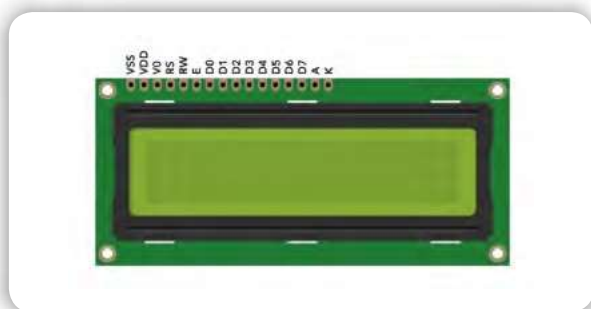


Geekcreit UNO R3 starter kit (5) - LCD in LM35

V tem poglavju bomo pokazali, kako povezati alfanumerični displej (LCD) z Arduino UNO ploščico in kako ga izkoristiti za prikaz napetosti in temperature, ki jo bere temperaturni senzor LM35.

Alfanumerični displej

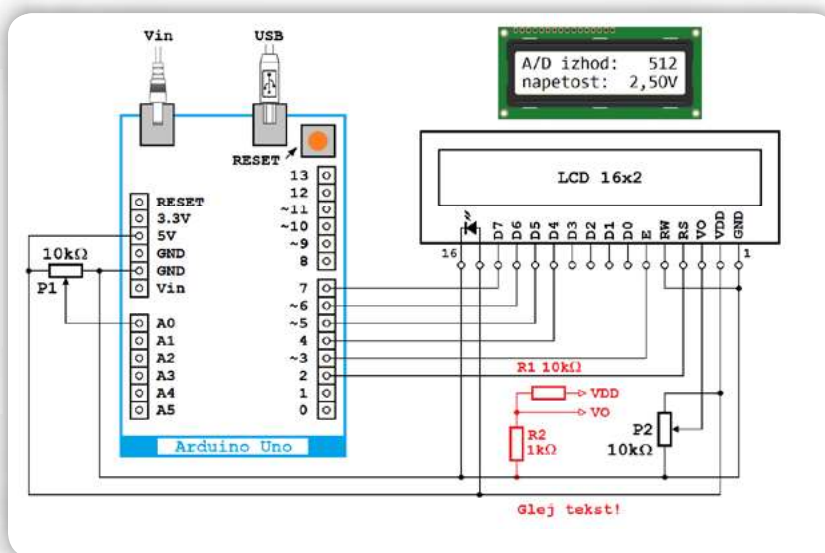


Slika 31: LCD displej iz Geekcreit kompleta.

Slika 31 prikazuje alfanumerični displej iz Geekcreit kompleta. Displej je vrste 16x2, kar pomeni, da lahko prikaže dve vrstici teksta s po 16 znakov v vsaki vrstici. Vsak znak se formira v matriki dimenzij 5x8 pikselov, v internem spominu so pa shranjene črke angleške abecede, številke 0-9, ločila in drugi znaki, ki jih običajno srečamo na tipkovnicah osebnih računalnikov, nekaj grških in nekaj sto kitajskih črk. LCD ima lasten kontroler, zato nam ni potrebno skrbeti za multipleksiranje in z njim je na splošno precej enostavneje delati, kot z LED displeji, ki smo spoznali v predhodnem poglavju.

LCD bomo z Arduino UNO ploščico povezali preko 6-bitnega vodila, po shemi na sliki 32. Linije D7-D4 prenašajo podatek ali ukaz, E in RS pa sta krmilni liniji. Na Arduino strani lahko vodilo povežemo na katere koli od 6 priključkov - mi smo izkoristili priključke 2-7. (Opomba: obstaja tudi drugi načini povezovanja, ki ga tukaj ne bomo analizirali!)

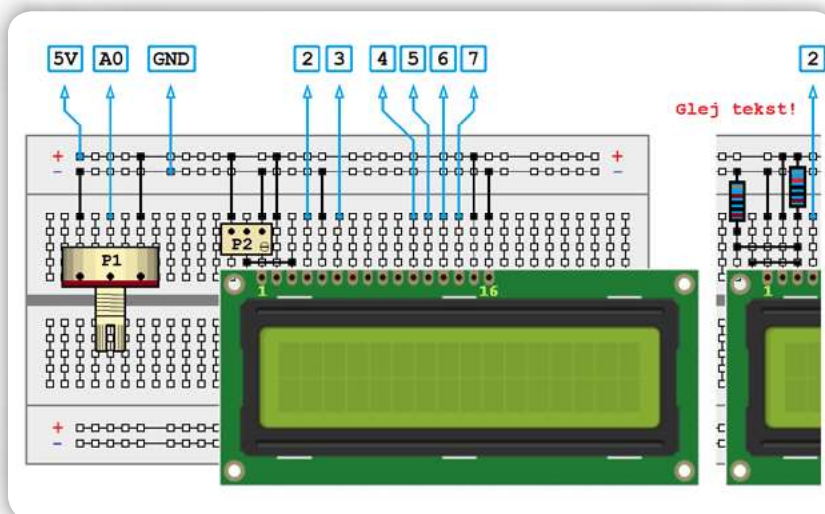
LCD iz Geekcreit kompleta uporablja napetost napajanja 5 V, in to za napajanje elektronike (priključka 1 in 2) in za osvetlitev ozadja (priključka 15 in 16). Osvetlitev ozadja troši 20-30 mA toka, medtem ko je poraba elektronskih vezij nekaj desetkrat manjša. Ker to ne predstavlja obremenitve za Arduino UNO ploščico, smo napetost napajanja LCD-a prav tako „potegnili“



Slika 32: Tako bomo povezali LCD z Arduino UNO ploščico.

iz Arduina. Čitljivost prikazanih znakov je odvisna od kontrasta, ki ga nastavljamo s trimerjem P2.

Slika 33 prikazuje kako bomo komponente s slike 32 postavili na veliko testno ploščico. Tukaj moramo poudariti, da v Geekcreit kompletu ne obstaja ustrezen trimer P2,



Slika 33: Tako bomo komponente s slike 32 postavili na veliko testno ploščico.

zato ga bomo morali dokupiti! Na testno ploščico bomo najlažje postavili večobratni trimer, ki je na sliki 33 tudi narisane. Lahko ga nadomestimo s fiksnima uporoma upornosti 10 kΩ in 1 kΩ iz kompleta (R1 in R2). Te komponenti sta na sliki 32 obarvani z rdečo barvo, modifikacija pa je prikazana na desni strani slike 33. Ker imata upora fiksno vrednost upornosti, nastavljanje kontrasta ne bo možno, vendar bi prikaz na displeju moral biti jasen.

Razen prikaza sporočila, ki mu ga je Arduino program poslal preko vodila, LCD pozna tudi nekaj ukazov, katerih pomen je zgoščeno prikazan v tabeli 3. Delo z alfanumeričnim displejem bomo ilustrirali v naslednjem programskem primeru.

ukaz	opis
begin()	Inicijalizira okolje na LCD displeju in definira dimenzije displeja
clear()	Briše obstoječi izpis na displeju, postavlja kurzor na naslov 0, 0 (začetek prve vrste)
home()	Postavlja kurzor na naslov 0, 0 (začetek prve vrste)
setCursor(x,y)	Postavlja kurzor na naslov y, x (y = vrsta, x = pozicija znotraj vrste)
print()	Izpisuje vsebino na displeju
display()	Vklopi prikaz na displeju po izklopu z ukazom noDisplay()
noDisplay()	Izklopi prikaz na displeju
cursor()	Vklopi prikaz kurzorja
noCursor()	Izklopi prikaz kurzorja
scrollDisplayLeft()	Premakne vsebino displeja eno pozicijo v levo
scrollDisplayRight()	Premakne vsebino displeja eno pozicijo v desno

Tabela 3. Popis osnovnih funkcij razreda LiquidCrystal za delo z AN displejem

9. programska naloga

Napišite program za vezje s slik 32 in 33, ki bo bral napetost na drsniku potenciometra P1 in jo prikazal na alfanumeričnem displeju na naslednji način:

- v gornjem vrstici je potrebno izpisati prebrano vrednost v obsegu 0-1023, to je v obliki, ki jo dobimo iz A/D pretvornika mikrokontrolerja;
- v spodnji vrstici je potrebno izpisati ustrezno napetost v obsegu 0,00-5,00 V!

Pričakovani prikaz na displeju vidimo tudi na sliki 32 zgoraj!

Arduino rešitev (program Geekcreit_9.ino)

Za delo s alfanumeričnim displejem moramo uporabiti knjižnico funkcije LiquidCrystal. Nato definiramo objekt z imenom lcd in sočasno dodamo komunikacijske priključke:

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(2, 3, 4, 5, 6, 7);
```

V funkciji setup() definiramo vrsto alfanumeričnega displeja, ki ga uporabljamo:

```
void setup() {
  lcd.begin(16, 2);
```

Parametri funkcije lcd.begin(16,2) ustrezajo displeju z 2 vrstama in s po 16 znakov v vsaki vrstici. Po definiciji displeja se izpis avtomatično prične od prvega mesta v gornji vrstici. Tekst in vrednosti spremenljivk izpisujemo na displeju s pomočjo funkcije lcd.print(). Mi bomo najprej izpisali sporočilo v gornjem levem vogalu displeja:

```
lcd.print(„A/D izhod:“);
}
```

V funkciji loop() bomo prebrali stanje na analognem vhodu A0 in ga shranili v spremenljivki potValue. Izpis prebrane vrednosti je potrebno pričeti od 13 stolpca prve vrste, na željeno pozicijo se bomo premaknili s funkcijo lcd.setCursor():

```
void loop() {
  int potValue = analogRead(A0);
  lcd.setCursor(12, 0);
```

Funkciji smo poslali vrednosti 12 in 0, ker stolpce in vrste šteje od ničle. Če želimo, je izpis številke poravnane v desno oziroma, da se številka enic vedno izpisuje kot zadnji znak v prvi vrstici. Da bi to dosegli, moramo odkriti koliko števil ima prebrana vrednost potenciometra. To bomo napravili s pomočjo if-elseif-else ukaza, v katerem sočasno izpisujemo tudi potrebno število praznih mesta:

```
if (potValue > 999 ) {
} else if (potValue > 99) {
  lcd.print(„ “);
} else if (potValue > 9) {
  lcd.print(„ “);
} else {
  lcd.print(„ “);
}
lcd.print(potValue);
```

Postavljamo kurzor na začetek druge vrste in izpisujemo začetni tekst in postavljamo kurzor na pozicijo, od katere bomo izpisali vrednost napetosti:

```
lcd.setCursor(0, 1);
lcd.print(„napetost:“);
lcd.setCursor(11,1);
```

Prebrano vrednost moramo preračunati v napetost v obsegu 0-5 V in prikazati z dvema decimalnima mestoma. Zato definiramo novo spremenljivko napetost vrste float:

```
float napetost = potValue * (5.0 / 1023.0);
```