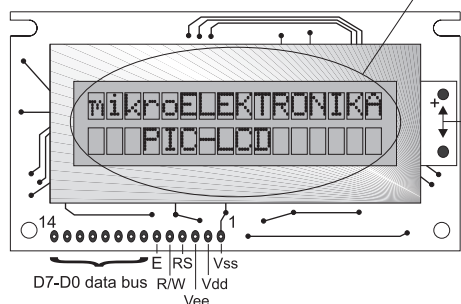


## Povezivanje LCD-a

Postoji više vrsta ovih displeja, ali najviše se koriste matrični tačkasti sa jednim, dva ili četiri reda. Mogu imati pozadinsko (*Back light*) osvetljenje, a neki rade i u grafičkom režimu. Obično imaju 14 priključnih pinova (slika 1.) od kojih su dva za napajanje ( $V_{dd}$ ,  $V_{ss}$ ), jedan za podešavanje kontrasta ( $V_{ee}$ ), tri su kontrolna, a preostalih osam su data bitovi. Na slici 1. je prikazan dvoredni smart LCD sa 16 matrica u jednom redu.

Upotreba pametnog LCD-a je jednostavna. Postoji dvosmerni data bus (D7 – D0), tri kontrolna pina E, RW i RS. Generisanjem opadajuće ivice na kontrolnom pinu E (*enable*) omogućava se pristup LCD-u. Pomoću RW (*read/write*) pina kontroliše se upis ili čitanje (RW=1 za čitanje, RW=0 za upis). Treći kontrolni RS pin određuje da li se obraćamo instrukcijskom ili data registru. Upisom podataka u instrukcijski registar (RS=0) definišemo mesto pojave karaktera na LCD-u, možemo obrisati ceo ekran, postaviti kursor na početak DDRAM-a itd. Kada je RS=1, vrši se upis ili čitanje DDRAM-a (*Display Data RAM*) ili CGRAM-a (*Character Generator RAM*). Upisom u DDRAM podaci se prikazuju na ekranu LCD-a. CGRAM sadrži matrične tačkaste prikaze karaktera koji se mogu prikazati na LCD-u. Sadržaj CGRAM-a je



**Sl.1** Dvoredni smart LCD sa 16 matrica u jednom redu.

moguće softverski menjati, tako da se mogu prikazati karakteri po želji korisnika. Na slici 1. prikazane su adrese vidljivih mesta na LCD-u na kojima se ispisuju karakteri, a koje se upisuju u instrukcijski registar. Primećuje se da drugi red ne počinje tamo gde se prethodni završava. U stvari u produžetku reda postoje mesta za ispis koja se ne vide i koriste se ako želimo da prikazemo neku poruku u pomeračkom (*shift*) modu. Tako npr. dvoredni displej sa po 16 vidljivih karaktera ima još 24 nevidljiva.

### Povezivanje Smart LCD-a sa mikrokontrolerom PIC16F84

Za povezivanje pametnog LCD-a, na prvi pogled potrebno je odvojiti 11 pinova nekog mikrokontrolera. Ako se opredelimo za Microchip-ov PIC16F84 koji ima svega 13 I/O linija (slika 2.), posle povezivanja sa LCD-om ostaju nam dve I/O linije sa kojima možemo uraditi. Ipak, pametni LCD je moguće povezati tako da ne koristimo svih osam linija data bus-a već četiri. Tako smo broj potrebnih linija sveli na 7 i podaci se ka displeju sada šalju u dva paketa po četiri bita. Ako se LCD koristi samo za prikaz poruka pin RW se može povezati na masu i broj potrebnih pinova je tako smanjen na 6. Primer povezivanja

U poslednje vreme sve je više mikrokontrolerskih uređaja koji za vizuelizaciju svog rada koriste pametne "SMART LCD" displeje sa tečnim kristalom. Razlozi tome su velika količina informacija koja se može prikazati i mala potrošnja, reda mA, te su nezamenjivi u uređajima sa baterijskim napajanjem.



smart LCD-a i mikrokontrolera PIC16F84 dat je na Slici 3. U vezi korišćenja RW pina treba nešto napomenuti. Naime, da bi "regulamo" koristili LCD potrebno je ispitivati BUSY bit (koji je ustvari D7 bit data bus-a LCD-a). Kada LCD spusti BUSY bit na "0" to znači da je završio sa prethodnim poslom i da mu se može "uspešno" pristupiti. Da bi proverili stanje B U S Y b i t a m o r a m o

koristiti mod isčitavanja tj. potreban nam je RW pin. Međutim to se može rešiti na zadovoljavajući način tako što se umesto proveravanja BUSY-ja ubacuju softverska kašnjenja dovoljno duga da za to vreme LCD može da obavi prethodni posao.

### Test program

Listing programa za mikrokontroler PIC16F84 koji je dat u okviru ovog članka koristi 4-bitni data interfejs i softverska kašnjenja umesto ispitivanja BUSY bita. Data bitovi LCD-a su respektivno povezani na pinove RB7, RB6, RB5, RB4. Pin E je na pinu RB3, a pin RS na pinu RB2. Ovim načinom povezivanja port A je ostao neiskorišćen, a na portu B ostala su "slobodna" još dva pina: RB0/INT kojim je moguće generisati spoljni interapt i RB1 koji možemo koristiti kao običnu I/O liniju. Program je prenosiv na sve PIC mikrokontrolere, samo na početku programa treba "uključiti" odgovarajuću INC datoteku koja se dobija uz MPASM.

Osnovu programa čine podprogrami `Send_Char` i `Send_Cmd`. Ove dve rutine su u osnovi iste, obe šalju podatak ka LCD

```
#include LCD.H ; Uključuje heder datateku LCD.H
#include PIC16F84.inc ; Uključuje datoteku u kojoj su definisane
; memorijske lokacije mikrokontrolera

org 0x00 ; Početak programa

page1
clrf TRISB ; PORTB je izlazni
page0
clrf PORTB
clrf TEMP_W
call Init ; Inicijalizacija LCD-a
movlw 0 ; Početak Look-up tabele
```

*;Glavni program*

```
Poruka movwf TEMP_W ; Sačuvaj sadržaj W-registra
call Tabela ; Uzmi karakter iz tabele
andlw 0xFF
btfsc STATUS, Z ; Da li je kraj poruke
goto Loop ; Ako jeste idi na Loop
call Send_Char ; Pošalji karakter na LCD
L1 movf TEMP_W, W ; Vрати sadržaj u W-registar
addlw 1 ; Uvećaj ga za 1
goto Poruka ; Idi na početak glavnog programa

Loop goto Loop ; Ostani ovde
```

*;Podprogram za prelazak u drugi red*

```
II_Red movlw 0xC0 ; Početno mesto drugog reda
call Send_Cmd
goto L1 ; Vрати se u glavni program
```

*;Podprogram za slanje karaktera na LCD (RS=1)*

*;Karakter koji se šalje na LCD nalazi se u W registru*

```
Send_Char movwf CHAR ; Sačuvaj sadržaj W registra
call Delay ; Trostruko kašnjenje zbog BUSY-a
call Delay
call Delay
movf CHAR, W ; Vрати sadržaj u W registar
andlw 0xF0 ; Maskiraj donja četiri bita
movwf PORTB ; Pošalji podatak na LCDport
bsf PORTB, RS ; Selektuj Data registar RS=1
nop
bsf PORTB, E ; Generisanje opadajuće ivice na pinu E
nop
bcf PORTB, E
swapf CHAR, W ; Zamena gornja i donja četiri bita
andlw 0xF0 ; Maskiraj donja četiri bita
movwf PORTB ; Pošalji podatak na LCDport
bsf PORTB, RS ; Selektuj Data registar RS=1
nop
bsf PORTB, E ; Generisanje opadajuće ivice na pinu E
nop
bcf PORTB, E
return ; Povratak iz podprograma
```

*;Podprogram za slanje komandi LCD-u (RS=0)*

```
Send_Cmd movwf CHAR
call Delay
call Delay
call Delay
movf CHAR, W
andlw 0xF0
movwf PORTB
bcf PORTB, RS ; Selektuj Instrukcijski registar RS=0
nop
bsf PORTB, E
nop
bcf PORTB, E
swapf CHAR, W
andlw 0xF0
movwf PORTB
bcf PORTB, RS ; Selektuj Instrukcijski registar RS=0
nop
bsf PORTB, E
nop
bcf PORTB, E
return
```

*;Podprogram za inicijalizaciju LCD-a*

```
Init movlw low_bit_2reda_5x7 ; Trostruka inicijalizacija zbog
call Send_Cmd ; mogućeg neuspešnog reseta LCD-a
movlw low_bit_2reda_5x7
call Send_Cmd
movlw low_bit_2reda_5x7
call Send_Cmd
movlw DisplayON_cursorOFF
call Send_Cmd
movlw Display_clear
call Send_Cmd
movlw Increment_mod_freeze_display
call Send_Cmd
return
```

```
Delay movlw 0xFF
movwf TEMP
L2 nop
decfsz TEMP, F
goto L2
return
```

*;Look-up tabela u kojoj se nalazi poruka za ispis*

```
Tabela addwf PCL, F
retlw 'm' ; Početak Look-up tabele
retlw 'i'
retlw 'k'
retlw 'r'
retlw 'o'
retlw 'E'
retlw 'L'
retlw 'E'
retlw 'K'
retlw 'T'
retlw 'R'
retlw 'O'
retlw 'N'
retlw 'I'
retlw 'K'
retlw 'A'
goto II_Red ; Prelazak na drugi red
retlw ' '
retlw ' ' ; Razmak
retlw 'P'
retlw 'I'
retlw 'C'
retlw '-'
retlw 'L'
retlw 'C'
retlw 'D'
retlw 0 ; Kraj poruke

end ; Kraj programa
```

#### LCD.H

```
#define page0 bcf 3,5
#define page1 bsf 3,5
```

#### ;Definicija memorijskih lokacija

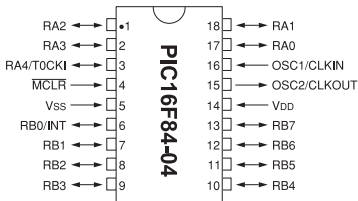
```
TEMP_W EQU 0x0C
CHAR EQU 0x0D
TEMP EQU 0x0E
```

#### ;Definicija pojedinih bitova

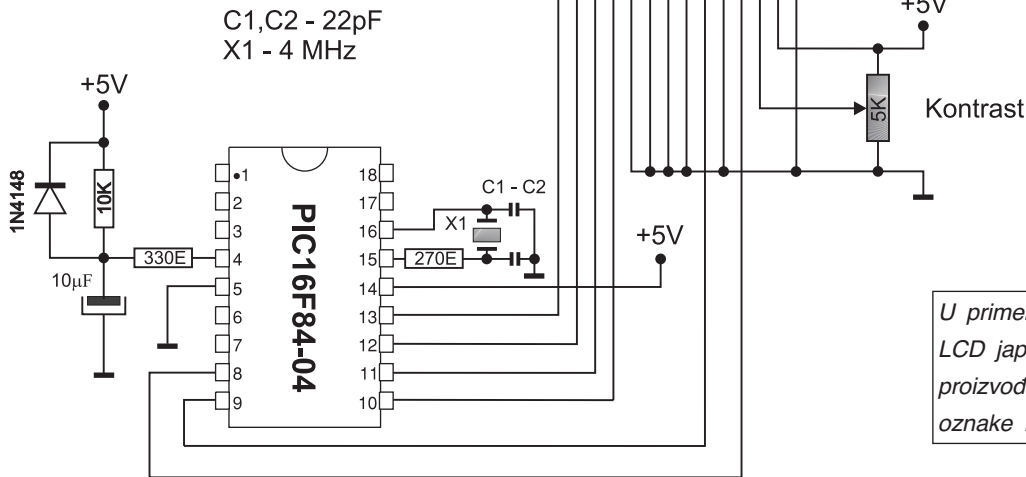
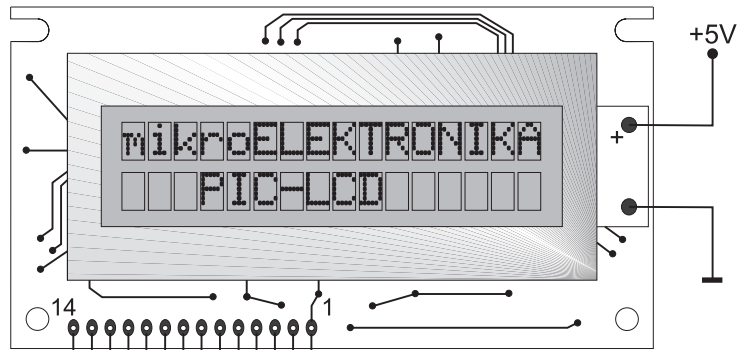
```
RS EQU 2
E EQU 3
```

#### ;Definicija konstanti

```
low_bit_2reda_5x7 EQU 0x28
Shift_display EQU 0x18
DisplayON_cursorOFF EQU 0x0C
DisplayOFF_cursorOFF EQU 0x08
DisplayON_cursorON EQU 0x0F
Increment_mod_freeze_display EQU 0x03
Display_and_cursor_home EQU 0x02
Display_clear EQU 0x01
```



**SI.2 Raspored izvoda mikro-kontrolera PIC16F84**



U primeru je korišćen LCD japanskog proizvođača SHARP oznake LM16X212

**SI.3 Povezivanje mikrokontrolera i PIC-a**

portu (port B) u vidu dva 4-bitna paketa, jedina razlika je u tome što se prvom rutinom obraćamo data registru, a drugom rutinom instrukcijskom registru. Prilikom reseta mikrokontrolera prvo se vrši inicijalizacija LCD-a pozivom podprograma **Init**. Prvo se izvrši setovanje 4-bitnog data interfejsa (*low\_bit\_2reda\_5x7*) i nakon toga sledi niz komandi kao što je uključivanje displeja i po potrebi kursora (*DisplayON\_CursorOFF*), brisanje ekrana i postavljanje kursora na početak DDRAM-a (*Display\_clear*), setovanje Increment freeze moda u kom se karakteri mogu pojavljivati samo na vidljivim mestima LCD-a s leva u desno (*Increment\_mod\_freeze\_display*).

Komande koje se šalju LCD-u definisane su u heder datoteci LCD.H i mogu se menjati i dodavati po želji korisnika, a u skladu sa uputstvom proizvođača. Nakon inicijalizacije displeja sledi ispisivanje karaktera koje glavni program (**Poruka**) uzima iz look-up tabele. U ovom slučaju u prvom redu ispisuje **mikroELEKTRONIKA**, a u drugom redu posle tri razmaka **PIC-LCD**. Program se "vrti" u petlji sve dok iz tabele ne izvuče vrednost "0" tj. kraj poruke, tada program "skače" na lokaciju **Loop** i tu ostaje dok ponovo ne resetujemo mikrokontroler.

**Dragan Andrić**

**Kontakt:**

Za sva pitanja u vezi teksta obratite se autoru na telefon redakcije svakog radnog dana između 18-20h



**POLAROTOR**

**Emisija o novim tehnologijama**

**3K RTS**

**Četvrtak 17 sati**

**POLAROTOR**

**Milentija Popovića 9**

**11070 Novi Beograd**

**E-mail: polarotor@beotel.yu**

**www.beotel.net\korisnici\polarotor\index.html**