



Arduino radionica

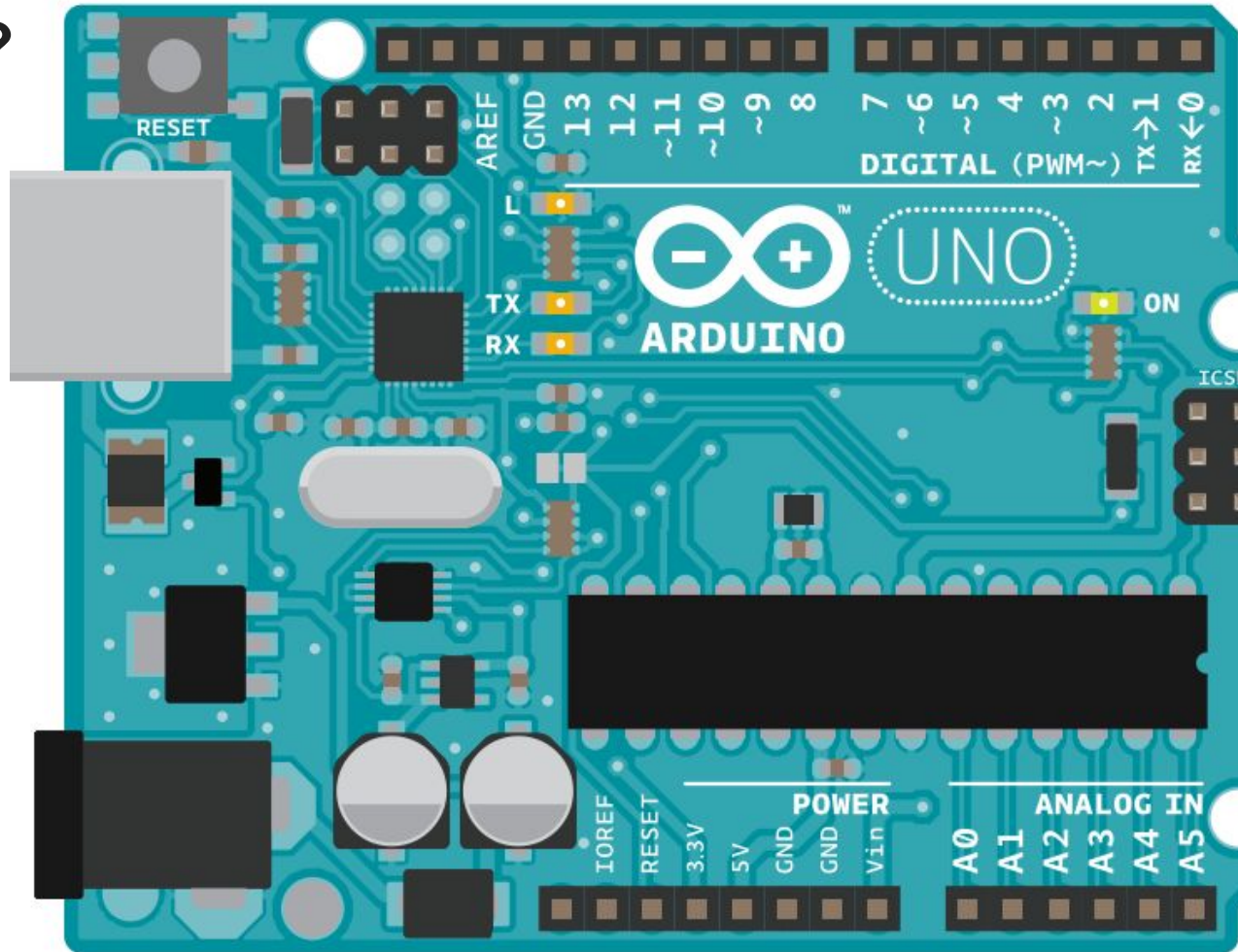
osnove

Plan motivacija

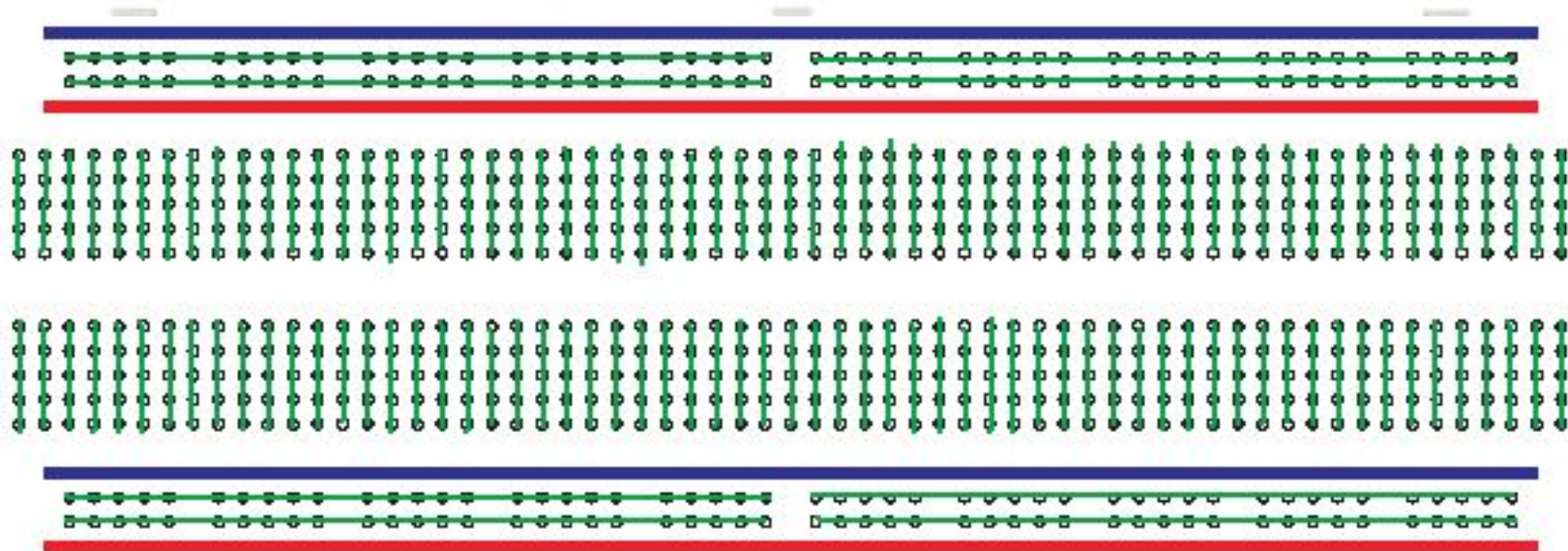
- Upoznati Arduino Uno pločicu te Arduino IDE (razvojno okruženje)
- Naučiti osnovne pojmove u programiranju
- Samostalno nadograditi postojeće primjere te riješiti zadatke

Što ćemo koristiti ?

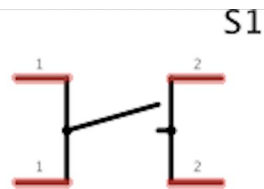
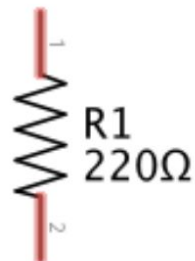
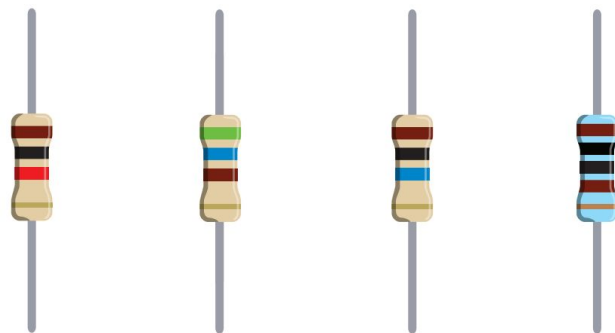
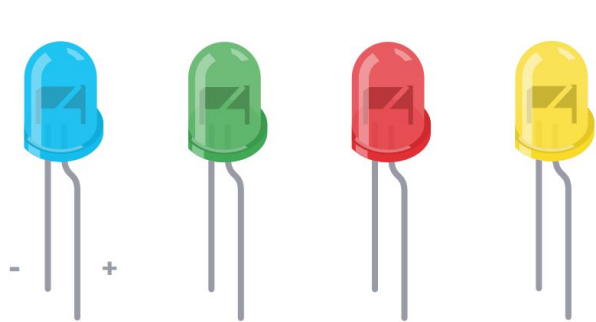
- Arduino Uno pločica



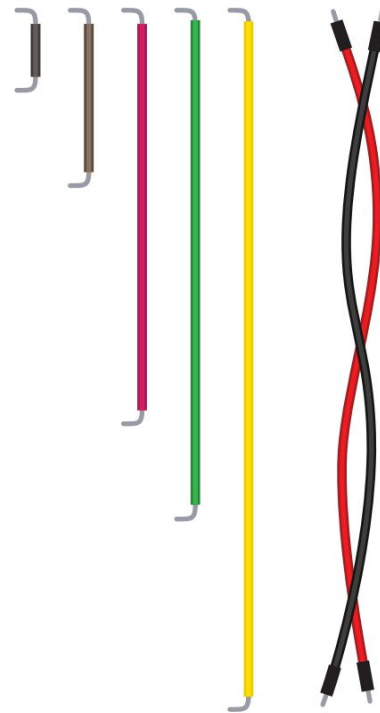
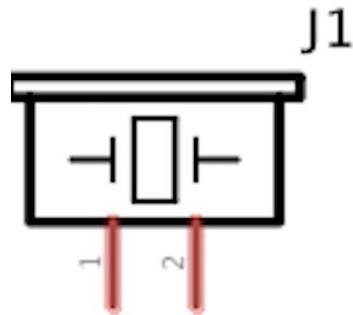
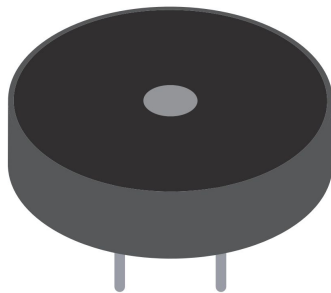
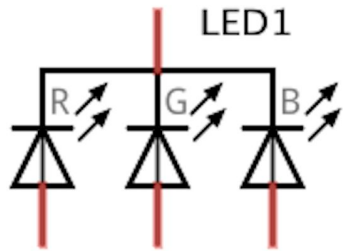
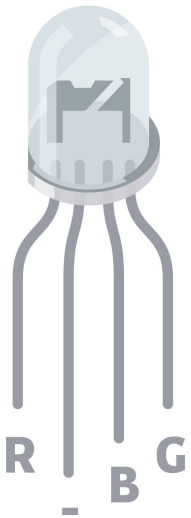
- Eksperimentalna pločica (engl. breadboard)



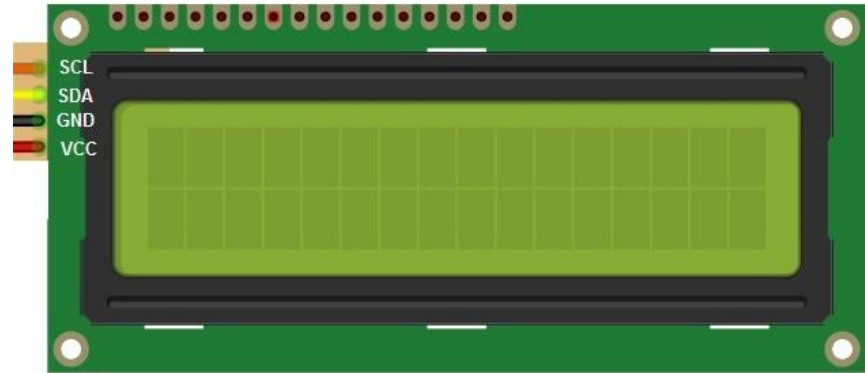
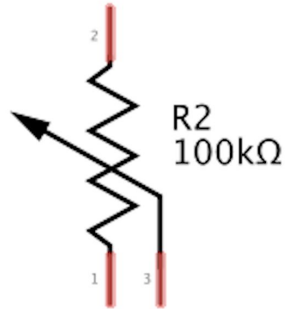
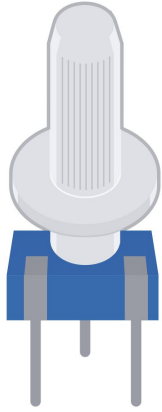
- Elektroničke komponente



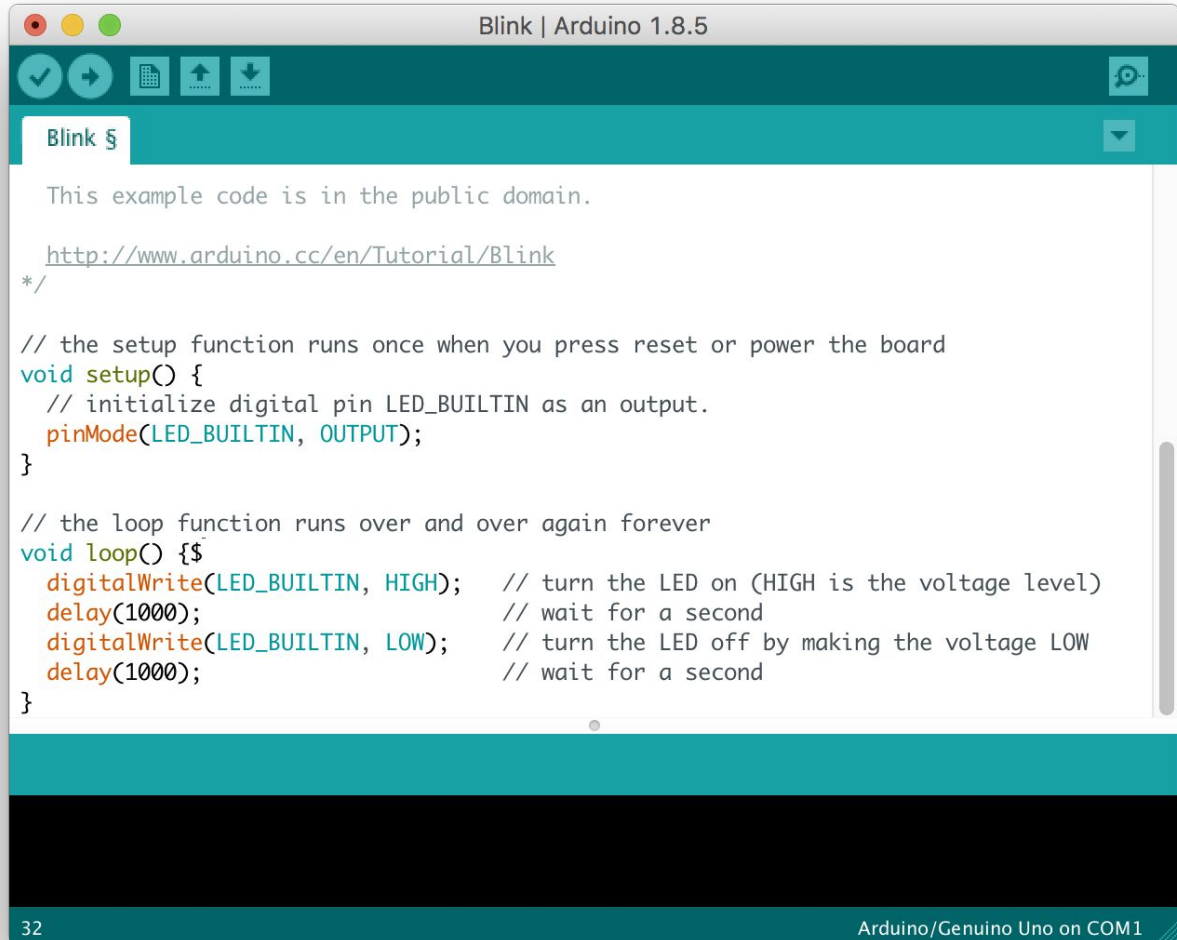
- Elektroničke komponente



- Elektroničke komponente



Arduino IDE



The screenshot shows the Arduino IDE window titled "Blink | Arduino 1.8.5". The interface includes a toolbar with icons for check, run, upload, and download, and a search icon. The main editor area displays the following code:

```
Blink §  
  
This example code is in the public domain.  
  
http://www.arduino.cc/en/Tutorial/Blink  
*/  
  
// the setup function runs once when you press reset or power the board  
void setup() {  
  // initialize digital pin LED_BUILTIN as an output.  
  pinMode(LED_BUILTIN, OUTPUT);  
}  
  
// the loop function runs over and over again forever  
void loop() {  
  digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the voltage level)  
  delay(1000); // wait for a second  
  digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the voltage LOW  
  delay(1000); // wait for a second  
}
```

At the bottom of the IDE, the status bar shows "32" on the left and "Arduino/Genuino Uno on COM1" on the right.

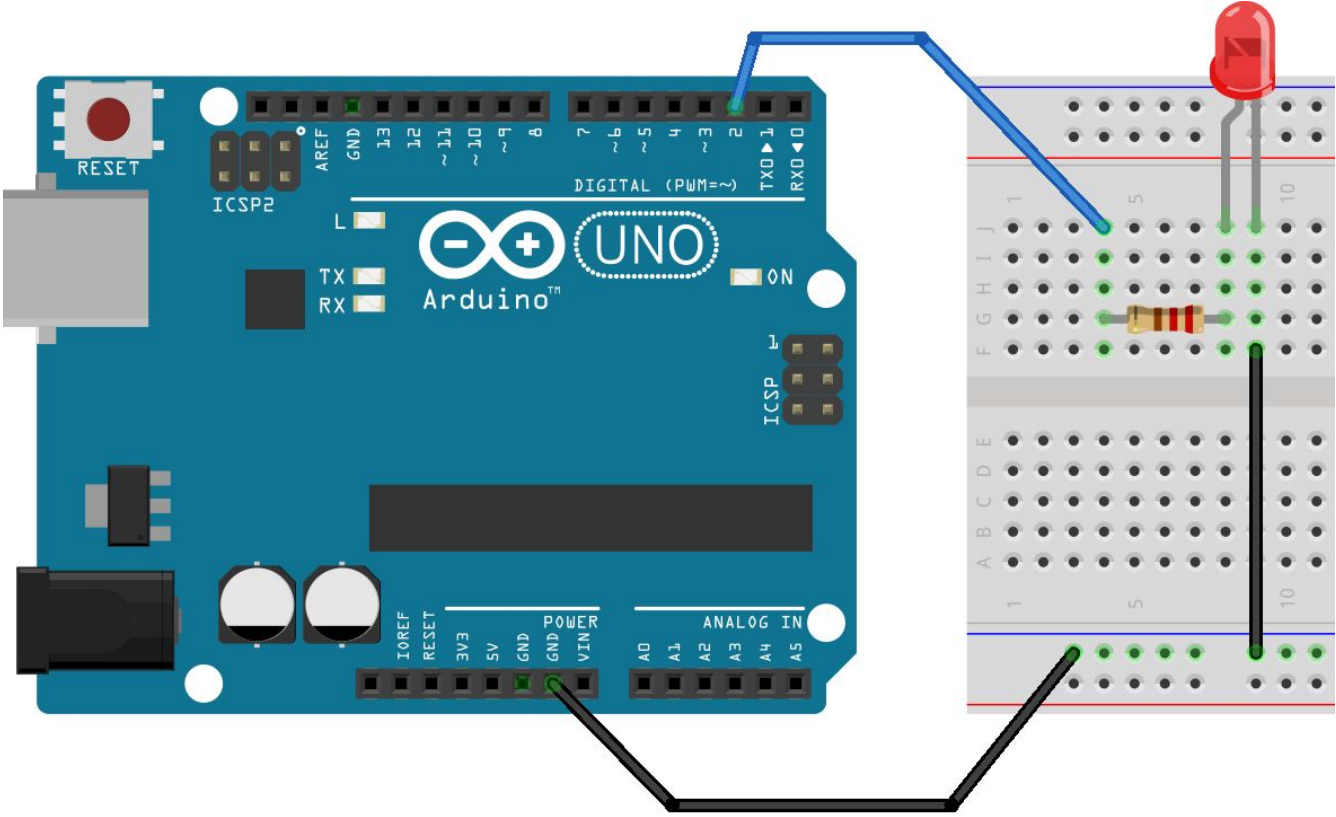
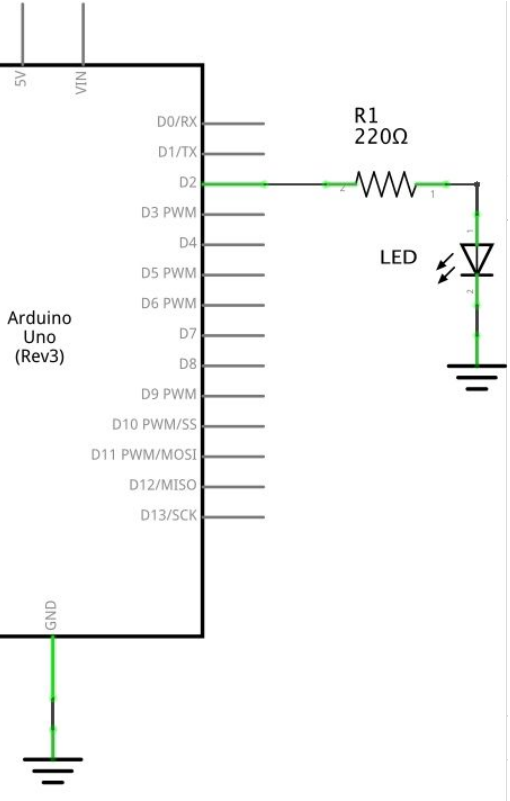
Minimalan osnovni kod

```
1 void setup() {  
2     // kod koji se treba izvršiti jednom  
3 }  
4  
5 void loop() {  
6     // kod koji se izvršava ponavljajući  
7 }
```

Primjer: Blink

```
1 void setup() {
2     // postavlja pin 13 kao izlaz
3     pinMode(LED_BUILTIN, OUTPUT);
4 }
5
6 void loop() {
7     digitalWrite(LED_BUILTIN, HIGH); // upali LED
8     delay(1000);                      // čekaj 1000ms
9     digitalWrite(LED_BUILTIN, LOW);  // ugasi LED
10    delay(1000);                       // čekaj 1000ms
11 }
```

Zadatak: Blink s vanjskom LED-icom na pinu 2



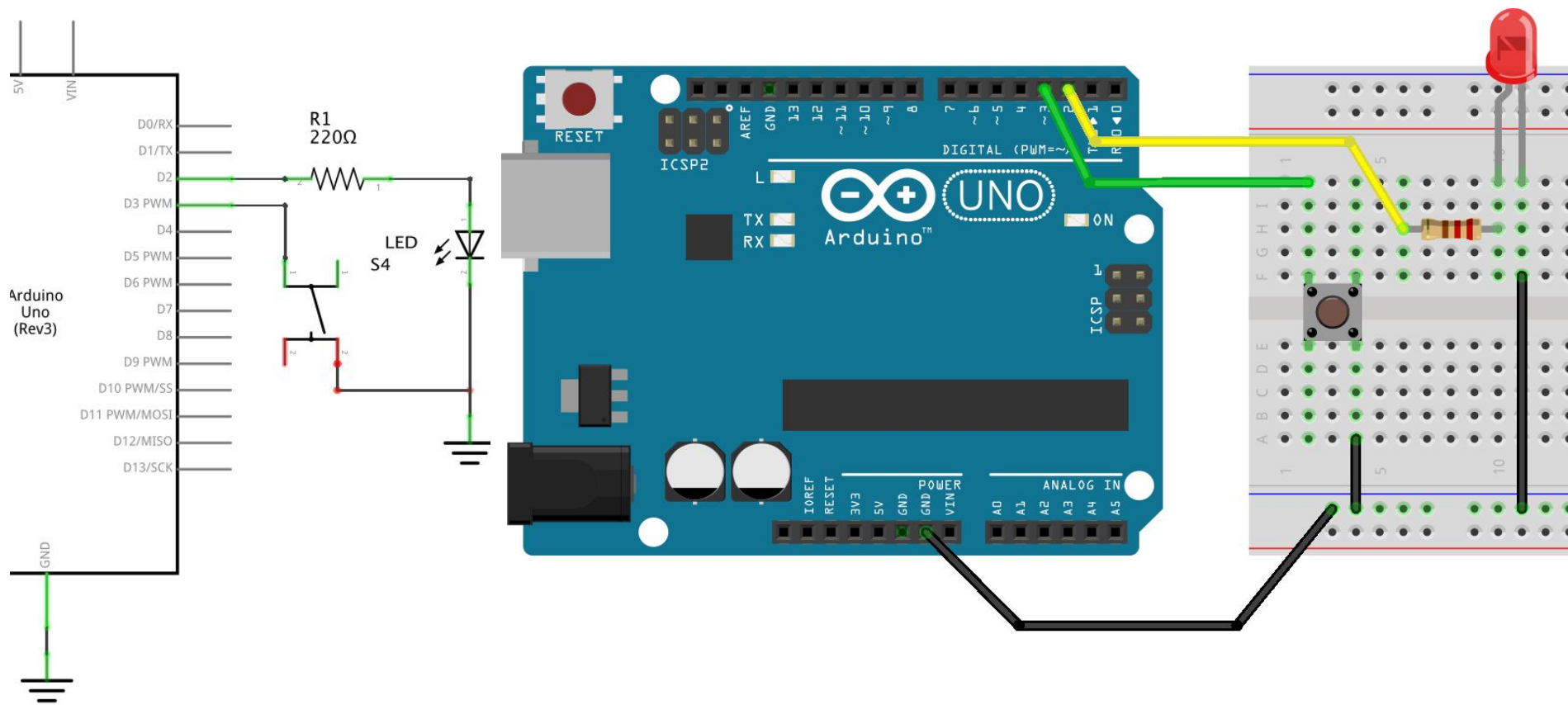
Zadatak: Blink s vanjskom LED-icom na pinu 2

```
1 int led = 2;
2
3 void setup() {
4     pinMode(led, OUTPUT);
5 }
6
7 void loop() {
8     digitalWrite(led, HIGH);
9     delay(1000);
10    digitalWrite(led, LOW);
11    delay(1000);
```

Zadatak 1: Trčeće svjetlo

Sad kada znamo spojiti i upravljati jednom LED-icom, možemo ostvariti i nešto kompleksniji sklop. Potrebno je spojiti tri LED-ice na tri zasebna pina te ih upogoniti u takozvano "trčeće svjetlo".

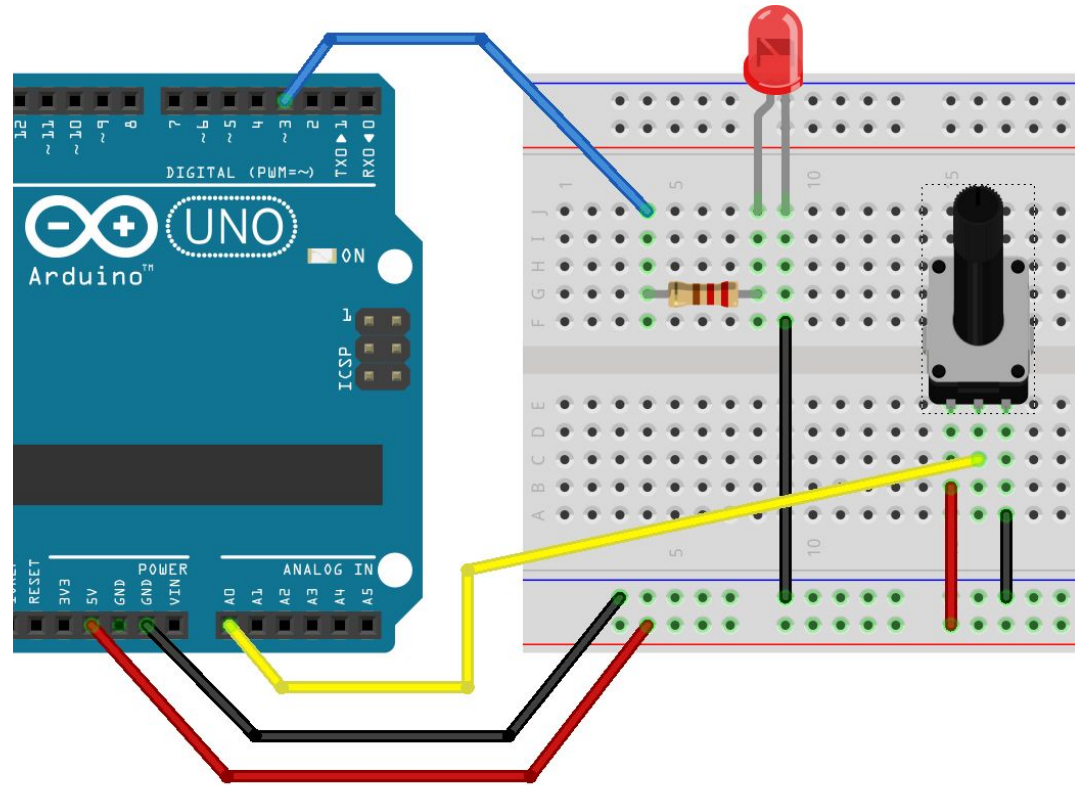
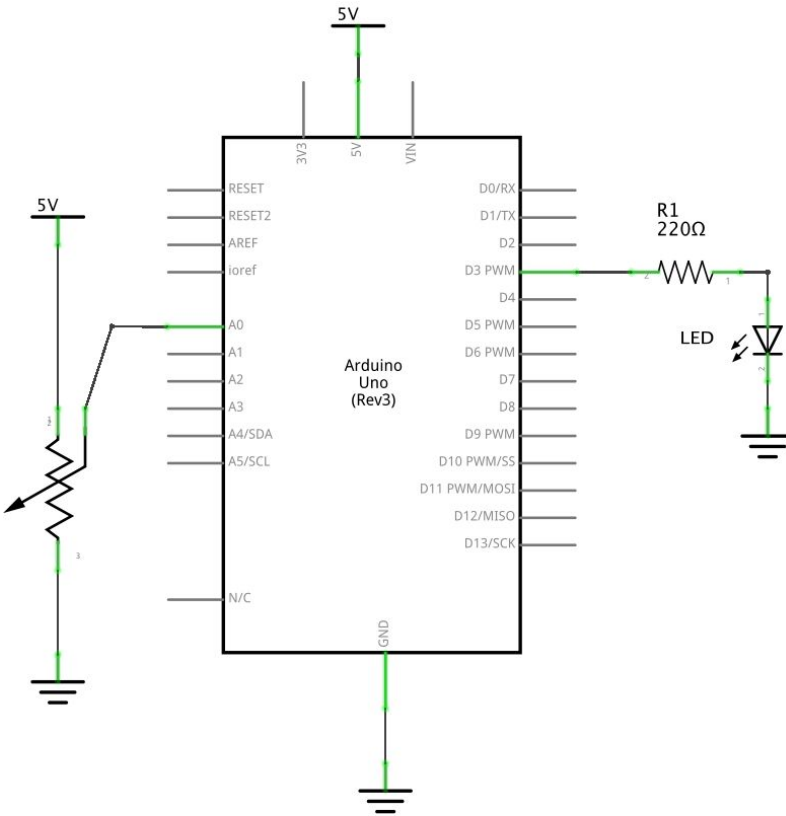
Primjer: Tipkalo i digitalni ulaz



Primjer: Tipkalo i digitalni ulaz

```
1 const int button = 3;
2 const int led = 2;
3 int button_state = 0;
4
5 void setup() {
6     pinMode(led, OUTPUT);
7     pinMode(button, INPUT_PULLUP);
8 }
9
10 void loop(){
11     button_state = digitalRead(button);
12     digitalWrite(led, button_state);
13 }
```

Primjer: Čitanje s analognog ulaza



Čitanje s analognog ulaza i funkcija map

```
// vraća vrijednost iz raspona od 0 do 1023
int analog = analogRead(A0);

// skalira vrijednost 'analog' iz raspona
// od 0 do 1023 u raspon od 0 do 100
int percentage = map(analog, 0, 1023, 0, 100);
```

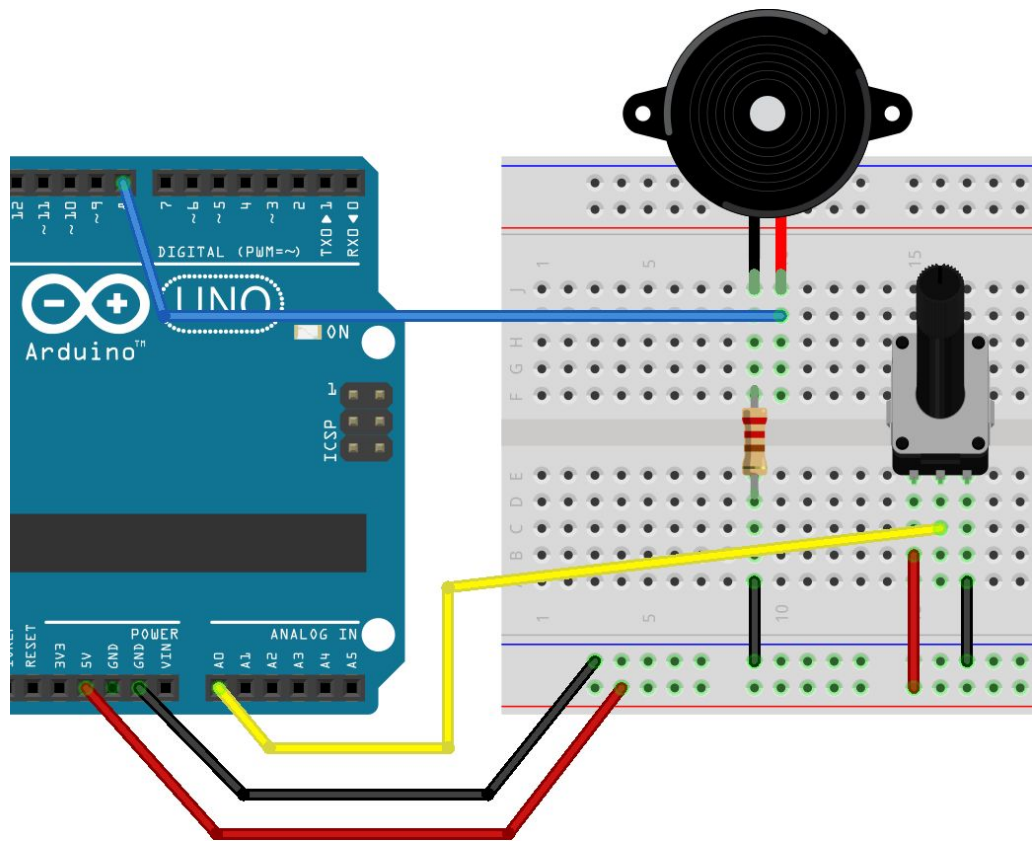
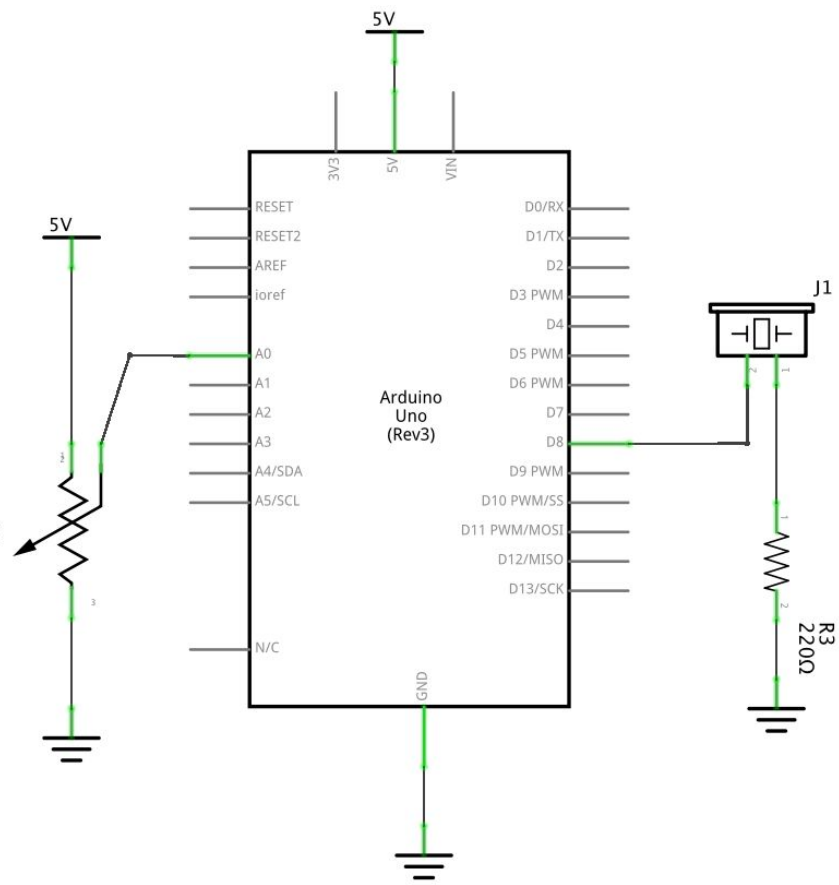
Primjer: Čitanje s analognog ulaza

```
1 const int led = 3;
2 const int pot_pin = A0;
3
4 void setup() {
5     pinMode(led, OUTPUT);
6 }
7
8 void loop(){
9     int value = analogRead(pot_pin);
10    digitalWrite(led, LOW);
11    delay(value);
12    digitalWrite(led, HIGH);
13    delay(value);
14 }
```

Zadatak 2: Funkcija map

Izmijenite prethodno napisani kod uz pomoć funkcije **map()** kako bi povećali frekvencijski raspon treperenja LED-ice.

Primjer: Analogni izlaz i zvučnik



Primjer: Analogni izlaz i zvučnik

```
1 const int speaker_pin = 8;
2 const int pot_pin = A0;
3
4 void setup() {
5     pinMode(speaker_pin, OUTPUT);
6 }
7
8 void loop() {
9     int pot_value = analogRead(A0);
10
11     int pitch = map(pot_value, 0, 1023, 200, 5000);
12     tone(speaker_pin, pitch, 20);
13     delay(10);
14 }
```

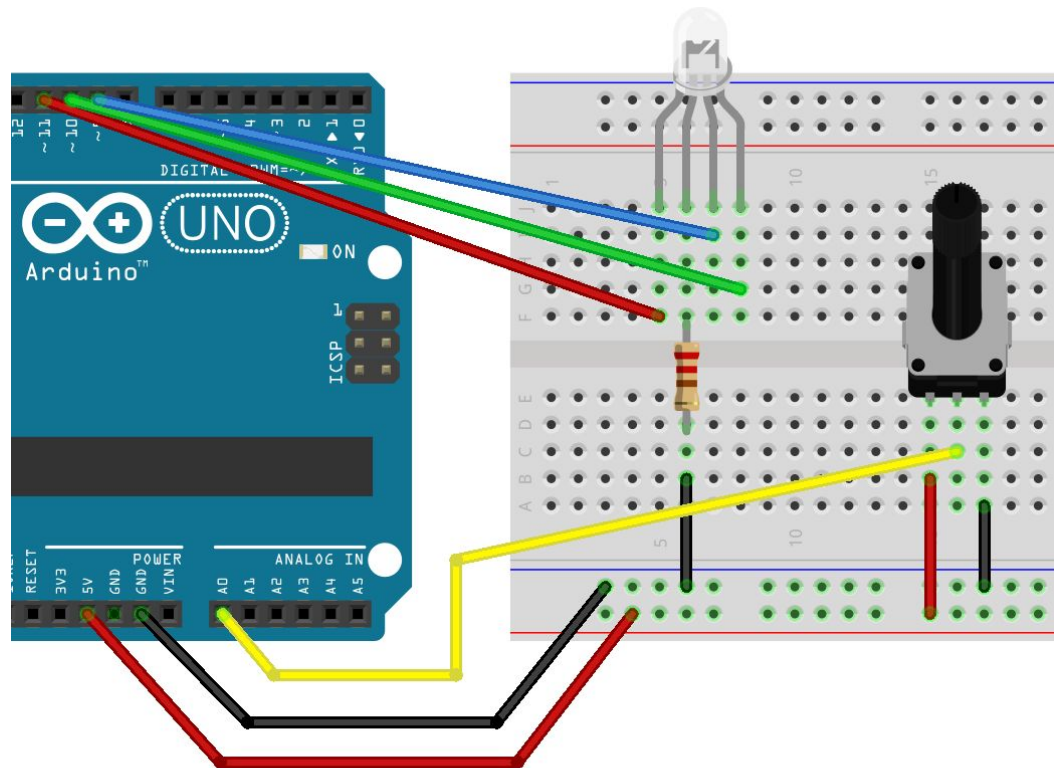
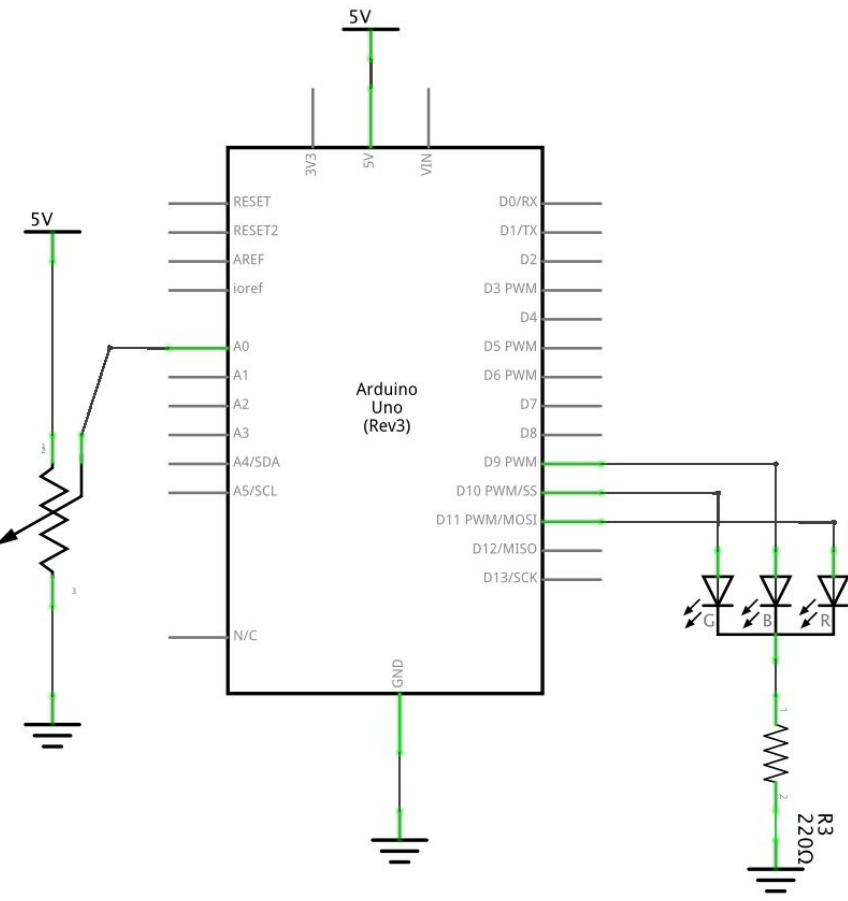
Zadatak 3: Zvučnik i tipkalo

Na navedenu shemu dodajte tipkalo spojeno na zaseban digitalni ulaz te izmijenite kod tako da se na pritisak tipkala mijenja frekvencija tona.

Kontrola toka programa

```
if(state < 8) {  
  // ukoliko je uvjet u zagradama istinit  
  // izvrsava se kod u ovom bloku  
} else if (state < 3) {  
  // ukoliko prvi uvjet nije bio istinit, a ovaj jest  
  // izvrsava se kod u ovom bloku  
} else {  
  // ukoliko nijedan od uvjeta do sad nije bio istinit  
  // izvrsava se kod u ovom bloku  
}
```

Primjer: Analogni izlaz i RGB dioda



Primjer: Analogni izlaz i RGB dioda

```
1 const int rgb_red_pin = 11;
2 const int rgb_green_pin = 10;
3 const int rgb_blue_pin = 9;
4 const int pot_pin = A0;
5
6 void setup()
7 {
8     pinMode(rgb_red_pin, OUTPUT);
9     pinMode(rgb_green_pin, OUTPUT);
10    pinMode(rgb_blue_pin, OUTPUT);
11 }
```

Primjer: Analogni izlaz i RGB dioda

```
12 void loop()  
13 {  
14     int pot_value = analogRead(pot_pin);  
15     int rgb_value = map(pot_value, 0, 1023, 0, 1535);  
16  
17     int red;  
18     int blue;  
19     int green;  
20  
21     if (rgbValue < 256) {  
22         red = 255;  
23         blue = rgbValue;  
24         green = 0;
```

Primjer: Analogni izlaz i RGB dioda

```
46     else {
47         red = 255;
48         blue = 0;
49         green = 1535 - rgbValue;
50     }
51
52     analogWrite(rgb_red_pin, red);
53     analogWrite(rgb_blue_pin, blue);
54     analogWrite(rgb_green_pin, green);
55 }
```



Drugi dio

Serijska komunikacija

- UART protokol
- Dijagnostika i otkrivanje pogrešaka
- Arduino IDE: Tools → Serial Monitor

```
1 void setup() {  
2     Serial.begin(9600);  
3 }  
4  
5 void loop(){  
6     Serial.println("Pozdrav");  
7 }
```

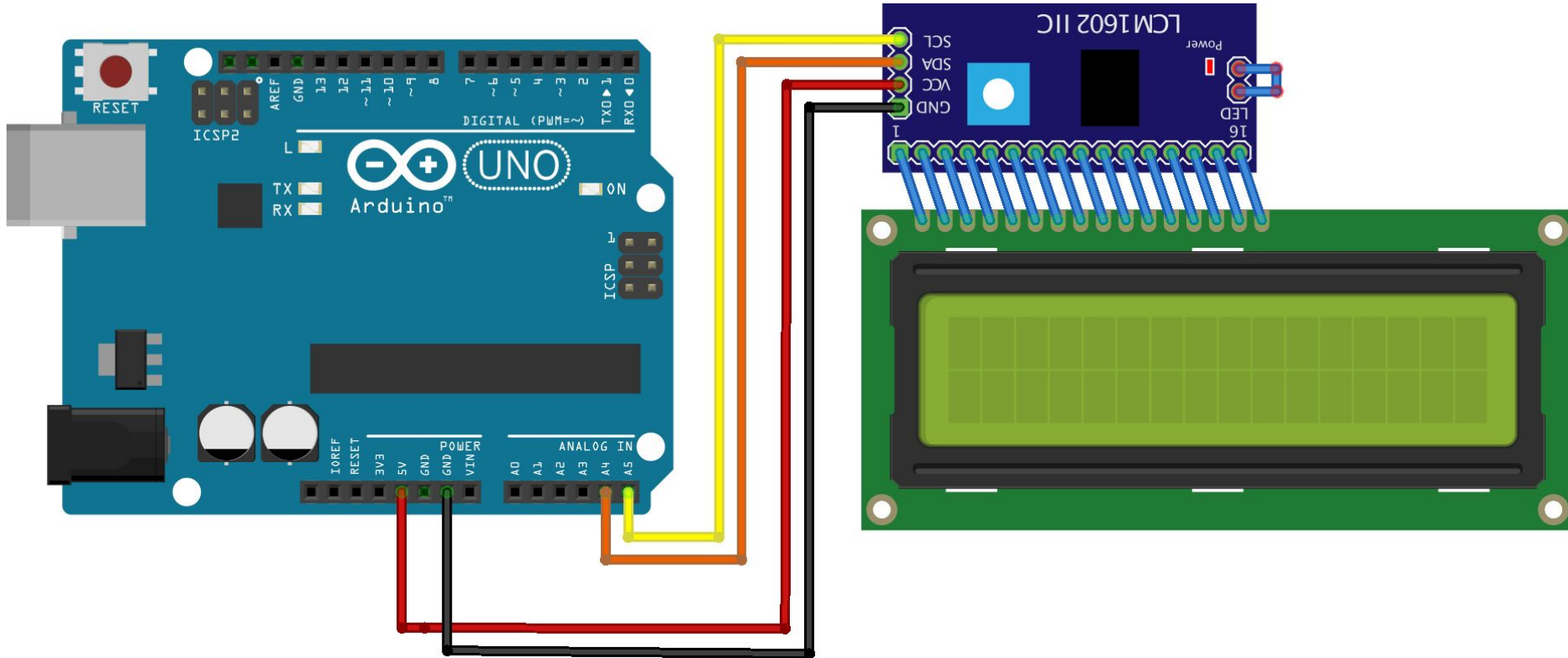
Primjer: Serijska komunikacija

Na postojeći primjer **Analogni izlaz i zvučnik** dodati ćemo serijsku komunikaciju

```
4 void setup() {
5     pinMode(led, OUTPUT);
6     //inicijalizira komunikaciju na 9600 baud
7     Serial.begin(9600);
8 }
9
10 void loop(){
11     int value = analogRead(pot_pin);
12     //salje vrijednost preko serijskog porta na konzolu
13     Serial.println(value);
14     digitalWrite(led, LOW);
15     delay(value);
16     digitalWrite(led, HIGH);
17     delay(value);
```

Primjer: LCD ispis

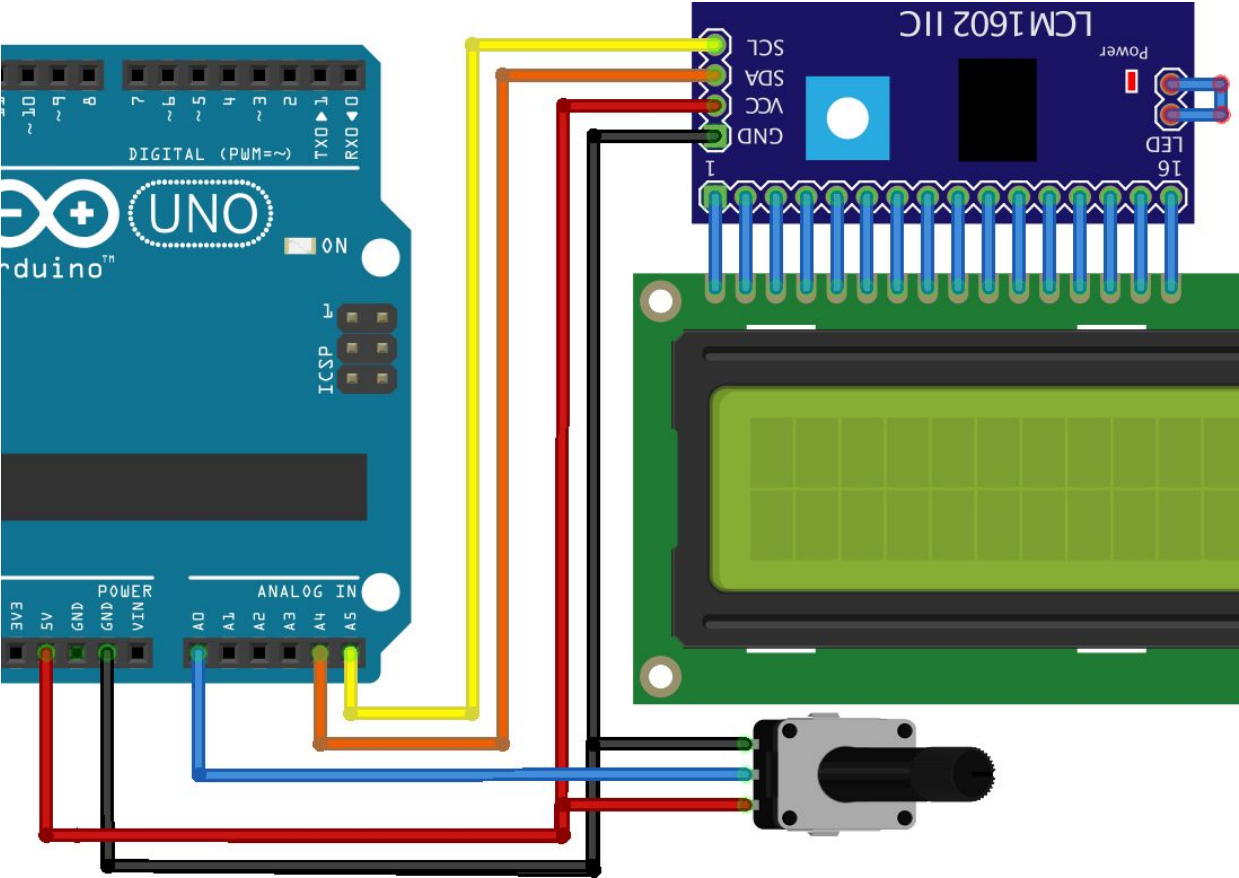
- I2C komunikacijski protokol



Primjer: LCD ispis

```
1 #include <LiquidCrystal_I2C.h>
2
3 LiquidCrystal_I2C lcd(0x3F,16,2);
4
5 void setup()
6 {
7     lcd.init();
8     lcd.backlight();
9     lcd.setCursor(1,0);
10    lcd.print("Pozdrav Jura");
11 }
12 void loop()
13 {
14 }
```


Zadatak 4: Analogni ulaz i kontinuirani ispis na LCD



Primjer: Novi znakovi (custom znakovi)

```
1 #include <LiquidCrystal_I2C.h>
2
3 LiquidCrystal_I2C lcd(0x27, 16, 2);
4
5 byte customChar[8] = {
6     0b00000,
7     0b01010,
8     0b11111,
9     0b11111,
10    0b01110,
11    0b00100,
12    0b00000,
13    0b00000
14 };
15
16 void setup()
17 {
18     lcd.init();
19     lcd.backlight();
20
21     lcd.createChar(0, customChar);
22     lcd.setCursor(2, 0);
23     lcd.write((byte)0);
24 }
25
26 void loop()
27 {
28 }
```

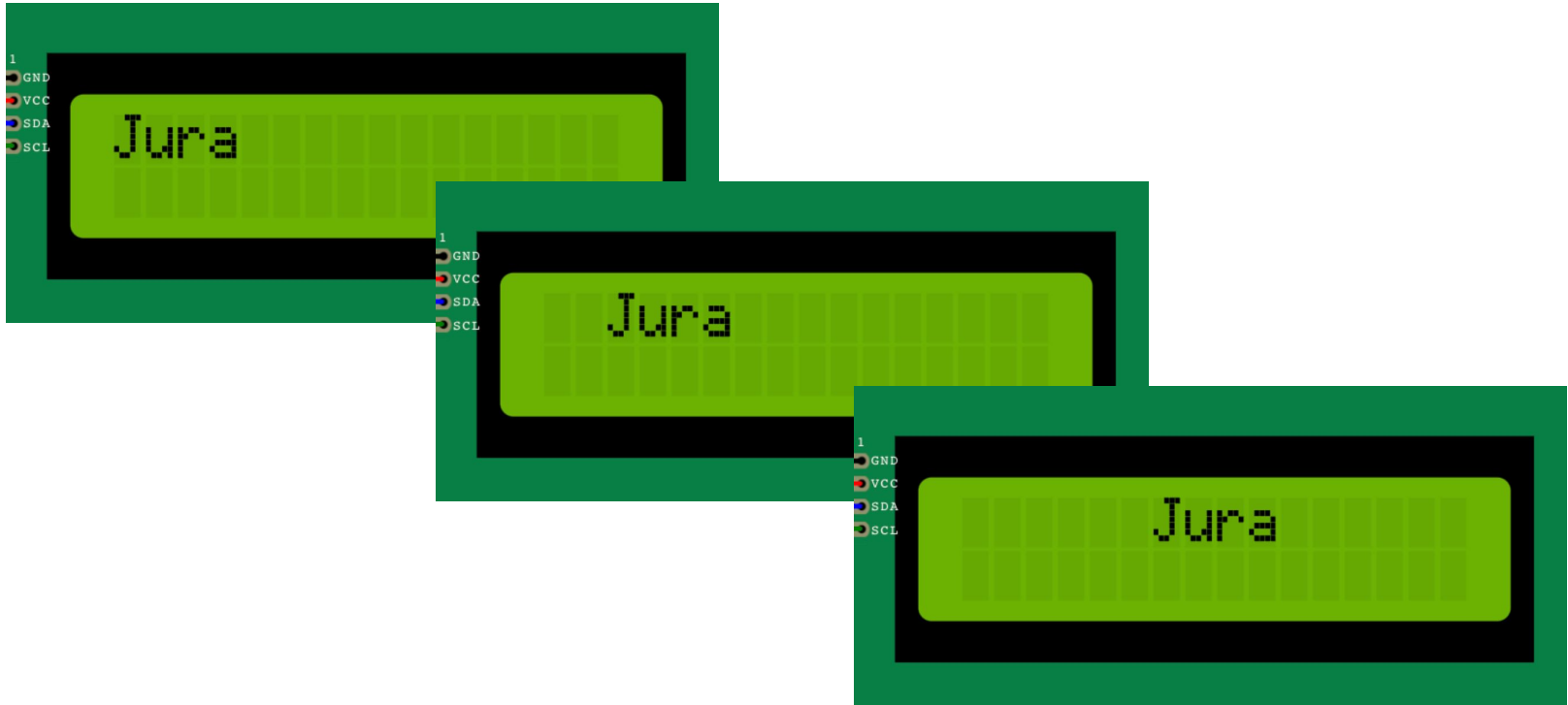
Zadatak 5

Ukoliko Vaše ime i prezime sadrži dijakritičke znakove, dodajte znakove i ispišite Vaše puno ime i prezime. Ukoliko to nije slučaj, dodajte potrebne znakove i ispišite ime Šimširpašić Sabahudin

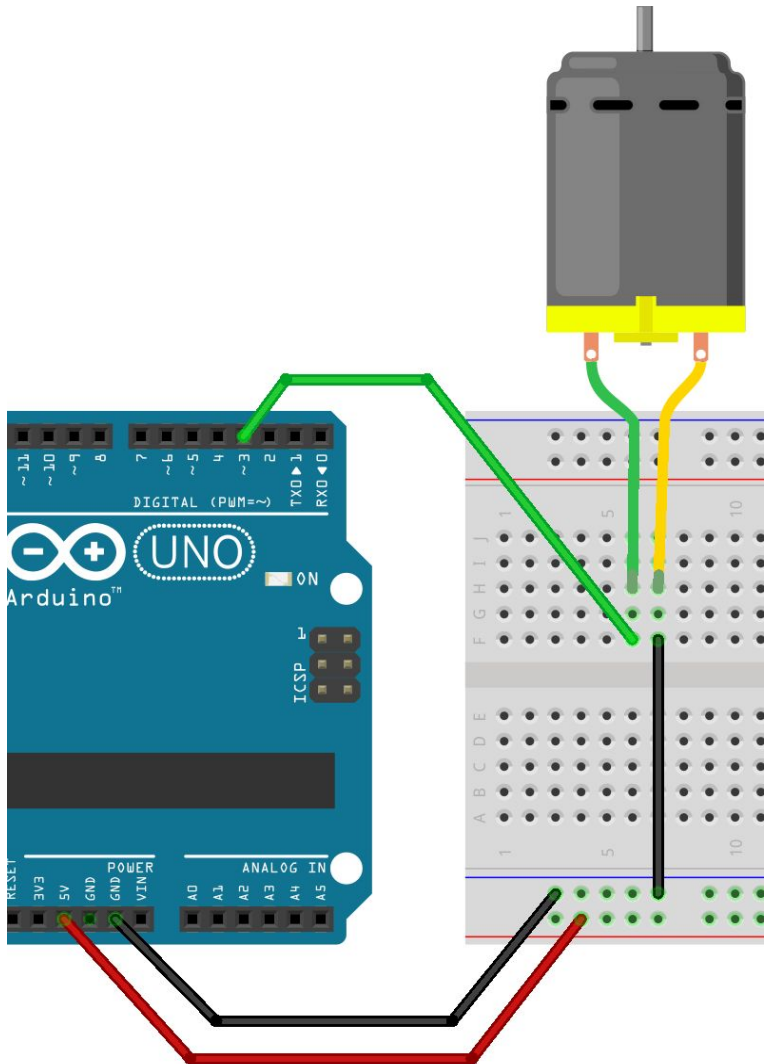


Bonus zadatak 6: Trčeći znakovi

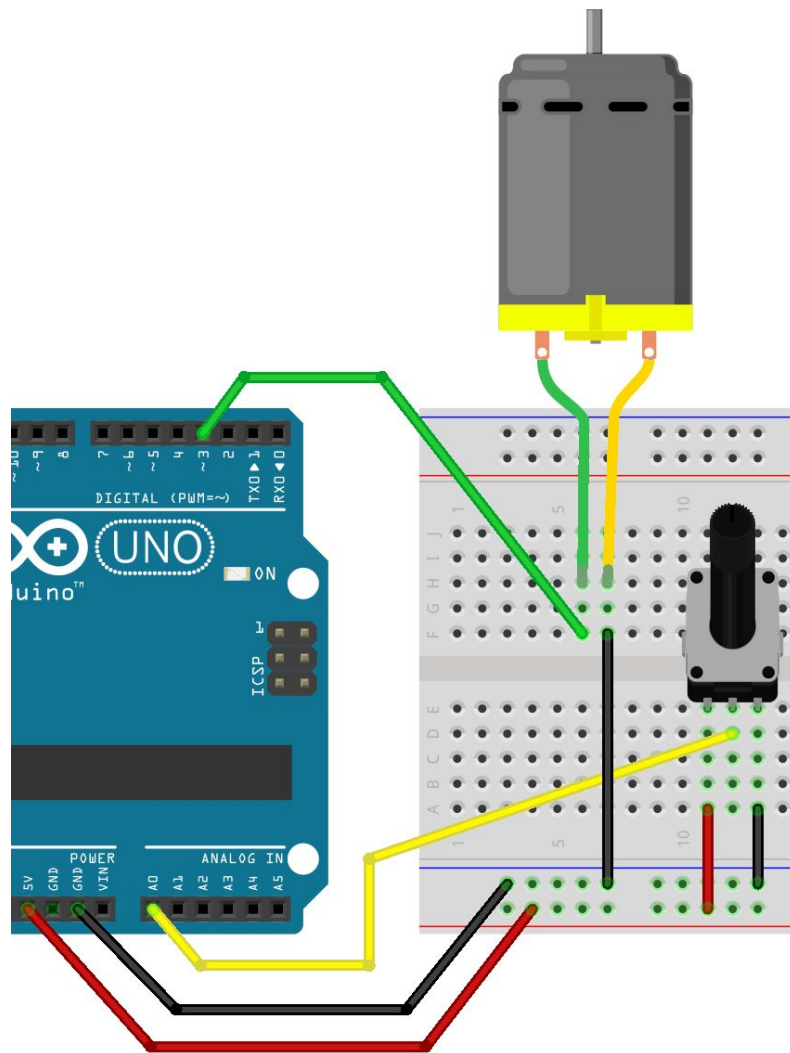
Koristeći znanje iz prethodnih zadataka, učinite da se ispisano ime svake dvije sekunde pomakne za jedno mjesto u desno.



Upravljanje DC motorom



Zadatak 7: Upravljanje DC motorom uz analogni ulaz





Hvala 😊

Krešimir Topolovec & Nikola Jerković