

DOKUMENTACIJA
PROJEKAT: PAMETNA ŠTOPERICA

Projektni menadžer:	Hardware:	Software:	Projektni asistent:
Said Karaosmanović	Tarik Redžić	Edhem Krivdić	Mirza Bašić

Bugojno, oktobar 2023. – maj 2024.

DOKUMENTACIJA PROJEKTA PAMETNA ŠTOPERICA

Nosioc autorskih prava:	Srednja tehnička škola Bugojno
Izdavač:	Srednja tehnička škola Bugojno
Za izdavača:	Said Karaosmanović
Lektor:	Mirza Bašić, Edhem Krivdić
Tehnički urednik:	Mirza Bašić, Tarik Redžić
Hardware:	Tarik Redžić
Software:	Edhem Krivdić
Broj stranica:	13
Izdanje:	Prvo izdanje
Cijena:	<i>besplatan primjerak</i>

Copyright © 2024 by Srednja tehnička škola Bugojno. All rights reserved.

Kopiranje, štampanje i dijeljenje ovog dokumenta krši zakon o autorskim pravima bez prethodne dozvole izdavača i nosioca autorskih prava (Službeni glasnik 63/10).

Uvod u projekat

Pametna štoperica je uređaj koji mogu koristiti profesori sportske struke. U pitanju je štoperica kojoj je cilj eliminisati ljudski faktor greške prilikom mjerenja.

Primjena

Štoperica se može primjeniti za mjerenje vremena trčanja učenika, ali i kao sredstvo za mjerenje brzine automobila.

Princip rada

Štoperica radi na jednostavan princip. Na početku i kraju nalaze se laseri i kontrolne ploče sa senzorom. Prilikom prekida lasera na STARTU, šalje se signal preko radio valova na drugu kontrolnu ploču koja pokreće štopericu. Sada, sve dok se ne prekine laser na KRAJU utrke, štoperica će i dalje raditi.

Limitacije

Postojanje su software-ske limitacije. Naime, funkcija `mills()` ima limit od 49 dana kada će se ponovo restartovati vrijeme.

Također, štoperica nije testirana u okruženjima sa promjenjivom vlagom i temperaturom, te otpor na oštećenja i deformacije.

Moramo napomenuti da štoperica koristi 433 MHz frekvenciju rada, što je frekvencija rada ključeva automobila, te pojedinih alarmnih sistema, te upozoravamo na pažljivo korištenje.

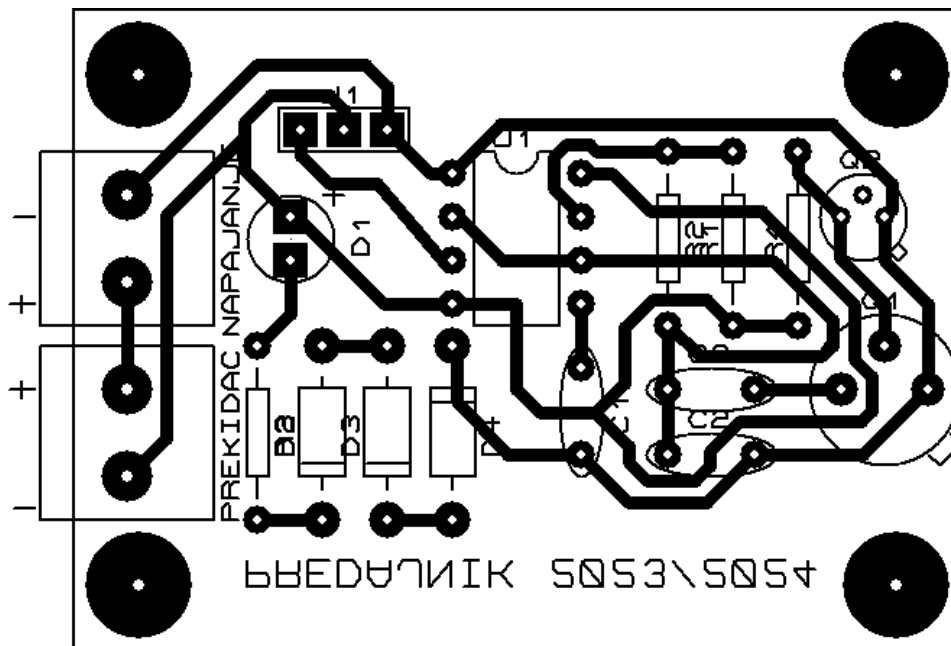
ZABRANJENA KORIŠTENJA DATE ŠTOPERICE U KRIMINALNE SVRHE, TE SVRHE ZLOUPOTREBE I KRŠENJA ZAKONA.

Popis korištenih komponenata

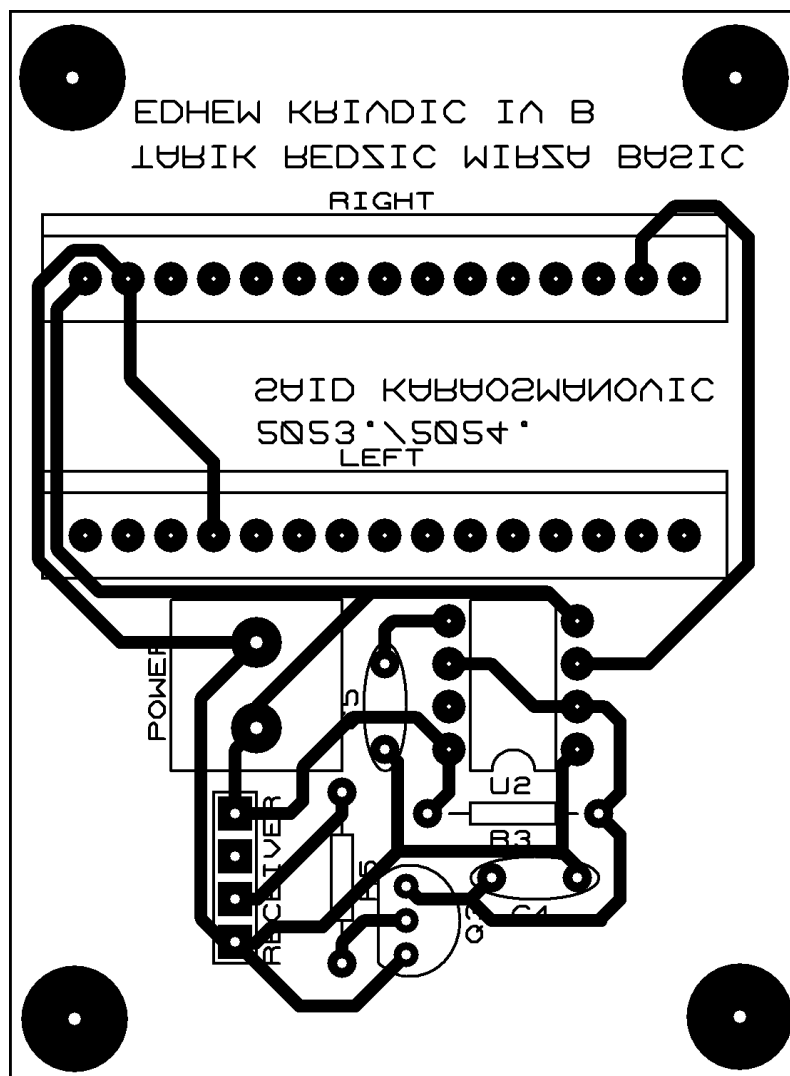
U ovom projektu su korištene sljedeće komponente.

- TBLOCK šarafni terminali
- Otpornici 15Ω , $1k\Omega$, $3k3\Omega$ i $10k\Omega$
- Kondenzatori $10nF$ i $100nF$
- Diode 1N4001 i 1N4733
- Tranzistori 2N2219
- LED (2 komada) za indicaciju
- Prekidač
- Laser (2 komada)
- Fototranzistor
- NE555 (2 komada)
- 1602 (16 karaktera x 2 reda) LCD
- CONN-SILL (muško-ženski konektori)
- Arduino Nano
- FS1000A i XY-MK-5V 433MHz kit predajnik i prijemnik

Za napajanje baterija koristiti AAA baterije (9 komada) i 9V baterija 6F22.



PCB predajnika



PCB prijemnika

Kod

Kod datog projekta dat je u produžetku:

```
const int start = 13, stopp = 6;
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;
#include <LiquidCrystal.h>
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
unsigned long trenutno_vrijeme, sekunde, milisekunde, interval, proslo_vrijeme = 0;
bool istina = true; // služi za provjeru da li se prvi put pokrene funkcija
stopperica u rekurziji iste funkcije
void stopperica();
void kraj();
void setup(){
    pinMode(stopp, INPUT);
    pinMode(start, INPUT);
    lcd.begin(16, 2);
    lcd.print("Rising Edge(TM)");
    delay(3000);
    lcd.clear();
}
void loop(){
    lcd.setCursor(0, 0);
    lcd.print("vrijeme:");
    lcd.print("0:");
    lcd.print("0");
    if (digitalRead(start) == HIGH) {
        istina = true;
        stopperica();
    }
}
void stopperica(){
    if (istina == true) {
        trenutno_vrijeme = millis();
        proslo_vrijeme = trenutno_vrijeme;
    }
    istina = false;
    lcd.clear();
    trenutno_vrijeme = millis();
    interval = trenutno_vrijeme - proslo_vrijeme;
    sekunde = interval / 1000;
    milisekunde = interval - sekunde * 1000;
    lcd.setCursor(0, 0);
    lcd.print("vrijeme: ");
    lcd.print(sekunde);
    lcd.print(":");
    lcd.print(milisekunde);
    delay(100);
    digitalRead(stopp) == HIGH ? stopperica() : kraj() ; // obrnuti kraj i stoppericu
    za projekat(signal se očitava low na finalnoj shemi) // prvo stopperica() pa kraj
}
void kraj(){
```

```

    lcd.clear();
    lcd.setCursor(0, 0);
    interval = trenutno_vrijeme - proslo_vrijeme;
    sekunde = interval / 1000;
    milisekunde = interval - sekunde * 1000;
    lcd.setCursor(0, 0);
    lcd.print("VRIJEME: ");
    lcd.print(sekunde);
    lcd.print(":");
    lcd.print(milisekunde);
    delay(10000);
    lcd.clear();
    interval = 0;
    sekunde = 0;
    milisekunde = 0;
    return;
}

```

OPIS KODA

Kod započinje deklaracijom pinova za LCD:

```

#include <LiquidCrystal.h>
const int rs = 12, en = 11, d4 = 5, d5 = 4, d6 = 3, d7 = 2;

```

što je uslijedilo sa inicijalizacijom korištenih pinova na Arduino:

```
const int start = 13, stopp = 6;
```

Inicijalizacija display-a vrši se preko komande:

```
LiquidCrystal lcd(rs, en, d4, d5, d6, d7);
```

Zatim su deklarirane/inicijalizirane korištene varijable

```

unsigned long trenutno_vrijeme, sekunde, milisekunde, interval, proslo_vrijeme = 0;
bool istina = true; // služi za provjeru da li se prvi put pokrene funkcija
stopperica u rekurziji iste funkcije

```

te su deklarirane korištene funkcije:

```

void stopperica();
void kraj();

```

Naravno, sada slijedi uređenje **setup** funkcije, gdje se deklariraju ulazni i izlazni pin-ovi, te formatiranje LCD kolona i redova:

```

void setup() {
    pinMode(stopp, INPUT);
    pinMode(start, INPUT);
    lcd.begin(16, 2);
    lcd.print("Rising Edge(TM)");
    delay(3000);
    lcd.clear();
}

```

Zatim slijedi formatiranje **loop** funkcije u kojoj se poziva **štoperica**:

```

void loop(){
  lcd.setCursor(0, 0);
  lcd.print("vrijeme:");
  lcd.print("0:");
  lcd.print("0");
  if (digitalRead(start) == HIGH) {
    istina = true;
    stopperica();
  }
}

```

Funkcija loop će se konstantno ponavljati, te u komandi

```
lcd.setCursor(0, 0);
```

kursor se postavlja na red 0, kolona 0. Zatim se vrijeme ispisuje koje je jednako 0, jer odbrojanje nije uopšte počelo.

```

lcd.setCursor(0, 0);
lcd.print("vrijeme:");
lcd.print("0:");
lcd.print("0");

```

Stanje će biti očitano sa pina `start` i ako je ono jednako `HIGH`, onda će varijabla „**istina**“ dobiti vrijednost `true` i poziva se funkcije `stopperica()`.

```

if (digitalRead(start) == HIGH) {
  istina = true;
  stopperica();
}

```

Ako uslov nije ispunjen funkcija `loop()` vrši rekurziju, a ako je ispunjen funkcija `stopperica()` vrši sljedeće.

Provjerava se varijabla „**istina**“ koja služi da compiler spozna da li je ovo prvi put da se funkcija poziva, odnosno da nije vršena rekurzija. Ako je „**istina**“ jednaka `true` onda će varijabla „**trenutno_vrijeme**“ dobiti vrijednost funkcije `millis()` čiji je povratni tip `unsigned long` i onda će varijabla „**proslo_vrijeme**“ dobiti vrijednost varijable „**trenutno_vrijeme**“. Obavezno je da se osigura da izvršavanje ovog dijela koda samo prvi put kako bi dobili tačno vrijeme i zbog toga se koristi varijabla „**istina**“.

```

if (istina == true) {
  trenutno_vrijeme = millis();
  proslo_vrijeme = trenutno_vrijeme;
}

```

Zatim radi osiguranja protiv ponavljanja ovog dijela koda varijabla „**istina**“ se postavlja na **false**.

```
istina = false;
```

Funkcijom

```
lcd.clear();
```


briše se sav trenutni ispis na LCD.

Vrijeme se formatira pomoću sljedećeg bloka koda koji koristi aritmetičke operatore.

```
trenutno_vrijeme = millis();  
interval = trenutno_vrijeme - proslo_vrijeme;  
sekunde = interval / 1000;  
milisekunde = interval - sekunde * 1000;
```

Kursor se postavlja na red 0 i kolonu 0 i ispisuje se vrijeme pomoću sljedećeg bloka koda.

```
lcd.setCursor(0, 0);  
lcd.print("vrijeme: ");  
lcd.print(sekunde);  
lcd.print(":");  
lcd.print(milisekunde);
```

Da bi se stanje sa LCD moglo očitati, odnosno da bi se osiguralo sporije osvježavanje LCD-a koristit će se delay od 100ms.

```
delay(100);
```

Koristi se ternarni operator da bi se provjerilo stanje varijable „stop“, odnosno da li je korisnik štoperice završio s trčanjem. Ukoliko jeste poziva se funkcija „kraj()“, a ako nije vrši se rekurzija, odnosno poziva se funkcija „stopperica()“.

Kao što je navedeno, ako je uslov ispunjen poziva se funkcija „kraj()“, a ona vrši sljedeće:

Napočetku čisti se postojeći ispis sa LCD-a, postavlja se kursor na red 0 kolona 0.

```
lcd.clear();  
lcd.setCursor(0, 0);
```

Ponovno formatiranje vremena:

```
interval = trenutno_vrijeme - proslo_vrijeme;  
sekunde = interval / 1000;  
milisekunde = interval - sekunde * 1000;
```

Ponovni ispis, ali ovog puta finalni:

```
lcd.setCursor(0, 0);  
lcd.print("VRIJEME: ");  
lcd.print(sekunde);  
lcd.print(":");  
lcd.print(milisekunde);
```

Postavlja se delay od 10000ms(10s) radi lakšeg očitavanja vremena s LCD-a.

```
delay(10000);
```

Varijable se resetuju (vrijednost im se postavlja na 0)

```
interval = 0;
```

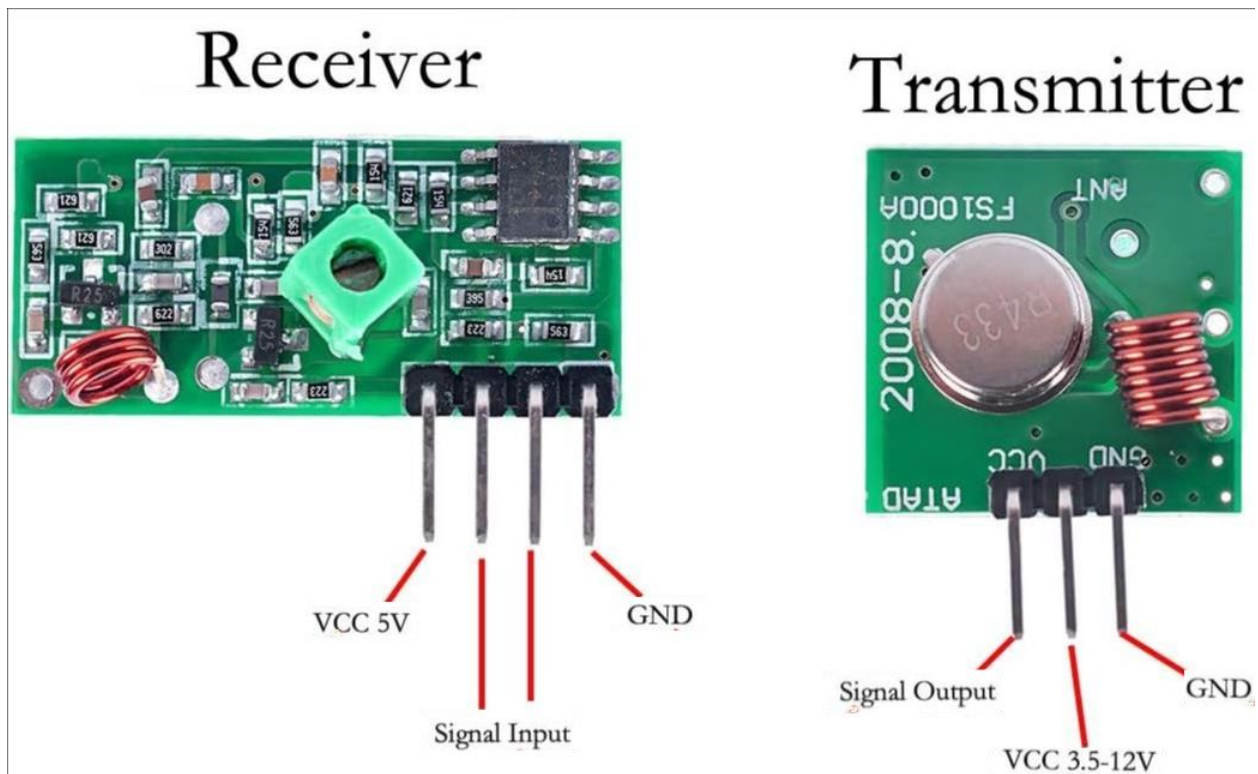
```
sekunde = 0;  
milisekunde = 0;
```

Te naredba return koja prekida void funkciju i kod se ponavlja ispočetka.

```
return;
```

Hardware-ska izvedba

FS1000A i XY-MK-5V



FS1000A i XY-MK-5V modul (set prijemnik i predajnik) su korišteni u hardverskoj izvedbi

FS1000A tehnički podaci:

Napon rada	+5 V
Frekvencija	433MHz
Daljina prenosa(bez/sa antenom)	3m / 100m
Tehnika modulacije	Amplitude Shift Keying
Brzina prenosa	10 Kbps
Struja rada	9-40 mA

FS1000A Pinout:

- DATA
- Vcc
- GROUND

XY-MK-5V tehnički podaci:

Napon rada	5V
Struja rada	4 mA
Osjetljivost	105 dB
Eksterna antena	Signalni provodnik

XY-MK-5V Pinout:

- Vcc
- 2 x DATA
- GROUND

Opći princip rada

Štoperica radi na sljedeći način:

- 2 lasera su postavljenja na početku i kraju utrke.
- Okidanjem prvog lasera, preko radio valova šalje se signal na kraj, te štoperica kreće sa radom. Okidanje prvog lasera nakon prvog puta ne radi ništa.
- Okidanjem drugog lasera, štoperica stane sa radom, na displej izbacuje vrijeme i to stoji 10s. Poslije 10s, ekran se automatski očisti, te štoperica je spremna za ponovni rad. Okidanjem drugog lasera nakon prvog puta ne radi ništa.

Projekat se može koristiti za mjerenje brzine, recimo, automobila na već poznatoj udaljenosti. Mora se napomenuti da domet hvatanja radio predajnika i prijemnika zavisi od kvaliteta antena korištenih. U ovom projektu, korištene su antene dometa cirka 100m.

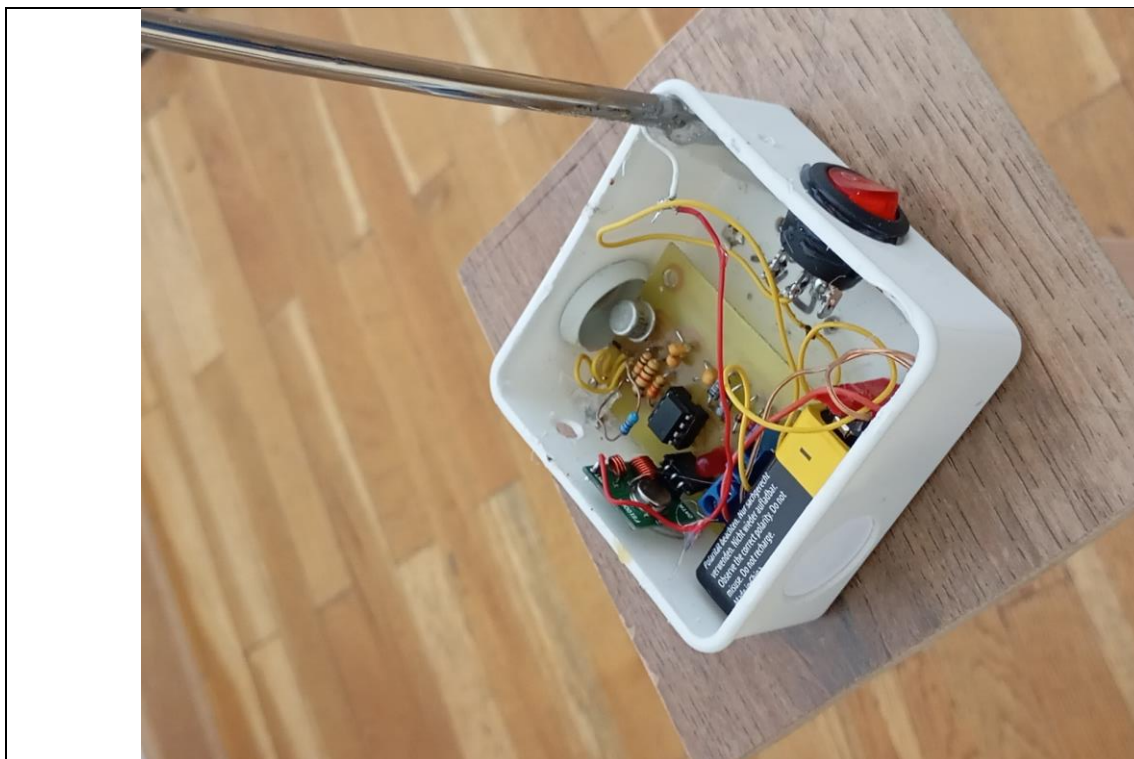
Slike projekta



Finalni izgled projekta



Izgled jedne od laserskih kutija



Izgled predajnika



Izgled prijemnika

KRAJ DOKUMENTA