



ARDUINO KROZ JEDNOSTAVNE PRIMJERE

- pripreme za natjecanja -

PRIPREMA 2 MALO TEORIJE I UPRAVLJANJE LED DIODAMA

Paolo Zenzerović, mag. ing. el.

Zagreb, 2013.

KAKO ZAPOČETI RADITI S MIKROKONTROLEROM?

Mikrokontroler za početak možemo zamisliti kao crnu kutiju koja ima određen broj izvoda kojima je moguće upravljati pomoću programa kojeg korisnik napiše na računalu i koji se onda izvodi na samom mikrokontroleru. Kako bi mogli napisane programe prebaciti u mikrokontroler na razvojnoj pločici koristimo se USB vezom s računalom.

Izvodi mikrokontrolera mogu se koristiti za upravljanje raznim uređajima ili za očitavanje stanja s različitih senzora. Tako kažemo da izvodi mogu raditi u dva režima rada – izlazni režim rada i ulazni režim rada.

Kod izlaznog režima rada svaki izvod mikrokontrolera može biti s jednom od dva moguća stanja – visoko i nisko. Kada je izvod u stanju visoko (HIGH) na njemu je potencijal od +5V s obzirom na masu sustava, a kada je izvod u stanju nisko na njemu je potencijal od 0V.

Kada izvod radi u ulaznom režimu rada onda mikrokontroler može očitati digitalno stanje koje je narinuto vanjskom komponentom na sami izvod mikrokontrolera.

Osim u ovom digitalnom režimu rada neki od izvoda mikrokontrolera mogu raditi i kao analogni izvodi. Kada analogne izvode koristimo kao ulazne izvode tada možemo s njih očitavati napon u rasponu od 0 do 5V.

Neke izlazne pinove možemo koristiti kao analogne izlaze pomoću PWM modulacije – više o tome u nekoj od sljedećih priprema.

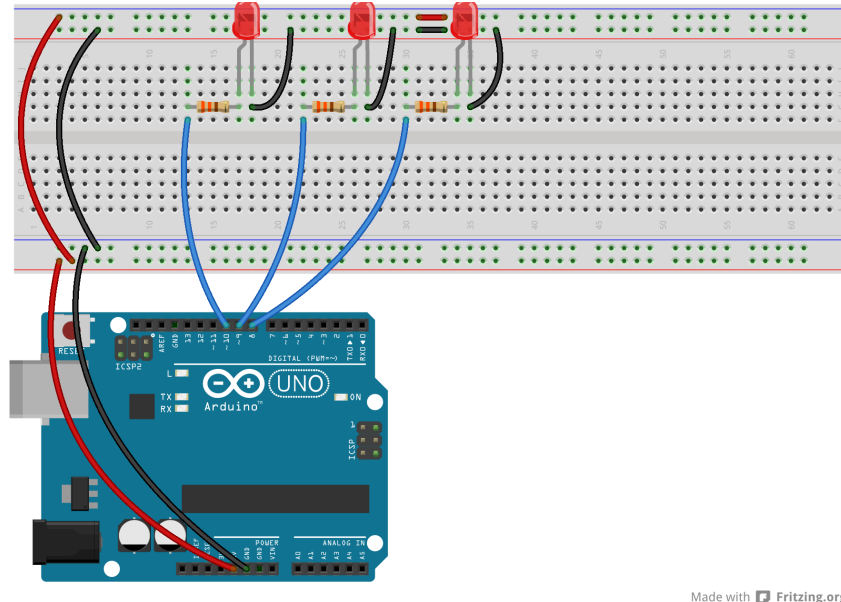
Svaki izvod na Arduino razvojnom okruženju ima svoju oznaku. Oznake su numeričke i idu od 0 do 13 za Arduino UNO izvode koji služe kao digitalni ulazi i izlazi, te A0 do A5 za analogne ulaze na Arduino UNO pločici. Kod Croduino pločice imamo nešto više analognih ulaza pa im oznake idu od A0 do A7. Na svakoj pločici izvodi su jasno obilježeni oznakama kraj konektora za svaki izvod.

Nakon ovog kratnog uvoda krenimo učiti kroz primjere i praksu!

UPRAVLJANJE LED DIODAMA

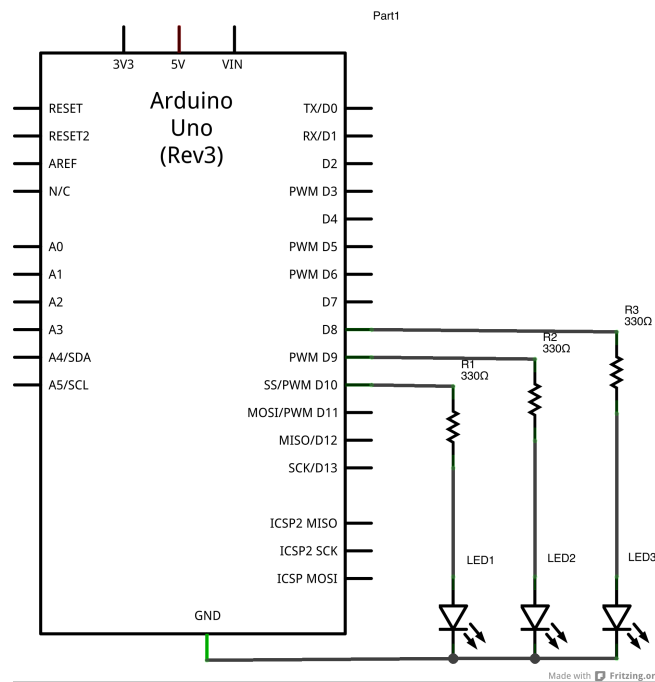
Ovaj primjer pokazuje nam na koji način možemo izvode mikrokontrolera koristiti kao izlazne te pomoću njih uključivati i isključivati nekoliko LED dioda.

Grafički prikaz spajanja na prototipnoj pločici:



Made with Fritzing.org

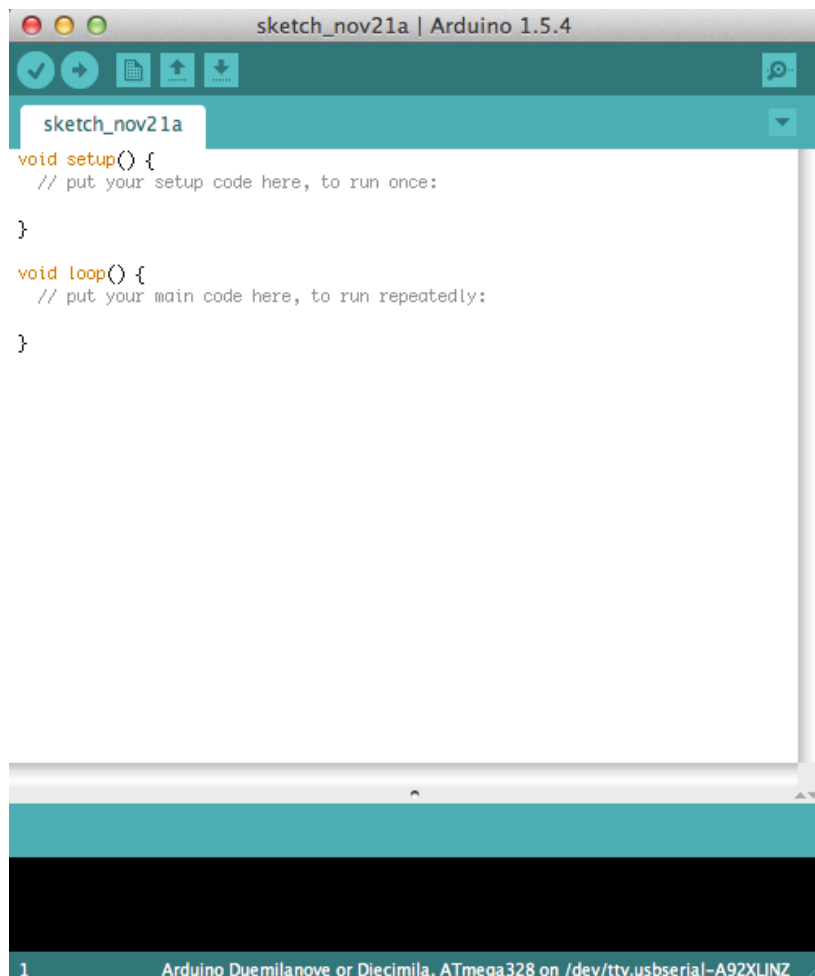
Električna shema:



Made with Fritzing.org

Tri LED diode spojili smo na izvode mikrokontrolera s oznakama 8, 9 i 10. Zbog ograničavanja struje ispred LED diode spojili smo predotpornike vrijednosti 330 Ohma.

Cilj ovog primjera je prikazati način podešavanja izvoda u izlazni režim rada te upravljati LED diodama.



Osnovno okruženje Arduino programskog paketa izgleda kao na gornjoj slici. Vidimo da kod stvaranja novog programa za mikrokontroler programski paket odmah nudi osnovnu strukturu u koju upisujemo naš kod.

Programi pisani za Arduino mikrokontrolere nazivaju se sketchevi. Svaki sketch se sastoji od dvaju dijelova – **setup** dio i **loop** dio.

Setup dio koda se izvodi samo jednom na početku izvođenja sketcha i u njega upisujemo početne postavke koje su nam potrebne za rad s mikrokontrolerom za pojedini sketch.

Loop dio se izvodi nakon setup dijela i on se izvodi cijelo vrijeme dok se mikrokontroler ne resetira ili isključi iz napajanja. U loop dio upisujemo većinu koda mikrokontrolera i tu zapravo definiramo što mikrokontroler radi.

Upišimo sljedeći sketch u Arduino IDE.

Sketch

```
int led1 = 8;
// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led1, OUTPUT);
}
// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led1, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);             // wait for a second
  digitalWrite(led1, LOW); // turn the LED off by making the voltage LOW
  delay(1000);             // wait for a second
}
```

Prvi dio koda označen zelenom bojom definira varijablu nazvanu led1 i postavlja u nju vrijednost 8. Ovo služi kako bismo kasnije kroz sketch mogli koristiti naziv led1 umjesto da koristimo broječanu oznaku izvoda (8). Jednostavnije, ovo nam govori da smo led diodu spojili na izvod broj 8.

Drugi dio koda označen plavom bojom postavlja režim rada izvoda mikrokontrolera. U ovom slučaju izvod led1 postavljen je u OUTPUT odnosno izlazni režim rada. Izvod smo postavili u izlazni režim rada jer njime želimo uključivati i isključivati LED diodu koja je na njega spojena. Naredba prima dva argumenta – prvi označava za koji izvod dajemo naredbu (u ovom slučaju za izvod led1 odnosno izvod 8), a drugi označava želimo li izvod postaviti u ulazni ili izlazni režim rada.

Treći dio koda označen crvenom bojom je glavni dio koda u kojemu definiramo što želimo da mikrokontroler radi. U ovom primjeru želimo uključivati i isključivati LED diode. Za to će nam poslužiti naredba digitalWrite, koja postavlja izvod mikrokontrolera u određeno logičko stanje – visoko ili nisko.

Kada izvod postavimo u visoko stanje (HIGH) led dioda koja je spojena na njega će se uključiti a kada izvod postavimo u nisko stanje LED diode će se isključiti.

Naredba delay() nam omogućava čekanje pri izvođenju sketcha i u nju upisujemo broj milisekundi za čekanje. U našem slučaju delay(1000) daje nam čekanje od 1000ms odnosno jedne sekunde.

Kako se kod u dijelu loop ciklički vrti vidimo da smo ovim sketchom postigli blinkanje jedne led diode.

Zadatak 1: Pokušajmo sada modificirati kod tako da blinkaju sve 3 LED diode.

Sketch

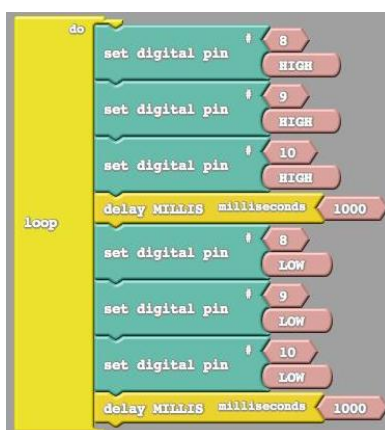
```
int led1 = 8;
int led2 = 9;
int led3 = 10;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led1, HIGH); // turn the LED on (HIGH is the voltage level)
  digitalWrite(led2, HIGH); // turn the LED on (HIGH is the voltage level)
  digitalWrite(led3, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000); // wait for a second
  digitalWrite(led1, LOW); // turn the LED off by making the voltage LOW
  digitalWrite(led2, LOW); // turn the LED off by making the voltage LOW
  digitalWrite(led3, LOW); // turn the LED off by making the voltage LOW
  delay(1000); // wait for a second
}
```

U sketch smo dodali naredbe za definiranje led1, led2 i led3 izvoda te smo ih postavili u izlazni režim rada. Unutar loop dijela koda dodali smo digitalWrite naredbe za sve izvode.

U prošloj uputi instalirali smo i alat za grafičko programiranje Arduino mikrokontrolera. Pogledajmo kako bi gornji sketch izgledao kada bismo ga grafički isprogramirali.



Pomoću Ardublock alata napravite grafički kod kao na gornjoj slici. Pomoću gumba Upload možete isprogramirati nacrtani program u mikrokontroler. Ardublock se sam brine o tome koji je izvod ulazni a koji izlazni u ovisnosti o tome što ste napisali u Vašem programu.

Zadatak 2: Napravite trčeće svjetlo. Neka se LED diode uključuju s lijeva na desno i svaka neka bude uključena pola sekunde.

Sketch

```
int led1 = 8;
int led2 = 9;
int led3 = 10;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led1, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(500);               // wait for half a second
  digitalWrite(led1, LOW);  // turn the LED off by making the voltage LOW

  digitalWrite(led2, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(500);               // wait for half a second
  digitalWrite(led2, LOW);  // turn the LED off by making the voltage LOW

  digitalWrite(led3, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(500);               // wait for half a second
  digitalWrite(led3, LOW);  // turn the LED off by making the voltage LOW
}
```

Na početku loop dijela uključili smo prvu LED diodu, potom smo čekali 500 milisekundi te smo isključili prvu LED diodu. Postupak smo ponovili za drugu i treću LED diodu i time smo postigli efekt trčećih LED dioda.

Zadatak 3: Napravite drugu vrstu trčećeg svijetla. Neka se LED diode uključuju s lijeva na desno svaku sekundu, bez gašenja prethodne LED diode. Kada se uključe sve LED diode i prođe jedna sekunda neka ciklus krene ispočetka.

Sketch

```
int led1 = 8;
int led2 = 9;
int led3 = 10;

// the setup routine runs once when you press reset:
void setup() {
  // initialize the digital pin as an output.
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
}

// the loop routine runs over and over again forever:
void loop() {
  digitalWrite(led1, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(led2, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second
  digitalWrite(led3, HIGH); // turn the LED on (HIGH is the voltage level)
  delay(1000);              // wait for a second

  digitalWrite(led1, LOW);  // turn the LED off by making the voltage LOW
  digitalWrite(led2, LOW);  // turn the LED off by making the voltage LOW
  digitalWrite(led3, LOW);  // turn the LED off by making the voltage LOW
}
```

Na početku uključujemo prvu LED diodu. Potom čekamo jednu sekundu pa uključujemo drugu LED diodu. Nakon jedne sekunde uključujemo i treću LED diodu. Sada sve tri LED diode svijetle zajedno jednu sekundu, nakon čega se sve tri LED diode isključuju i proces kreće ispočetka.