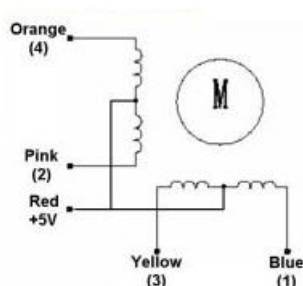
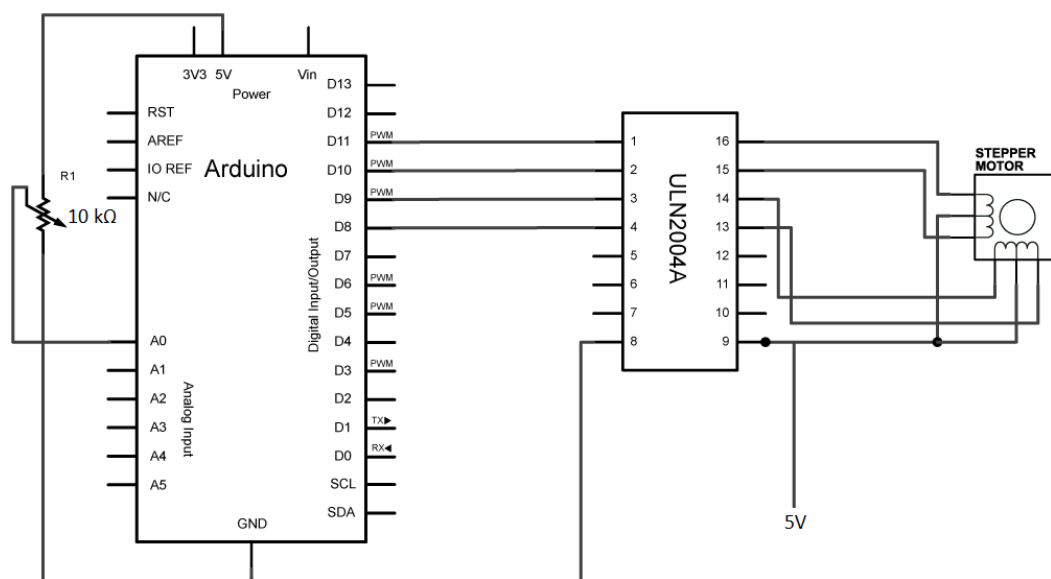


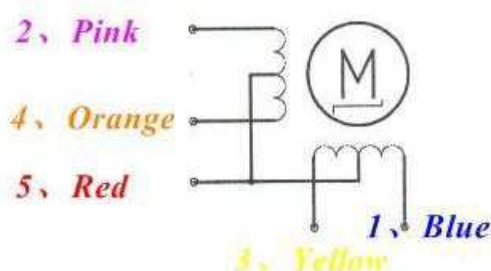
## 18. Stepper motor – nastavak

**ZADATAK 1.** Spoji komponente prema shemi na slici:



Lead Wire Color	---> CW Direction (1-2 Phase)								
	1	2	3	4	5	6	7	8	
4 Orange	-	-							-
3 Yellow		-	-	-					
2 Pink				-	-	-			
1 Blue						-	-	-	

Na slijedećem linku (<https://goo.gl/NV6Wfy>) preuzmi program koji pokreće stepper koristeći *port manipulation* prema tablici (zapravo je tablica djelomično pogrešna jer je motor drugačije spojen za razliku od ovog na shemi iznad – vidi shemu ispod).



Prilagodi program tako da sadrži funkciju koja okrene motor za jedan korak (kao argument prima smjer vrtnje i trenutnu brzinu) te funkciju za zadavanje brzine i akceleracije.

**ZADATAK 2.** Iskoristi prethodno napisani program i prilagodi ga kako bi korisnik mogao zadavati naredbe preko *Serial Monitora* (program strukturiraj koristeći funkcije, kako bi se kasnije lagano moglo prilagoditi program tako da umjesto sa *Serial Monitora* dobiva instrukcije sa npr. bežičnog

modula). Korisnik u početku bira jedan od 3 „modova“ rada: Stalna vrtnja (unos se smjer, početna brzina, maksimalna brzina i akceleracija – brzina je strogo pozitivna, akceleracija može biti bilo koji cijeli broj, smjer očito poprima samo dvije vrijednosti), zakretanje za određen kut (argumenti su kut i brzina) te „prelaženje“ određene udaljenosti (argumenti su radijus priključenog kotača (**double**), udaljenost (**double**), brzina (**unsigned int**) i akceleracija (**int**)) – hint: pređeni put je ukupni kut zakreta \* radijus kotača. Udaljenost i radijus su naravno > 0.

Ako korisnik odabere mod **Stalna vrtnja** zadana brzina se može mijenjati preko potencijometra (vidi prvu shemu).

Korištenje zasebnog argumenta za zadavanje smjera je nepraktično, umjesto toga pametnije je preko predznaka brzine zadati smjer vrtnje (po konvenciji, brzina>0 – obrnuto od smjera kazaljke na satu, brzina<0 – u smjeru kazaljke na satu).

## SPREMI PROGRAM

### ZA BRZE:

**ZADATAK 3.** Prilagodi prethodni program tako da korisnik unosi gotovu sekvencu pokreta (npr. zakreni za 60°, zakreni za -25°, okreći se obrnuto od smjera kazaljke na satu brzinom 2 okr/sek...). Priključi na Arduino gumb kojim će se prisilno zaustaviti stepper odnosno prekinuti tok sekvence (hint – interrupti).

**ZADATAK 4.** Treba napisati program koji mjeri brzinu okretaja, tj. brzinomjer (pogledaj na ploči).

HINT pomoću funkcije *millis()* mjeri period signala sa fotootpornika (ulaz A0). Reći ćemo da je period vrijeme između trenutaka kada vrijednost na ulazu prijeđe neku graničnu vrijednost (vjerojatno MAX\_VRIJ/2, eksperimentiraj sa različitim vrijednostima, kako to utječe na točnost mjerenja).

---

Rated voltage : 5VDC  
Number of Phase 4  
Speed Variation Ratio 1/64  
Stride Angle 5.625° /64  
Frequency 100Hz  
DC resistance 50Ω±7%(25°C)  
Idle In-traction Frequency > 600Hz  
Idle Out-traction Frequency > 1000Hz  
In-traction Torque >34.3mN.m(120Hz)  
Self-positioning Torque >34.3mN.m  
Friction torque 600-1200 gf.cm  
Pull in torque 300 gf.cm

(1 N·m = 10197.1621297793 gf·cm)