

Tesina Di Maturità



Alunno : Galota Mario.

Classe : 5[^]B I.T.I.S.

Indirizzo: E.Elettrotecnica.

Anno Scolastico: 2014 – 2015 .

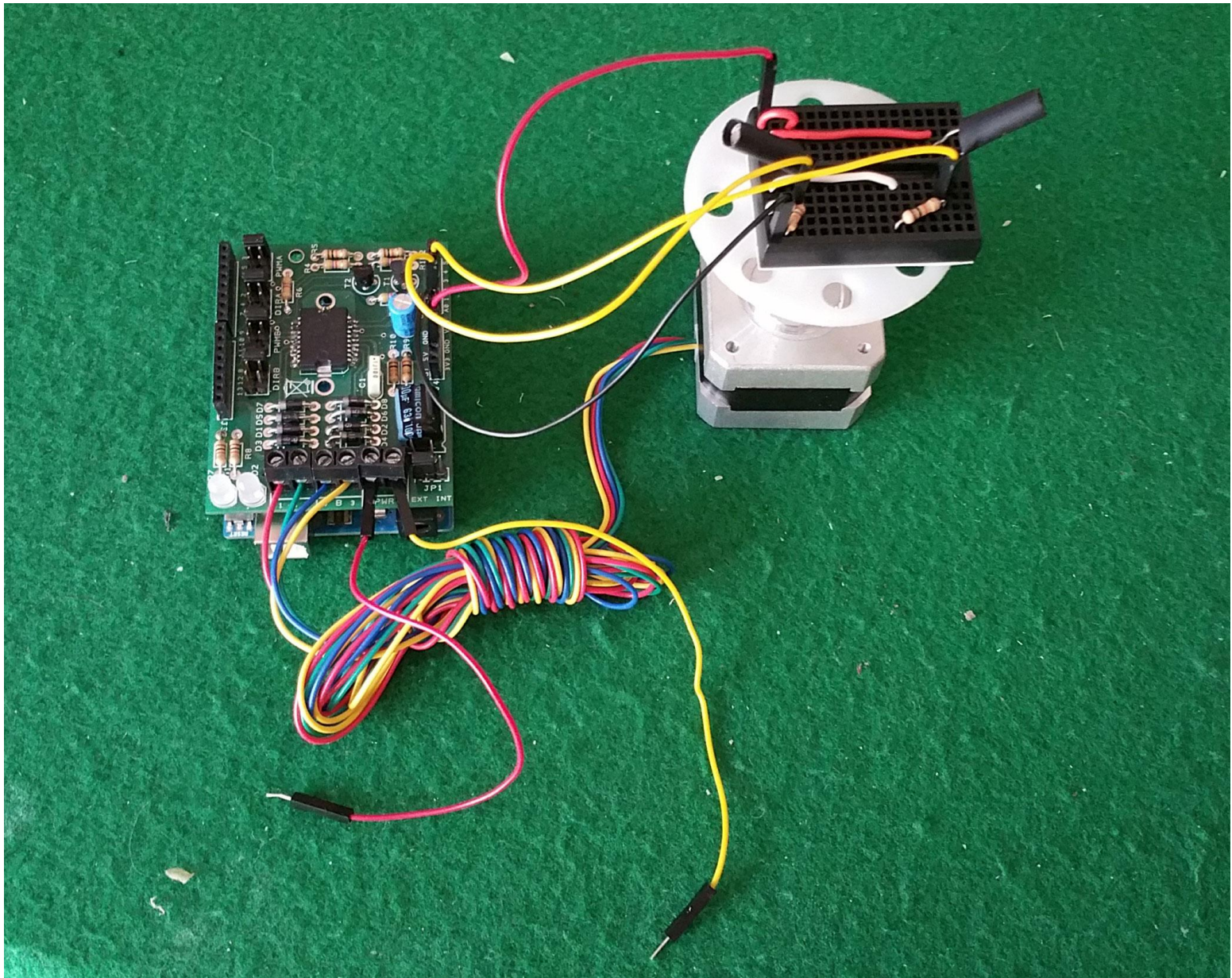
➤ T.D.P. : Inseguitore di luce

In this project I have examined the data distribution system, in particular the stepper motors which are devices that are fed by a sequence of bits.

➤ Obiettivo del progetto :

L'obiettivo di questo progetto era di realizzare un inseguitore di luce capace di individuare il punto più luminoso all'interno di una stanza e quindi una volta individuato il punto più luminoso l'inseguitore si fermerà verso la fonte. Inoltre per rendere il progetto più completo si è cercato di creare un software capace di memorizzare il valore più elevato che viene rilevato durante le misurazioni e successivamente se questo valore cambia il dispositivo inizia di nuovo a cercare la fonte che emette più luce e quindi si verificherà uno spostamento del dispositivo. Per migliorare ulteriormente il progetto sono state utilizzate due foto resistenze e a secondo della foto resistenza più luminosa il motore girerà dal lato corrispondente.

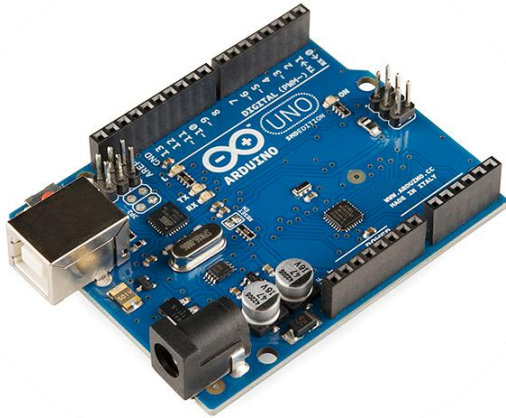




➤ Componenti utilizzati:

Per realizzare questo progetto è stato necessario utilizzare i seguenti componenti:

1. Arduino uno.
2. Motor Shield.
3. 2 foto resistenze.
4. Motore passo passo bipolare.

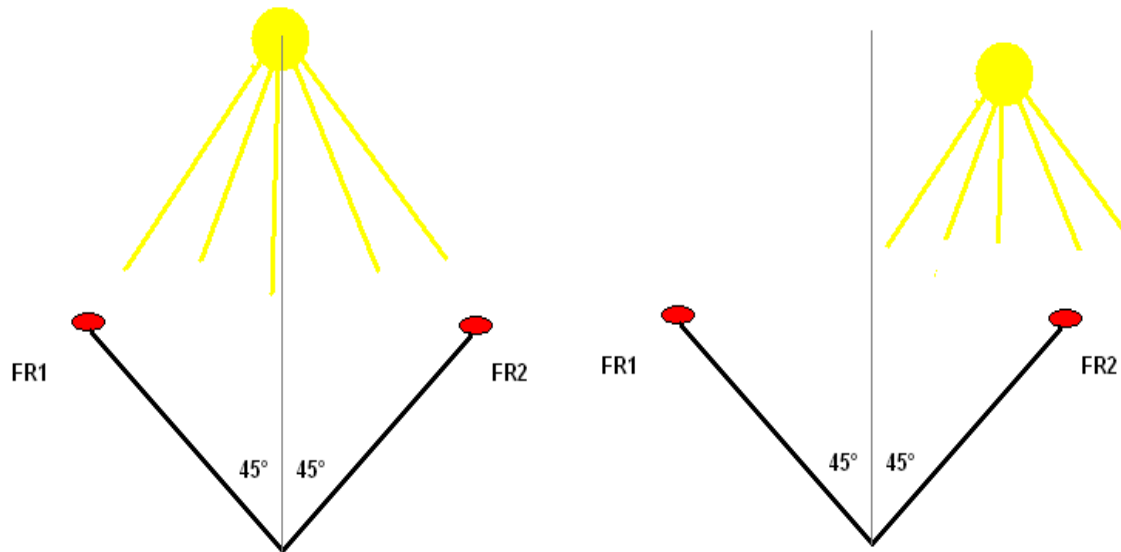


➤ Principio di funzionamento

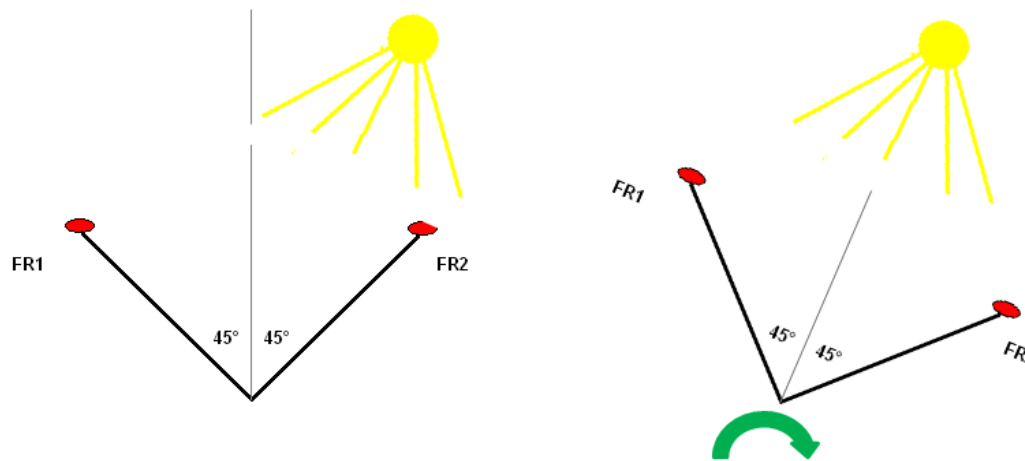
Come già affermato il progetto che ho realizzato, è un inseguitore di luce, quindi il principale scopo è quello di individuare la fonte più luminosa e qualora la posizione della fonte più luminosa variasse il dispositivo va ad individuare dove si trova la nuova fonte luminosa. Il principio di funzionamento è abbastanza semplice e si basa sul seguente ragionamento:



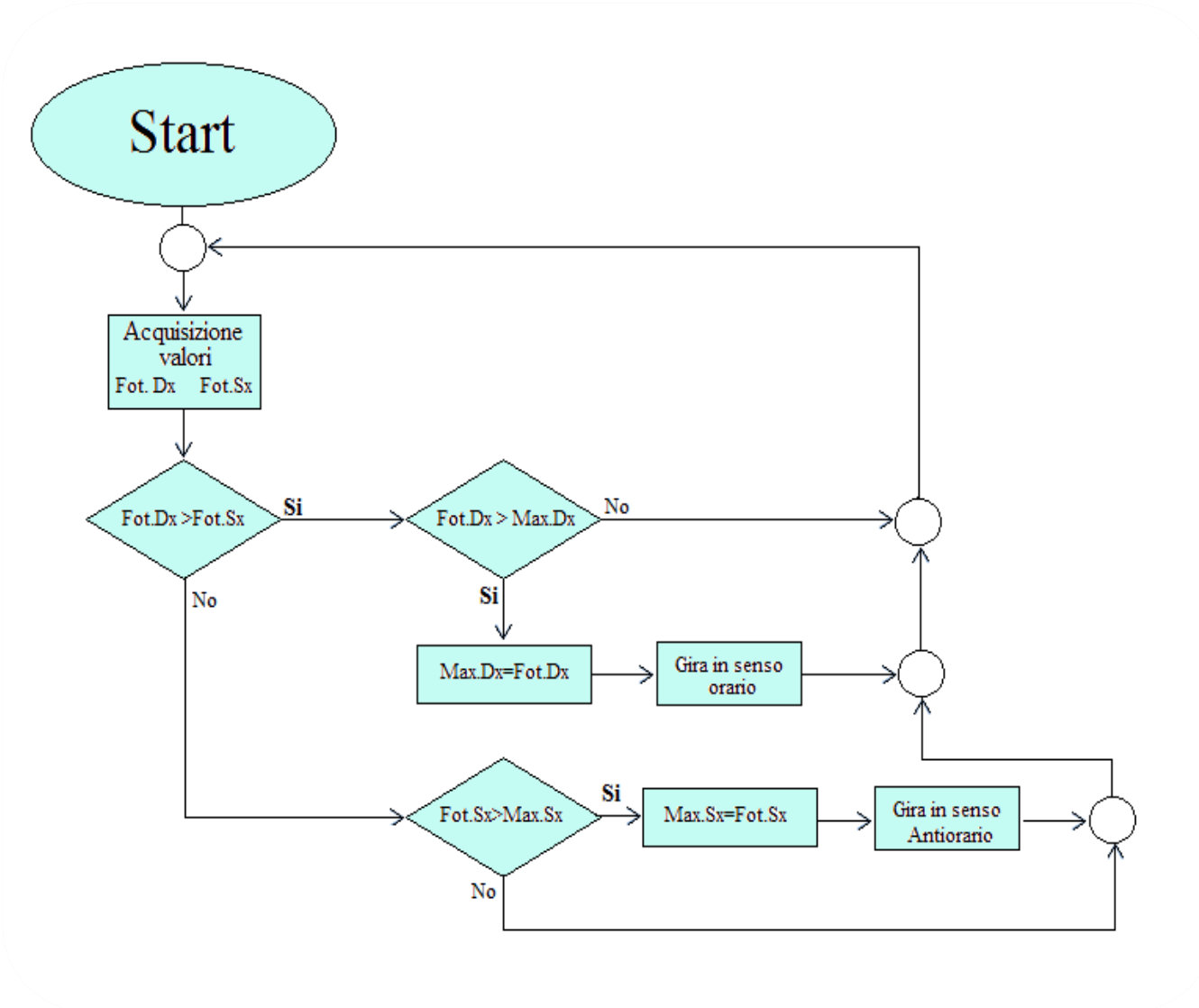
Le due foto resistenze devono essere posizionate in modo da risultare a 90° l'una rispetto all'altra in modo che quando l'illuminazione è verticale presentino lo stesso valore di resistenza, mentre i valori di resistenza si sbilanciano allo spostamento della fonte di luce.



Quando la fonte di luce si muove e di conseguenza il valore delle foto resistenze varia, Arduino deve rilevare il cambiamento rispetto alla condizione iniziale e far muovere il Motore passo passo nella direzione della foto resistenza che riceve più luce, finché la luminosità rilevata non sarà uguale al valore massimo.



Flow Chart



Spiegazione:

Il flow chart che è stato realizzato rappresenta nelle linee generali il procedimento logico che segue il programma che è stato caricato sul microcontrollore, i passaggi che sono eseguiti sono i seguenti:

- Inizialmente dopo aver avviato il microcontrollore (alimentandolo), vengono rilevati i valori di luminosità delle rispettive foto resistenze.
- In seguito viene fatto un confronto tra questi valori per vedere in quale foto resistenza si presenta maggiore luminosità.
- Successivamente si possono verificare due condizioni che permettono di fare azionare il motore o verso destra o verso sinistra.



- Ipotizzando la prima condizione (più luce sulla foto resistenza destra) il μc confronta questo valore con il valore massimo che è riuscito a trovare fino a quel momento, che viene contenuto in una variabile, e successivamente se il valore di luminosità ricevuto è maggiore di quello contenuto nella variabile allora viene aggiornata la variabile Max.Dx e quindi il motore non si muoverà dalla posizione raggiunta fino a quando non varieranno nuovamente le condizioni. Se invece la condizione iniziale che viene presa in considerazione risulta falsa il dispositivo rimarrà fermo perché già si trova in una posizione dove si presenta un'elevata luminosità.
- Ipotizzando la seconda condizione (più luce sulla foto resistenza sinistra) il μc confronta questo valore con il valore massimo che è riuscito a trovare fino a quel momento, che viene contenuto in una variabile, e successivamente se il valore di luminosità ricevuto è maggiore di quello contenuto nella variabile allora viene aggiornata la variabile Max.Sx e quindi il motore non si muoverà dalla posizione raggiunta fino a quando non varieranno nuovamente le condizioni. Se invece la condizione iniziale che viene presa in considerazione risulta falsa il dispositivo rimarrà fermo perché già si trova in una posizione dove si presenta un'elevata luminosità.
- Quindi questo ciclo si ripete per tutto il tempo durante il quale il dispositivo viene alimentato.

Da ciò si deduce che con due foto resistenze si ha il vantaggio di individuare immediatamente quale sia la zona maggiormente illuminata, e quindi il punto maggiormente illuminato viene individuato in maniera più rapida in quanto il dispositivo inizia a girare dalla parte più illuminata.



➤ Lo Sketch:

Di seguito vengono riportate le parti principali del programma che è stato caricato sul microcontrollore :

questa è la parte iniziale dello sketch dove vengono indicate le variabili, e quindi ne viene specificata la loro tipologia; successivamente si procede con il «void Setup» dove vengono impostati i piedini se sono utilizzati come input o come output.

```
const int sensorpinD =A0;
const int sensorpinS =A1;
int motorPinDirA = 2;
int motorPinDirB = 8;
int motorPinPwmA = 3;
int motorPinPwmB = 9;
int sensorvalueDx=0;
int sensorvalueSx=0;
int MotorvalueDx=0;
int MotorvalueSx=0;
int delayTime = 100;
int MaxDx=0;
int MaxSx=0;

void setup(){
  Serial.begin(9600); //Inizializzazione seriale
  //Definizione dei pin
  pinMode(sensorpinD, INPUT);
  pinMode(sensorpinS, INPUT);
  pinMode(motorPinDirA, OUTPUT);
  pinMode(motorPinDirB, OUTPUT);
  pinMode(motorPinPwmA, OUTPUT);
  pinMode(motorPinPwmB, OUTPUT);
}
```



```

void loop()
{
  sensorvalueDx=analogRead(sensorpinD);
  sensorvalueSx=analogRead(sensorpinS);

  MotorvalueDx= map(sensorvalueDx,500,1200,0,255);
  MotorvalueDx= constrain(MotorvalueDx,0,255);
  MotorvalueSx= map(sensorvalueSx,500,1200,0,255);
  MotorvalueSx= constrain(MotorvalueSx,0,255);

  //più luce fotoresistenza destra rotazione in senso orario
  if (sensorvalueDx > sensorvalueSx){

  if (sensorvalueDx >MaxDx) { MaxSx=0;
MaxDx=sensorvalueDx;
digitalWrite(motorPinDirA, HIGH);
digitalWrite(motorPinDirB, LOW);
digitalWrite(motorPinPwmA, HIGH);
digitalWrite(motorPinPwmB, LOW);
delay(delayTime);

digitalWrite(motorPinDirA, LOW);
digitalWrite(motorPinDirB, HIGH);
digitalWrite(motorPinPwmA, LOW);
digitalWrite(motorPinPwmB, HIGH);
delay(delayTime);

digitalWrite(motorPinDirB, LOW);
digitalWrite(motorPinPwmA, HIGH);
digitalWrite(motorPinPwmB, LOW);
delay(delayTime);

digitalWrite(motorPinDirA, LOW);
digitalWrite(motorPinDirB, LOW);
digitalWrite(motorPinPwmA, LOW);
digitalWrite(motorPinPwmB, HIGH);
delay(delayTime);
}
}
}

```

questa parte rappresenta il programma principale all'interno del quale vengono svolti i controlli per capire se si presenta maggiore luce nel trasduttore destro o in quello sinistro; in particolare si presenta la prima condizione quando risulta essere maggiormente illuminata la fotoresistenza di destra.



```

// più luce fotoresistenza sinistra ruota in senso antiorario
else {
    if (sensorvalueSx >MaxSx) { MaxDx=0;
MaxSx=sensorvalueSx;
digitalWrite(motorPinDirA, LOW );
digitalWrite(motorPinDirB, HIGH);
digitalWrite(motorPinPwmA, LOW );
digitalWrite(motorPinPwmB, HIGH);
delay(delayTime);

digitalWrite(motorPinDirA,HIGH);
digitalWrite(motorPinDirB,LOW);
digitalWrite(motorPinPwmA,HIGH);
digitalWrite(motorPinPwmB,LOW);
delay(delayTime);

digitalWrite(motorPinDirA, LOW);
digitalWrite(motorPinDirB, LOW);
digitalWrite(motorPinPwmA, LOW);
digitalWrite(motorPinPwmB, HIGH);
delay(delayTime);

digitalWrite(motorPinDirA, LOW);
digitalWrite(motorPinDirB, LOW);
digitalWrite(motorPinPwmA, HIGH);
digitalWrite(motorPinPwmB, LOW);
delay(delayTime);}

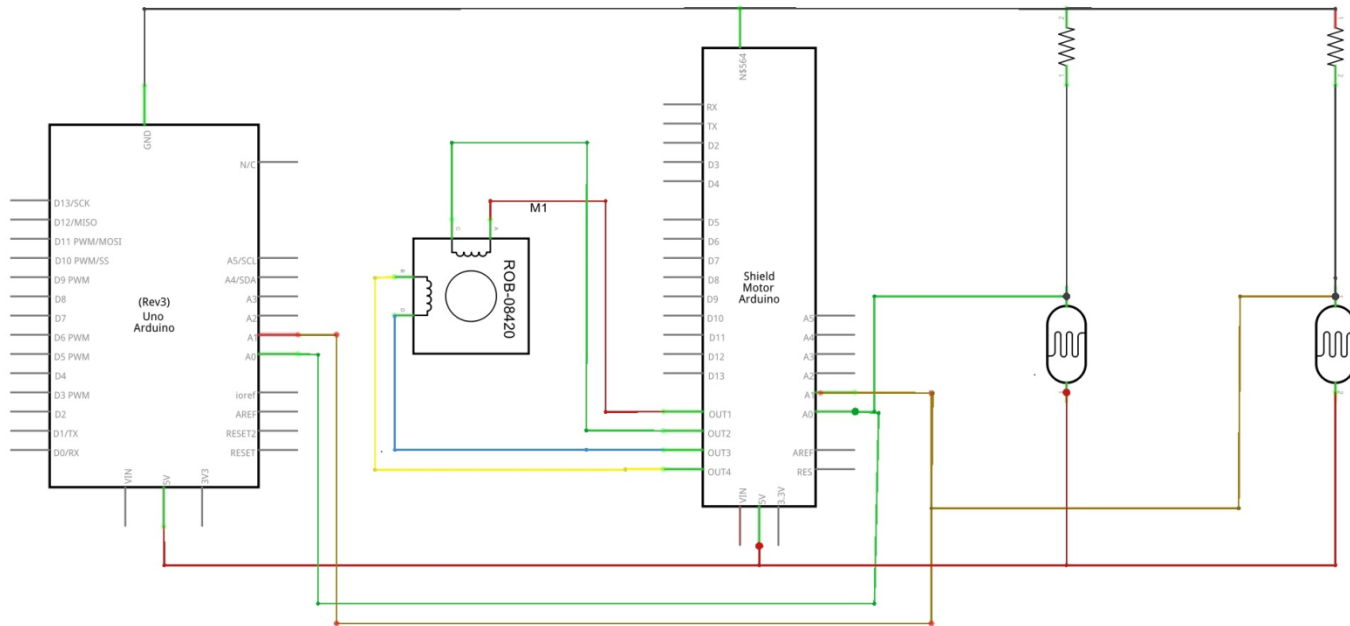
Serial.print("sensor 1 ");
Serial.println(sensorvalueDx);
Serial.print("sensor 2 ");
Serial.println(sensorvalueSx);
Serial.println(MotorvalueDx);
Serial.println(MotorvalueSx);}
}

```

Questa invece è la seconda parte, e si riferisce alla seconda condizione che si può verificare, cioè quella di avere maggiore intensità luminosa nella fotoresistenza di sinistra e di conseguenza il motore si azionerà in senso antiorario; inoltre alla fine del programma prima di mettere l'ultimo end troviamo i comandi "serial print ln" che vengono utilizzati per stampare i valori dei trasduttori nell'apposito monitor seriale.



Schema elettrico

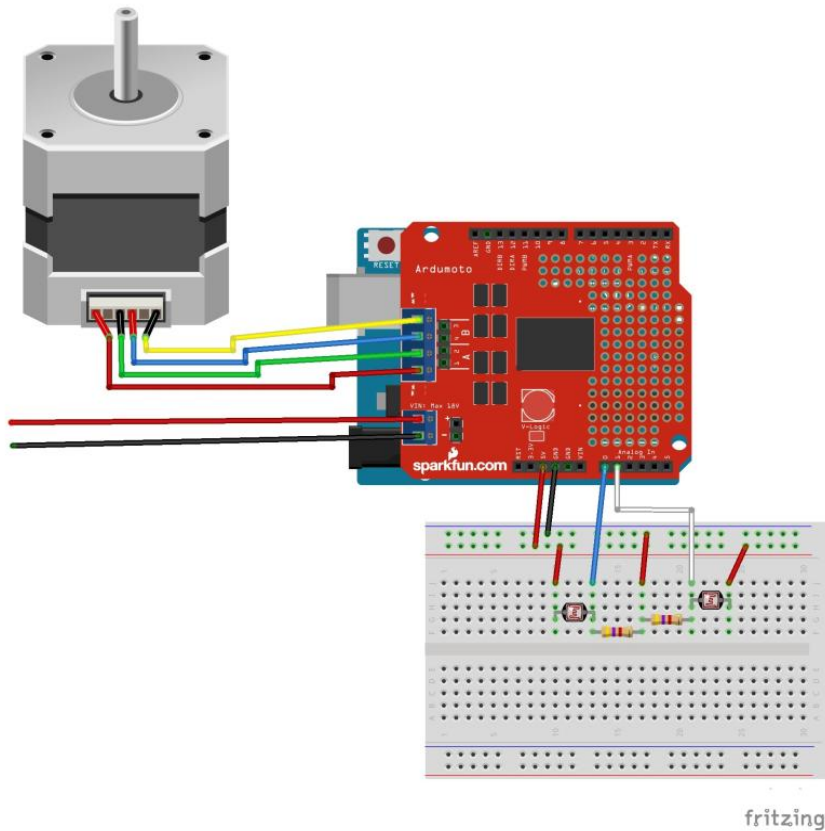


fritzing

Questo è lo schema elettrico dell'inseguitore di luce che è stato realizzato, e come si può notare la scheda Arduino viene collegata in cascata con un'altra scheda che è la Motor shield che in questo caso consente di pilotare il motore passo passo.

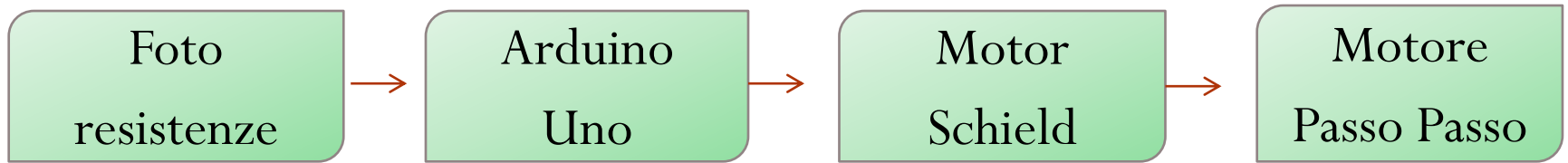


Schema di montaggio



Lo schema riportato a fianco rappresenta il modo in cui è stato montato il circuito, quindi si può vedere com'è stata inserita la Motor Shield, infatti è stata inserita sull'Arduino stesso. Si nota inoltre che dalla scheda partono diversi fili, i quattro colorati rappresentano l'alimentazione delle bobine del motore, mentre quello nero e quello rosso servono per alimentare il tutto.

Schema a blocchi



Adesso iniziamo con la descrizione dei blocchi approfondendo in modo particolare il blocco riguardante il motore passo passo.



Primo Blocco → Fotoresistenze

I trasduttori fotoelettrici trasformano variazioni di intensità luminosa in variazioni di grandezza elettrica. A parte i dispositivi fotoemissivi e le celle fotovoltaiche che, rispettivamente, trovano applicazioni nelle misure fotometriche in astronomia e negli esposimetri fotografici, sono gli elementi fotoconduttori che hanno più larghe applicazioni come trasduttori fotoelettrici. Agli elementi fotoconduttori appartengono le fotoresistenze, i fotodiodi e i fototransistori.



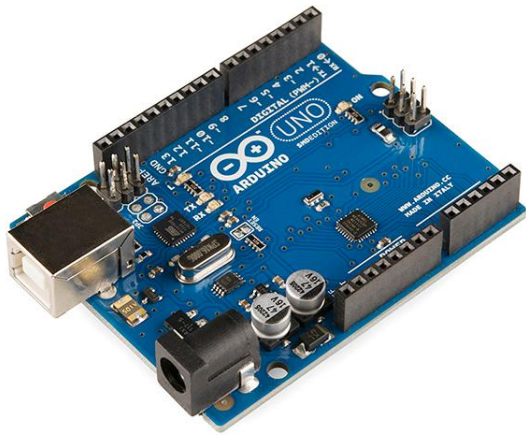
Fotoresistenza CdS



Le fotoresistenze traducono l'informazione luminosa in una variazione di resistenza. Sono realizzate con materiali semiconduttori composti (drogati di tipo N). Le più comuni sono le CdS (solfuro di cadmio) e le PbS (solfuro di piombo). Tra i pregi delle fotoresistenze vi è l'elevata sensibilità e la robustezza. Sono anche economiche e possono essere alimentate sia in corrente continua che alternata.



Secondo blocco → Arduino

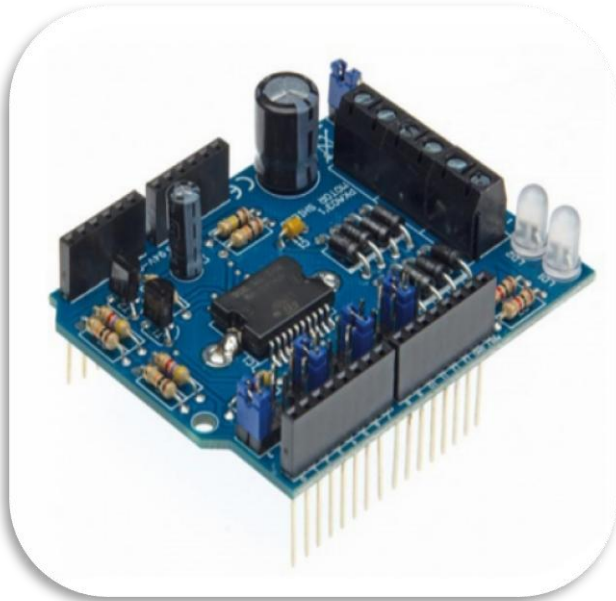


Arduino è una piattaforma OPEN-SOURCE. Arduino è una semplice scheda di Input/Output con un μc . Implementa Hardware e software facile da utilizzare. Per la programmazione viene utilizzato un linguaggio semplice adatto anche ai principianti. Può essere interfacciato con un PC , tramite cavetto USB. Come già abbiamo visto in precedenza con la scheda Arduino si posso realizzare molti sistemi di controllo dai più semplici ai più complessi.

Approfondimento → 



Terzo blocco → Motor shield



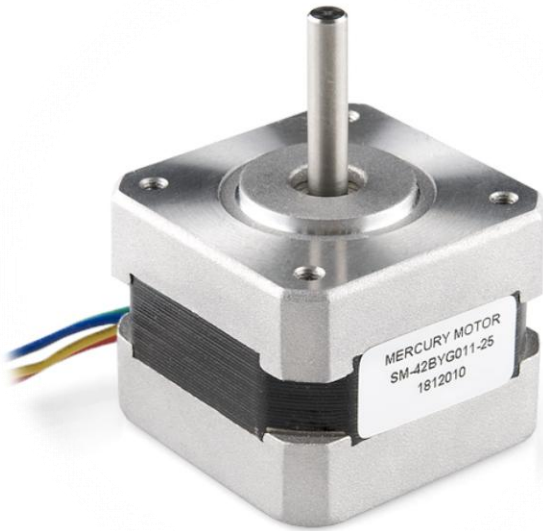
- Questa scheda è basata su l'integrato doppio full bridge L298P in grado di pilotare sia motori DC sia passo-passo.
- La scheda ha una morsettiera per il collegamento di due motori DC e un motore passo-passo bipolare.
- È munita pin strip maschio/femmina che consentono l'inserimento in cascata di altre schede.
- L'alimentazione può essere fornita dall'esterno oppure dalla scheda stessa con una corrente massima di 2 A per fase

Approfondimento → 



Quarto blocco → Motore passo passo bipolare.

I motori passo passo sono dei particolari motori in corrente continua la cui rotazione è ottenuta mediante l'eccitazione in sequenza degli avvolgimenti in esso contenuti. Le differenze maggiori con i comuni motori in corrente continua sono:



- la mancanza di spazzole,
- la presenza di un maggior numero di fili,
- la possibilità di conoscere la posizione dell'albero senza l'utilizzo degli encoder.
- Le caratteristiche del motore, che di solito sono riportate dal costruttore ci permettono di capire quale può essere l'impiego.



Classificazione dei motori passo passo

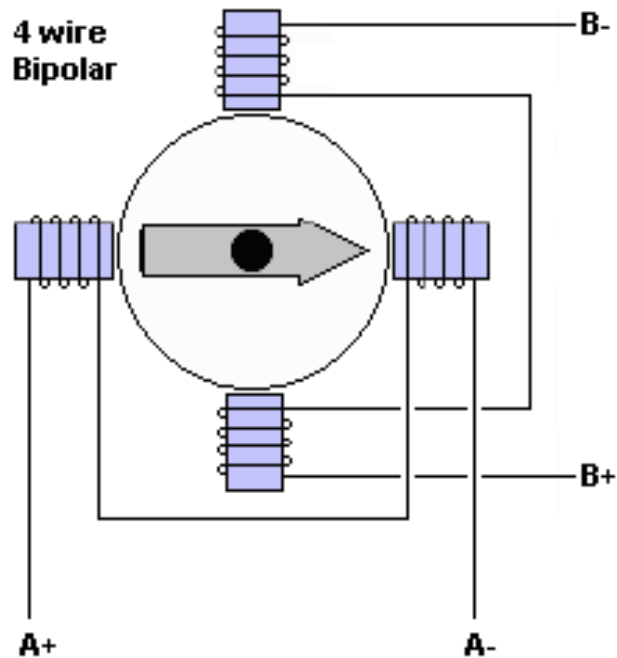
I motori passo passo possono essere classificati in due differenti tipologie :

- **Unipolari:** sono i più semplici da pilotare in quanto la sequenza della rotazione è espressa dal succedersi dell'eccitazione delle bobine di cui è composto.
- **Bipolari:** sono più complessi rispetto ai primi in quanto la rotazione avviene con un campo magnetico generato dalla corrente che percorre in entrambi i sensi le bobine.

C'è da dire però che i bipolari sviluppano maggiore forza rispetto agli unipolari in quanto la rotazione è prodotta da due avvolgimenti contemporaneamente. I motori passo passo si caratterizzano anche per il numero di fili di cui sono fatti che possono essere: 4, 5, 6, 8; quindi per quelli unipolari avremo (5, 6, o 8 conduttori) o bipolare (4, 8 conduttori).



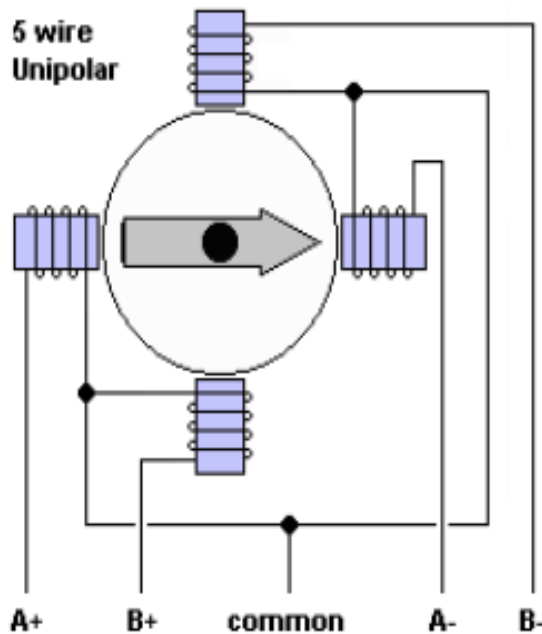
Motori passo passo a 4 fili



In figura è rappresentato un motore passo passo bipolare a 4 fili, pilotare questi motori è possibile facendo percorrere alla corrente gli avvolgimenti in entrambi i versi in modo alternato, in modo da formare un campo magnetico necessario alla corretta rotazione, inoltre c'è da dire che questi motori non possono essere pilotati in altro modo.



Motori passo passo a 5 fili

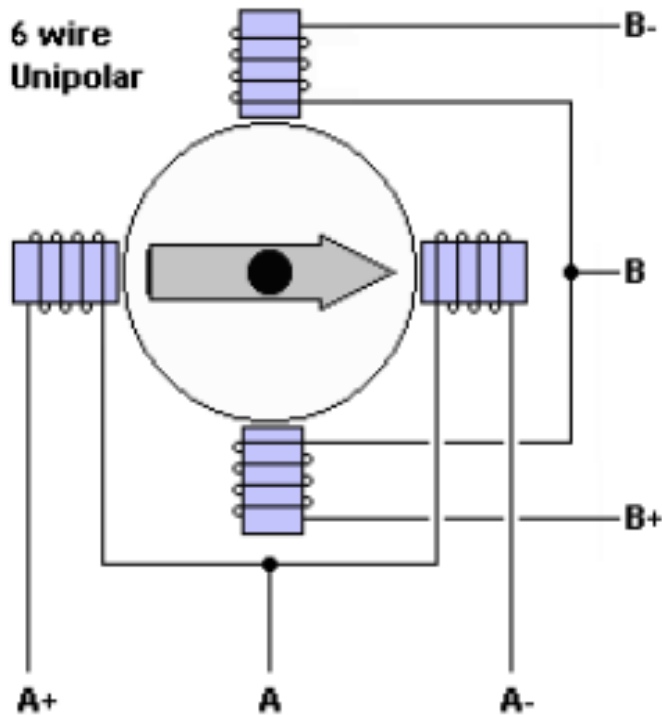


In figura è rappresentato un motore a 5 fili, questi motori sono di tipo unipolare. Per pilotare questi motori è necessario alimentare le fasi nella sequenza desiderata, ad esempio per produrre una rotazione in senso orario dell'albero partendo da una situazione come quella illustrata in figura si deve applicare una tensione nella sequenza B+,A+,B-,A-; in senso antiorario avremo quindi la sequenza: B-,A+,B+,A- .



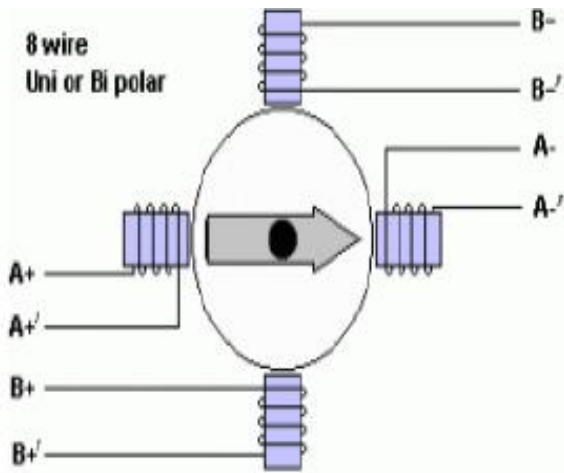
Motori passo passo a 6 fili

Il motore illustrato in figura è un motore passo passo unipolare a sei fili, la figura mostra lo schema costruttivo evidenziando come gli avvolgimenti sono uniti tra di loro. Per pilotare questi motori si può adottare una tecnica simile a quella dei motori a cinque fili, in quanto i comuni delle due fasi (A e B) possono essere connessi assieme rendendo il motore simile a quello a cinque fili. Inoltre è possibile gestire le fasi A e B in maniera indipendente non solo attraverso i terminali A+, A-, B+, B- ma applicando corrente in modo indipendente su A e B.



Motori passo passo a 8 fili

I motori a 8 fili possono essere utilizzati sia in configurazione unipolare che in configurazione bipolare. Lo schema di costruzione riportato in figura mostra le due bobine ciascuna con due terminali indipendenti. Questi motori vengono configurati in base alle necessità che si hanno nel progetto, e quindi in base al lavoro che devono compiere.



GRAZIE PER L'ATTENZIONE!

Galota Mario