



# ARDUINO KROZ JEDNOSTAVNE PRIMJERE

*- pripreme za natjecanja -*

PRIPREMA 7 - 2015  
IZRADA SENZORA BOJA

Paolo Zenzerović, mag. ing. el.

Zagreb, 2015.

## SENZOR ZA BOJE

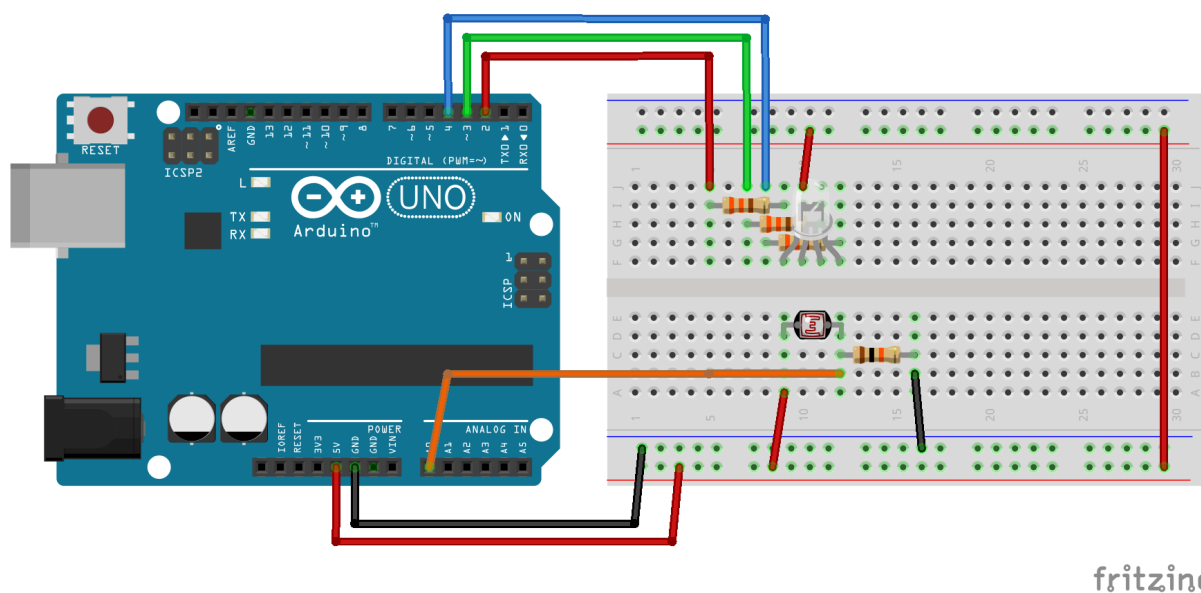
Eto jedne opsežnije i zaista zanimljive pripreme :)

Izraditi ćemo jednostavni i jeftini senzor za detekciju boja s Arduinoom. Senzor se sastoji od jedne RGB diode i jednog fotootpornika – sve komponente koje smo već upoznali. Senzor je temeljen na izvornim kodu koji možete naći na sljedećem linku:

<http://www.instructables.com/id/Using-an-RGB-LED-to-Detect-Colours/?ALLSTEPS>

Senzor se temelji na fizikalnom principu da se svjetlost određene boje bolje odbija od objekta iste te boje nego od objekta druge boje (pojednostavnjeno). RGB diodom obasjamo objekt čiju boju želimo odrediti i mjerimo količinu svjetlosti koja upadne na fotootpornik te njezinim mjerenjem određujemo boju.

Pogledajmo kako spojiti naš senzor boja.



fritzing

Ja ću u ovoj pripremi koristiti RGB diodu sa zajedničkom anodom. Zbog toga sam morao malo promjeniti izvorni kod sa gornjeg linka.

Obzirom da je programski kod veći nego uobičajeni u našim pripremanama neću ga prikazivati u pripremi već ga možete preuzeti na sljedećem linku:

[https://drive.google.com/file/d/0B\\_F4EH9ejJbSeGdGX1Nyb0VzSTg/view?usp=sharing](https://drive.google.com/file/d/0B_F4EH9ejJbSeGdGX1Nyb0VzSTg/view?usp=sharing)

Kada ste sklop spojili prema gornjem prikazu prebacite kod s gornjeg linka u mikrokontroler i pokrenite Serial monitor alat.

Pripremite nekoliko papira raznih boja. Treba vam bijeli i crni papir za kalibraciju i papiri u boji za testiranje senzora. Neka papiri budu mat, a ne sjajni. Ukoliko nemate crni papir poslužite se nekim crnim objektom koji nije sjajan.

Kako bi senzor mogao raspoznavati boje potrebno ga je prvo kalibrirati.

Kada pokrenete Serial monitor dobiti ćete poruku da postavite bijeli objekt iznad senzora. Postavite bijeli objekt i pričekajte dok RGB dioda ne zasvjetli u svim bojama. Nakon toga dobiti ćete poruku da postavite crni objekt iznad senzora i pričekajte dok RGB dioda ponovno ne zasvjetli u svim bojama. Ovime ste odradili kalibraciju senzora. Program je očitao vrijednosti za bijelu i crnu boju i kasnije će ih koristiti za određivanje drugih boja.

Malo nakon kalibracije program počinje s očitavanjem boja. Očitavanje se odvija svake 2 sekunde. Nakon svakog očitavanja program će serijskim putem ispisati vrijednost pojedine osnovne boje u očitanoj boji. Ove će vrijednosti varirati OTPRILIKE od 0 do 255.

Evo ispisa sa mog Serial monitor alata:

Postavi bijeli objekat ispred senzora	
Postavi crni objekat ispred senzora	
R = 253 G = 255 B = 258	BIJELO
R = 253 G = 253 B = 253	BIJELO
R = 253 G = 252 B = 252	BIJELO
R = 4 G = 7 B = 6	CRNO
R = 7 G = 3 B = 10	CRNO
R = 7 G = 3 B = 12	CRNO
R = 232 G = 109 B = 133	CRVENO
R = 241 G = 110 B = 134	CRVENO
R = 241 G = 112 B = 135	CRVENO
R = 84 G = 178 B = 166	ZELENO
R = 87 G = 178 B = 163	ZELENO
R = 88 G = 179 B = 167	ZELENO
R = 64 G = 99 B = 124	PLAVO
R = 66 G = 100 B = 129	PLAVO
R = 67 G = 101 B = 130	PLAVO
R = 163 G = 217 B = 232	PRAZAN SENZOR
R = 168 G = 222 B = 238	PRAZAN SENZOR
R = 174 G = 224 B = 238	PRAZAN SENZOR

Ja sam za probu koristio 150 gramski mat papir u bojama.

Nakon kalibracije napravio sam tri uzorka bijele boje, zatim tri uzorka crne, zatim crvene, pa zelene i plave. U posljednja tri uzorka nije bilo objekta ispred senzora.

Tekst koji je gore upisan ljubičastom bojom nisam dobivao Serijskim putem već sam ga ručno dopisao kako biste se lakše snašli u vrijednostima.

Iz podataka je vidljivo da senzor vrlo dobro raspoznaje osnovne boje.

Obzirom da je kod nešto veći i kompliciraniji malo sam ga pojednostavnio za naše potrebe. Pojednostavnjeni kod koji dalje možete koristiti i kojeg ćemo dalje pojasniti možete preuzeti na sljedećem linku:

[https://drive.google.com/file/d/0B\\_F4EH9ejJbSQjluQjM1ZWpWdk0/view?usp=sharing](https://drive.google.com/file/d/0B_F4EH9ejJbSQjluQjM1ZWpWdk0/view?usp=sharing)

Pogledajmo što se nalazi u programu:

```
int ledArray[] = {2,3,4}; //UPIŠI IZVODE NA KOJE JE SPOJENA RGB DIODE
int fotootpornik = A0; //UPIŠI IZVOD NA KOJI JE SPOJEN FOTOTPORNIK

//INICIJALIZACIJA VARIJABLI - NE DIRATI
float colourArray[] = {0,0,0};
float whiteArray[] = {0,0,0};
float blackArray[] = {0,0,0};
int avgRead;
int red, green, blue;
//KRAJ INICIJALIZACIJE VARIJABLI - NE DIRATI
```

U početnom djelu iskorišteno je polje kako bi se definiralo gdje je spojena RGB dioda. Ukoliko R izvod diode spojite na izvod broj 2, G izvod na broj 3 i B izvod na broj 4 mikrokontrolera onda ne morate mijenjati prvu naredbu. Ukoliko RGB diodu želite spojiti na druge izvode onda promijenite brojeve u prvoj naredbi.

Setup dio:

```
void setup(){

//POSTAVLJANJE IZVODA SENZORA - NE DIRATI
pinMode(ledArray[0],OUTPUT);
pinMode(ledArray[1],OUTPUT);
pinMode(ledArray[2],OUTPUT);
digitalWrite(ledArray[0], HIGH);
digitalWrite(ledArray[1], HIGH);
digitalWrite(ledArray[2], HIGH);
//KRAJ POSTAVLJANJA IZVODA SENZORA - NE DIRATI

Serial.begin(9600); //INICIJALIZACIJA SERIJSKE KOMUNIKACIJE

setBalance(); //KALIBRACIJA SENZORA - OBAVEZNA

}
```

U setup dijelu smo osim djela za postavljanje izvoda na koje je spojena RGB dioda i i sključivanje svih segmenata diode započeli serijsku komunikaciju. Na kraju smo pozvali funkciju “setBalance”. Nju je potrebno pozvati jednom na početku rada i ona služi za kalibraciju senzora. Kalibraciju napravite kako je prethodno objašnjeno.

Loop dio:

```
void loop(){
    checkColour(); //očitaj boju

    Serial.print("R = ");
    Serial.print(red);
    Serial.print("  G = ");
    Serial.print(green);
    Serial.print("  B = ");
    Serial.println(blue);

    delay(2000);
}
```

Unutar loop dijela programa očitavamo boju i ispisujemo rezultat svake dvije sekunde. Kako bismo to učinili pozivamo funkciju "checkColour". Ta funkcija očitava boju i pohrani vrijednosti u varijable "red", "green" i "blue". Nakon toga možemo ih jednostavno ispisati serijskim putem kako god želimo.

Osim početnog, setup i loop dijela na kraju programa postavljene su funkcije koje služe za rad senzora.

```
//FUNKCIJE ZA RAD SENZORA - NE DIRATI
void setBalance(){
    Serial.println("Postavi bijeli objekt ispred senzora");
    delay(5000);
    for(int i = 0;i<=2;i++){
        digitalWrite(ledArray[i],LOW);
        delay(100);
        getReading(5);
        whiteArray[i] = avgRead;
        digitalWrite(ledArray[i],HIGH);
        delay(100);
    }
    Serial.println("Postavi crni objek ispred senzora");
    delay(5000);
    for(int i = 0;i<=2;i++){
        digitalWrite(ledArray[i],LOW);
        delay(100);
        getReading(5);
        blackArray[i] = avgRead;
        digitalWrite(ledArray[i],HIGH);
        delay(100);
    }
    delay(5000);
}
```

```

void checkColour(){
  for(int i = 0;i<=2;i++){
    digitalWrite(ledArray[i],LOW);
    delay(100);
    getReading(5);
    colourArray[i] = avgRead;
    float greyDiff = whiteArray[i] - blackArray[i];
    colourArray[i] = (colourArray[i] - blackArray[i])/(greyDiff)*255;
    digitalWrite(ledArray[i],HIGH);
    delay(100);
  }
  red = int(colourArray[0]);
  green = int(colourArray[1]);
  blue = int(colourArray[2]);
}

void getReading(int times){
  int reading;
  int tally=0;
  for(int i = 0;i < times;i++){
    reading = analogRead(fotootpornik);
    tally = reading + tally;
    delay(10);
  }
  avgRead = (tally)/times;
}
//KRAJ FUNKCIJA ZA RAD SENZORA - NE DIRATI

```

Ove funkcije načelno nije potrebno modificirati, ali za one znatizeljne ukratko ćemo objasniti svaku funkciju:

#### Funkcija **setBalance**:

Ova funkcija služi za kalibraciju senzora. Funkcija očitava stanja fotootpornika za bijeli uzorak te potom za crni uzorak. Vrijednosti se spremaju u dva zasebna polja i služe za određivanje graničnih vrijednosti za rad senzora. Vrijeme koje senzor čeka između javljanja korisniku da postavi uzorak i samog očitavanja postavljeno je na 5 sekundi ali ga je moguće mijenjati promjenom dvaju naredbi `delay(5000)`. Osim toga nakon očitavanja obje boje senzor čeka dodatnih 5 sekundi. Ovo je moguće promijeniti mijenjanjem poslnjenje `delay` naredbe u toj funkciji.

#### Funkcija **checkColour**:

Služi za očitavanje boje. Pozovite ju svaki put kada želite očitati boju na senzoru boja.

#### Funkcija **getReading**:

Služi za očitavanje napona na fotootporniku više puta, kako bi se vrijednost usrednjila i kako bi se smanjila mogućnost pogreške. – Nemojte ju sami pozivati

Kada želite koristiti senzor za boje u vašem projektu dovoljno je da u njega iskopirate dijelove koji su nužni za njegovo funkcioniranje – inicijalizaciju varijabli, postavljanje izvoda, te korištene funkcije. Također obavezno pozovite funkciju za kalibraciju u setup dijelu i nakon toga koristite funkciju za detekciju boja u loop dijelu.

Ukoliko Vam je jednostavnije možete krenuti raditi s već gotovim kodom (drugi link u ovoj pripremi) i samo nadopisati dodatan kod koji vam treba :)

**Zadatak:** Modificirajte program za detekciju boja na način da osim što ispisuje vrijednosti crvene, zelene i plave boje da ispiše je li objekt koji je očitana crveni, zeleni ili plavi.

Potrebno je prvo pokrenuti ovaj program i provesti kalibraciju. Potom ćemo objekte čije boje želimo odrediti skenirati i zapisati ćemo rezultate koje dobivamo. Evo mojih rezultata:

Postavi bijeli objekt ispred senzora

Postavi crni objek ispred senzora

R = 225 G = 62 B = 68

R = 75 G = 172 B = 153

R = 44 G = 68 B = 93

Prvi rezultat odnosi se na crveni uzorak, drugi na zeleni i treći na plavi. Primjećujemo da je vrijednost očitane boje najveća za upravo onu boju koja je očitana. Sada je samo potrebno napraviti nekoliko if uvjeta kako bismo odlučili koje je boje objekt:

Mijenja se samo loop dio pa ću njega prikazati: -

```
void loop(){
  checkColour(); //očitaj boju
  Serial.print("R = ");
  Serial.print(red);
  Serial.print("  G = ");
  Serial.print(green);
  Serial.print("  B = ");
  Serial.print(blue);
  Serial.print("\t\t");
  if (red>green && red>blue) {
    Serial.println("CRVENA");
  } else if (green>red && green>blue) {
    Serial.println("ZELENA");
  } else if (blue>green && blue>red) {
    Serial.println("PLAVA");
  }
  delay(2000);
}
```

Pogledajmo rezultate:

Postavi bijeli objekt ispred senzora	
Postavi crni objek ispred senzora	
R = 230 G = 61 B = 74	CRVENA – postavljena crvena - OK
R = 84 G = 184 B = 173	ZELENA – postavljena zelena - OK
R = 61 G = 99 B = 133	PLAVA – postavljena plava - OK
R = 256 G = 257 B = 260	PLAVA – postavljena crna - GREŠKA
R = 255 G = 257 B = 259	PLAVA – postavljena crna - GREŠKA
R = 256 G = 257 B = 259	PLAVA – postavljena crna - GREŠKA
R = 16 G = 17 B = 22	PLAVA – postavljena bijela - GREŠKA
R = 16 G = 17 B = 26	PLAVA – postavljena bijela - GREŠKA

Program radi dobro dok na senzor dolaze samo uzorci poznate boje – crvene, zelene ili plave. Kada na senzor postavimo neku drugu boju senzor javlja pogrešnu boju.

Kako ovome doskočiti? U ovoj ćemo se pripremi ograničiti na rješenje da dodamo vrijednosti za sve boje koje ćemo koristiti u IF petlju.

Zadatak: Nadogradite prethodni zadatak tako da raspoznaje bijelu, crnu, crvenu, zelenu i plavu boju.

```
void loop(){
  checkColour(); //očitaj boju
  Serial.print("R = ");
  Serial.print(red);
  Serial.print("  G = ");
  Serial.print(green);
  Serial.print("  B = ");
  Serial.print(blue);
  Serial.print("\t\t");
  if (red>200 && green>200 && blue>200) {
    Serial.println("BIJELA");
  } else if (red<20 && green<20 && blue<20) {
    Serial.println("CRNA");
  } else if (red>green && red>blue) {
    Serial.println("CRVENA");
  } else if (green>red && green>blue) {
    Serial.println("ZELENA");
  } else if (blue>green && blue>red) {
    Serial.println("PLAVA");
  }
  delay(2000);
}
```

Moguće je da zbog svjetlosnih uvjeta i udaljenosti vaših papira od senzora kao i samih papira vrijednosti koje dobivate nisu iste kao u pripremi. Dovoljno je da shvatite princip rada i promjenite vrijednosti u if – else if strukturi.