



# **ARDUINO KROZ JEDNOSTAVNE PRIMJERE**

*- pripreme za natjecanja -*

## **PRIPREMA 7 MJERENJE VREMENA**

Paolo Zenzerović, mag. ing. el.

Zagreb, 2014.

## MJERENJE VREMENA

Mjerenje vremena je vrlo zanimljivo područje koje se često zanemaruje u početnom korištenju mikrokontrolera, a može puno doprinijeti funkcionalnosti projekata koji se izrađuju s mikrokontrolerima. Srećom, Arduino okruženje nam nudi vrlo jednostavan način mjerenja vremena.

Arduino na vrlo jednostavan način mjeri vrijeme koje je proteklo od pokretanja izvršavanja programa na samom mikrokontroleru pomoću funkcije `millis()`. Pogledajmo sljedeći sketch:

```
unsigned long vrijeme;

void setup(){
  Serial.begin(9600);
}

void loop(){
  Serial.print("Vrijeme: ");

  // izmjeri koliko je vremena prošlo od pokretanja programa
  vrijeme = millis();
  // ispiši vrijeme
  Serial.println(vrijeme);
  // čekaj jednu sekundu
  delay(1000);
}
```

Kako vidimo iz gornjeg primjera Arduino mjeri vrijeme i ispisuje koliko je vremena proteklo od pokretanja mikrokontrolera. Ispis je vidljiv na računalu putem pogleda na podatke sa serijskog porta.

Kako bi mogli mjeriti vrijeme potrebno je na početku sketcha definirati jednu varijablu u kojoj ćemo spremati vrijeme. Ta varijabla uvijek mora biti `unsigned long` tipa. U gornjem sketchu ona se naziva "vrijeme". Funkcija `millis()` vraća nam vrijednost vremena koje je proteklo od pokretanja mikrokontrolera izraženo u milisekundama.

Kako bismo ograničili količinu podataka koja se ispisuje na računalu putem serijskog porta dodali smo naredbu `delay(1000)` da se podatci ispisuju svakih jednu sekundu.

Zadatak: Pokušajmo napraviti projekt koji će mjeriti vrijeme između dva događaja. Na dva izvoda mikrokontrolera spojite dva tipkala (ako se ne sjećate kako smo spajali tipkala zavirite u pripremu o digitalnim ulazima). Napravimo sketch koji će mjeriti vrijeme koje je proteklo između pritiska prvog i drugog tipkala. Vrijeme ćemo ispisati na računalu pomoću serijske veze.

```

unsigned long poc_vrijeme;
unsigned long kraj_vrijeme;
int tipkalo1 = 12;
int tipkalo2 = 11;

int stanje_tipkala;

void setup(){
  Serial.begin(9600);
  pinMode(tipkalo1, INPUT);
  pinMode(tipkalo2, INPUT);
}

void loop(){

  stanje_tipkala=digitalRead(tipkalo1);
  if (stanje_tipkala==HIGH){
    poc_vrijeme = millis();
    Serial.println("Pritisnuto je prvo tipkalo!");
    delay(100);
  }

  stanje_tipkala=digitalRead(tipkalo2);
  if (stanje_tipkala==HIGH){
    kraj_vrijeme = millis();
    Serial.println("Pritisnuto je drugo tipkalo!");
    Serial.print("Proteklo vrijeme je:");
    Serial.println(kraj_vrijeme-poc_vrijeme);
    delay(100);
  }
}

```

Na početku programa definirali smo dvije varijable – poc\_vrijeme i kraj\_vrijeme. One nam služe za pamćenje početnog i krajnjeg vremena koje očitavamo pri pritisku tipkala 1 odnosno tipkala 2.

Kada pritisnemo tipkalo 1 proteklo vrijeme očitano pomoću funkcije millis() pohraniti ćemo u varijablu poc\_vrijeme. Kada pritisnemo tipkalo vrijeme ćemo pohraniti u varijablu kraj\_vrijeme. Također, kada je pritisnuto drugo tipkalo potrebno je ispisati koliko je vremena proteklo od pritiska prvog do pritiska drugog tipkala. To vrijeme ćemo lako dobiti odizimanjem varijabli kraj\_vrijeme i poc\_vrijeme. Za ispis smo se poslužili naredbom Serial.println(kraj\_vrijeme-poc\_vrijeme).

Naredbe delay(100) služe nam kako bi malo usporili tijek odvijanja programa kod pritiska tipkala i donekle izbjegli duplo ocitanje pritiska.

Zadatak: Modificirajte sketch prethodnog zadatka tako da program mjeri vrijeme između pritiska tipkala 1 i 2. Ako je vrijeme proteklo od pritiska prvog tipkala do pritiska drugog tipkala manje od jedne sekunde neka program ispiše "Wow, brz si!" ako je prošlo više od jedne sekunde neka ispiše "Budi brzi!".

```

unsigned long poc_vrijeme;
unsigned long kraj_vrijeme;
int tipkalo1 = 12;
int tipkalo2 = 11;

int stanje_tipkala;

void setup(){
  Serial.begin(9600);
  pinMode(tipkalo1, INPUT);
  pinMode(tipkalo2, INPUT);
}

void loop(){

  stanje_tipkala=digitalRead(tipkalo1);
  if (stanje_tipkala==HIGH){
    poc_vrijeme = millis();
    delay(100);
  }

  stanje_tipkala=digitalRead(tipkalo2);
  if (stanje_tipkala==HIGH){
    kraj_vrijeme = millis();
    delay(100);

    if ((kraj_vrijeme-poc_vrijeme)<1000){
      Serial.println("Wow, brz si!");
    }
    else {
      Serial.println("Budi brzi!");
    }
  }
}

```

Kada detektiramo pritisak drugog tipkala izračunamo vrijeme koje je proteklo od pritiska prvog do pritiska drugog tipkala. Pomoću if naradbe odredimo je li vrijeme bilo manje ili veće od 1000 ms te prema tome ispisujemo poruku putem serijske veze.