

LJETNI ROBOTIČKI KAMP

«PETICA» 2017.

ARDUINO RADIONICA - početna

Hrvatski robotički savez

Doc. dr. sc. Ana Sović Kržić

Ivanić – Grad, srpanj 2017.

1. Little Bits

ZADATAK 1. Moj prvi strujni krug.

Svaki strujni krug počinje baterijom i elementom za napajanje. Pronađite ih u kutiji i spojite.

Pronađite i dugačku LED-icu. Spojite i nju.

Uključite sklopku na napajanju.

Što se dogodilo?



ZADATAK 2. Prekidač za svjetlo.

Pronađite tipkalo i dodajte ga u strujni krug između napajanja i LED-ice.

Kada je tipkalo pritisnuto:

- a) LED-ica svijetli,
- b) LED-ica ne svijetli,
- c) nema razlike kada je pritisnuto i kada nije.



ZADATAK 3. Potenciometar.

Pronađite potenciometar. Ubacite ga u strujni krug umjesto tipkala.

Pomičite ručicu potenciometra lijevo – desno.

Što se događa sa svjetлом?

- a) Jednako svijetli u svim položajima,
- b) svijetli jače kada je lijevo,
- c) svijetli jače kada je desno.



ZADATAK 4. Sastavite strujni krug koji se sastoji od: baterije, elementa za napajanje, LED-ice, tipkala i potenciometra. Neka LED svijetli ako je tipkalo pritisnuto. Pomicanjem ručice potenciometra odredite jačinu svjetla.

Kojim redom ste spojili elemente? Upišite brojeve.

_____ baterija _____ element za napajanje _____ LED-ica _____ tipkalo _____ potenciometar

ZADATAK 5. Brojač.

Pronađite 7-segmentni zaslon. Spojite strujni krug: napajanje, tipkalo, zaslon.

Prekidač na zaslonu postavite u položaj «count – up». Tipkalo pritisnite nekoliko puta.

Koji najveći broj možete dobiti? _____



Prekidač na zaslonu postavite u položaj «count – down». Tipkalo pritisnite nekoliko puta.

Koji najmanji broj možete dobiti? _____

ZADATAK 6. Zaslon i potenciometar.

Spojite strujni krug koji se sastoji od napajanja, potenciometra i 7-segmentnog zaslona. Prekidač na zaslonu postavite u položaj «read – value». Pomičite ručicu potenciometra lijevo – desno.

Što se ispisuje na zaslonu? _____

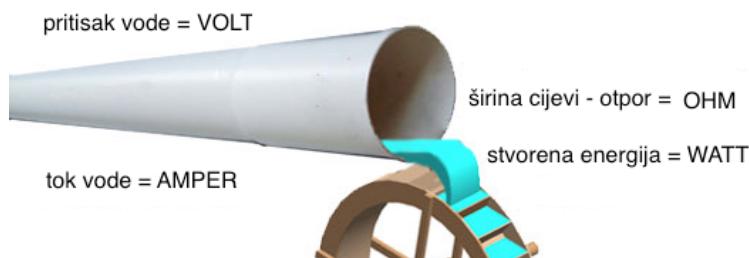
Na kraj dodajte LED-icu. Koji broj piše na ekranu kada LED-ica svijetli jako? _____

Koji broj piše na ekranu kada LED-ica prestaje svijetliti? _____

Prebacite prekidač na zaslonu u položaj «read – volts». Koji brojevi sada pišu?

Što su to volti?

“Ohm's Law”



ZADATAK 7. Senzor temperature.

Pronađite senzor temperature. Spojite strujni krug: napajanje, senzor temperature, zaslon. Prekidač na zaslonu postavite u položaj «read – value».

Kolika je temperatura zraka? _____



Pritisnite prst na senzor. Je li temperatura porasla? DA NE

ZADATAK 8. Senzor osvjetljenja.

Pronađite senzor osvjetljenja. Spojite strujni krug: napajanje, senzor osvjetljenja, zaslon. Prekidač na zaslonu postavite u položaj «read – value».

Što morate napraviti kako bi na zaslonu vidjeli broj 0?



Koji najveći broj možete dobiti? _____

Što ste napravili kako bi to postigli? _____

ZADATAK 9. Generator impulsa.

Pronađite generator impulsa. Spojite strujni krug: napajanje, generator impulsa, LED-ica.

Kako sada LED-ica svijetli?



- a) Skoro nevidljivo,
- b) vrlo jako,
- c) pali se i gasi se.

ZADATAK 10. Inverter.

Spojite krug: napajanje, tipkalo i RGB dioda.

Što se događa kada pritisnete tipkalo?

- a) Dioda svijetli,
- b) dioda ne svijetli.

Kojom bojom svijetli dioda? _____



Dodajte sada inverter između tipkala i RGB diode.



Kada dioda svijetli?

- a) Kada pritisnemo tipkalo,
- b) kada tipkalo nije pritisnuto.

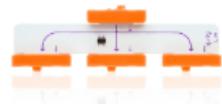
ZADATAK 11. Policijska svjetla.

Spojite strujni krug koji će naizmjence paliti i gasiti 2 LED svjetla (RGB dioda i dugačka LED-ica). Kada je jedno svjetlo upaljeno, drugo mora biti ugašeno i obratno.

Navedite koje elemente ste koristili i kojim redoslijedom.

ZADATAK 12. Policijska sirena.

Pronađite element za grananje i buzzer.



Na napajanje spojite element za grananje. U jednu granu elementa za grananje spojite buzzer. U drugu granu spojite elemente iz prošlog zadatka (policijska svjetla).

Svira li i svijetli li vaš uređaj istovremeno? DA NE

Koji element morate koristiti kako bi mogli mijenjati glasnoću sirene?



Spojite taj element i testirajte.

ZADATAK 13. Napravite alarm za torbu. Kada je alarm u mračnoj torbi, on šuti. Kada se torba otvorí i svjetlo padne na alarm, on počne svirati.

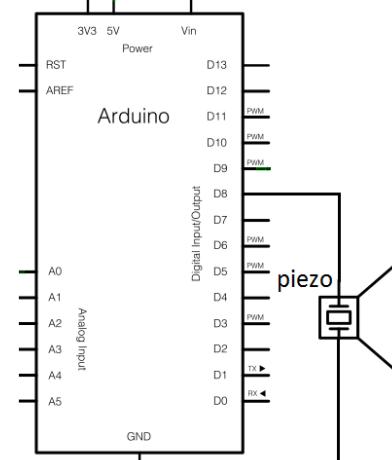
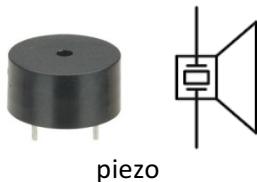
ZADATAK 14. Od dijelova u kutiji sastavite vozilo koristeći kotače i motore.

ZADATAK 15. Nadogradite vozilo tako da se vozi samo kada ga obasjate svjetlom sa mobitela.

ZADATAK 16. Napravite uređaj koji će crtati.

2. Svirajmo

ZADATAK 1. Spoji Arduino i piezo element prema shemi.



ZADATAK 2. Programiraj Arduino prema uputama.

```
#define NOTA_C4 262
#define NOTA_D4 294
#define NOTA_E4 330
#define NOTA_F4 349
#define NOTA_G4 392
#define NOTA_A4 440
#define NOTA_B4 466
#define NOTA_H4 494
#define NOTA_C5 523

int PinPiezo = 8;

void setup(){
    pinMode(PinPiezo, OUTPUT);
}

void loop(){
    tone(PinPiezo, NOTA_C4, 100);
    delay(100);
    tone(PinPiezo, NOTA_A4, 200);
    delay(200);

    noTone(PinPiezo);
    delay(300);
}
```

Testiraj. Što se dogodilo?

ZADATAK 3. Arduino program se sastoji od sljedećih dijelova:

1. **definicija** konstanti i varijabli
2. **inicijalizacija** koja će se izvršiti samo jednom prilikom prvog pokretanja programa
3. **glavni program** koji se neprestano izvršava

Zaokruži i obilježi na gornjem programu navedene dijelove.

ZADATAK 4. Poveži objašnjenje s naredbom.

tone();	definiranje na kojem je pinu spojen ulazni ili izlazni elektronički element
delay();	odsviraj notu
pinMode();	pauza

ZADATAK 5. Poveži objašnjenja s dijelovima naredbe.

```
tone(piezoPin, NOTA_A4, 200);
```

koju notu odsvirati	koliko dugo svirati tu notu u ms	na koji pin je spojen piezo element
---------------------	----------------------------------	-------------------------------------

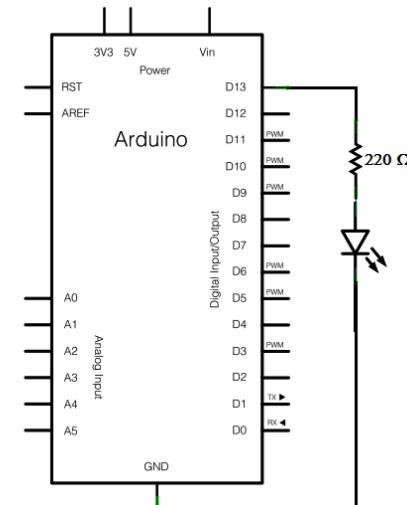
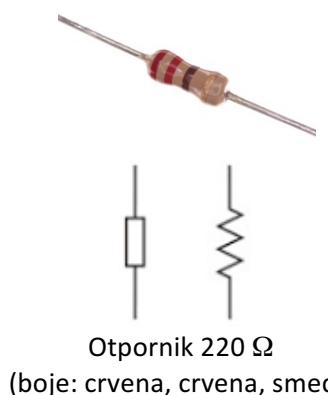
ZADATAK 6. Odsviraj „Sretan rođendan“.

Koliko mora biti trajanje najbrže note da bi pjesma bila brzine koju poznaješ?

ZADATAK 7. Skladaj svoju pjesmu!

3. Kako radi semafor

ZADATAK 1. Spoji crvenu LED diodu na Arduino prema shemi.



ZADATAK 2. Programiraj Arduino prema uputama.

```
int LEDPin1 = 13;

void setup(){
    pinMode(LEDPin1, OUTPUT);
    digitalWrite(LEDPin1, LOW);
}

void loop(){
    digitalWrite(LEDPin1, HIGH);
    delay(500);
    digitalWrite(LEDPin1, LOW);
    delay(500);
}
```

Testiraj. Što se dogodilo?

ZADATAK 3. Poveži objašnjenje s naredbom.

- | | |
|------------------------------|---|
| <code>digitalWrite();</code> | definiranje na kojem je pinu spojen ulazni ili izlazni elektronički element |
| <code>delay();</code> | uključi ili isključi LED diodu |
| <code>pinMode();</code> | pauza |

ZADATAK 4. Koji od navedenih objašnjenja su točni?

- a. `digitalWrite(LEDPin1, LOW);` Neka LED dioda svijetli
- b. `digitalWrite(LEDPin1, LOW);` Neka se LED dioda ugasi
- c. `delay(500);` Pričekaj 500 ms (pola sekunde)

ZADATAK 5. Promijeni program tako da LED dioda blinka svakih 1 sekundu.

ZADATAK 6. Spoji sada jednu žutu LED diodu, na pin broj 12. Nemoj zaboraviti spojiti i otpornik od $220\ \Omega$. Prilikom spajanja elektroničkih elemenata, isključi Arduino.

ZADATAK 7. Programiraj Arduino tako da obje diode blinkaju istovremeno.

Naredbe koje koristimo su:

```
int LEDPin2 = 12;
pinMode(LEDPin2, OUTPUT);
digitalWrite(LEDPin2, LOW);
digitalWrite(LEDPin2, HIGH);
digitalWrite(LEDPin2, LOW);
```

Dopiši ih na pravo mjesto u programu:

```
int LEDPin1 = 13;

void setup(){
    pinMode(LEDPin1, OUTPUT);

    digitalWrite(LEDPin1, LOW);

}

void loop(){
    digitalWrite(LEDPin1, HIGH);

    delay(500);
    digitalWrite(LEDPin1, LOW);

    delay(500);
}
```

ZADATAK 8. Pozovi voditelja koji će provjeriti program i spoj. Ako je sve točno spojeno, uključi Arduino.

Prebac program. Blinkaju li LED diode kako smo htjeli? DA NE

ZADATAK 9. Promijeni program kako bi LED diode blinkale NAIZMJENIČNO.

ZADATAK 10. Dodaj jednu zelenu LED diodu na pin broj 11. U seriju spoji jedan otpornik od $220\ \Omega$.

ZADATAK 11. U program iz zadatka 8 dodaj naredbe kako bi i zelena LED dioda blinkala zajedno s crvenom i žutom diodom.

ZADATAK 12. Promijeni program tako da diode svijetle naizmjenično: svaka po 300 ms.

ZADATAK 13. Promijeni program tako da prvo svijetli crvena dioda 3 sekunde, zatim žuta i crvena zajedno 1 sekundu, pa samo zelena 3 sekunde. Za kraj svijetli samo žuta 1 sekundu. Ovo treba ponavljati sve dok se Arduino ne isključi.

Čestitamo – napravio si semafor!

ZADATAK 14. Napravi semafor kojem će na početku sve LED diode svijetlili. Redom isključuj jednu po jednu diodu, dok ostale ostaju svijetliti. Dobio si trčeću tamu!

ZADATAK 15. Napravi „trčeću tamu“ koristeći 5 crvenih dioda.

ZADATAK 16. Animiraj 5 crvenih dioda tako da su početno sve isključene. Neka se prvo upale vanjske dvije na 600 ms. Nakon što se one ugase, upale se dvije diode pored njih, na 400 ms. Srednja dioda neka svijetli sama i to 500 ms.

ZADATAK 17. Napravi cvijet koji svijetli. Rasporedi 5 crvenih dioda u krug i jednu žutu diodu u sredinu. Neka latice cvijeta naizmjenično svijetle, svaka po 400 ms. Sredina cvijeta neka svijetli bez prestanka.

ZADATAK 18. Napravi cijelo križanje: jedan semafor za aute (crveno, žuto, zeleno), te jedan semafor za pješake (crveno, zeleno). Programiraj Arduino tako da diode svijetle kao pravo križanje.

ZADATAK 19. Rasporedi diode raznih boja na protoboard, te isprogramiraj smajlića.

4. Tipkalo

ZADATAK 1. Spoji tipkalo i LED diodu na Arduino prema shemi.

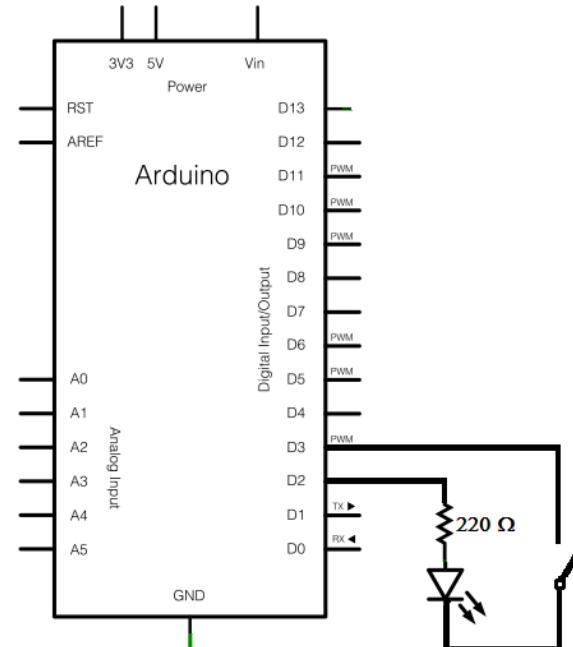


tipkalo

4 nožice, po dvije kratko spojene
prijetlog: na Arduino spojiti
dijagonalne nožice



Otpornik $220\ \Omega$
(boje: crvena, crvena, smeđa)



ZADATAK 2. Prepiši program u Arduino.

```
int PinLed = 2;
int PinTipkalo = 3;
int StanjeTipkala;

void setup() {
    pinMode(PinLed, OUTPUT);
    pinMode(PinTipkalo, INPUT_PULLUP);
    digitalWrite(PinLed, LOW);
}

void loop() {
    StanjeTipkala = digitalRead(PinTipkalo);
    if (StanjeTipkala == LOW) {
        digitalWrite(PinLed, HIGH);
    } else {
        digitalWrite(PinLed, LOW);
    }
}
```

Što program radi?

Na koji pin je spojeno tipkalo? _____

Je li tipkalo ulazni ili izlazni element? _____

Kojom naredbom si to definirao u programu? _____

Znak = je znak pridruživanja. DA NE

Znak == se koristi za provjeru je li nešto jednako. DA NE

Kada je tipkalo pritisnuto, njegovo stanje je LOW. DA NE

Kada LED dioda svijetli, njeno stanje je LOW. DA NE

ZADATAK 3. Promijeni program tako da LED dioda svijetli kada je tipkalo otpušteno.

ZADATAK 4. Dodaj jednu zelenu LED diodu. Neka obje LED diode svijetle kada je tipkalo pritisnuto. Kada je tipkalo otpušteno, neka LED diode ne svijetle.

ZADATAK 5. Promijeni program tako da crvena LED dioda svijetli kada je tipkalo pritisnuto, a zelena LED dioda kada je tipkalo otpušteno.

ZADATAK 6. Spoji još jedno tipkalo. Napiši program koji će paliti crvenu LED diodu kada je pritisnuto prvo tipkalo, a gasiti crvenu LED diodu kada je pritisnuto drugo tipkalo.

ZADATAK 7. Napiši program koji na pritisak prvog tipkala pali crvenu LED diodu i gasi zelenu LED diodu. Na pritisak drugog tipkala radi suprotno: gasi crvenu LED diodu i pali zelenu LED diodu.

ZADATAK 8. Napiši program koji će na pritisak prvog tipkala upaliti crvenu diodu na pola sekunde, pa zelenu na pola sekunde. Na pritisak drugog tipkala će upaliti crvenu diodu na 2 sekunde, pa zelenu na 2 sekunde.

ZADATAK 9. Napiši program koji će na svaki pritisak tipkala mijenjati stanje LED dioda: kada se prvi puta pritisne, diode će svijetliti. Kada se sljedeći puta pritisne, diode će se ugasiti.

Pomoć: nakon provjere je li tipkalo pritisnuto, potrebno je provjeriti je li LED dioda ugašena ili upaljena. Nakon pritiska tipkala, potrebno je promijeniti stanje diode.

```
int PinLed1 = 2;
int PinLed2 = 4;
int PinTipkalo1 = 3;
int StanjeTipkala1;
int StanjeLED = LOW;

void setup() {
    pinMode(PinLed1, OUTPUT);
    pinMode(PinLed2, OUTPUT);
    pinMode(PinTipkalo1, INPUT_PULLUP);
    digitalWrite(PinLed1, LOW);
    digitalWrite(PinLed2, LOW);
}

void loop() {
    StanjeTipkala1 = digitalRead(PinTipkalo1);
    if (StanjeTipkala1 == LOW) {
        if (StanjeLED == LOW) {
            StanjeLED = HIGH;
        } else {
            StanjeLED = LOW;
        }
        delay(50);
        digitalWrite(PinLed1, StanjeLED);
        digitalWrite(PinLed2, StanjeLED);
    }
}
```

Radi li spoj svaki puta ispravno? DA NE

Koji problemi se javljaju? _____

Kako bismo riješili ovaj problem, prije promjene stanja diode dodaj čekanje od 50 ms.

5. Morseov kod

Morseov kod je izmislio Samuel Morse za šifriranje i prenošenje slova, brojeva i znakova koristeći odgovarajući niz udaraca, tonova ili svjetlosnih treptaja različitih dužina trajanja. Morseovom kodom se koriste spasilačke službe, radioamateri i vojnici. Npr. ostali ste zarobljeni na planini, po noći. Helikopter kruži oko planine i traži vas. Kako ćete mu signalizirati gdje se nalazite?

Oznaka za svako slovo se sastoji od određenog broja crtica i točkica. Kod prenošenja poruke vrijede pravila:

1. Točka je jedan kratak udarac, ton ili svjetlosni signal.
2. Crtica je traje 3 puta dulje od točke.
3. Razmak između pojedinih točaka ili crtica je trajanja 1 točke.
4. Pojedina slova se odvajaju tišinom u trajanju tri točke.

E .	T -	A · -	N - -	A ● -	U ● . ● -
I ...	M - -	U ... -	D - - -	B ● ● -	V ● . ● -
S ... -	O - - -	V ... - -	B - - - -	C ● ● -	W ● - -
H				D ● - -	X - - ● -
Z - - - -	G - - -	L - - - -	X - - - -	E ●	Y - - -
C - - - -	K - - -	F - - - -	Y - - - -	F ● - -	Z - - - -
P - - - -	J - - - -	R - - -	Q - - - -	G ● - -	
1 - - - - -	6 - - - -			H ● - -	
2 - - - - -	7 - - - -			J ● - -	
3 - - - - -	8 - - - - -			K - - -	
4 - - - - -	9 - - - - -			L - - -	
5 - - - - -	0 - - - - -			M - - -	
				N - - -	
				O - - -	
				P - - -	
				Q - - -	
				R - - -	
				S - - -	
				T - - -	
				1 ● - -	
				2 ● - -	
				3 ● - -	
				4 ● - -	
				5 ● - -	
				6 - - -	
				7 - - -	
				8 - - -	
				9 - - -	
				0 - - -	
				Č - - -	LJ - - -
				Ć - - -	NJ - - -
				DŽ - - -	Š - - -
				Đ - - -	Ž - - -

ZADATAK: osmislite tajnu poruku koju ćete prenijeti prijatelju koristeći Morseov kod i ton.

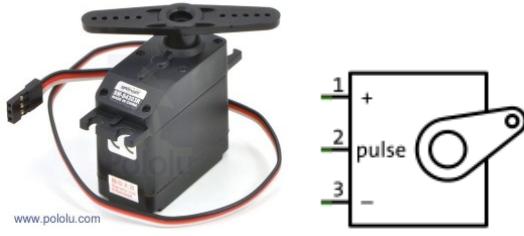
U ovom projektu ćemo prenijeti tajnu poruku koristeći tipkalo i piezo element. Spoji piezo element i tipkalo na Arduino. Kada je tipkalo pritisnuto čuje se nota A, kada tipkalo nije pritisnuto ne čuje se zvuk. Pošalji tajnu poruku prema tablici Morseovih kodova. Je li prijatelj uspio dekodirati?

POSLANA PORUKA: _____

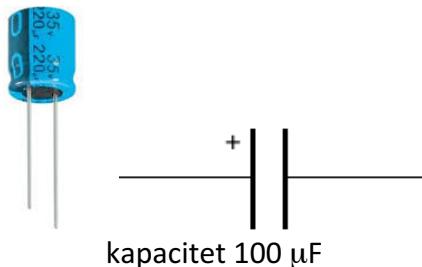
PORUKA PRIMLJENA OD PRIJATELJA: _____

6. Što će biti za večeru? (servomotor)

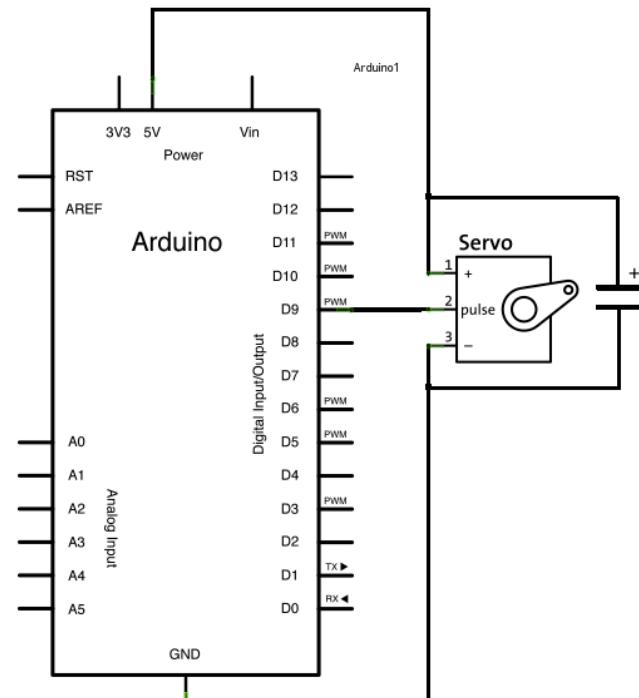
ZADATAK 1. Spoji servomotor na Arduino prema shemi.



servomotor
crvena žica – na 5V
bijela žica – na pin 9
crna žica – na GND



kraća nožica – minus strana – na GND
duža nožica – plus strana – na 5V



ZADATAK 2. Prepiši program koji će kazaljku servomotora postaviti na točno zadani kut.

Poveži naredbe sa značenjem:

```
#include <Servo.h>
Servo MojServo;
int kut;
void setup() {
    MojServo.attach(9);
}
void loop() {
    kut = 90;
    MojServo.write(kut);
}
```

slanje servomotora na točno određeni kut

uključi Servo.h biblioteku – omogući Arduinu korištenje naredbi koje su vezane uz servomotor

stvaranje objekta MojServo – svako kasnije pozivanja naziva MojServo direktno će biti razgovor sa servomotorom

zadajemo pin na koji je spojen servo motor

Koje kutove (položaje) servomotor može primiti? _____

ZADATAK 3. Promijeni program tako da kazaljka pokazuje kut 0 – početni položaj servo motora.

Poklapa li se taj položaj s položajem za koji si mislio da je početni položaj? DA NE

Što moraš učiniti kako bi kazaljka u početnom položaju bila okrenuta u smjeru u kojem želiš?

ZADATAK 4. Napiši program koji će naizmjence pomicati kazaljku na 0°, pa na 90°.

ZADATAK 5. Napiši program koji će pomicati kazaljku od 179° prema 0°, svaku milisekundu za 1 stupanj.

ZADATAK 6. Napiši program koji će pomicati kazaljku od 179° prema 0° , svaku milisekundu za 1 stupanj. Kada dođe do 0° neka krene iz početka, od 179° .

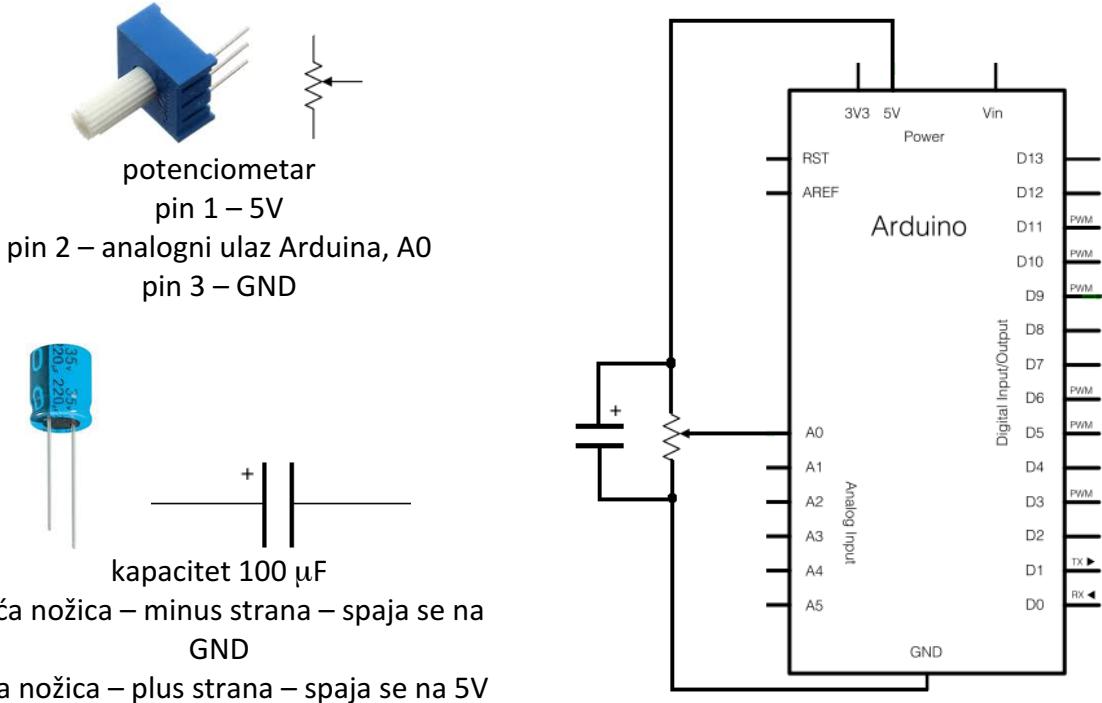
ZADATAK 7. Napravi uređaj koji će proricati što će biti za večeru. Izreži iz kartona jedan polukrug i podijeli ga u više kružnih isječaka. U svaki isječak unesi jednu vrstu hrane. Kazaljka servomotora se mora polako pomicati i pokazivati na sve vrste hrane. Kada se pritisne tipkalo, kazaljka mora stati iznad jednog jela. Ono što će kazaljka pokazati – to će biti za večeru!

Upute: Spoji tipkalo na Arduino. Kada je tipkalo pritisnuto, kazaljka se pomiče kao u prošlom zadatku. Kada se tipkalo pusti – kazaljka stane na mjestu na kojem se tipkalo pustilo.



7. Potenciometar i serijski monitor

ZADATAK 1. Spoji potenciometar na Arduino prema shemi.



ZADATAK 2. Potenciometar je otpornik koji mijenja vrijednost okretanjem svoje ručice. Arduino može očitati u kojem se točno položaju nalazi ručica. Kako bismo očitali koji je to trenutno položaj, omogućit ćemo Arduinu zapisivanje očitane vrijednosti na ekran računala. Prepišite program i povežite koja naredba se odnosi na potenciometar, a koja na zapisivanje na ekran računala.

```
int const PinPot = A0;
int polozaj;

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

void loop() {
    polozaj = analogRead(PinPot);

    Serial.print("Vrijednost potenciometra: ");
    Serial.print(polozaj);

    delay(1000);
}
```

POTENCIOMETAR

PISANJE NA EKRAN RAČUNALA



Kako bismo vidjeli što se ispisalo na zaslon računala, nakon prebacivanja programa u Arduino pritisnite oznaku za serijski monitor (Serial monitor).

Vrti ručicu potenciometra. Što se ispisuje na ekran računala?

Ispisuje li se sav tekst u jedan red ili svako očitanje u novi red? ISTI RED NOVI RED

Kako bismo omogućili pisanje u novi red umjesto `Serial.print` treba upisati `Serial.println`. Testirajte. Koju najmanju vrijednost potenciometar može poprimiti? _____

Koju najveću vrijednost potenciometar može poprimiti? _____

Upiši koja naredba obavlja opisane funkcije:

definicija pina na koji je spojen potenciometar _____

čitanje vrijednosti s potenciometra _____

omogućavanje pisanja na ekran računala vrijednosti koje su pročitane na Arduinu _____

pisanje teksta na ekran računala BEZ prelaska u novi red _____

pisanje teksta na ekran računala SA prelaskom u novi red _____

čekanje _____

ZADATAK 3. Napravi program koji će odrediti što će biti za večeru koristeći potenciometar i servomotor. Odgovori prije spajanja i programiranja:

Potenciometar daje vrijednosti između _____ i _____.

Servomotor se može pozicionirati na kutove između _____ i _____.

Naredba koja će automatski pretvoriti minimalnu vrijednost s potenciometra na minimalnu vrijednost za servomotor i maksimalnu vrijednost s potenciometra na maksimalnu vrijednost za servomotor je

`kut = map(polozaj, minPotenciometar, maxPotenciometar, minServo, maxServo);`

Upiši na crtice točne brojeve:

`kut = map(polozaj, _____, _____, _____, _____);`

Ubaci ovu naredbu u program i ispiši na ekran računala vrijednost kuta. Testiraj. Naredbe koje moraš dodati u program su s desne strane. Obilježi kamo si ih dodao.

<pre>int const PinPot = A0; int položaj; void setup() { Serial.begin(9600); } void loop() { položaj = analogRead(PinPot); Serial.print("Vrijednost potenciometra: "); Serial.println(položaj); delay(1000); }</pre>	<pre>int kut; kut = map(položaj, 0, 1023, 0, 179); Serial.print("Vrijednost kuta: "); Serial.println(kut);</pre>
---	---

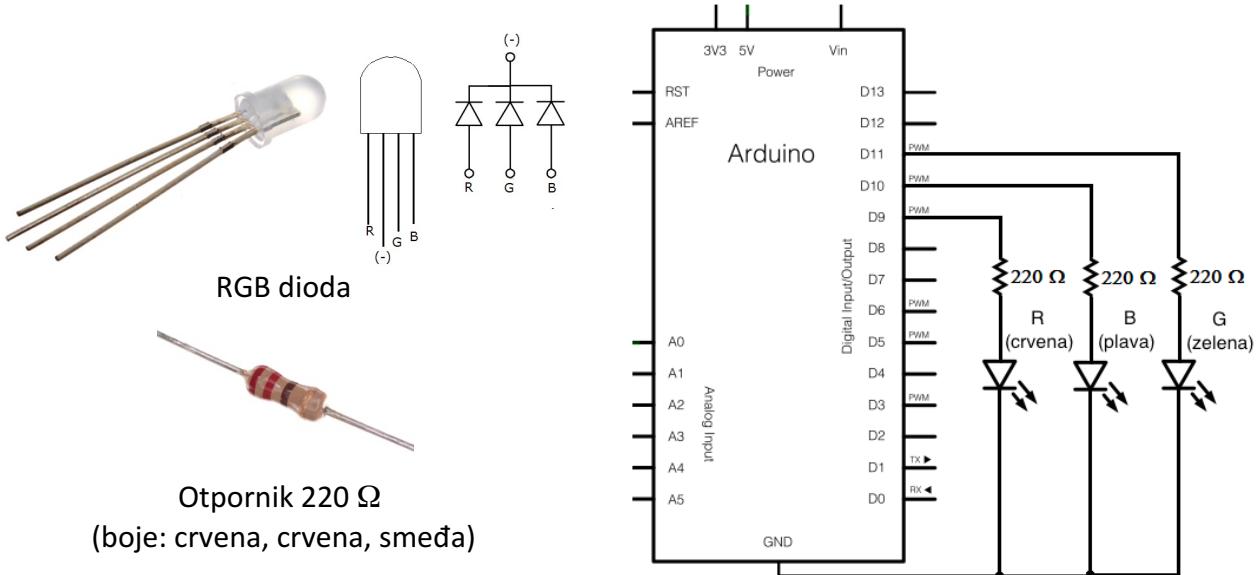
Spoji servomotor na Arduino. U postojeći program dodaj naredbe koje pokreću servomotor. Testiraj i provjeri miće li se servomotor na položaj koji si zadao pomoću potenciometra.

ZADATAK 4. Napravi instrument za sviranje koristeći potenciometar i piezo element. Koraci:

1. Spoji potenciometar i piezo element na Arduino
2. Napiši program koji sadrži:
 - naredbe koje čitaju vrijednosti sa potenciometra
 - naredba koja pretvara vrijednosti sa potenciometra u note za piezo element (frekvencije koje ljudi čuju su između 20 Hz i 20000 Hz, no u pjesmama se najčešće koriste 262 Hz do 523 Hz)
 - naredbe koje šalju note na piezo element
3. Testiraj
4. Napravite mali koncert. Odsviraj svoju pjesmu drugima.
5. Napravite orkestar i svirajte svi istu pjesmu istovremeno.

8. Disko kugla (RGB dioda)

ZADATAK 1. Spoji RGB diodu na Arduino prema shemi.



ZADATAK 2. Prepiši program koji određuje kojom bojom će svijetliti RGB dioda.

```
const int PinCrveni = 9;
const int PinPlavi = 10;
const int PinZeleni = 11;

void setup() {
    pinMode(PinCrveni, OUTPUT);
    pinMode(PinZeleni, OUTPUT);
    pinMode(PinPlavi, OUTPUT);
}

void loop() {
    analogWrite(PinCrveni, 155);
    analogWrite(PinPlavi, 155);
    analogWrite(PinZeleni, 0);
}
```

Kojom bojom svijetli LED dioda? _____

Koji brojevi moraju pisati u naredbama `analogWrite` kako bi dioda svijetlila bojama prema tablici:

boja	pin Crveni	pin Plavi	pin Zeleni
crvena	255	0	0
plava			
zelena			
žuta			
ljubičasta			
bijela			

ZADATAK 3. Napravite program koji će mijenjati boju RGB diode redom: crvena, žuta, zelena, crvena, žuta,... svakih 1 sekundu. Izrežite ping-pong lopticu i napravite kapicu iznad RGB diode. Dobili ste disk kuglu koja mijenja boje!

Napravi svoju verziju disk kugle koja će mijenjati boje koje ti voliš, brzo kako se tebi sviđa!

Zamračite prostoriju i napravite malu igru svjetla.

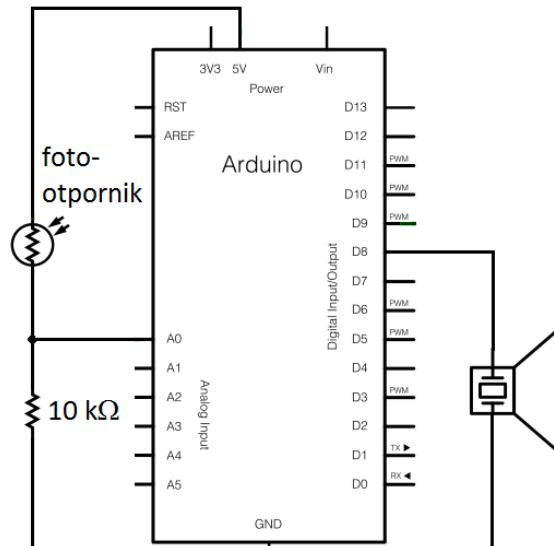
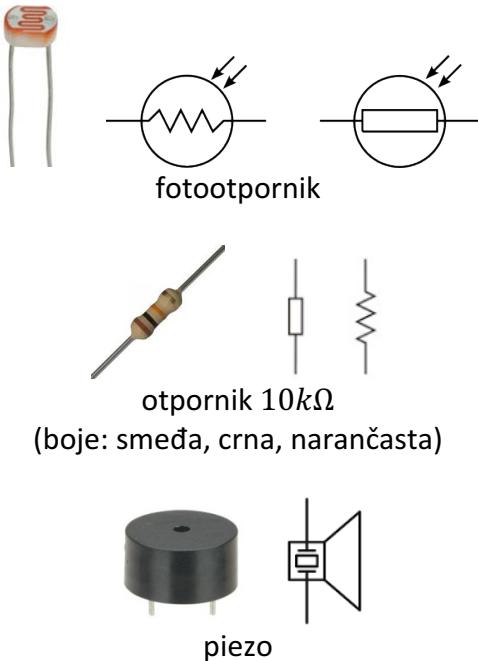
ZADATAK 4. Na Arduino spojite 3 potenciometra. Okretanjem prvog od njih definirajte koliko crvene boje neka bude u svjetlu RGB diode, okretanjem drugog odredite koliko plave boje, a okretanjem trećeg koliko zelene boje neka bude u svjetlu RGB diode.

Podsjetnik: potenciometar daje vrijednosti između 0 i 1023, dok RGB dioda za svaku svoju boju može primiti vrijednosti između 0 i 255.

Omogući serijski monitor kako bi mogao pratiti položaj potenciometra i vrijednosti boja koje šalješ RGB diodi.

9. Svjetlosno sviranje (fotoootpornik)

ZADATAK 1. Spoji fotoootpornik na Arduino prema shemi.



Fotoootpornik je otpornik koji je osjetljiv na svjetlo. Njegov otpor je manji ako je veći intenzitet ulazne svjetlosti. Spaja se na analogni ulaz Arduina. Arduino čita njegove vrijednosti kao broj između 0-1023.

ZADATAK 2. Napiši program koji će svirati različite note (frekvencije) ovisno koliko ruku približiš ili udaljiš od fotoootponika. Fotoootponik će biti naš „daljinski upravljač“.

Prije programiranja, složi ispravan redoslijed:

- _____ odsviraj ton
- _____ očitaj sa senzora
- _____ pretvori očitanu vrijednost sa senzora u frekvenciju
- _____ pričekaj prije sljedećeg očitanja

Prepiši program. Poveži ulogu pojedinih naredbi.

```
const int PinPiezo = 8;
int svjetlo;
int nota;

void setup() {
    pinMode(PinPiezo, OUTPUT);
}

void loop() {
    svjetlo = analogRead(A0);

    nota = map(svjetlo, 0, 1023, 50, 4000);

    tone(PinPiezo, nota, 20);

    delay(10);
}
```

pretvaranje vrijednosti sa senzora u frekvencije

definicija varijabli

inicijalizacija pinova

čitanje vrijednosti sa senzora

sviranje

definicija konstanti

ZADATAK 3. Promijeni minimalnu i maksimalnu frekvenciju u naredbi `map()`. Koje frekvencije moraš izabrati kako bi tvoj instrument svirao samo duboke tonove?

Koje frekvencije moraš izabrati kako bi tvoj instrument svirao samo visoke tonove?

Koja je najniža, a koja najviša frekvencija koju može čuti? Istraži na Internetu.

ZADATAK 4. Omogući ispis očitanja sa fotootpornika na zaslonu računala koristeći serijski monitor.

Koja je najniža vrijednost koju senzor može postići? _____

Je li tada mrak ili svjetlo? _____

Koja je najviša vrijednost koju senzor može postići? _____

Je li tada mrak ili svjetlo? _____

Promijeni naredbu `map()` kako bi pri sviranju koristili samo one vrijednosti očitanja senzora svjetla koje stvarno postoje.

ZADATAK 5. Neka svatko podesi drugačije frekvencije rada svojih instrumenata. Skladajte zajedničku pjesmu!

ZADATAK 6. Dodaj tipkalo. Neka instrument počne raditi nakon pritiska tipkala.

ZADATAK 7. KALIBRACIJA FOTOOTPORNika. Kako bi si olakšali posao i kako ne bismo trebali svaki puta čitati na serijskom monitoru vrijednosti osvjetljenja, moguće je napraviti automatsku kalibraciju fotootpornika. Kalibracija traje 5 sekundi i u tom vremenu trebalo osvijetliti fotootpornik svim mogućim osvjetljenjima kakva nas mogu okružiti za vrijeme sviranja. Program tada izgleda ovako:

```
int senzorDonja = 1023;
int senzorGornja = 0;

const int PinLed = 13;
const int PinPiezo = 8;

int svjetlo;
int nota;

void setup() {
    pinMode(PinPiezo, OUTPUT);

    pinMode(PinLed, OUTPUT);
    digitalWrite(PinLed, HIGH);

    while (millis() < 5000) {
        svjetlo = analogRead(A0);
        if (svjetlo > senzorGornja) {
            senzorGornja = svjetlo;
        }
        if (svjetlo < senzorDonja) {
            senzorDonja = svjetlo;
        }
    }
    digitalWrite(PinLed, LOW);
}

void loop() {
    svjetlo = analogRead(A0);
    nota = map(svjetlo, senzorDonja, senzorGornja, 50, 4000);

    tone(PinPiezo, nota, 20);
    delay(10);
}
```

10. LCD displej

ZADATAK 1. Spoji LCD displej na Arduino prema shemi.



Potenciometar – otpornik koji mijenja svoju vrijednost
spajanje nožica: zemlja – V0 LCD displeja – 5V

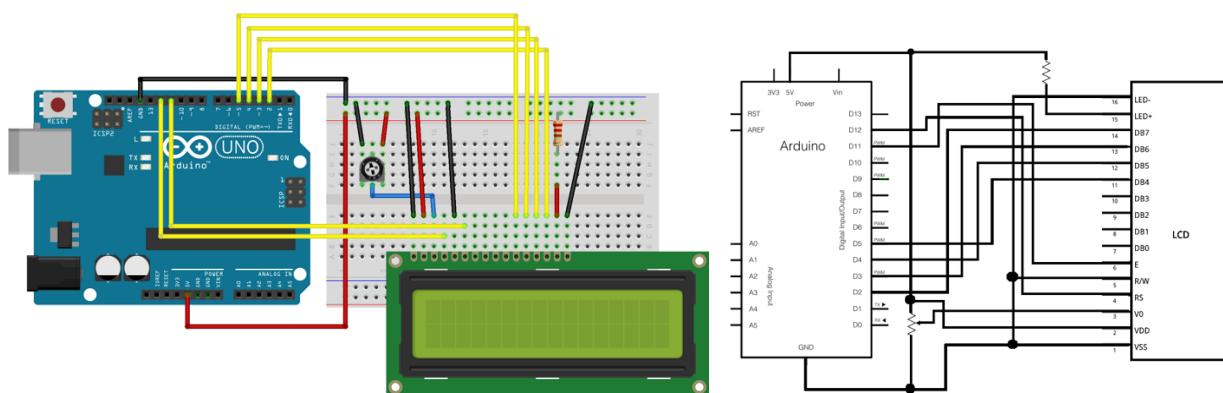
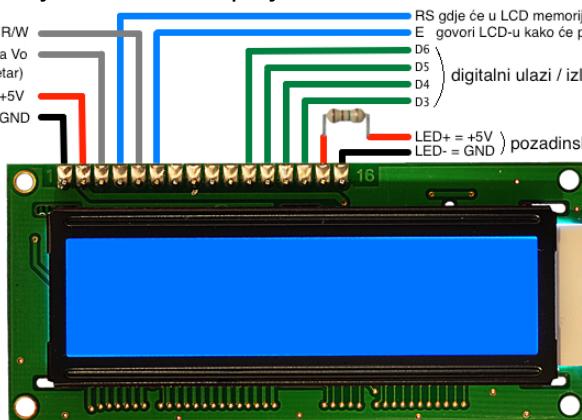


Otpornik $220\ \Omega$
(boje: crvena, crvena, smeđa)

čitanje (+5V) / pisanje (GND) = R/W
kontrast displeja V0
(na potenciometar)
napajanje ($V_{cc} = +5V$)
 $V_{ss} = GND$

RS gdje će u LCD memoriji biti zapisani podaci (na pin 12 Arduinoa)
E govorit LCD-u kako će primiti podatke (na pin 11 Arduinoa)

D6, D5, D4, D3 digitalni ulazi / izlaz
LED+ = +5V pozadinsko osvjetljenje
LED- = GND



ZADATAK 2. Zaokruži DA ili NE.

LCD displej se napaja s 5V.

DA NE

Potenciometar koristimo za podešavanje osvjetljenja zaslona.

DA NE

Za prijenos podataka se koristi 5 pinova.

DA NE

ZADATAK 3. Programiraj Arduino prema uputama.

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup() {
    lcd.begin(16, 2);
    lcd.print("Dobar dan.");
    lcd.setCursor(0, 1);
    lcd.print("Ja sam LCD!");
}

void loop() {
```

ZADATAK 4. Poveži naredbu i objašnjenje.

#include <LiquidCrystal.h>	ispis teksta na LCD displej
lcd.begin(16, 2);	pozivanje naredbi vezanih u LCD displej
lcd.print("Dobar dan.");	definiranje veličine LCD displeja
lcd.setCursor(0, 1);	postavljanje pokazivača na željenu poziciju pisanja na displej

ZADATAK 5. Promijeni program tako da u sredinu drugog retka ispiše: PETICA.

Na koju poziciju si morao postaviti cursor? _____

ZADATAK 6. Prepiši program.

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

void setup() {
    lcd.begin(16, 2);
}

void loop() {
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("PETICA");
    delay(1000);
    lcd.clear();
    lcd.setCursor(1, 0);
    lcd.print("PETICA");
    delay(1000);
}
```

Što radi naredba `lcd.clear()`?

- a) ispisuje na ekran «clear»
- b) briše sve na ekranu
- c) postavlja ispisivanje teksta na početak ekrana

Što radi naredba `delay(1000)`?

- a) ispisuje na ekran «delay»
- b) briše sve na ekranu
- c) čeka 1 sekundu prije izvođenja sljedeće naredbe

Što se ispisuje na ekranu?

ZADATAK 7. Napišite program koji će riječ «PETICA» kao da putuje s lijeve strane ekrana na desnu.

ZADATAK 8. Napišite program koji će riječ «PETICA» kao da putuje s desne strane ekrana na lijevu.

ZADATAK 9. Kreiraj svoj znak.

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);

byte smiley[8] = {
    B00000,
    B10001,
    B00000,
    B00000,
    B10001,
    B01110,
    B00000,
};

void setup() {
    lcd.createChar(0, smiley);
    lcd.begin(16, 2);
    lcd.write(byte(0));
}

void loop() {
```

Što prikazuje znak koji si kreirao? _____

ZADATAK 10. Promijeni izgled svog znaka.

ZADATAK 11. Neka tvoj znak popunjava LCD displej: prvi stupac, drugi stupac, treći stupac...

ZADATAK 12. Kreiraj dva znaka. Popuni cijeli ekran naizmjence sa tvoja dva znaka.

ZADATAK 13. Neka se u prvom retku, 5 stupac naizmjenice pojavljuju tvoja dva znaka.