



ARDUINO KROZ JEDNOSTAVNE PRIMJERE

- pripreme za natjecanja -

**PRIPREMA 9 - 2015
KORIŠTENJE KORAČNIH MOTORA**

Paolo Zenzerović, mag. ing. el.

Zagreb, 2015.

KORAČNI MOTORI

U ovoj ćemo pripremi naučiti koristiti koračne motore s Arduino mikrokontrolerom.

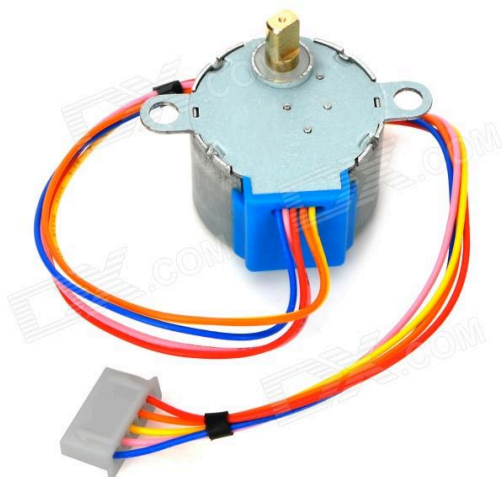
Koračni motor (eng. stepper motor) je posebna vrsta motora koja nam omogućava da na jednostavan način zakrećemo osovinu motora u malim diskretnim pomacima – koracima. Koračni su motori zapravo posebna vrsta DC motora. Postoje dvije vrste koračnih motora – unipolarni i bipolarni koračni motori. U ovoj ćemo pripremi razmatrati samo unipolarne motore.

Tijekom pripreme koristiti ćemo koračni motor oznake 28BYJ-48 koji radi na 5V. Ovaj je motor lako dobavljiv u HR i možete ga kupiti u E-Radionici:

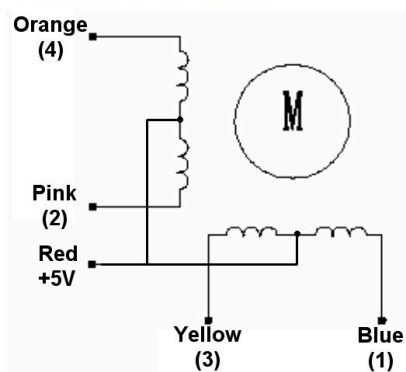
<http://www.e-radionica.com/collections/diy-croduino/products/stepper-motor-driver-stepper-motor>

Koračni motori unutar sebe imaju nekoliko zavojnica koje kada kroz njih poteče struja postaju elektromagneti. Rotor koračnog motora na sebi ima permanentne magnete koje privlači magnetsko polje zavojnica kada kroz njih teče struja. Propuštanjem struje kroz zavojnice koračnog motora određenim redoslijedom rotor koračnog motora možemo zakretati korak po korak u jednom ili drugom smjeru.

Kod unipolarnih koračnih motora najčešće nalazimo dvije zavojnice sa srednjim izvodom. Svaka zavojnica ima dva krajnja izvoda i srednji izvod. Kod 28BYJ-48 koračnog motora srednji izvodi obje zavojnice spojeni su zajedno pa motor ima 5 izlaznih žica za spajanje kako je prikazano na sljedećoj slici:



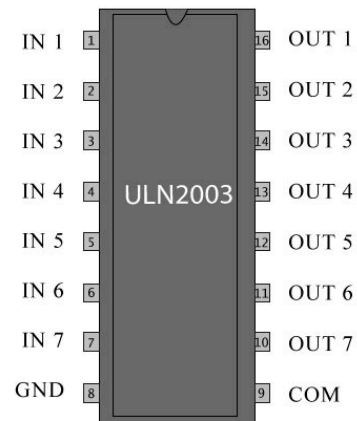
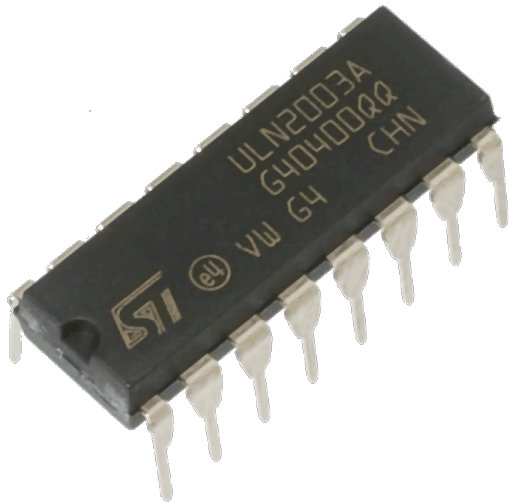
WIRING DIAGRAM



Na lijevoj slici prikazan je motor, a na desnoj su slici prikazane zavojnice u motoru i njihovi izvodi prema bojama žica.

Nažalost, koračne motore nije moguće priključiti direktno na mikrokontroler jer je struja koja je potrebna da bi motor radio znatno veća od one koju može dati jedan izvod mikrokontrolera.

Zbog toga ćemo za spajanje mikrokontrolera i koračnog motora iskoristiti integrirani krug ULN2003. Ovaj integrirani krug je vrlo popularno rješenje za spajanje koračnih motora i relela na mikrokontrolere.

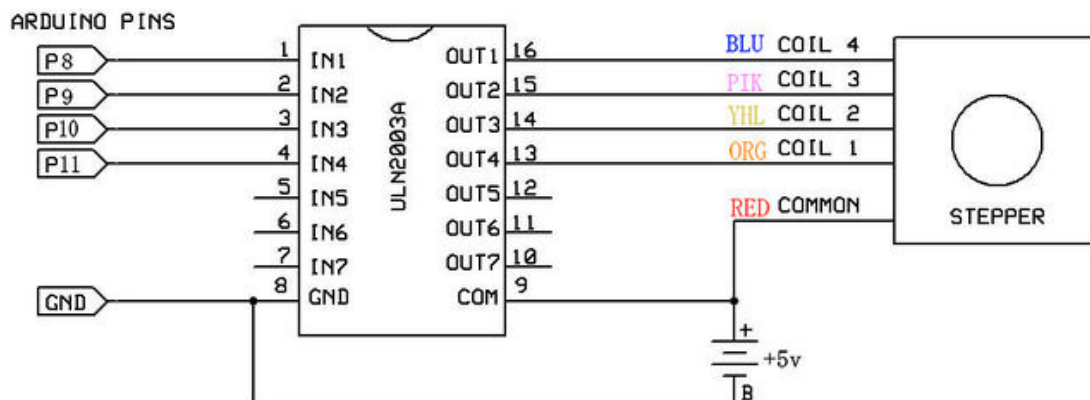
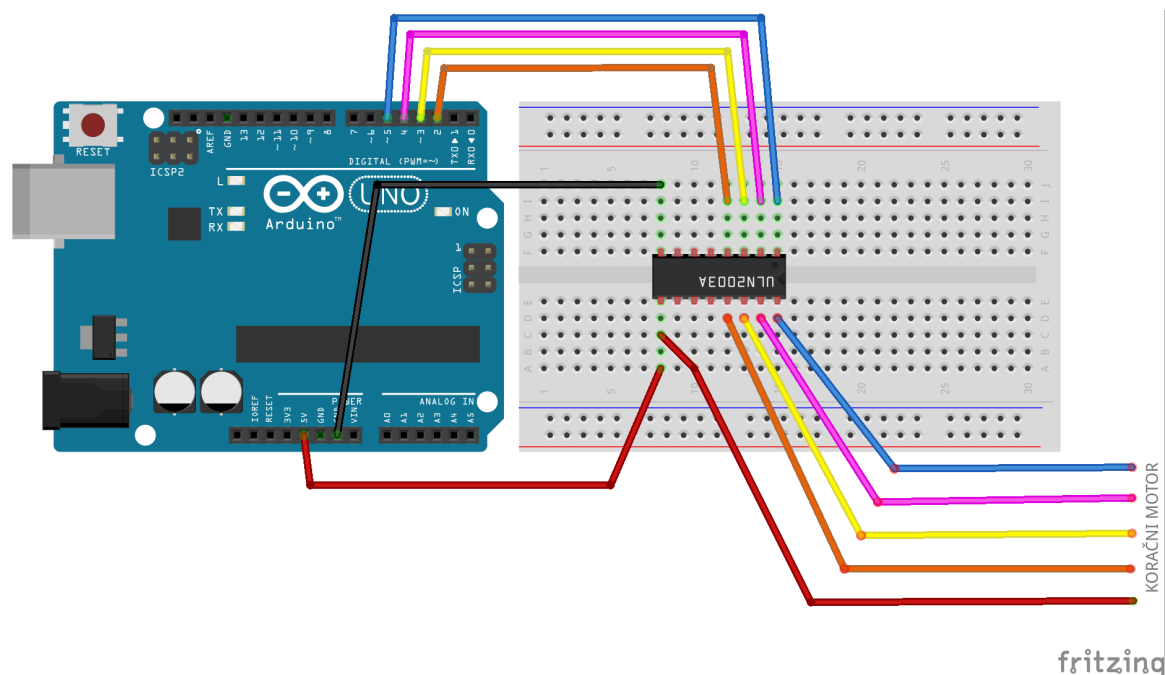


Na lijevoj slici prikazan je izgled integriranog sklopa a na desnoj njegovi izvodi.

Ulazni izvodi 1-7 obilježeni sa IN1, IN2... do IN7 spajaju se na mikrokontroler, dok se izlazni izvodi, obilježeni sa OUT1, OUT2... do OUT7 spajaju na koračni motor. Na izvod GND potrebno je spojiti masu napajanja mikrokontrolera.

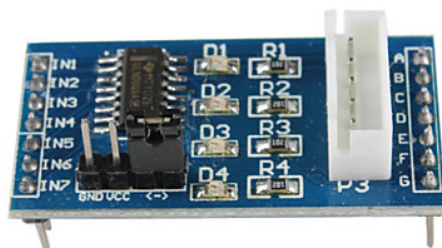
Bilo bi najbolje da je napajanje koračnog motora izdvojeno od napajanja mikrokontrolera, ali obzirom da ovaj koračni motor nema potrebu za velikom strojom pri radu mi ćemo ga napajati s istog napajanja kao i mikrokontroler pa ćemo na COM izvod spojiti 5V s Arduino pločice, a na GND izvod ULN2003 integriranog kruga GND izvod s Arduino pločice.

Spoj koristeći direktno ULN2003 integrirani krug i Arduino pločicu nalazi se na sljedećoj stranici.



Izvode mikrokontrolera na koje ćete spojiti motor naravno možete birati prema vašoj želji.

Osim direktnog spoja koristeći ULN2003 integrirani sklop (kojeg je moguće zasebno kupiti u Chipoteci ili sl.) možete iskoristiti i pločicu s ULN2003 sklopom koju dobijete uz kupljeni motor. Izgled pločice prikazan je na sljedećoj slici:



Ako koristite pločicu, na izvode IN1 do IN4 spojite Arduino izvode, na izvode GND i VCC spojite GND i 5V izvode a motor možete priključiti direktno na bijeli konektor na samoj pločici.

Sada kad smo uspješno spojili motor na driver i na Arduino pogledajmo kako ćemo upravljati motorom.

Postoji više načina pokretanja koračnog motora ali mi u ovoj pripremi nećemo ići u detalje oko načina pokretanja već ćemo iskoristiti gotov primjer iz Stepper knjižnice za Arduino.

```

/*
Stepper Motor Control - one revolution
This program drives a unipolar or bipolar stepper motor.
The motor is attached to digital pins 8 - 11 of the Arduino.
The motor should revolve one revolution in one direction, then
one revolution in the other direction.
Created 11 Mar. 2007
Modified 30 Nov. 2009
by Tom Igoe
*/

#include <Stepper.h>

const int stepsPerRevolution = 32; // change this to fit the number of steps per revolution
// for your motor

// initialize the stepper library on pins 2 through 5:
Stepper myStepper(stepsPerRevolution, 2,4,3,5);

void setup() {
  // set the speed at 60 rpm:
  myStepper.setSpeed(300);
  // initialize the serial port:
  Serial.begin(9600);
}

void loop() {
  // step one revolution in one direction:
  Serial.println("clockwise");
  myStepper.step(stepsPerRevolution*64);
  delay(500);

  // step one revolution in the other direction:
  Serial.println("counterclockwise");
  myStepper.step(-stepsPerRevolution*64);
  delay(500);
}

```

Gitiv je primjer malo modificiran kako bi radio snažim motorom: Upisali smo da motor ima 32 koraka po okretu i promijenili smo redosljed izvoda u inicijalizaciji koračnog motora – zbog sekvence paljenja pozivamo – 2, 4, 3 pa 5 zbog načina rada knjižnice.

Pri korištenju naredbe myStepper.step pomnožili smo broj koraka s 64 jer ovaj motor na sebi ima ugrađen dodatni prijenos u omjeru 1:64.

Sretno u igri s koračnim motorima :)