM kartice, startuje njegovo izvršavanje i ispita ga koristeći programsku podršku opisanu u 3 poglavlju, u nastavku je opisana sekvenca radnji koje korisnik obavlja u cilju izvršenja prethodnog.

4.2.1 PREVOĐENJE KORISNIČKOG PROGRAMA I FORMIRANJE BINARNE DATOTEKE KOJA SE UČITAVA U RAM KARTICE

Prvi korak po završetku formiranja ASCII asemblerske datoteke korisničkog programa je prevođenje u binarnu datoteku koja se učitava u RAM kartice. Primer koje opcije treba koristiti pri prevođenju korisničkog programa i formiranju binarne datoteke čiji se sadržaj učitava u RAM kartice je "batch" program d:\196demo\primer\ukfbz.bat. Ovaj program prevodi korisnički program ukrozfbz.a96, formira datoteku ukrozfbz.lst u kojoj se nalaze poruke o greškama i tabela simbola (nazivi registara i labela) sa adresama simbola, formira datoteku ukrozfbz.m96 u kojoj se nalaze adrese upotrebljenih memorijskih lokacija u korisničkom programu (korisnik pregleda ovaj program da bi utvrdio da li se njegov segment registara i segmenti koda i podataka ne međusobno nepreklapaju), na kraju "batch" program formira datoteku ukrozfbz.bin koja se upisuje u RAM kartice. Sadržaj ovog "batch" programa je:

```
ASM96 ukrozfbz.a96 nodb  
RL96 ukrozfbz.OBJ TO ukrozfbz.ABS  
OH ukrozfbz.abs TO ukrozfbz.hex  
hexbin ukrozfbz.hex ukrozfbz.bin I
```

4.2.2 REDOSLED POSTUPAKA KORISNIKA PRI RADU SA LABORATORIJSKOM STANICOM U CILJU ISPITIVANJA SVOG PROGRAMA

- a) Korisnik uključi laboratorijsku stanicu pridržavajući se postupka opisanog u uputstvu na serveru Kiklop datoteka x:\nastava\eg4ev\uputstvo.pdf, i uključi PC računar koji je u sklopu stanice (korisniku koji prvi put radi sa laboratorijskom stanicom pomaže asistent određen od strane profesora Slobodana Vukosavića), po uključenju stanice, signalizacione diode na mikroprocesorskoj kartici počinju da svetle na način opisan u 3 poglavlju.
- b) Korisnik se pozicionira u direktorijum d:\196demo\primer\ i startuje program ukfbz.bat opisan u poglavlju 4.2.1 radi formiranja datoteke ukrozfbz.bin, koja se učitava u RAM kartice. Od ove tačke se zahteva samostalan rad korisnika.
- c) Po završetku rada programa ukfbz.bat, otvori datoteku ukrozfbz.lst komandom:

edit ukrozfbz.lst <ENTER>

pozicionira se na kraj datoteke i primeti poruku o greškama u toku prevodenja, u ovom slučaju pošto nema grešaka poruka će biti:

ASSEMBLY COMPLETED, NO ERROR(S) FOUND

Kako je ovaj program pisan da radi samostalno bez mogućnosti rada nadglednog programa, onda nije potrebno pregledati tabelu simbola i beležiti adrese promenljivih koje program koristi.

- d) Korisnik otvori datoteku ukrozfbz.m96 u kojoj se nalaze adrese upotrebljenih memorijskih lokacija od strane korisničkog programa (korisnik pregleda ovu datoteku da bi utvrdio da li se njegov segment registara i segmenti koda i podataka ne preklapaju međusobno i da li ne zalaze u lokacije rezervisane za nadgledni program) komandom:

edit ukrozfbz.m96 <ENTER>

Pozicionira se na kraj datoteke i primeti poruku o greškama u toku povezivanja programa, u ovom slučaju pošto nema grešaka poruka će biti:

RL96 COMPLETED, 0 WARNING(S), 0 ERROR(S)

- e) Primeti postojanje datoteke ukrozfbz.bin
- f) Aktivira prilagodni program komandom:

serial com2 <ENTER>

- g) Zada komandu za učitavanje korisničkog programa u RAM memoriju kartice, to je komanda broj 4 sa liste komandi:

wf<ENTER>

posle čega se na ekranu ispiše poruka:

Navedi ime fajla u *.bin formatu, početna adresa upisa je #8000h:

u nastavku poruke se dopiše ime binarne datoteke koja se upisuje u RAM:

ukrozfbz.bin<ENTER>

u toku upisa datoteke u RAM, nadgledni program vrši proveru ispravnosti upisa i na ekran šalje niz poruka koje obavestavaju korisnika o stanju serijske komunikacije, i pali svetleću diodu L3 dok su ostale ugašene što znači da je transfer u toku. Po završetku transfera pojavljuje se poruka da je transfer završen i da je sistem spreman za zadavanje sledeće komande.

- h) Sada je potrebno promeniti mapu adresnog prostora, to se postiže komandom broj 3:

rstb<ENTER>

primetite da je posle zadavanja ove komande svetleća dioda L4 upaljena, to znači da je stanje konfiguracionog bita R/E#=1, posledica ove komande je početak izvršavanja korisničkog programa upisanog u RAM, od ove tačke korisnik nema mogućnost kontrole izvršavanja svog programa preko PC računara.

- i) Podesi autotransformator koji je u sklopu laboratorijske stanice na 200 podeoka, vodite računa autotransformator nesmete da podesite preko 220 podeoka, zbog reagovanja prenaponske zaštite!!!
Resetuje prekostrujnu zaštitu invertora pritiskom na crveni taster koji se nalazi na ivici kartice prekostrujne i prenaponske zaštite K7.

- j) Korisnik može da proveri ispravnost rada programa:

- posmatranjem signala za uključenje tranzistora pomoću osciloskopa (korisnik priključuje sonde na odgovarajuće tačke koje se nalaze na upaljačkim karticama, pogledaj uputstvo.pdf);
- okretanjem potencijometra T-POT preko koga se zadaje učestanost fundamentalna PWM signala, na taj način se kontroliše brzina obrtanja motora.

- k) Deaktivira korisnički program pritiskom na reset taster mikroprocesorskog sistema, time se menja mapa adresnog prostora, tj. u R/E# se upisuje nula.

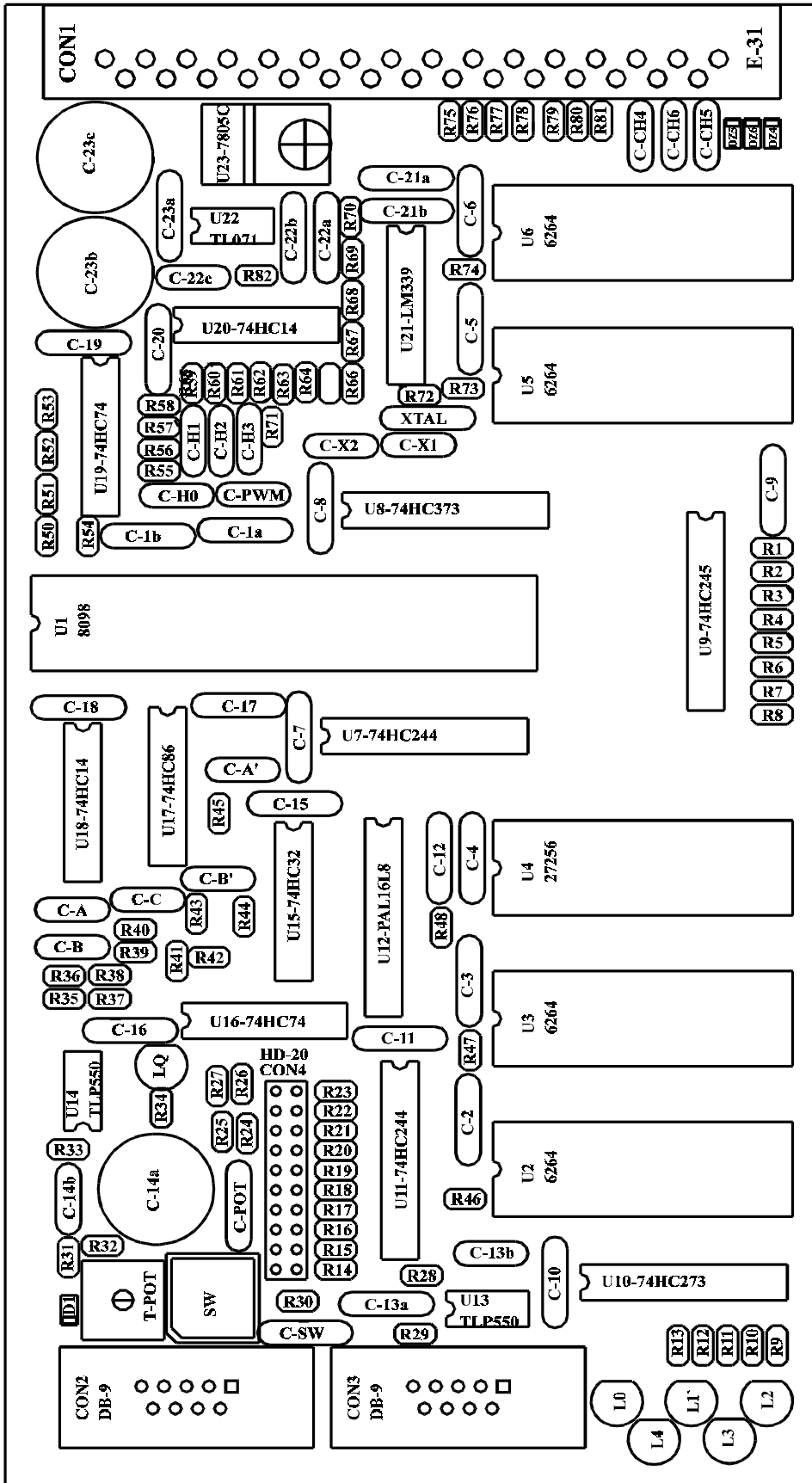
- l) Vrati se u DOS naredbom broj 5:

q<ENTER>

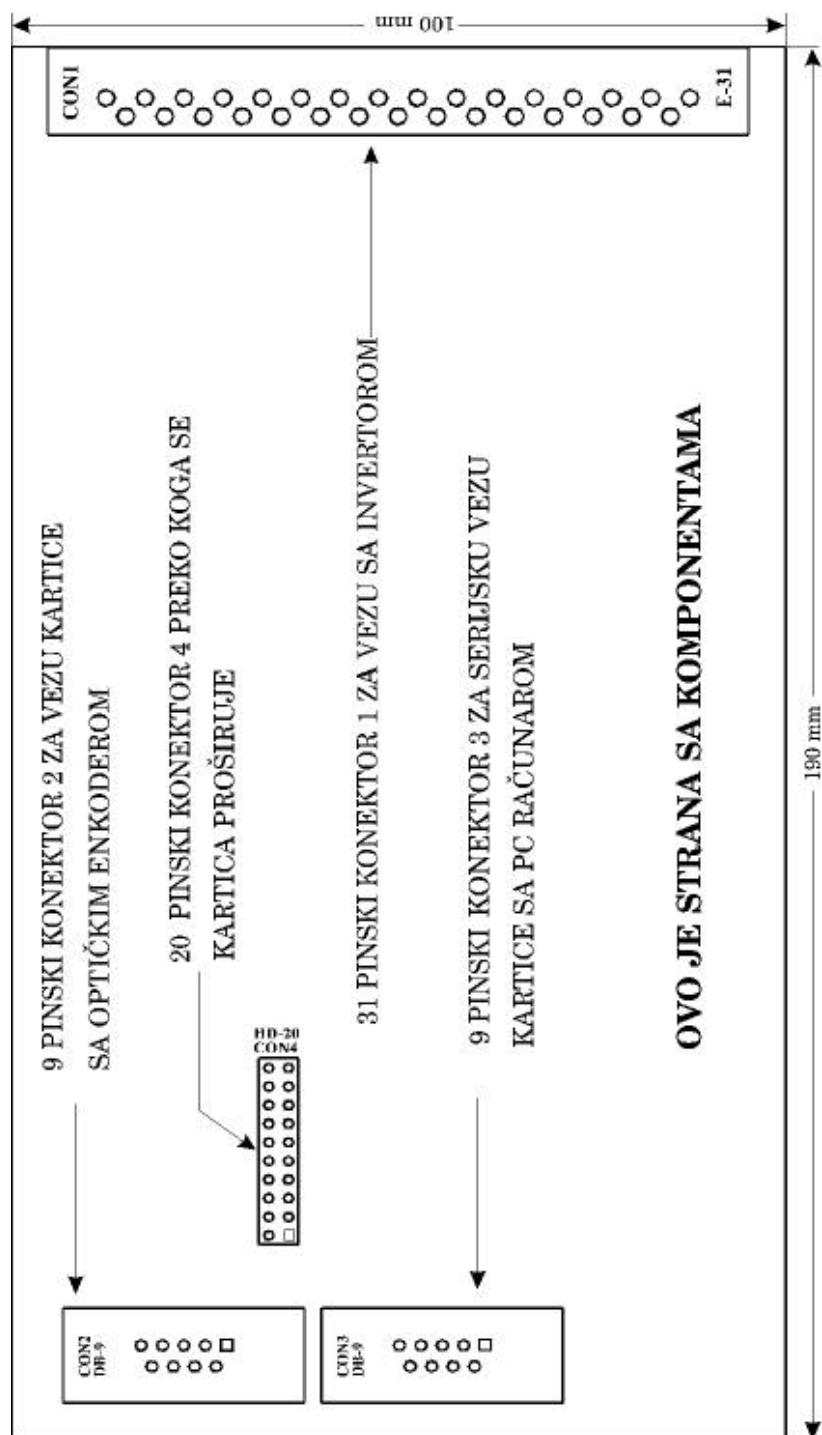
- m) Isključi laboratorijsku stanicu pridržavajući se postupka opisanog u datoteci uputstvo.pdf

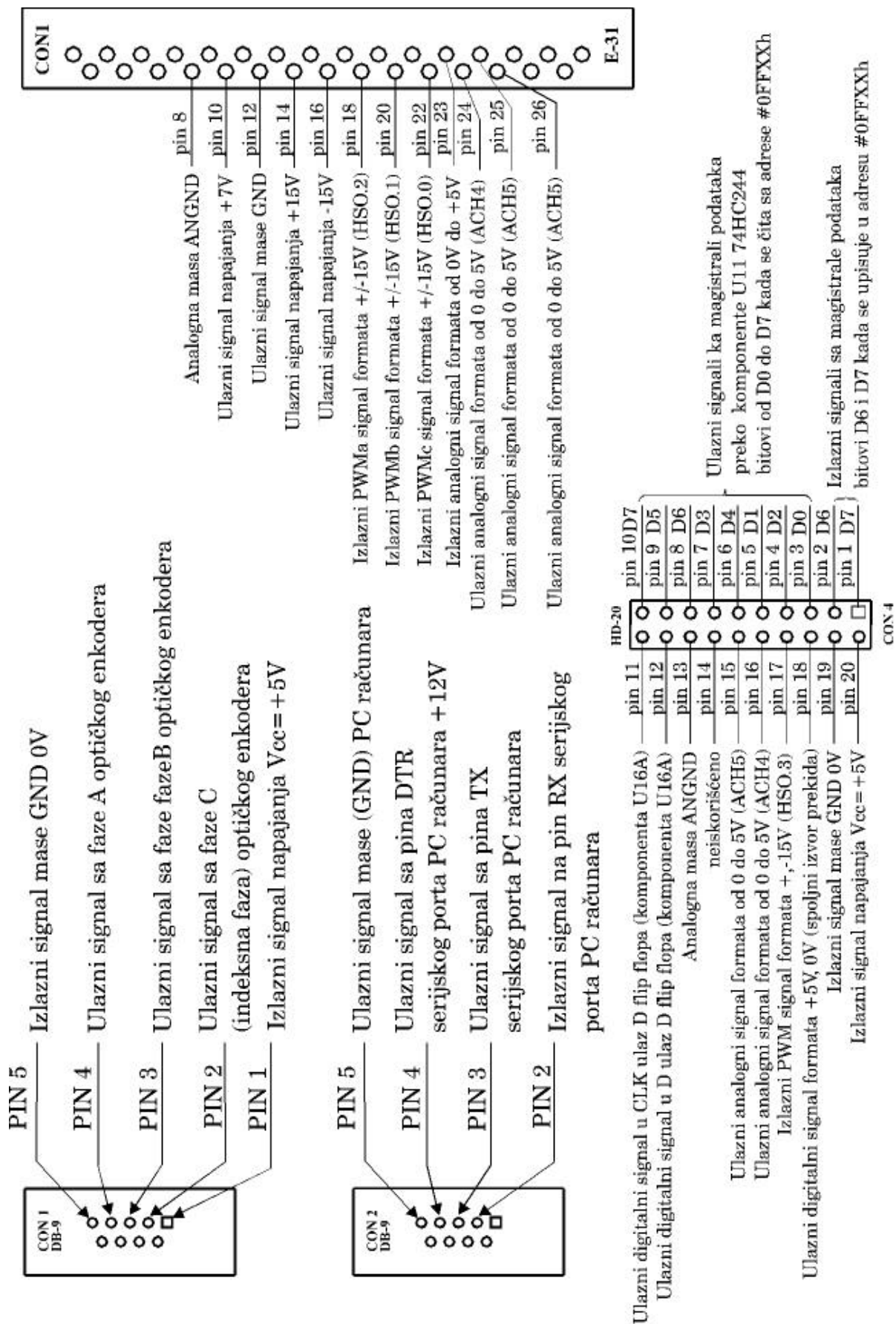
- n) Obriše sve datoteke koje je napravio u toku rada sa laboratorijskom stanicom.

> SKICA KARTICE



> RASPORED KONEKTORA





> SPISAK DELOVA UGRAĐENIH U KARTICU

BR.	DEO	KOMADA	OZNAKA
1.	Podnožje za procesor dip-48	1	
2.	Podnožje za memorije dip-28	5	
3.	Podnožje dip-20	6	
4.	Podnožje dip-14	7	
5.	Podnožje dip-8	3	
6.	Konektor E-31 muški (print)		CON1
7.	Konektor DB-9 ženski (print)	2	CON2, CON3
8.	MC8098-12	1	U1
9.	RAM memorija 6264	4	U2, U3, U5, U6
10.	EPROM memorija 27256	1	U4
11.	74HC244	2	U7, U11
12.	74HC373	1	U8
13.	74HC245	1	U9
14.	74HC273	1	U10
15.	PAL16L8	1	U12
16.	Optokapler TLP550	2	U13, U14
17.	74HC32	1	U15
18.	74HC74	2	U16, U19
19.	74HC86	1	U17
20.	74HC14	2	U18, U20
21.	LM339	1	U21
22.	TL071	1	U22
23.	7805C-TO220	1	U23
24.	Trimer potencijometar, vertikalni (print)	1	T-POT
25.	Reset taster, vertikalni (print)	1	SW
26.	Kristal 12 MHz	1	XTAL
27.	Konektor HEADER-20pinski	1	CON4
28.	LED dioda zelena	5	L0, ..., L4
29.	LED dioda crvena	1	LQ
30.	Zener dioda 5V1	3	DZ-4, 5, 6
31.	Dioda 4148	1	D1
32.	Keramički kondenzator 100 nF	24	C-1a, C-2, ..., C-12, C13a, C-15, ..., C-20, C-21a, C-21b, C-22a, C-22b, C-23a.
33.	Kondenzator 15 pF	4	C-14b, C13-b , C-A', C-B'.
34.	Kondenzator 220 pF	3	C-A, C-B, C-C.
35.	Kondenzator 470 pF	4	C-H3, C-H2, C-H1, C-H0.

36.	Kondenzator 220 nF	3	C-POT, C-SW, C-22C
37.	Kondenzator 6n8	3	C-CH4, C-CH5, C-CH6.
38.	Elektrolit 22 μ F	3	C-14a, C-23b, C-23c.
39.	Elektrolit 1 μ F	1	C-1b
40.	Kondenzator 33 pF	3	C-PWM, C-Xtal1, C-Xtal2.
41.	Otpor 10 k Ω	22	R1, ..., R8, R44, R46, R47, R52, R53, R65, ..., R70, R73, R74
42.	Otpor 1 k Ω	12	R9, ..., R13, R34, ..., R40
43.	Otpor 100 Ω	32	R14, ..., R27, R41, R42, R48, R51, R54, ..., R64, R72, R78
44.	Otpor 2k2	8	R28, R32, R79, R80, R81, R75, R76, R77, R71.
45.	Otpor 470 Ω	1	R29
46.	Otpor 330 k Ω	1	R30
47.	Otpor 47 Ω	1	R31
48.	Otpor 3k3	1	R33
49.	Otpor 100 k Ω	3	R43, R45, R82
50.	Otpor 220 Ω	1	R50

> ŠEME MIKROPROCESORSKOG SISTEMA

SADRŽAJ:

- 1 Ova strana;
- 2 Interfejs sa HSI i serijskim urđajem;
- 3 Reset kolo, napajanje i konektor za proširenje;
- 4 HSO izlaz, analogni ulazi, analogni izlaz i glavni konektor;
- 5 Mikroprocesor, programabilana logika i logika za promenu adresnog prostora;
- 6 Memorija.

