



ARDUINO KROZ JEDNOSTAVNE PRIMJERE

- pripreme za natjecanja -

PRIPREMA 4 ANALOGNI ULAZI I IZLAZI

Paolo Zenzerović, mag. ing. el.

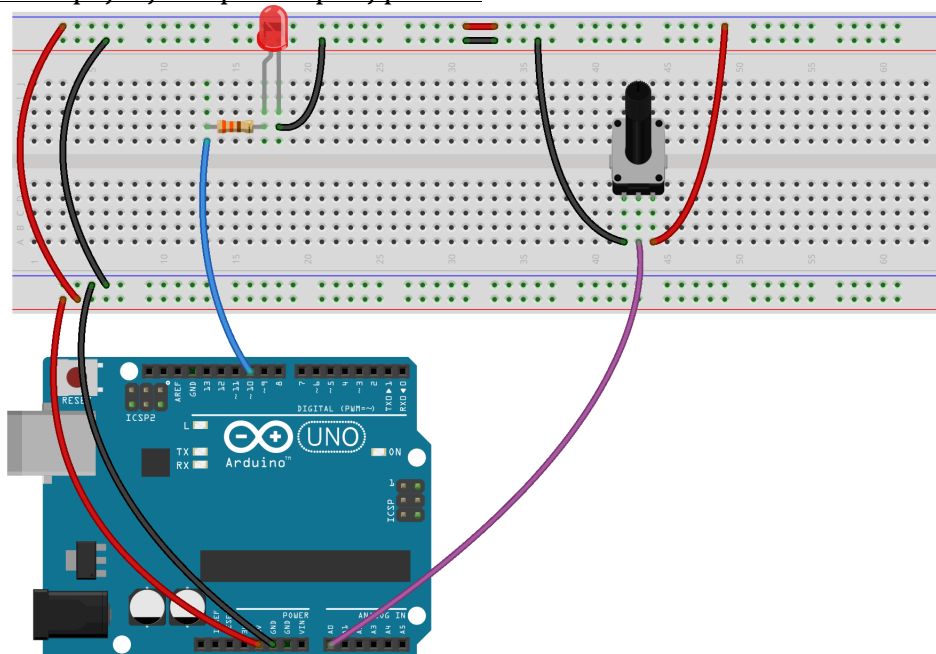
Zagreb, 2013.

OČITAVANJE STANJA DIGITALNIH ULAZA

U ovome nastavku naučiti ćemo kako očitavati stanja analognih ulaza – kako mjeriti napon na određenim ulazima mikrokontrolera. Također, govoriti ćemo o izlazima mikrokontrolera koji mogu funkcionirati kao “analogni” pomoću pulsno-širinske modulacije.

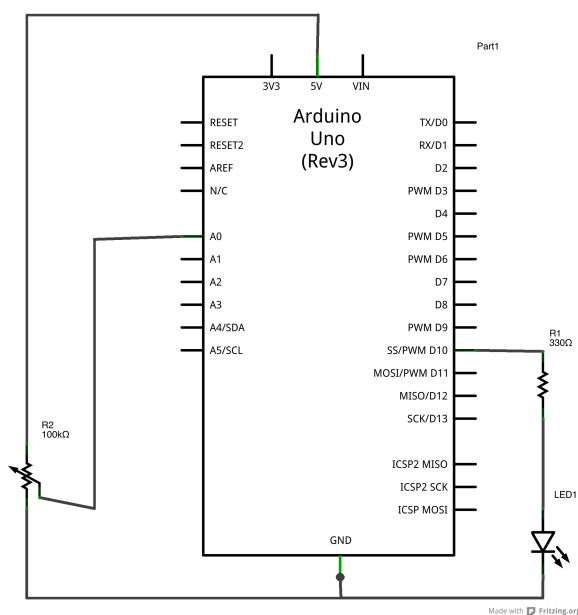
Započnimo sa jednom jednostavnom shemom:

Grafički prikaz spajanja na prototipnoj pločici:



Made with Fritzing.org

Električna shema:

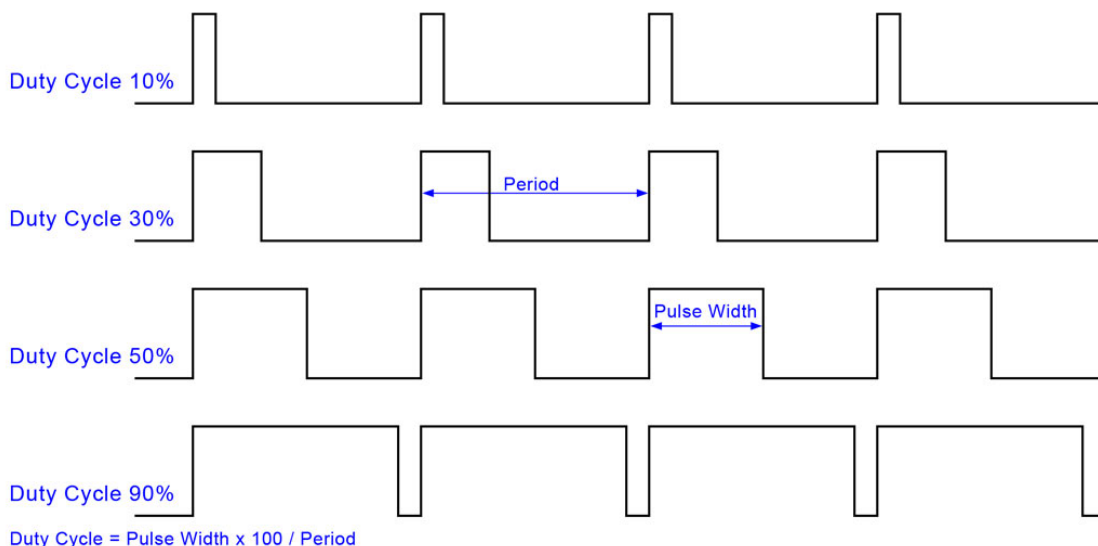


Made with Fritzing.org

Analogni ulazi na Arduino i Croduino pločicama odvojeni su od digitalnih ulaza i obilježeni jedinstvenim imenima A0, A1, A2 itd.

Na analogni ulaz A0 mikrokotrolera spojili smo potenciometar kao djelilo napona. Okretanjem potenciometra na ulazu A0 varira napon u granicama od 0 do 5V. Unutar mikrokotrolera postoji pretvornik koji pretvara taj analogni signal u digitalni i reprezentira ga u obliku broja od 0 do 1023, gdje 0 označava napon od 0V, a 1023 označava napon od 5V. Iznos mjerenog napona lako možemo dobiti formulom: $Umj=5/1024*izmjereni_broj$.

Digitalni izlazi koji nam mogu poslužiti kao “analogni” su na Arduino i Croduino pločicama naznačeni znakom apostrofa kraj broja izvoda za Croduino pločicu odnosno znakom tilde ispod broja izvoda za Arduino pločicu. Ti digitalni izvodi mogu generirati PWM (pulse-width modulation) signale. PWM modulirani signali mogu poslužiti za upravljanje jačinom svjetlosti LED diode, brzinom okretanja motora itd. Nekoliko PWM signala je prikazano na sljedećoj slici:



Kako bi generirali PWM signal moramo imati odabranu frekvenciju odnosno period signala - on je konstantan. Mijenja se širina impulsa (Pulse Width) koju želimo generirati, pa se prema tome mijenja i odnos između vremena u kojem je signal u stanju visoko i vremena u kojem je signal u stanju nisko. U gornjem primjeru ti omjeri su 10%, 30% 50% i 90%. Kako se PWM modulacija izvodi na većim frekvencijama (recimo 5 kHz) tada priključeno trošilo može ove promjene “doživjeti” kao analognu promjenu iznosa napona.

Kako bi prikazali PWM izlaz spojili smo na izvod 10 LED diodu preko otpornika za ograničenje struje. Pogledajmo prvi sketch:

Sketch:

```
int analogniulaz = A0;
int ledPin = 10;
int ocitanavrijednost = 0;
int izlaznavrijednost;

void setup() {
  // declare the ledPin as an OUTPUT:
  pinMode(ledPin, OUTPUT);
}

void loop() {
  // read the value from the sensor:
  ocitanavrijednost = analogRead(analogniulaz);
  izlaznavrijednost = map(ocitanavrijednost, 0, 1023, 0, 255);
  analogWrite(ledPin, izlaznavrijednost);
  delay(10);
}
```

Na početku definiramo varijable za analogni ulaz i LED diodu. Osim toga stvorili smo dvije varijable u kojima ćemo pohraniti i obraditi vrijednost analognog očitavanja. U setup dijelu postavljamo samo ledPin kao izlazni. Ulazni analogni izvod nije potrebno podešavati jer su izvodi *po defaultu* ulaznog tipa.

Pomoću naredbe analogRead očitavamo vrijednost ulaza A0 te tu vrijednost pohranimo u varijablu ocitanavrijednost. Kako se očitana vrijednost kreće u razmjerima od 0 do 1024 a PWM modulacija prihvaća za izlaznu vrijednost raspon od 0 do 255 potrebno je skalirati ulaznu vrijednost. To činimo pomoću funkcije map u kojoj kao argumente dajemo varijablu koju želimo skalirati, te minimale i maksimalne ulazne i izlazne vrijednosti.

Na kraju smo dodali delay(10) kako bi se ažuriranje vrijednosti obavljalo svakih 10 milisekundi.

Ovim sketchem postigli smo regulaciju intenziteta svjetlosti LED diode zakretanjem potenciometra.

Zadatak: Napravite jednostavni mjerač napona. Na 5 izvoda mikrokontrolera spojite 5 LED dioda zajedno s potrebnim predotpornicima. Na jedan analogni ulaz mikrokontrolera spojite potenciometar kao naponsko djelilo tako da napon na ulazu mikrokontrolera možete mijenjati između 0 i 5 volti. Napišite shetch koji će mjeriti ulazni napon i u skladu s njim uključivati LED diode. Kada je ulazni napon između 0 i 1 volt neka svijetli prva LED dioda, kada je napon između 1 i 2 volta neka svijetle prva i druga LED dioda itd. Kada napon između 4 i 5 volti svijetle sve LED diode.

```
int analogniulaz = A0;
int led1 = 13;
int led2 = 12;
int led3 = 11;
int led4 = 10;
int led5 = 9;
int ocitanavrijednost = 0;

void setup() {
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
  pinMode(led4, OUTPUT);
  pinMode(led5, OUTPUT);
}

void loop() {

  ocitanavrijednost = analogRead(analogniulaz);

  if ((ocitanavrijednost>0) && (ocitanavrijednost<205)) {
    digitalWrite(led1, HIGH);
    digitalWrite(led2, LOW);
    digitalWrite(led3, LOW);
    digitalWrite(led4, LOW);
    digitalWrite(led5, LOW);
  } else if ((ocitanavrijednost>206) && (ocitanavrijednost<410)) {
    digitalWrite(led1, HIGH);
    digitalWrite(led2, HIGH);
    digitalWrite(led3, LOW);
    digitalWrite(led4, LOW);
    digitalWrite(led5, LOW);
  } else if ((ocitanavrijednost>411) && (ocitanavrijednost<615)) {
    digitalWrite(led1, HIGH);
    digitalWrite(led2, HIGH);
    digitalWrite(led3, HIGH);
    digitalWrite(led4, LOW);
    digitalWrite(led5, LOW);
  } else if ((ocitanavrijednost>616) && (ocitanavrijednost<820)) {
    digitalWrite(led1, HIGH);
    digitalWrite(led2, HIGH);
    digitalWrite(led3, HIGH);
    digitalWrite(led4, HIGH);
    digitalWrite(led5, LOW);
  } else {
    digitalWrite(led1, HIGH);
    digitalWrite(led2, HIGH);
    digitalWrite(led3, HIGH);
    digitalWrite(led4, HIGH);
  }
}
```

```
digitalWrite(led5, HIGH);
}
delay(10);
}
```

Gornji program radi na principu usporedbe vrijednosti očitane s ulaznog analognog izvoda mikrokontrolera i neke unaprijed definirane vrijednosti – u našem slučaju granični naponi od 1V, 2V itd.

Kako znamo da 1023 predstavlja 5V lako možemo izračunati da je granica za 1V $1023/5=205$, za 2V $1023/5*2=410$, za 3V $1023/5*3=615$, te za 4V $1023/5*4=820$.

Zadatak: Napravite trčeće svijetlo sa tri LED diode kao iz pripreme 2 ali s mogućnošću upravljanja brzinom efekta trčanja. Za promjenu brzine trčanja iskoristite jedan potencijometar spojen na analogni ulaz mikrokontrolera. Napomena: Kako biste mijenjali brzinu efekta mijenjajte pauzu između uključivanja LED dioda, a za promjenu pause iskoristite očitane vrijednosti sa analognog ulaza.

```
int analogniulaz = A0;
int led1 = 13;
int led2 = 12;
int led3 = 11;
int pauza = 0;

void setup() {
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);
  pinMode(led3, OUTPUT);
}

void loop() {
  pauza = analogRead(analogniulaz);

  digitalWrite(led1, HIGH);
  delay(pauza);
  digitalWrite(led1, LOW);

  digitalWrite(led2, HIGH);
  delay(pauza);
  digitalWrite(led2, LOW);

  digitalWrite(led3, HIGH);
  delay(pauza);
  digitalWrite(led3, LOW);
}
```