

Državna razina natjecanja mladih tehničara

Automatika

Pripreme

Dasduino CORE

Magnetni senzor AS 5600

Školska godina 2025. / 2026.

Radni materijal za pripremu učenika osnovnih škola

Hrvoje Vrhovski, Hrvatska zajednica tehničke kulture

Sadržaj

Sadržaj	1
1. Dasduino CORE.....	4
Zašto je odabrano upravo ovo sučelje	4
Zaključak	5
2. Tipkalo	6
Tipkalo u sustavu	6
Kako tipkalo radi	6
Problem – odbijanje kontakta (debouncing)	6
Kako izgleda problem	7
Posljedice	7
Rješenje – debouncing	7
Softversko rješenje.....	7
Primjer koda.....	8
Zašto je to važno u ovom sustavu	8
Debounce bez delay() (pravilna izvedba)	8
Zašto ne koristiti delay()	8
Princip rada	9
Univerzalni debounce (preporučena verzija)	9
Kod	9
Korištenje u loop()	9
Važna napomena (ovo je često greška)	10
Pravi “klik” (one-shot trigger)	10
Edge detekcija.....	10
Što smo dobili	10
Zaključak	10
3. AS 5600 magnetski sensor položaja	12
Osnovne karakteristike.....	12
Način rada	12
Zašto je odabran ovaj senzor.....	12
Ograničenja i izazovi	13
Zaključak	13

4. Blok kondenzator (100 nF)	14
Blok kondenzator u elektroničkom sustavu	14
Zašto je potreban	14
Uloga blok kondenzatora	14
Gdje se postavlja	15
Zašto baš 100 nF	15
Što ako ga nema	15
Zaključak.....	15
5. Kalibracija AS 5600	16
Zašto je kalibracija potrebna	16
Ideja kalibracije.....	16
Postupak kalibracije	16
Primjer s konkretnim vrijednostima	17
Što smo postigli.....	17
Važne napomene	17
Napredna napomena (hint).....	18
Zaključak.....	18
Primjer koda za kalibraciju	19
Što ovaj kod radi	20
Bez kalibracije	20
S kalibracijom	20
Mapiranje	20
Grafički prikaz	20
Važna napomena	21
Integracija	22
Završne napomene	22
Ishodi učenja (kompetencijski okvir).....	22
Kompetencije	22
Zaključak.....	23
Namjena dokumenta	23

Uvod

Na državnoj razini Natjecanja mladih tehničara u području Automatika koristiti će se mikroupravljačko sučelje Dasduino CORE te magnetni senzor AS5600. U ovoj skripti ćemo obraditi spajanje i upotrebu ovih elemenata.

Napomena:

Svi programski primjeri u skripti pisani su i ispitani u Arduino IDE okruženju uz korištenje DASDUINO CORE mikroupravljačkog sučelja.

Ako koristite klasični Arduino, kod će u pravilu raditi jednako, ali nazivi pločica i odabir procesora u izborniku mogu se razlikovati.

Cilj je postupno razumjeti:

- Što je i kako se spaja magnetni senzor AS5600
- kako očitavati, pretvarati i upotrebljavati podatke s AS5600
- zašto je potrebno podesiti očitavanja s AS5600 i kako se to radi

Gradivo je strukturirano tako da učenik najprije razumije osnovni princip rada magnetnog senzora AS5600, nauči spojiti senzor i zatim očitati podatke s njega

Nakon toga prelazimo na uviđanje i razumijevanje mogućih netočnih očitavanja podataka te načine rješavanja tih problema.

U završnom dijelu skripte prikazujemo podatke na odgovarajućem izlaznom uređaju.

Na taj način prolazimo cjelovit put:

Spajanje → očitavanje podataka → obrada podataka → prikaz podataka.

Takav pristup odražava tipične zadatke iz područja automatike i čini temelj za uspješno rješavanje složenijih natjecateljskih zadataka.

1. Dasduino CORE

Dasduino CORE mikroupravljačko sučelje

Dasduino CORE je mikroupravljačko sučelje temeljeno na platformi kompatibilnoj s Arduino sustavom. Namijenjeno je edukaciji i razvoju elektroničkih sustava, a omogućuje jednostavno povezivanje različitih senzora, aktuatora i prikaznih elemenata.

Sučelje omogućuje:

- upravljanje digitalnim i analognim ulazima i izlazima
- komunikaciju s uređajima putem I2C, SPI i UART protokola
- obradu podataka u stvarnom vremenu
- razvoj i testiranje programa u Arduino okruženju

Zašto je odabrano upravo ovo sučelje

Za potrebe ovog zadatka odabrano je Dasduino CORE mikroupravljačko sučelje zbog sljedećih svojstava:

- [Edukativna pristupačnost](#)

Sučelje je jednostavno za korištenje i pogodno za učenike koji već imaju osnovno iskustvo s Arduino platformom.

- [Dovoljno ulazno/izlaznih pinova](#)

Zadatak uključuje:

- upravljanje 4-znamenkastim 7-segmentnim displejem (više pinova)
- povezivanje senzora putem I2C sučelja
- korištenje tipkala

Dasduino CORE omogućuje istovremeno korištenje svih potrebnih priključaka.

- [Podrška za I2C komunikaciju](#)

Senzor koji se koristi u zadatku komunicira putem I2C protokola.

Dasduino CORE ima ugrađenu podršku za ovu vrstu komunikacije, što pojednostavljuje povezivanje i programiranje.

- Stabilno napajanje i rad

Sučelje omogućuje pouzdan rad u kombinaciji s baterijskim napajanjem, što je važno za mobilne i samostalne uređaje.

- Kompatibilnost s postojećim znanjem

Učenici su već upoznati s Arduino načinom programiranja, pa se mogu usmjeriti na rješavanje problema zadatka, a ne na učenje novog alata.

- Hrvatski proizvod

Dasduino CORE je rezultat višegodišnjeg rada i iskustva hrvatskih stručnjaka i lako je dostupan.

Zaključak

Odabirom Dasduino CORE sučelja omogućeno je:

- jednostavno povezivanje svih elemenata sustava
- pouzdano očitavanje senzora
- stabilan prikaz podataka
- fokus na rješavanje problema, a ne na tehničke prepreke

Odabir sučelja nije slučajna, već je prilagođen zahtjevima zadatka kako bi se omogućila optimalna ravnoteža između tehničke složenosti i izvedivosti u vremenskom okviru natjecanja.

2. Tipkalo

Tipkalo u sustavu

Tipkalo je jednostavan elektronički element koji omogućuje korisniku unos naredbi u sustav.

U ovom sustavu koristi se za:

- postavljanje početne referentne točke
- postavljanje krajnje referentne točke
- postavljanje početne točke (nuliranje)

Pritiskom tipkala mikroupravljač izvršava određenu radnju, npr. pohranjuje trenutnu vrijednost senzora.

Kako tipkalo radi

Tipkalo ima dva osnovna stanja:

- otvoreno (nije pritisnuto)
- zatvoreno (pritisnuto)

U digitalnom sustavu:

- očitava se kao logička 0 ili 1

Najčešće se koristi:

pull-up otpornik

što znači:

- kada tipkalo nije pritisnuto → logička 1
- kada je pritisnuto → logička 0

Problem – odbijanje kontakta (debouncing)

Tipkalo nije idealan element.

Pri pritisku:

- kontakti se ne spoje odmah stabilno

- dolazi do brzog “titranja” između 0 i 1

To se naziva:

odbijanje (bounce)

Kako izgleda problem

Umjesto jednog pritiska:

0 → 1

dobijemo:

0 → 1 → 0 → 1 → 0 → 1

u vrlo kratkom vremenu (nekoliko milisekundi)

Posljedice

Ako se problem ne riješi:

- jedna tipka može se registrirati više puta
- referentne vrijednosti mogu biti pogrešno postavljene
- sustav može preskočiti korake
- ponašanje postaje nepredvidivo

Rješenje – debouncing

Debouncing je postupak uklanjanja neželjenih promjena stanja tipkala.

Softversko rješenje

Najčešće rješenje je:

☞ vremensko kašnjenje (debounce)

(a) Primjer logike:

1. detektira se promjena stanja
2. čeka se kratko vrijeme (npr. 20–50 ms)
3. ponovno se provjeri stanje
4. ako je stabilno → prihvaća se kao valjan pritisak

Primjer koda

```
const int tipkalo = 2;
bool zadnjeStanje = HIGH;
unsigned long vrijeme = 0;
const int debounce = 40;

void loop() {
  bool stanje = digitalRead(tipkalo);

  if (stanje != zadnjeStanje) {
    vrijeme = millis();
  }

  if ((millis() - vrijeme) > debounce) {
    if (stanje == LOW) {
      // valjan pritisak
    }
  }

  zadnjeStanje = stanje;
}
```

Zašto je to važno u ovom sustavu

U ovom zadatku tipkalo služi za:

- spremanje referentnih vrijednosti

Ako se pritisak registrira više puta:

- kalibracija postaje netočna
- cijeli sustav daje pogrešne rezultate

Jedan fizički pritisak tipkala ne znači nužno jedan digitalni signal.

Debounce bez delay() (pravilna izvedba)**Zašto ne koristiti delay()**

Korištenje delay():

- zaustavlja program
- usporava prikaz (multiplex!)
- može uzrokovati preskakanje očitavanja

Zato koristimo:

vrijeme (millis) umjesto čekanja

Princip rada

1. detektira se promjena stanja
2. zapamti se vrijeme promjene
3. čeka se kratko vrijeme (npr. 40 ms)
4. ako je stanje i dalje isto → prihvaća se

Univerzalni debounce (preporučena verzija)

Kod

```
const int BTN_MIN = 2;
const int BTN_MAX = 3;

bool stanjeMin = HIGH;
bool stanjeMax = HIGH;

bool zadnjeMin = HIGH;
bool zadnjeMax = HIGH;

unsigned long vrijemeMin = 0;
unsigned long vrijemeMax = 0;

const unsigned long debounce = 40;

bool debounceRead(int pin, bool &zadnje, unsigned long &vrijeme) {

    bool trenutno = digitalRead(pin);

    if (trenutno != zadnje) {
        vrijeme = millis();
        zadnje = trenutno;
    }

    if ((millis() - vrijeme) > debounce) {
        return trenutno;
    }

    return zadnje;
}
```

Korištenje u loop()

```
void loop() {

    bool btnMin = debounceRead(BTN_MIN, zadnjeMin, vrijemeMin);
    bool btnMax = debounceRead(BTN_MAX, zadnjeMax, vrijemeMax);

    if (btnMin == LOW) {
        // postavi početnu referencu
    }

    if (btnMax == LOW) {
        // postavi krajnju referencu
    }

}
```

Važna napomena (ovo je često greška)

Ovaj kod:

detektira stanje tipke

ALI:

može se izvršavati više puta dok je tipka pritisnuta

Pravi "klik" (one-shot trigger)

Ako želimo da se radnja izvrši **samo jednom po pritisku**, treba dodati:

Edge detekcija

```
bool stariMin = HIGH;
bool stariMax = HIGH;

void loop() {

    bool btnMin = debounceRead(BTN_MIN, zadnjeMin, vrijemeMin);
    bool btnMax = debounceRead(BTN_MAX, zadnjeMax, vrijemeMax);

    // detekcija pritiska (HIGH → LOW)
    if (stariMin == HIGH && btnMin == LOW) {
        // izvrši samo jednom
    }

    if (stariMax == HIGH && btnMax == LOW) {
        // izvrši samo jednom
    }

    stariMin = btnMin;
    stariMax = btnMax;
}
```

Što smo dobili

- ✓ nema blokiranja programa
- ✓ multiplex radi glatko
- ✓ senzor se očitava kontinuirano
- ✓ tipkalo radi pouzdano
- ✓ akcija se izvršava samo jednom

Zaključak

Tipkalo je jednostavan, ali nesavršen element.

Za pouzdan rad sustava potrebno je:

- pravilno ga spojiti
- razumjeti njegovo ponašanje
- primijeniti debouncing

Tipkalo nije problem — način na koji ga čitamo je problem.

3. AS 5600 magnetski sensor položaja

AS5600 je digitalni sensor koji omogućuje mjerenje kuta rotacije pomoću magnetskog polja. Sensor očitava položaj magneta postavljenog iznad njega i pretvara ga u numeričku vrijednost koja odgovara kutu zakreta.

Za razliku od mehaničkih senzora (npr. potenciometra), AS5600 nema fizički kontakt s pokretnim dijelom, što omogućuje preciznije i dugotrajnije mjerenje.

Osnovne karakteristike

- mjerenje kuta u rasponu: **0° – 360°**
- rezolucija: **12-bit (0 – 4095)**
- komunikacija: **I2C protokol**
- beskontaktno mjerenje (magnet + sensor)
- visoka osjetljivost i ponovljivost mjerenja

Način rada

Senzor detektira orijentaciju magnetskog polja koje stvara mali permanentni magnet postavljen iznad čipa.

Promjenom položaja magneta:

- mijenja se magnetsko polje
- senzor izračunava kut
- rezultat se šalje mikroupravljaču

Dobivena vrijednost:

- može biti sirova (0 – 4095)
- ili pretvorena u stupnjeve

Zašto je odabran ovaj senzor

- Beskontaktno mjerenje

Nema mehaničkog trošenja, što omogućuje dugotrajan i pouzdan rad.

- Visoka rezolucija

12-bitna rezolucija omogućuje vrlo precizno očitavanje kuta.

- Jednostavna komunikacija

Korištenje I2C protokola omogućuje jednostavno povezivanje s mikroupravljačem uz mali broj vodova.

- Pogodan za rotacijske sustave

Senzor je idealan za mjerenje položaja rotirajućih dijelova, što ga čini prikladnim za ovaj zadatak.

Ograničenja i izazovi

U stvarnoj primjeni senzor ne daje uvijek idealne rezultate.

Razlozi:

- magnet nije savršeno centriran
- magnet nije potpuno paralelan sa senzorom
- mehanička konstrukcija ima tolerancije

Posljedice:

- očitavanje može odstupati
- sustav može biti nelinearan
- potrebna je kalibracija

Zaključak

AS5600 omogućuje:

- jednostavno i precizno mjerenje kuta
- beskontaktni rad
- jednostavno povezivanje s mikroupravljačem

Međutim, zbog realnih mehaničkih ograničenja potrebno je dodatno obraditi podatke kako bi se dobio točan rezultat.

Očitana vrijednost senzora ne mora uvijek odgovarati stvarnom kutu, zbog čega je potrebno provesti kalibraciju sustava.

4. Blok kondenzator (100 nF)

Blok kondenzator u elektroničkom sustavu

Blok kondenzator (najčešće vrijednosti 100 nF) koristi se za stabilizaciju napajanja elektroničkih komponenti.

Njegova osnovna uloga je **smanjenje smetnji i stabilizacija naponske linije**.

Zašto je potreban

U elektroničkom sustavu, osobito kada se koriste:

- mikroupravljači
- senzori
- LED displeji

dolazi do:

- naglih promjena struje
- šuma u napajanju
- kratkotrajnih padova napona

Ti poremećaji mogu uzrokovati:

- nestabilan rad senzora
- pogrešna očitavanja
- treperenje prikaza
- nepredvidivo ponašanje sustava

Uloga blok kondenzatora

Blok kondenzator djeluje kao mali spremnik energije.

Kada dođe do naglog pada napona:

- kondenzator “daje” energiju sustavu

Kada se pojavi šum:

- kondenzator ga “upija” i smanjuje

Na taj način:

- napajanje postaje stabilnije
- rad sustava pouzdaniji

Gdje se postavlja

Blok kondenzator se postavlja **što bliže napajanju komponente (VCC i GND)**

U ovom sustavu:

- uz senzor

Zašto baš 100 nF

Vrijednost od 100 nF je standardna jer:

- učinkovito filtrira visokofrekventni šum
- brzo reagira na promjene
- koristi se kao univerzalna vrijednost za većinu digitalnih sklopova

Što ako ga nema

Ako blok kondenzator nije ugrađen:

- očitavanje senzora može biti nestabilno
- vrijednosti mogu “skakati”
- sustav može povremeno raditi pogrešno
- prikaz može treperiti

Problemi se javljaju “nasumično”, što ih čini teškima za dijagnostiku

Zaključak

Blok kondenzator:

- ne utječe izravno na funkcionalnost sustava
- ali je ključan za njegov stabilan i pouzdan rad

Sustav može raditi i bez blok kondenzatora, ali neće raditi pouzdano.

5. Kalibracija AS 5600

Zašto je kalibracija potrebna

U idealnim uvjetima senzor bi davao točnu vrijednost kuta.

Međutim, u stvarnom sustavu:

- magnet nije savršeno centriran
- magnet nije potpuno paralelan sa senzorom
- mehanička konstrukcija ima tolerancije

Zbog toga očitana vrijednost ne odgovara stvarnom kutu.

Ideja kalibracije

Umjesto da vjerujemo senzoru, definiramo:

- početnu referentnu točku
- krajnju referentnu točku

Na temelju njih prilagođavamo očitavanje stvarnom sustavu

Postupak kalibracije

1. Početni položaj

- mehanizam se postavi u početni položaj (zatvoren)
- pritisne se tipkalo
- sprema se vrijednost senzora

Označimo je kao:

minVal

2. Krajnji položaj

- mehanizam se postavi u krajnji položaj (otvoren)
- pritisne se drugo tipkalo
- sprema se vrijednost senzora

Označimo je kao:

maxVal

3. Izračun raspona

$\text{raspon} = \text{maxVal} - \text{minVal}$

4. Izračun kuta

Za svako novo očitavanje:

$\text{kut} = (\text{raw} - \text{minVal}) * 180 / \text{raspon}$

Primjer s konkretnim vrijednostima

Pretpostavimo:

- početna vrijednost: **minVal = 1200**
- krajnja vrijednost: **maxVal = 3000**

- Izračun raspona

$\text{raspon} = 3000 - 1200 = 1800$

- Novo očitavanje

$\text{raw} = 2100$

- Izračun kuta

$\text{kut} = (2100 - 1200) * 180 / 1800$

$\text{kut} = 900 * 180 / 1800$

$\text{kut} = 90^\circ$

Što smo postigli

Bez kalibracije:

- senzor daje “svoje” vrijednosti

S kalibracijom:

- sustav daje stvarni kut

Važne napomene

Zbog mehanike raspon neće biti idealan:

- neće biti točno 0–4095
- ali to nije problem

- Kalibracija je obavezna

Bez nje:

- rezultat nije pouzdan
- Vrijednosti mogu biti obrnute

U nekim slučajevima:

- maxVal može biti manji od minVal

Potrebno je to uzeti u obzir u programu

Napredna napomena (hint)

Ako se referentne točke nalaze blizu granice ($0^\circ/360^\circ$), jednostavno oduzimanje možda neće dati ispravan rezultat.

Zaključak

Kalibracijom:

- prilagođavamo sustav stvarnim uvjetima
- uklanjamo utjecaj mehaničkih pogrešaka
- dobivamo pouzdano mjerenje

Senzor ne mora biti savršeno postavljen — dovoljno je da ga znamo ispravno kalibrirati.

Primjer koda za kalibraciju

Ovaj primjer prikazuje:

- spremanje referentnih točaka pomoću tipkala
- izračun kuta
- osnovnu zaštitu od pogreške

```
#include <Wire.h>
#include <AS5600.h>

AS5600 as5600;

// tipkala
const int BTN_MIN = 2;
const int BTN_MAX = 3;

// varijable kalibracije
int minVal = 0;
int maxVal = 4095;

bool minSet = false;
bool maxSet = false;

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

  pinMode(BTN_MIN, INPUT_PULLUP);
  pinMode(BTN_MAX, INPUT_PULLUP);
}

void loop() {

  int raw = as5600.getRawAngle();

  // spremanje početne reference
  if (digitalRead(BTN_MIN) == LOW) {
    minVal = raw;
    minSet = true;
    Serial.println("Postavljena početna referenca");
    delay(300); // jednostavni debounce
  }

  // spremanje krajnje reference
  if (digitalRead(BTN_MAX) == LOW) {
    maxVal = raw;
    maxSet = true;
    Serial.println("Postavljena krajnja referenca");
    delay(300); // jednostavni debounce
  }

  // izračun samo ako su obje reference postavljene
  if (minSet && maxSet) {

    int raspon = maxVal - minVal;

    // zaštita od dijeljenja s nulom
    if (raspon != 0) {
```


Važna napomena

Ako:

- $\maxVal < \minVal$
ili
- sustav prelazi preko 0°

Potrebno je dodatno prilagoditi izračun

Kalibracija zapravo znači pretvaranje stvarnog (nesavršenog) sustava u matematički model.

Integracija

Završne napomene

Ishodi učenja (kompetencijski okvir)

Nakon rada s ovim materijalom učenik će moći:

1. Objasniti upotrebu tipkala u automatiziranim elektroničkim sustavima.
2. Objasniti problem debouncinga i načine rješavanja tog problema.
3. Objasniti razloge i načine korištenja kondenzatora u automatiziranim elektroničkim sustavima.
4. Objasniti funkciju i upotrebu magnetnog senzora AS5600.
5. Objasniti probleme pri očitavanju podataka s magnetnog senzora AS5600 te načine rješavanja nepreciznih očitavanja sa senzora.
6. Samostalno prilagoditi programsko rješenje novom natjecateljskom zadatku.

Kompetencije

Radom na ovim zadacima učenici razvijaju:

- logičko i algoritamsko razmišljanje
- sposobnost analize tehničkog sustava
- razumijevanje vremenskog upravljanja u mikroupravljačkim sustavima
- preciznost u radu s digitalnim i analognim signalima
- sposobnost organizacije podataka
- sposobnost optimizacije programskog rješenja

Ove kompetencije čine temelj za uspješno sudjelovanje na natjecanjima iz područja automatike i tehničke kulture.

Zaključak

U ovoj skripti prošli smo cjelovit put razumijevanja rada elektroničkih elemenata, od jednostavnih (tipkalo) do implementacije različitih elemenata u samostalan i cjelovit mjerni elektronički sustav.

Učenik je postupno:

- razumio kako radi tipkalo, koji problemi se pojavljuju i kako ih se može riješiti,
- uočio problem zajedničkih vodova kod višeznamenkastog displeja,
- uočio probleme naglih promjena struje, šuma u napajanju i kratkotrajnih padova napona,
- savladao način problema naglih promjena struje, šuma u napajanju i kratkotrajnih padova napona,
- razumio kako se spaja i kako se očitavaju podaci s magnetnog senzora AS5600,
- razumio probleme nepreciznih očitavanja s magnetnog senzora AS5600,
- savladao način rješavanja problema nepreciznih očitavanja s magnetnog senzora AS5600,
- povezao sve elektroničke elemente u funkcionalan model elektroničkog mjernog uređaja.

Takav način rada razvija:

- analitičko razmišljanje,
- razumijevanje uzročno-posljedičnih veza,
- sposobnost organizacije programa,
- preciznost u radu s mikroupravljačkim sustavima.

Upravo takva znanja i način razmišljanja čine temelj uspješnog rješavanja natjecateljskih zadataka iz područja automatike.

Namjena dokumenta

Ovaj dokument izrađen je kao radni materijal za pripremu učenika za državnu razinu Natjecanja mladih tehničara iz područja automatike.

Materijal je namijenjen:

- učenicima viših razreda osnovne škole,
- mentorima tehničke kulture,
- voditeljima izvannastavnih aktivnosti iz područja automatike i mikroupravljačkih sustava.

Dokument se može koristiti kao:

- pripremni materijal za natjecanje,
- dodatni obrazovni sadržaj,
- temelj za daljnju nadogradnju projekata iz područja automatike.