

University of Montenegro
Faculty of Electrical Engineering
Podgorica

Laboratorijske vježbe iz predmeta Industrijska elektronika

Očitavanje temperature i sile, timer interrupt
Sampling temperature and force, timer interrupt
(Vježba 3)

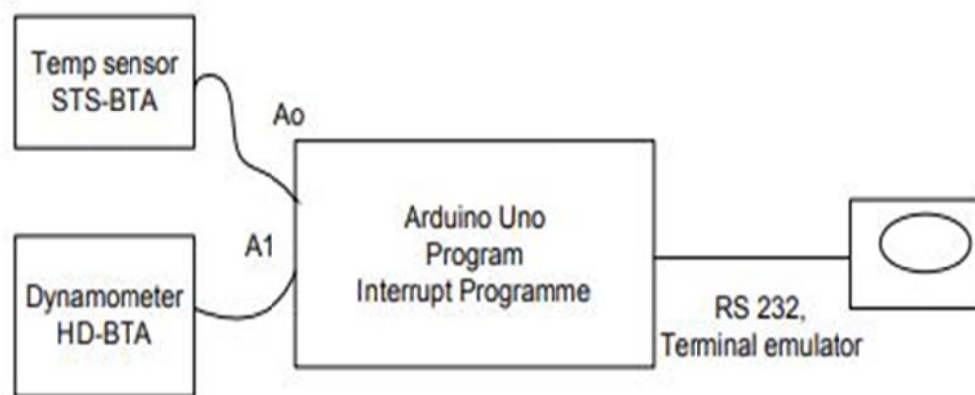
Predmetni nastavnik/Professor:

prof. Dr Radovan Stojanović

Mart 2018.

Grupa I (Mirko Kalezić, Tamara Koljenšić, Marija Milinković, Savo Pejović): Povezati temperaturni senzor na Arduino. Vršiti odabiranje signala sa periodom 1s, 0.5s i 0.2s. Vrijeme odabiranja generisati sa Tajmer interruptom. Prikazati odabirke preko serijskog terminala.

Grupa II (Božidar Andrović, Vesna Mandić, Ivan Martinović, Danko Petrić, Boris Turković): Povezati dinamometar na Arduino. Vršiti odabiranje signala sa periodom 1s, 0.5s i 0.2s. Vrijeme odabiranja generisati sa Tajmer interruptom. Prikazati odabirke preko serijskog terminala.



Abstract:

Timer interrupt is used for performing task at specified time intervals regardless of other programme parts. Interrupts are useful in measuring signal at equal time intervals, calculating those intervals, sending a signal of a specific frequency etc.

We apply it here, along with temperature sensor and dynamometer, to show measured values at specified time intervals.

We use surface temperature sensor and hand dynamometer to display measured values on Arduino Uno and carry them over long distance.

Surface temperature sensor is used for temperature measurements and it is useful in medicine, industry and commercial purposes.

Hand dynamometer is used to measure force. The most common usage of it is in medicine because it can measure grip and pinch strength.

Measurement results are shown on graphical monitor and terminal emulator of Arduino. Also, LED diode has function of alarm and it is turned on if measured value is above specified.

For showing measured values of both sensors we used the same Arduino code. Given code is uploaded on first Arduino – one on the receiving side, one that sends data about measured temperature and force.

For Arduino to make difference which one sensor is used, we call function `Vernier.sensorNumber()` that gives information about sensor ID number - different for different sensors. Temperature sensor's ID is 10, and dynamometer's ID is 67.

If measured value is above specified, LED diode is ON. Specified limit value for temperature sensor is 30°C and 250N for dynamometer.

Objašnjenje šeme rešenja:

Za prikazivanje izmjerenih vrijednosti oba senzora koristili smo isti Arduino kod. Ovaj Arduino kod se nalazi na predajnoj strani, tj. na Arduino koji šalje podatke o temperaturi ili sili.

Da bi Arduino razlikovao senzor sa kog prima podatke koristili smo funkciju `Vernier.sensorNumber()` koja daje podatke o ID broju senzora koji je različit za različite senzore. Za temperaturni senzor ID je jednak 10, a za dinamometar je 67.

Ukoliko vrijednost očitana sa senzora pređe preko specificirane (alarmantne) vrijednosti, pali se LED dioda. Alarmantna vrijednost temperaturnog senzora je 25°C, odnosno 250N za dinamometar.

Nakon toga, poslednji dio Arduino koda služi za prikazivanje izmjerenih vrijednosti temperature i sile. Promjene tih vrijednosti smo prikazali grafički (Slika 3. –za temperaturu, Slika 6. –za silu) i na serijskom monitoru (Slika 4. –za temperature, Slika 7. –za silu).

Timer interrupt je realizovan dijelom koda u funkciji `setup()`. Pojedinačne, značajne linije koda su objašnjenje u komentarima.

Kod:

```
#include "VernierLib.h"
VernierLibVernier;
floatsensorReading;
int LED=8;
intslanje=3;
int y;
#define freq1 // koristi se za definisanje frekvencije, uzimamo tri vrijednosti: 1, 2 i 5Hz, po zadatim periodama

void setup()
{
  pinMode(LED, OUTPUT);
  pinMode(slanje,OUTPUT);
  Serial.begin(9600);
  Vernier.autoID();
  while (!Serial);

  cli(); //zaustaviti prekide
  //postavljanje timer1 prekidana<freq>Hz

  TCCR1A = 0;
  TCCR1B = 0;
  TCNT1 = 0; //stavljanje brojača na 0
  OCR1A = (16000000) / (freq*1024) - 1; //vrijednost za poredjenje, koja nam daje traženu frekvenciju
  //upaliti CTC
  TCCR1B |= (1 << WGM12);
  //Postaviti CS12 i CS10 da bi imali dijeljenje frekvencije od 1024
  TCCR1B |= (1 << CS12) | (1 << CS10);
  //dozvoliti tajmerske prekide
  TIMSK1 |= (1 << OCIE1A);

  sei(); //početi sa prekidima
}

ISR(TIMER1_COMPA_vect){
  Serial.print("(Frekvencijaodabiranja: ");
  Serial.print(freq);
  Serial.print("Hz ");
  Serial.println(sensorReading);
}

void loop()
{
  sensorReading =Vernier.readSensor();
  if(Vernier.sensorNumber()==10){
    y = map(sensorReading,0,50,0,255);
    analogWrite(slanje,y);
  }
}
```

```
else{
  y = map(sensorReading,0,600,0,255);
  analogWrite(slanje,y);
}
if(Vernier.sensorNumber()==10 &&sensorReading>25){
  digitalWrite(LED, HIGH);
}
else if(Vernier.sensorNumber()==67 &&sensorReading>250){
  digitalWrite(LED, HIGH);
  delay(500);
}
else{
  digitalWrite(LED,LOW);
}
delay(500);
}
```

Reference

1. T. E. Kissell, Industrial Electronics, Third edition, Prentice Hall, 2003
2. S. A. Karr, T. E. Kissell, R. C. Overstreet. T.W. Wylie, Laboratory Manual to accompany Industrial Electronics, Third edition, Prentice Hall, 2003
3. <http://apeg.ac.me/nastava/Indel%20Sinro%20and%20Power%20Out%20Circuits.pdf>
4. <https://playground.arduino.cc/Main/ACPhaseControl>
5. http://www.bristolwatch.com/ele2/zero_crossing.htm