
ELEKTRONIKA - zbirka vaj

Ta zbirka vaj je namenjena študentom Pedagoške fakultete,
Univerze v Ljubljani.

dr. David Rihtaršič

2020-02-18 13:11

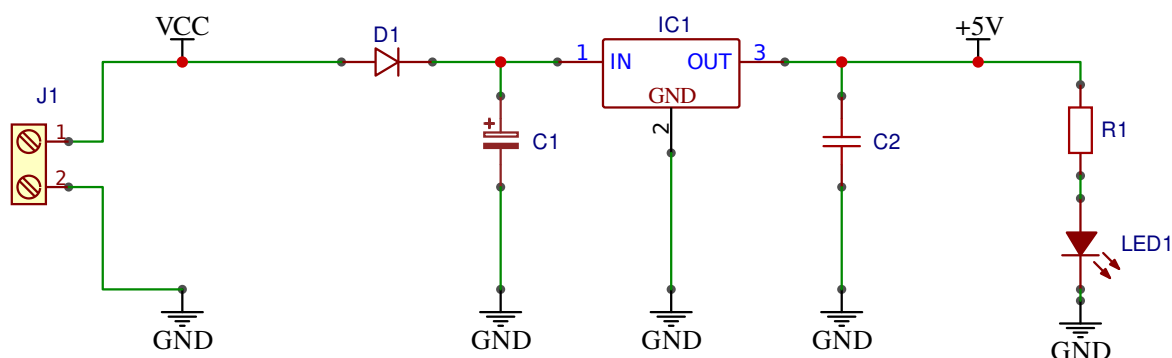
Kazalo

1	Napajanje	3
1.1	Napetostni potencial	4
1.2	Napetost	5
2	Krmilnik Arduino Nano	7
2.1	Testni program "BLINK.ino"	7
2.2	Napajanje krmilnika Arduino Nano	8
2.3	Model semaforja	10
2.4	Analiza vezja	11
3	Dodatki in pogosta vprašanja	13
3.1	Barvno kodiranje upornosti	13

1 Napajanje

Vsako električno vezje potrebuje napajanje. Pogosto vezja napajamo z baterijami (prenosne električne naprave) ali pa z uporabo omrežnega napajanja. Za današnje vezje bomo uporabili omrežno napajanje. S pomočjo 9 V napajalnika in 5 mm napajalnega priključka (J1) dobimo napajalno napetost 9 V. Oba napetostna potenciala (+9 V in 0 V) napajalne napetosti priključimo na testno ploščico in od tam zgradimo vezje, ki bo zagotavljalo 5 V napajanje za naše nadaljnja vezja.

1 VAJA: Izgradite vezje po shemi.



Slika 1.1: Shema napajalnega vezja.

V pomoč bomo uporabili program **Fritzing**, s katerim lahko učitelj zelo nazorno pokaže kako moramo povezati elemente med seboj. S pomočjo Fritzing programa dobimo naslednjo skico *realnega* vezja.

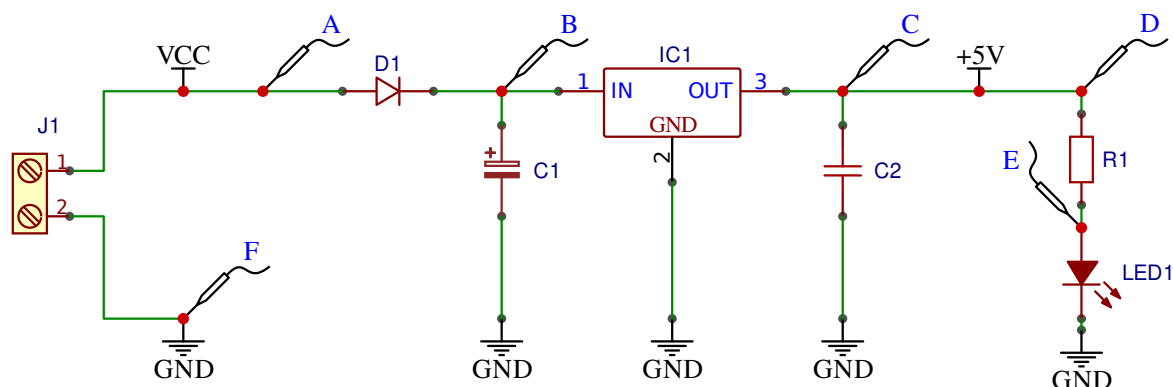


Slika 1.2: Skica realnega vezja.

1.1 Napetostni potencial

1 VAJA: Izmeri napetostne potenciale, ki so vrisani v naslednji shemi.

Točka v vezju	Nap. potencial [V]
A	
B	
C	
D	
E	
F	



Slika 1.3: Shema električnega vezja 5 V napaljalne napetosti.

1.2 Napetost

V vezju imamo imamo kar nekaj elektronskih elementov. Na shemi 1.1 so različno označeni, npr.:

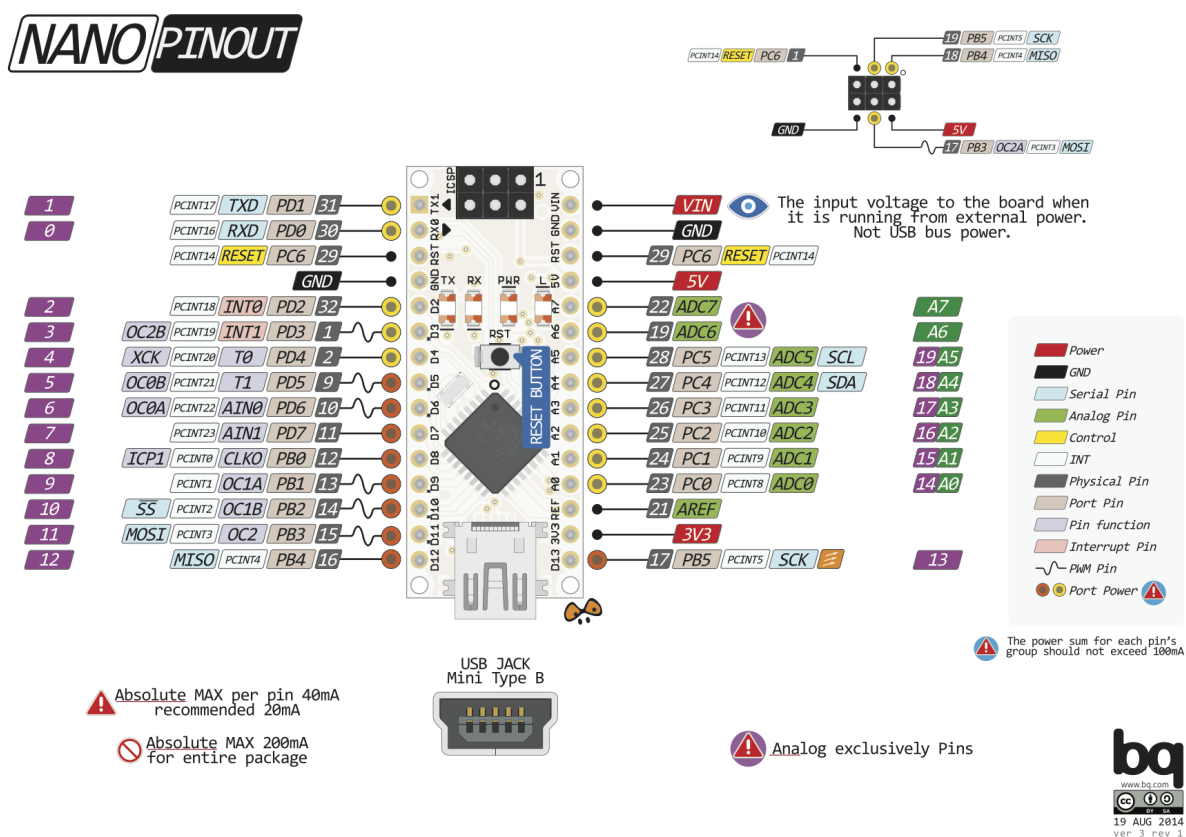
- polprevodniška dioda - D1,
- elektrolitski kondenzator - C1,
- keramični kondenzator - C2,
- upor - R1,
- svetleča dioda - LED1

1 **VAJA:** Za vse naštete elemente najprej izračunajte kolikšna napetost je na njih, nato pa izračun preverite z inštrumentom.

Element	$U_{izr}[V]$	$U_{izm}[V]$
D1		
C1		
C2		
R1		
LED1		

2 Krmilnik Arduino Nano

Krmilnik Arduino Nano je relativno cenovno ugoden (cca. 3-5€) in ker je programirljiv, ga lahko uporabimo v najrazličnejših aplikacijah. Razporeditev njegovih priključkov pa pa lahko vidimo na naslednji sliki 2.1.



Slika 2.1: Razporeditev priključkov na krmilniku Arduino Nano.

2.1 Testni program “BLINK.ino”

Preden bomo krmilnik uporabili v našem vezju, ga bomo preizkusili. S programskim orodjem **Arduino IDE** bomo na krmilnik naložili program `blink.ino` in s tem preverili, da vse komponente na krmilniku

delujejo pravilno. To je priporočljivo narediti pred vsakim projektom.

- ```
1 VAJA: Preizkus delovanja krmilnika Arduino Nano.
2 1. Krmilnik Arduino Nano povežite z računalnikom preko USB povezave,
3 2. zaženite program Arduino IDE in ga pravilno nastavite:
4 - Tools -> Processor : Arduino Nano,
5 - Tools -> Port : USB2
6 3. Odprite primer 01-BLINK.ino in
7 4. prenesite program na krmilnik.
```

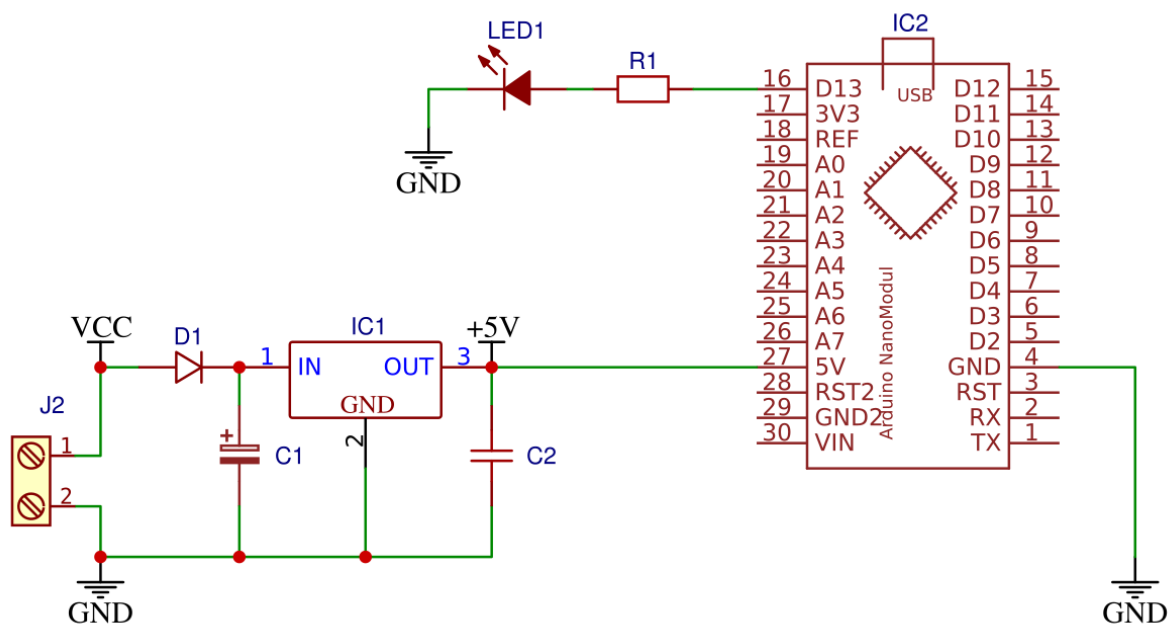
Program je napisan v programskem jeziku C++, ki uporablja nekaj funkcij za lažje rokovanje s krmilnikom.

```
1 void setup() {
2 // initialize digital pin LED_BUILTIN as an output.
3 pinMode(LED_BUILTIN, OUTPUT);
4 }
5
6 // the loop function runs over and over again forever
7 void loop() {
8 digitalWrite(LED_BUILTIN, HIGH); // turn the LED on (HIGH is the
9 voltage level)
10 delay(1000); // wait for a second
11 digitalWrite(LED_BUILTIN, LOW); // turn the LED off by making the
12 voltage LOW
13 delay(1000); // wait for a second
14 }
```

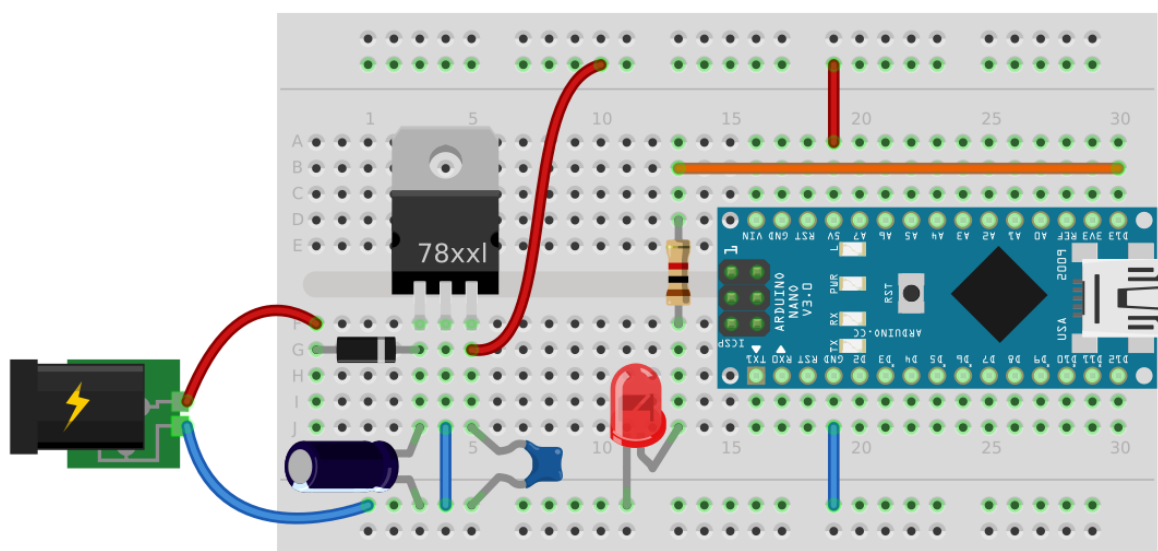
## 2.2 Napajanje krmilnika Arduino Nano

Krmilnik Arduino Nano lahko vstavimo tudi v prototipno ploščico in ga napajamo z zunanjim napajanjem.

- ```
1  VAJA: Uporaba krmilnika Arduino Nano na prototipni ploščici.  
2  Vstavite krmilnik Arduino Nano v prototipno ploščico in ga povežite kot  
3  prikazuje naslednja shema.  
4  Priključite tudi upor in LED na priključek 13 .
```



Slika 2.2: Priključitev napajanja in dodatne LED na izhodni priključek.

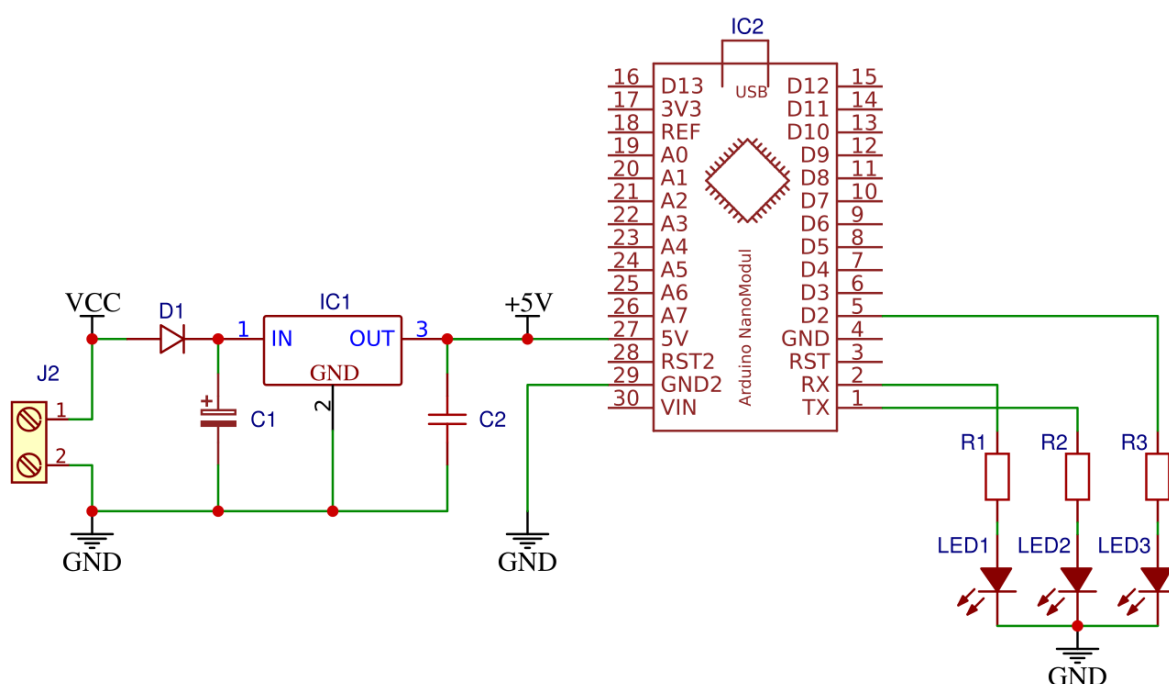


Slika 2.3: Shema vezave krmilnika Arduino Nano na prototipni ploščici.

2.3 Model semaforja

Vezje bomo preoblikovali tako, da bo delovalo kot semafor na cestnem križišču. Uporabili bomo tri LED svetila različnih barv in preoblikovali program.

- 1 VAJA: Model semaforja.
- 2 Preoblikujte vezje po shemi, ki je prikazana na naslednji sliki. In uporabite naslednji program ter ga ustrezno preoblikujte. Program, ki zagotavlja podobno delovanje kot pri šestnem semaforju dokumentirajte in komentirajte uporabljenej programske stavke (t.j. programske ukaze).



Slika 2.4: Shema vezave treh LED na krmilnik Arduino Nano.

Preskustite naslednji program in ga ustrezno preoblikujte.

```

1 void setup() {
2   pinMode(0, OUTPUT);
3   pinMode(1, OUTPUT);
4   pinMode(2, OUTPUT);
5 }
6
7 void loop() {
8   digitalWrite(0, HIGH);
9   digitalWrite(1, HIGH);

```

```
10     digitalWrite(2, HIGH);  
11     delay(1000); digitalWrite(0, LOW);  
12     digitalWrite(1, LOW);  
13     digitalWrite(2, LOW);  
14     delay(1000);  
15 }
```

2.4 Analiza vezja

Elektronski elementi so omejeni z njihovo največjo dopustno električno moč. Če to električno moč prekoračimo, jih bomo najverjetneje uničili.

Naprimera: Največja dopustna moč, ki se še lahko troši na uporih, ki jih uporabljate (premer upora = 2.4 mm) je 0,25 W.

Električno moč lahko izračunamo po enačbi:

$$P = UI$$

Pri nekaterih drugih elementih (kot naprimer pri LED) pa so omejitveni pogoji postavljeni če s samim tokom.

Naprimera za običajne 5mm LED je najpogosteje največji tok, ki lahko teče skozi 20 mA.

Tok skozi element lahko izračunamo po Ohmovem zakonu:

$$I_R = \frac{U_R}{R}$$

Če ne vemo kolikšno upornost ima element (tako kot je to v primeru LED), si največkrat pomagamo z izračunom toka skozi drug zaporedno vezan element. Kajti v tem primeru je tok isti.

- 1 VAJA: Električni tok skozi elemente.
- 2 Izračunajte kolikšen električni tok teče skozi elemente R1, R2, R3, LED1, LED2 in LED3 ter preverite kakšne so električne omejitve tega elementa.
- 3 Izračunajte tudi električno moč, ki se troši na tem elementu.

Element	U [V]	I [V]	P[W]
R1			
R2			
R3			
LED1			
LED2			
LED3			

3 Dodatki in pogosta vprašanja

3.1 Barvno kodiranje upornosti

Upornosti na uporih so bravno kodirane in sicer:

- 0. • črna,
- 1. • rjava,
- 2. • rdeča,
- 3. • oranžna,
- 4. • rumena,

