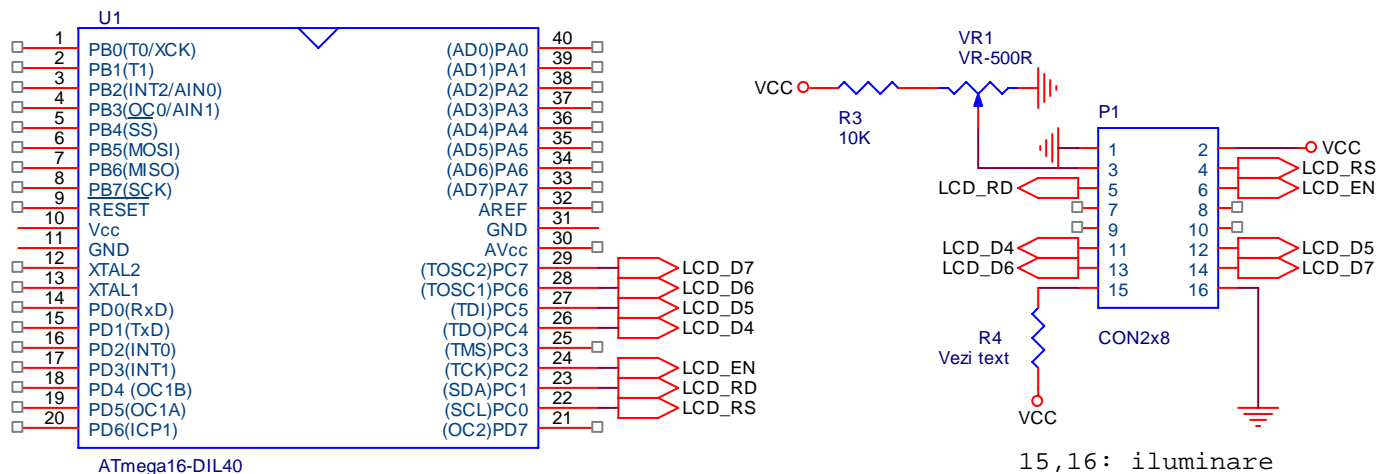


Folosirea afisajelor LCD alfanumerice
rev. 2
M. Stanciu, 2009



Conectarea unui afisaj LCD alfanumeric la microcontroller este foarte simpla, dupa schema de mai jos:



De notat ca pe piata exista un mare numar de LCD-uri, de la diversi producatori, cu dimensiuni diferite (de la o linie la 4 linii, de la 8 la 20 sau mai multe caractere pe linie) si preturi de la cca. 5 dolari in sus; din fericire, toate sint bazate pe un controller HD44780, KS0070 sau echivalent, de aceea se programeaza la fel si se conecteaza aproape la fel la AVR. Gasiti pe site datasheet-ul pentru un modul reprezentativ.

Interfata de conectare se bazeaza pe 4 linii de date: D4..D7 si 3 linii de control: RS, RD si EN. Practic, mai exista si liniile de date D0..D3 dar cum LCD-ul suporta controlul pe doar 4 linii, se prefera aceasta varianta pentru a economisi 4 pini de procesor. Conectarea a 8 linii in loc de 4 aduce avantajul dublarii vitezei de acces, dar aceasta nu este critica in cele mai multe aplicatii, in care informatiile se afiseaza relativ rar (puteti sa o folositi daca aveti o aplicatie de viteza, dar va trebui sa folositi o alta biblioteca decit cea standard din CvAVR).

Atentie, inainte sa conectati LCD-ul: in figura este desenat un conector *generic* pentru LCD – cel mai des intilnit. De notat ca exista mai multe variante, in care difera de obicei ordinea pinilor de alimentare (cel mai des 1=GND si 2=Vcc dar am gasit si invers!), pe cite rinduri sint – fie pe 2 rinduri in lateralul modului, fie pe 1 rind pe latura mai lunga - precum si pozitia si ordinea pinilor pentru alimentarea iluminarii (backlight). De aceea, inainte de a cabla conectorul, este *esential* sa consultati datasheet-ul modului disponibil. **Conectarea pe dos a alimentarii duce la arderea LCD-ului!**

Daca nu aveti datasheet-ul exact, o *identificare simpla a pinului de masa* este urmatoarea: folosind un ohmetru, se masoara care dintre pini 1, 2 este legat electric la partea metalica a LCD-ului. De obicei exista

niste “urechi” de tabla care fac contact electric pe niste pad-uri de pe spatele LCD-ului. In 99% din cazuri partea metalica este conectata la masa, deci asa identificati pinul de masa.

Daca constatati că urechile nu fac contact cu nici unul din pinii 1,2, înseamnă că fabricantul LCD-ului a lăsat la alegerea utilizatorului dacă masa va fi conectată la partea metalica sau nu. Pentru aceasta, exista o pereche de pad-uri (dreptunghiulare sau rotunjite) pe care se lipește o rezistență SMD de 0 ohmi dacă se dorește conectarea masei la partea metalică, dar rezistența nu e lipita din fabrică. Se identifică aceste paduri (căutați un traseu care se leagă vizibil la un pad de sub o ureche) și se vede la care din bornele 1,2 se leagă prin acea rezistență de 0 ohmi – aceea este masa.

Verificarea faptului ca s-a identificat corect se face astfel: se conecteaza pinii Vcc si GND deja identificati la sursa de 5V, si contrastul (pinul V0 sau VLCD, de obicei 3) se leaga la masa; restul pinilor se lasa neconectati. Daca totul e OK, pe LCD se va aprinde linia de sus sub forma unor patrate pline, iar linia de jos va fi goala. Acesta este modul “default” de afisare a controllerului din LCD (comanda “init”, primita de la procesor, va sterge ecranul). Dacă nu apare nimic, deconectați imediat LCD-ul, e posibil să nu se fi ars încă.

Pentru **iluminare**: In functie de fabricant, exista sau nu backlight; aici diferentele sint cele mai mari, uneori pinii de backlight sint complet separati si ramin 14 pini pe conectorul de baza. Culoarea si tensiunea de alimentare sint specificate in datasheet. Iluminarea se face cel mai des cu LED-uri; caderea de tensiune si curentul depind de combinatia serie/paralel si numarul de leduri. Daca nu este clar in datasheet, puteti verifica in felul urmator:

- se verifica ca pe LCD, de obicei intr-o extremitate, mai departe de conectorul de 16 pini, sint marcate explicit 2 pad-uri “A” si “K” semnificand anodul (+) si catodul (-) iluminarii. Cu un ohmetru, se identifica care dintre cele 2 (de obicei K), este legat la 15 si 16. Pe schema mea este 16. In felul acesta s-a identificat catodul/GND. Daca nu exista nici o conexiune, vezi punctul 3 mai jos.
- tot cu ohmetrul se verifica daca intre celalalt pin de pe conector si pad-ul “A” este o rezistenta diferita de 0. In functie de rezultat se procedeaza astfel:
 1. Daca rezistenta este de ordinul zecilor de ohmi, cel mai probabil este ca este deja dimensionata pentru alimentarea iluminarii la 5V. Se conecteaza pinul respectiv (in exemplul meu 15) la 5V, celalalt la GND si lumina ar trebui sa se aprinda
 2. Daca rezistenta este aproape de 0 ohm inseamna ca producatorul a scos la conector direct anodul si catodul si a lasat utilizatorul sa dimensioneze rezistenta serie. Atunci se conecteaza catodul la GND si o rezistenta serie (R4 pe schema mea) intre 5V si pinul respectiv (15), prin tatonarea valorii. Se incerca mai intii o rezistenta de 100 ohm, daca iluminarea e foarte slaba (sau deloc) se scade la 80, 50 etc. Ne oprim cind iluminarea este convenabila. Daca producatorul specifica un curent maxim pt. iluminare, se verifica ne-depasirea lui cu un ampermetru. De obicei trebuie un curent ceva mai mic decit cel maxim specificat.
 3. Daca rezistenta este infinita, atunci este probabil ca nu exista iluminare. Modulele se fac folosind un singur tip de cablaj (ca sa fie ieftin) si se insereaza sau nu panoul cu LED-uri si se lipeste sau nu rezistenta serie; deci un modul care se vinde fara acest panou are in continuare marcajele A, K dar ele nu sint legate la nimic. La Comet Electronic, codul modulelor care au Backlight contine și prescurtarea B/L în descrierea produsului.

Semnificatia pinilor este:

- Vcc (sau Vdd), GND (sau Vss): alimentarea modulului (5V).
- V0: reglajul contrastului; valoarea de 0V inseamna contrast maxim si deci cea mai ieftina metoda este de a lega V0 la GND. Totusi, pentru citirea usoara se recomanda conectarea unui potentiometru ca in schema; rezistenta R3 pusa in serie cu potentiometrul are rolul de a asigura ca intreaga plaja de rotire a potentiometrului este in zona de tensiuni mici (practic, nu are rost sa aplicam mai mult de circa 0.5V intrucit peste aceasta valoare nu se mai vede nimic pe afisaj).

- D0..D7 = magistrala de date pe care se transmit comenzi spre si dinspre LCD. De obicei este suficienta folosirea niblului superior D4..D7, in care caz fiecare octet este transmis ca 2 nibluri succesive
- RS = Register Select; specifica semnificatia octetului care urmeaza: RS=0 inseamna instructiune si RS=1 inseamna date
- RD sau R/W = Read/Write; 1 inseamna operatie de citire si 0 inseamna scriere; in mod normal scrierea este cel mai des folosita (scopul principal este sa afisam informatii pe LCD), dar trebuie si citit cel putin cuvintul de stare al afisajului pentru a vedea daca este gata sau nu sa primeasca comenzi.
- EN = Enable; se comanda inainte de a transmite date/comenzi.

Conectarea la AVR se face ca in schema; se poate folosi alt port in afara de C, dar indiferent de portul folosit, trebuie respectata ordinea pinilor din port (adica LCD_RS va fi PC0, PB0, etc dar nu la PB7!).

Programarea afisajului este simpla, intrucit pentru aproape orice compilator si orice limbaj se gasesc pe internet librarii de functii deja facute, care permit scrierea pe afisaj, stergerea acestuia, si eventual modificarea setului de caractere, fara a ne mai preocupa de modul de functiile low-level si de comanda pinilor de control enumerati mai sus. Compilatorul CodeVision AVR are si el functii incluse; extras din help-ul lui CVAVR:

```
unsigned char lcd_init(unsigned char lcd_columns)
```

initializes the LCD module, clears the display and sets the printing character position at row 0 and column 0. The numbers of columns of the LCD must be specified (e.g. 16). No cursor is displayed.

The function returns 1 if the LCD module is detected and 0 if it is not.

This is the first function that must be called before using the other high level LCD Functions.

```
void lcd_clear(void)
```

clears the LCD and sets the printing character position at row 0 and column 0.

```
void lcd_gotoxy(unsigned char x, unsigned char y)
```

sets the current display position at column x and row y. The row and column numbering starts from 0.

```
void lcd_putchar(char c)
```

displays the character c at the current display position.

```
void lcd_puts(char *str)
```

displays at the current display position the string str, located in SRAM.

```
void lcd_putsf(char flash *str)
```

displays at the current display position the string str, located in FLASH.

In program trebuie definit la ce port (PORTA, PORTB, PORTC, etc) este conectat LCD-ul; ordinea pinilor trebuie sa fie ca pe figura (unde este exemplificat PORTC). Se observa ca pinul 3 al portului ales este nefolosit.

Pentru mai multe detalii consultati programul din directorul de instalare `\cvavr\examples\lcddemo`, precum si helpul online.