



ARDUINO KROZ JEDNOSTAVNE PRIMJERE

- pripreme za natjecanja -

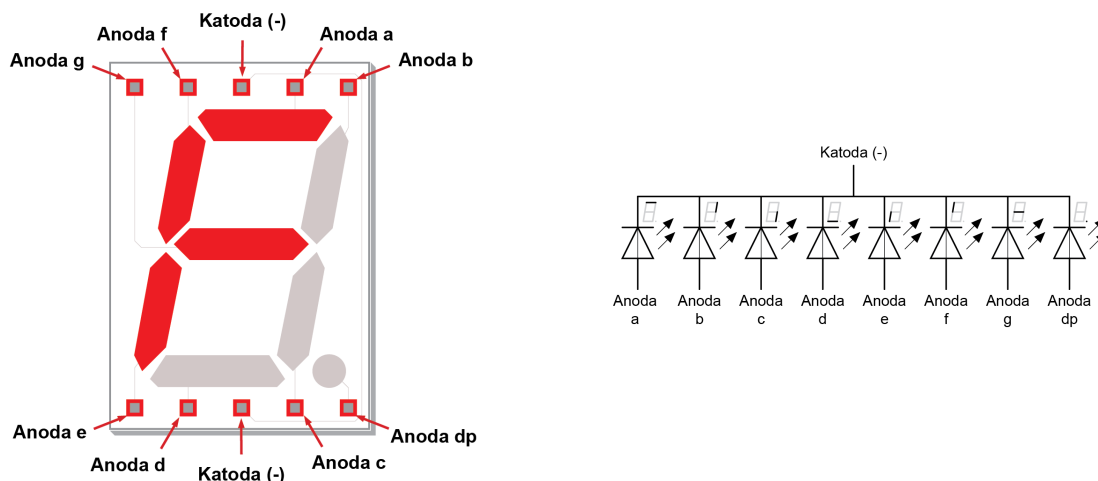
**PRIPREMA 3 - 2015
SEDAM SEGMENTNI EKRANI**

Paolo Zenzerović, mag. ing. el.

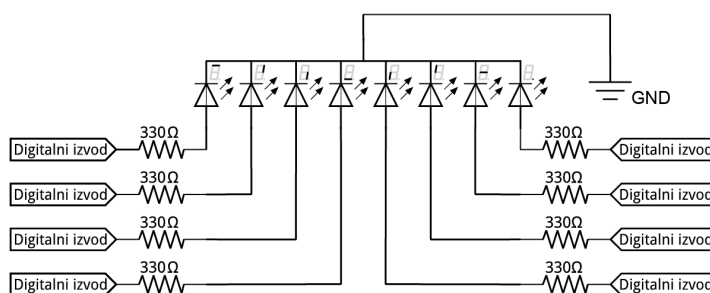
Zagreb, 2015.

SEDAM SEGMENTNI EKRANI

Sigurno ste već na puno mjesta vidjeli ovakve ekrane – u satovima, vagama, brojačima i slično. Ovi su ekrani zapravo skupina svjetlećih dioda smještenih u jedno kućište na takav način da bi se njihovom određenom kombinacijom mogli prikazivati neki znakovi. Na sljedećoj slici prikazan je sedam segmentni ekran i njegov simbol u Fritzing programu.

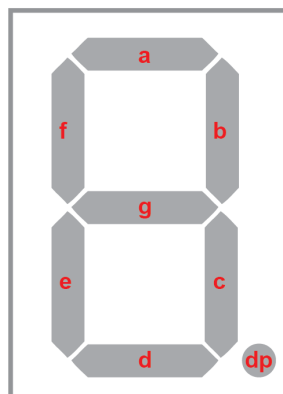


Postoje dvije vrste ekrana, oni sa zajedničkom katodom i oni sa zajedničkom anodom. U ovoj ćemo pripremi raditi sa sedam segmentnim ekranima sa zajedničkom katodom. Na sljedećoj slici pogledajte način spajanja ekrana sa zajedničkom katodom na mikrokontroler.



Svaki od segmenata ima svoj naziv. Segmenti se nazivaju slovima engleske abecede od a do g. Neki sedam segmentni ekrani dodatno imaju i jednu točkastu svjetleću diodu za prikaz decimalne točke. Taj se izvod naziva dp što dolazi od engleskog naziva za decimalnu točku – *dot point*.

Na sljedećoj slici prikazani su segmenti sedam segmentnog ekrana i njihovi nazivi.



Sljedeća tablica prikazuje koje segmente je potrebno uključiti kako bi se na sedam segmentnom ekranu prikazale dekadске znamenke:

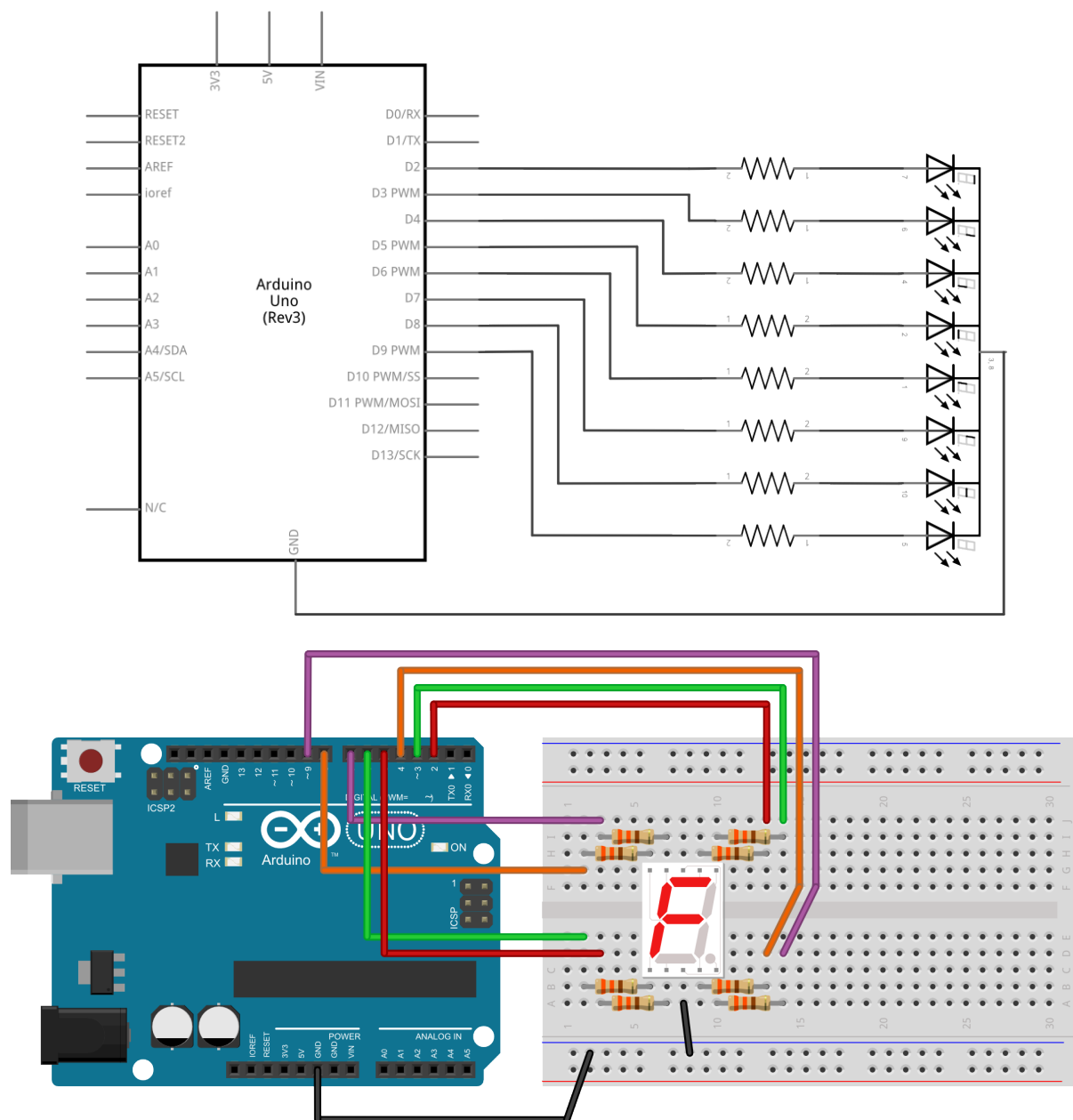
	a	b	c	d	e	f	g
0	1	1	1	1	1	1	0
1	0	1	1	0	0	0	0
2	1	1	0	1	1	0	1
3	1	1	1	1	0	0	1
4	0	1	1	0	0	1	1
5	1	0	1	1	0	1	1
6	1	0	1	1	1	1	1
7	1	1	1	0	0	0	0
8	1	1	1	1	1	1	1
9	1	1	1	1	0	1	1

Vrijednost 1 u tablici znači da segment mora biti uključen a vrijednost 0 znači da segment mora biti isključen.

Trčanje segmenata

Z1: Na mikrokontroler spojite jedan sedam segmentni pokazivač. Redom uključujte segmente a,b,c,d,e,f. Neka je svaki segment uključen 500 milisekundi. Kada ciklus završi neka trčanje krene ispočetka.

Slijedi prikaz spajanja:



Programski kod rješenja:

```

int seg_a = 2;           //definiraj seg_a = 2
int seg_b = 3;           //definiraj seg_b = 3
int seg_c = 4;           //definiraj seg_c = 4
int seg_d = 5;           //definiraj seg_d = 5
int seg_e = 6;           //definiraj seg_e = 6
int seg_f = 7;           //definiraj seg_f = 7
int seg_g = 8;           //definiraj seg_g = 8

void setup() {
pinMode(seg_a, OUTPUT); //postavi izvod seg_a kao izlazni
pinMode(seg_b, OUTPUT); //postavi izvod seg_b kao izlazni
pinMode(seg_c, OUTPUT); //postavi izvod seg_c kao izlazni
pinMode(seg_d, OUTPUT); //postavi izvod seg_d kao izlazni
pinMode(seg_e, OUTPUT); //postavi izvod seg_e kao izlazni
pinMode(seg_f, OUTPUT); //postavi izvod seg_f kao izlazni
pinMode(seg_g, OUTPUT); //postavi izvod seg_g kao izlazni
digitalWrite(seg_a, LOW); //isključi a segment - početno stanje
digitalWrite(seg_b, LOW); //isključi b segment - početno stanje
digitalWrite(seg_c, LOW); //isključi c segment - početno stanje
digitalWrite(seg_d, LOW); //isključi d segment - početno stanje
digitalWrite(seg_e, LOW); //isključi e segment - početno stanje
digitalWrite(seg_f, LOW); //isključi f segment - početno stanje
digitalWrite(seg_g, LOW); //isključi g segment - početno stanje
}

void loop() {
digitalWrite(seg_a, HIGH); //uključuje a segment
delay(500);                //čekaj 500 ms
digitalWrite(seg_a, LOW);  //isključi a segment
digitalWrite(seg_b, HIGH); //uključuje b segment
delay(500);                //čekaj 500 ms
digitalWrite(seg_b, LOW);  //isključi b segment
digitalWrite(seg_c, HIGH); //uključuje c segment
delay(500);                //čekaj 500 ms
digitalWrite(seg_c, LOW);  //isključi c segment
digitalWrite(seg_d, HIGH); //uključuje d segment
delay(500);                //čekaj 500 ms
digitalWrite(seg_d, LOW);  //isključi d segment
digitalWrite(seg_e, HIGH); //uključuje e segment
delay(500);                //čekaj 500 ms
digitalWrite(seg_e, LOW);  //isključi e segment
digitalWrite(seg_f, HIGH); //uključuje f segment
delay(500);                //čekaj 500 ms
digitalWrite(seg_f, LOW);  //isključi f segment
digitalWrite(seg_g, HIGH); //uključuje g segment
delay(500);                //čekaj 500 ms
digitalWrite(seg_g, LOW);  //isključi g segment
}

```

Iz gornjeg programskog koda vidimo kako je programiranje uključivanja i isključivanja segmenata sedam segmentnog ekrana isto kao i programiranje uključivanja i isključivanja svijetlećih dioda.

Brojimo

Z2: Na mikrokontroler spojite jedan sedam segmentni pokazivač. Uključujte segmente pokazivača tako da se na njemu ispisuju brojevi od 1 do 3. Neka svaki broj bude jednu sekundu prikazan na pokazivaču.

Kako su shema i spoj na eksperimentalnoj pločici isti kao i kod prethodnog zadatka, nećemo ih ponovno prikazivati.

Programski kod rješenja:

```
int seg_a = 2;           //definiraj seg_a = 2
int seg_b = 3;           //definiraj seg_b = 3
int seg_c = 4;           //definiraj seg_c = 4
int seg_d = 5;           //definiraj seg_d = 5
int seg_e = 6;           //definiraj seg_e = 6
int seg_f = 7;           //definiraj seg_f = 7
int seg_g = 8;           //definiraj seg_g = 8

void setup() {
  pinMode(seg_a, OUTPUT); //postavi izvod seg_a kao izlazni
  pinMode(seg_b, OUTPUT); //postavi izvod seg_b kao izlazni
  pinMode(seg_c, OUTPUT); //postavi izvod seg_c kao izlazni
  pinMode(seg_d, OUTPUT); //postavi izvod seg_d kao izlazni
  pinMode(seg_e, OUTPUT); //postavi izvod seg_e kao izlazni
  pinMode(seg_f, OUTPUT); //postavi izvod seg_f kao izlazni
  pinMode(seg_g, OUTPUT); //postavi izvod seg_g kao izlazni
  digitalWrite(seg_a, LOW); //isključi a segment - početno stanje
  digitalWrite(seg_b, LOW); //isključi b segment - početno stanje
  digitalWrite(seg_c, LOW); //isključi c segment - početno stanje
  digitalWrite(seg_d, LOW); //isključi d segment - početno stanje
  digitalWrite(seg_e, LOW); //isključi e segment - početno stanje
  digitalWrite(seg_f, LOW); //isključi f segment - početno stanje
  digitalWrite(seg_g, LOW); //isključi g segment - početno stanje
}

void loop() {
  digitalWrite(seg_b, HIGH); //uključuje b segment
  digitalWrite(seg_c, HIGH); //uključuje c segment
  delay(1000); //čekaj 1 s - prikaz broja 1
  digitalWrite(seg_b, LOW); //isključuje b segment
  digitalWrite(seg_c, LOW); //isključuje c segment
  digitalWrite(seg_a, HIGH); //uključuje a segment
  digitalWrite(seg_c, HIGH); //uključuje c segment
  digitalWrite(seg_g, HIGH); //uključuje g segment
  digitalWrite(seg_e, HIGH); //uključuje e segment
  digitalWrite(seg_d, HIGH); //uključuje d segment
  delay(1000); //čekaj 1 s - prikaz broja 2
  digitalWrite(seg_a, LOW); //isključuje a segment
  digitalWrite(seg_c, LOW); //isključuje c segment
  digitalWrite(seg_g, LOW); //isključuje g segment
  digitalWrite(seg_e, LOW); //isključuje e segment
  digitalWrite(seg_d, LOW); //isključuje d segment
  digitalWrite(seg_a, HIGH); //uključuje a segment
  digitalWrite(seg_b, HIGH); //uključuje b segment
  digitalWrite(seg_c, HIGH); //uključuje c segment
  digitalWrite(seg_d, HIGH); //uključuje d segment
  digitalWrite(seg_g, HIGH); //uključuje g segment
}
```

```

delay(1000);           //čekaaj 1 s - prikaz broja 3
digitalWrite(seg_a, LOW); //isključí a segment
digitalWrite(seg_b, LOW); //isključí b segment
digitalWrite(seg_c, LOW); //isključí c segment
digitalWrite(seg_d, LOW); //isključí d segment
digitalWrite(seg_g, LOW); //isključí g segment
}

```

Dodatni zadatci:

Z3: Modificirajte zadatak dva tako da program broji od 0 do 9 i prikazuje to na sedam segmentnom ekranu.

Z4: Možete li na jednom sedam segmentnom ekranu prikazivati brojeve od 0 do 15? Napišite program koji će to učiniti. (Hint: brojavni sustavi :)

Z5: Na mikrokontroler spojite dva sedam segmentna ekrana i jedno tipkalo. Napravite program koji će kada stisnemo tipkalo početi odbrojivati 60 sekundi unatrag i prikazivati stanje na dva sedam segmentna ekrana. Kada brojanje završi neka na ekranima ostane ispisano 00. Ako korisnik ponovno pritisne tipkalo proces kreće ispočetka.