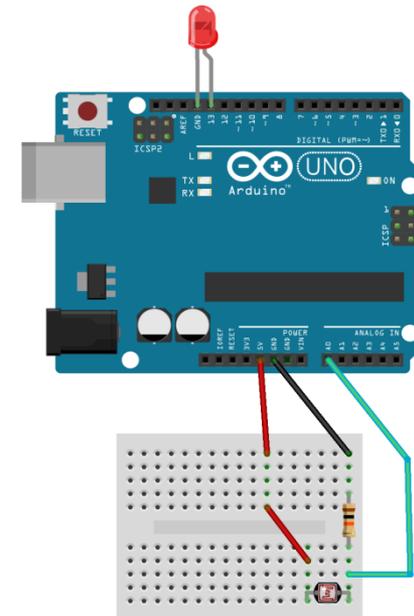
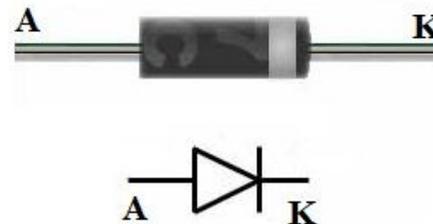
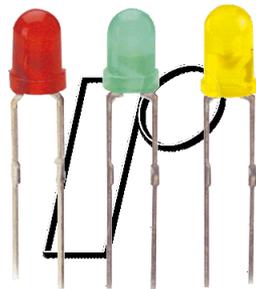
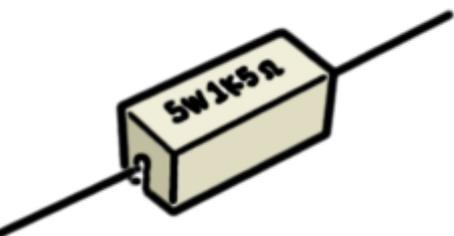


Robotica elementare

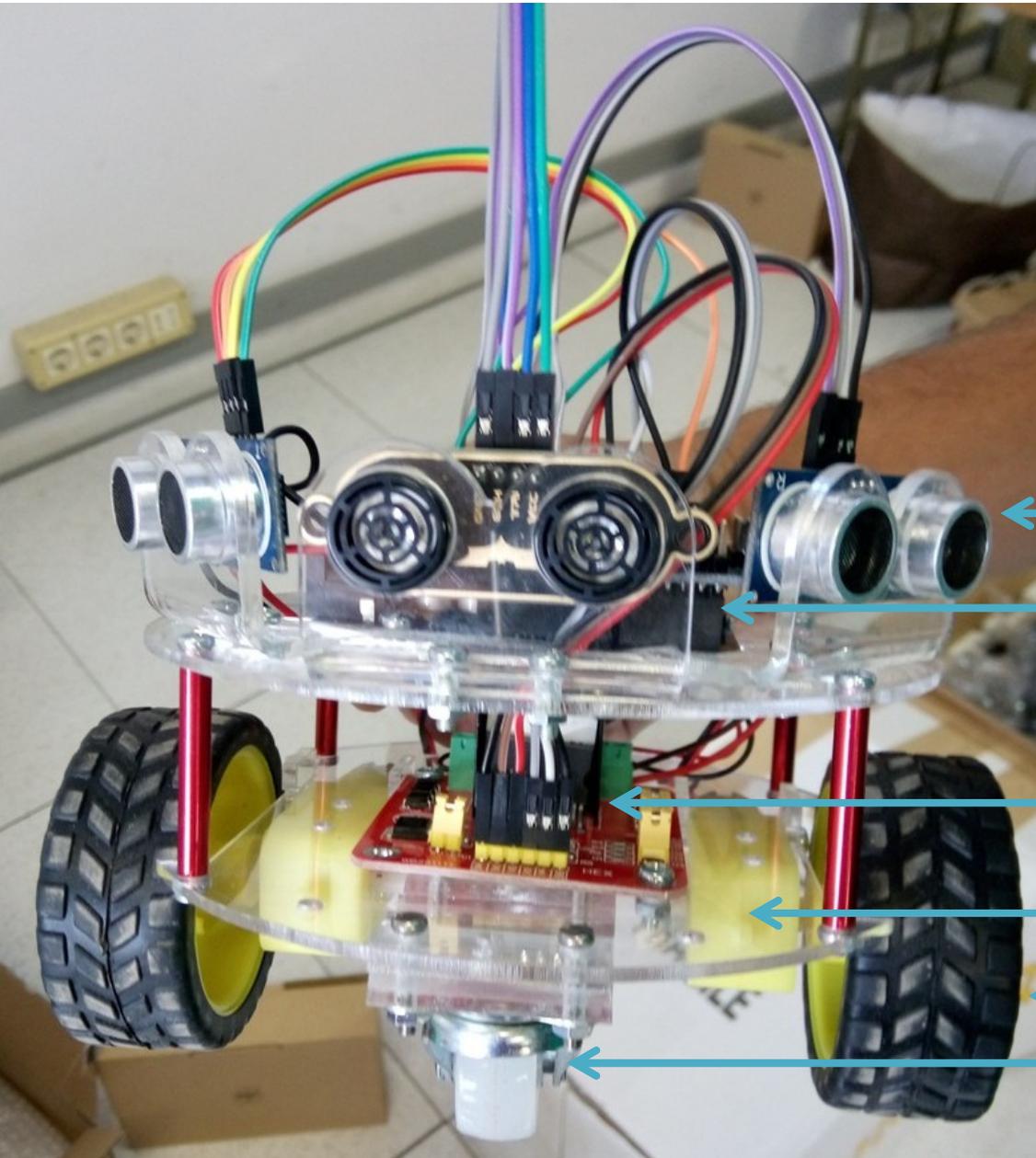
Prof. Giorgio De Nunzio giorgio.denunzio@unisalento.it



Introduzione

- Il corso parlerà di....
 - Com'è fatto un robot su ruote: ROBi-1 KENOBI
 - Pilotaggio dei motori DC
 - Sensori US (file HC-SR04-GDN.pdf)

Il robot: ROBI-1 Kenobi



Sensore US HC-SR04

Arduino

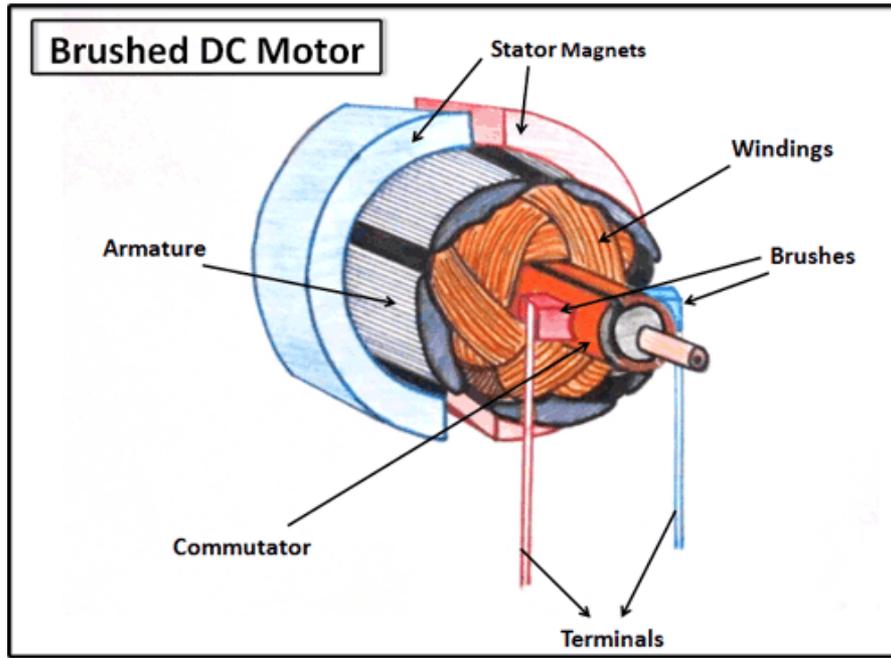
Controller Motori

Motore DC

Ruota traente

Ruotino libero

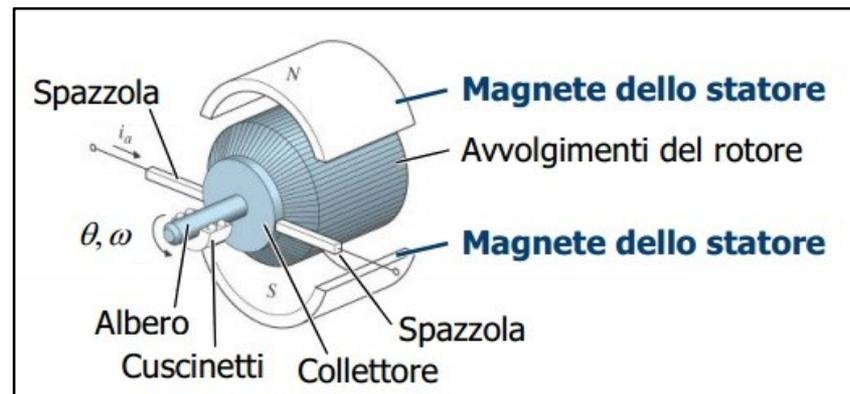
I motori a spazzole in corrente continua (DC brushed motors)



Il motore a spazzole in corrente continua è costituito da due parti, il rotore e lo statore.

Lo statore ha lo scopo di generare un campo magnetico costante.

Il rotore, alimentato in corrente continua, si comporta come un insieme di spire percorse da corrente in un campo magnetico, e quindi si allinea con esso; nella rotazione, la presenza delle spazzole fa sì che il verso della corrente nel rotore si inverta, e quindi esso sia portato ad allinearsi nuovamente con il campo magnetico, ruotando ancora, e così via.



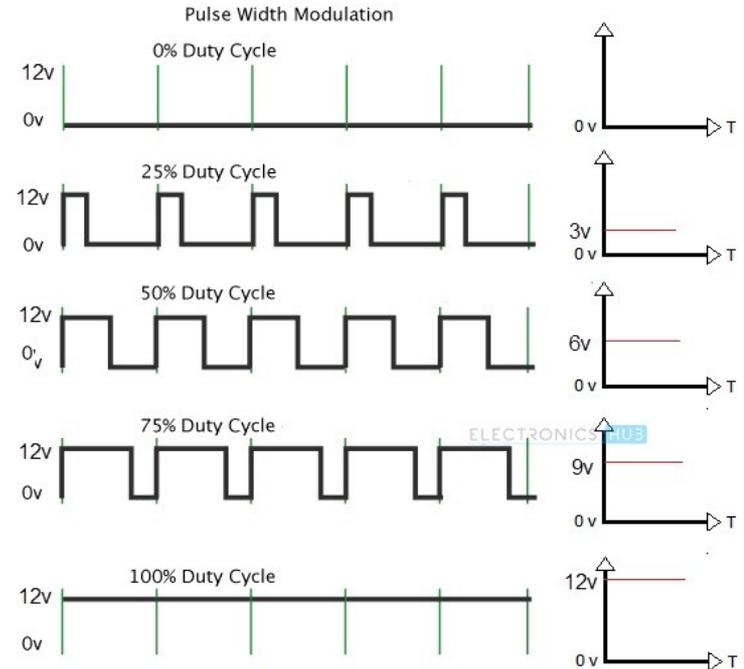
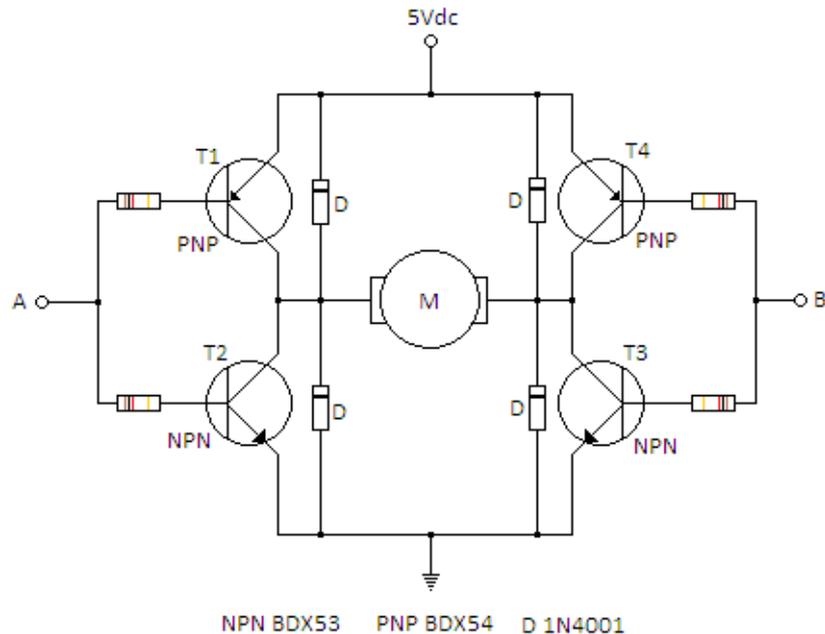
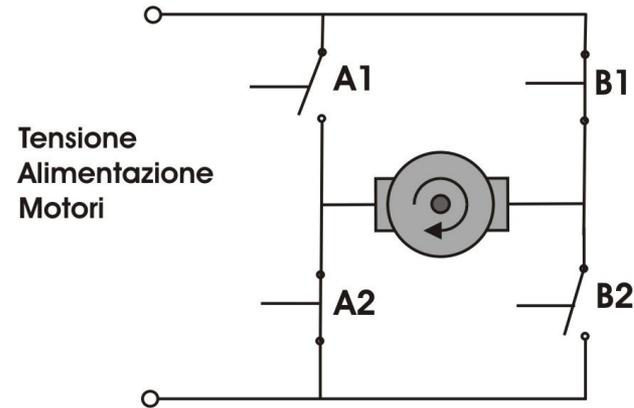
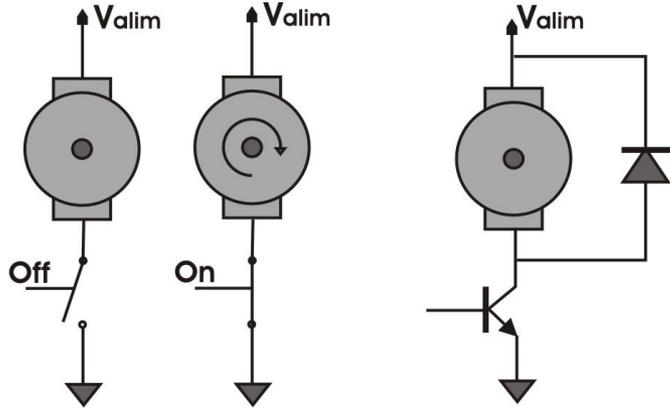
I motori a spazzole in corrente continua (DC brushed motors)

Un motore DC ha grande velocità di rotazione, ma piccola “coppia”.

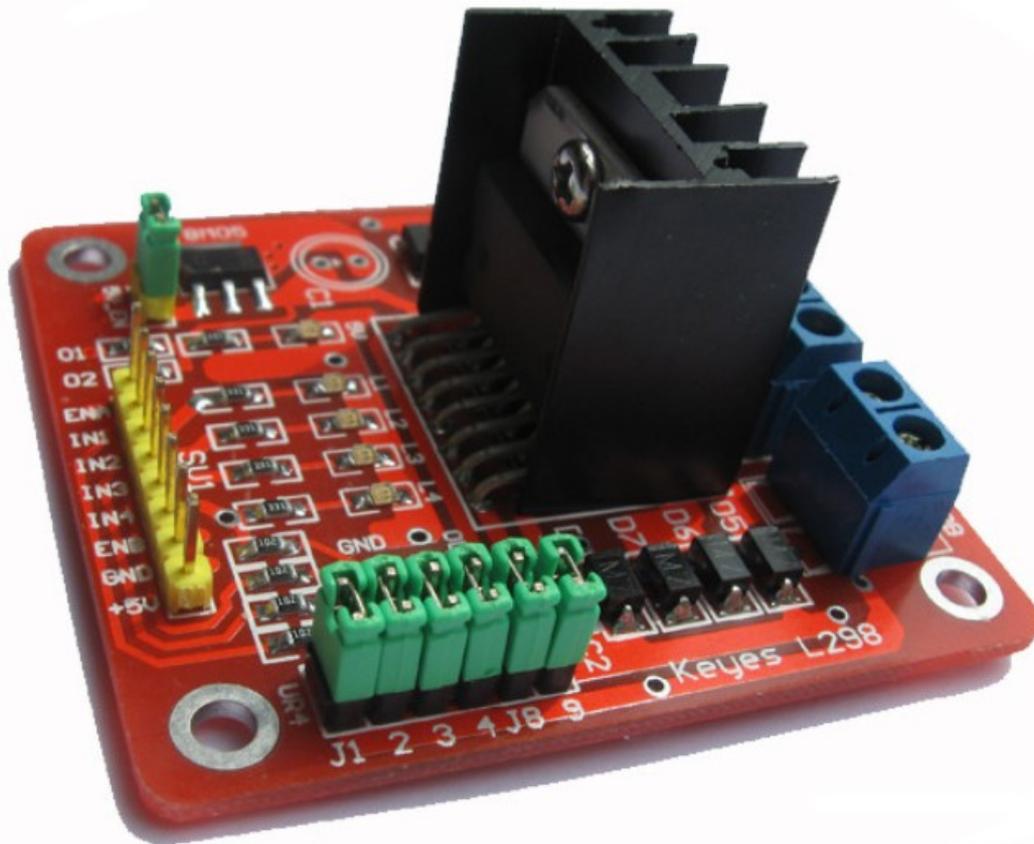
Motoriduttori: motori con ingranaggio di riduzione velocità.



I motori a spazzole in corrente continua (DC brushed motors): PWM



Dual H-Bridge Motor Driver L298



Modulo per il controllo di motori a spazzole in corrente continua o per motori passo-passo (*stepper*).

Leggero, piccole dimensioni
LED per l'indicazione della direzione di moto

Può funzionare con tensioni da 5 ai 35 volt e fornire 2A di corrente per ogni uscita.

Dual H-Bridge Motor Driver L298

Perché abbiamo bisogno del modulo L298 per pilotare i motori?

I motori assorbono molta corrente, e Arduino non può fornirne abbastanza dai suoi pin, per cui è necessario interporre un sistema che, pilotato da Arduino, tragga dalla batteria sufficiente corrente per il funzionamento dei motori.

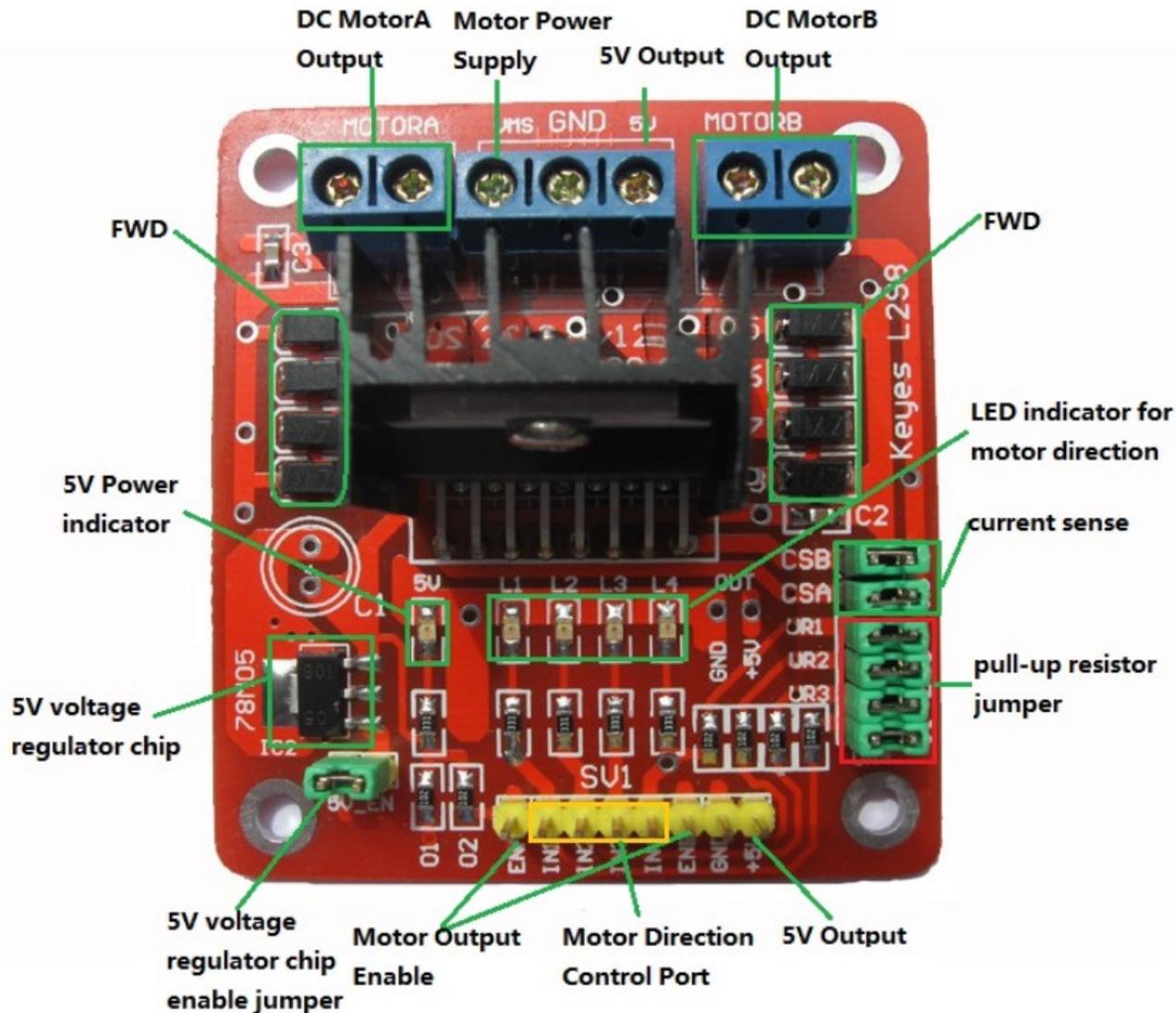
Come funziona?

http://www.fabbrimarco.com/droboitalia/PWM_tutorial_1_0.pdf

Il PWM

Non tutti i pin di Arduino sono PWM ready, cioè abilitati a fornire questo tipo di segnale. Nella maggior parte dei casi (Arduino che montano i processori ATmega168 o ATmega328), questa funzione è abilitata sui pin 3, 5, 6, 9, 10 e 11.

Dual H-Bridge Motor Driver L298



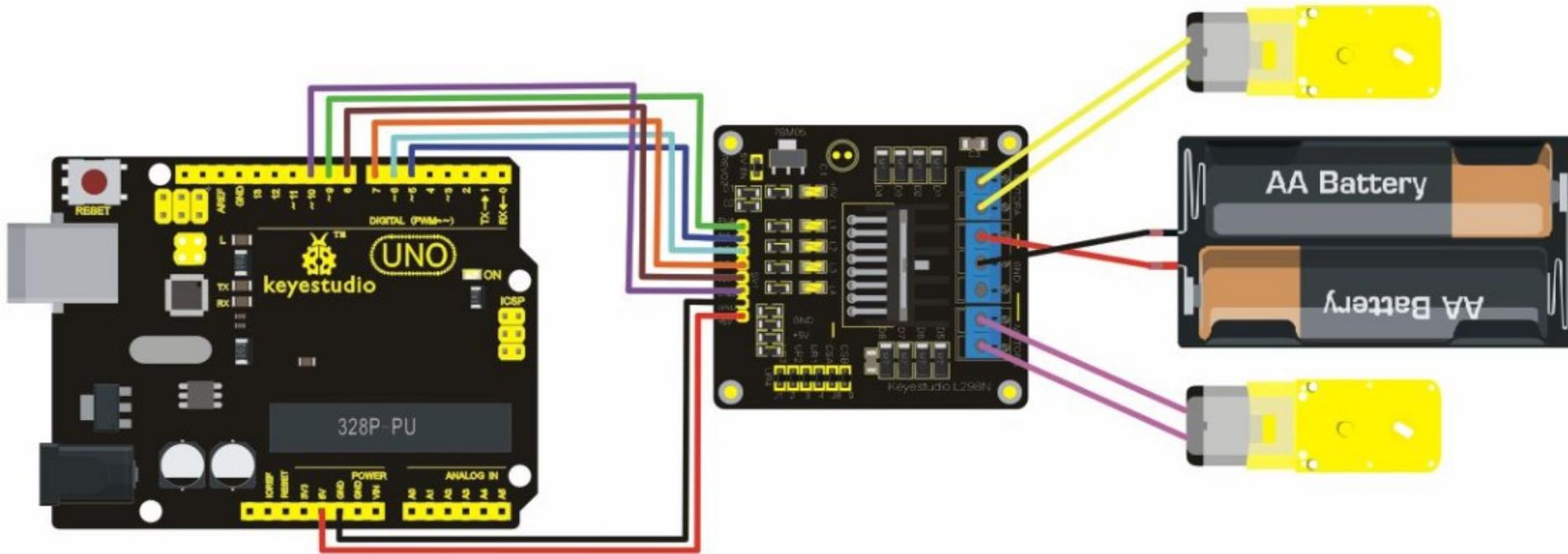
Dual H-Bridge Motor Driver L298

Primo esperimento: pilotare un singolo motore DC, variandone la velocità con un potenziometro.

<http://www.lombardoandrea.com/l298n-pilotare-un-motore-dc-con-arduino/>

Dual H-Bridge Motor Driver L298

Secondo esperimento: pilotare due motori DC, variandone direzione e velocità da programma.



Controllo di due motori DC a spazzole.

Dual H-Bridge Motor Driver L298

```
int IN1 = 5;
int IN2 = 6;
int IN3 = 7;
int IN4 = 8;
int ENA = 9;
int ENB = 10;

void setup(){
  for(int i=5; i<11; i++){
    pinMode(i,OUTPUT);
  }
}
```

Controllo di due motori DC a spazzole.

```
void loop(){
  //rotateCW
  digitalWrite(IN1,LOW);
  digitalWrite(IN2,HIGH);
  analogWrite(ENA,150);
  digitalWrite(IN3,LOW);
  digitalWrite(IN4,HIGH);
  analogWrite(ENB,150);
  delay(1000);
  //pause for 1S
  analogWrite(ENA,0);
  analogWrite(ENB,0);
  delay(1000);
  //rotateCCW
  digitalWrite(IN1,HIGH);
  digitalWrite(IN2,LOW);
  analogWrite(ENA,100);
  digitalWrite(IN3,HIGH);
  digitalWrite(IN4,LOW);
  analogWrite(ENB,100);
  delay(1000);
  //pause for 1S
  analogWrite(ENA,0);
  analogWrite(ENB,0);
  delay(1000);
}
```

Dual H-Bridge Motor Driver L298

Riorganizzare il codice scrivendo funzioni per

- 1) Muovere la base mobile “avanti”: `void forward(int leftSpeed, int rightSpeed)`
- 2) Muovere la base mobile “indietro” : `void backward(int leftSpeed, int rightSpeed)`
- 3) Far ruotare la base mobile sul posto, in senso orario
- 4) Far ruotare la base mobile sul posto, in senso antiorario
- 5) Fermare i motori

Scrivere il codice in modo che la base mobile vada avanti per un secondo, si fermi per un secondo, ruoti sul posto per 0,5 secondi, verso destra, e così via, scegliendo un percorso

Può succedere che la base mobile, a causa della non perfetta eguaglianza di caratteristiche tra i motori, non proceda su traiettoria rettilinea (per un percorso di qualche metro) quando si dia il comando `forward()`. Individuare, allora, la giusta scelta di velocità del motore destro e del motore sinistro affinché la base mobile vada dritto, a una velocità complessiva di circa 0,50m/s.

Dual H-Bridge Motor Driver L298

Terzo esperimento: pilotare un motore DC in base all'ostacolo visto dal sensore HC-SR04.

Programma

- ROBOTICA.pdf
- HC-SR04-GDN.pdf
- Esercizi: A partire dal codice ThreeUSSensorsV2.ino:
 - Studiare il codice
 - Compilare ed eseguire il codice così com'è; ricordare di spegnere il robot prima di caricare il software; staccare il cavetto, poi mettere il robot in posizione sul pavimento in un luogo sicuro in vicinanza di muri e ostacoli, e accenderlo; osservarne il comportamento
 - Decomentando la riga del codice che si riferisce al proprio robot, tarare i motori in modo che il robot vada il più possibile diritto
 - Verificare il funzionamento delle funzioni mRotateRight90() etc, già definite ma non usate nel codice: i valori delle costanti di tempo è corretto o da modificare?
 - Verificare il funzionamento della funzione getDistance(TrigM, EchoM); inviandone il valore letto sul canale seriale e leggendolo con il monitor seriale (o il plotter); disabilitare i motori e tenere naturalmente il robot connesso al pc
 - Fissate le funzioni di movimento avanti, rotazione etc, definire un frammento di codice (da mettere nella loop) che faccia andare il robot avanti per un metro, lo fermi, lo faccia ruotare a sinistra, andare avanti per un metro, ruotare a sinistra, e così via, cercando di riportarlo nella posizione di partenza percorrendo dunque un quadrato.
 - Scrivere il codice che fa andare il robot avanti finché libero da ostacoli, fermi il robot se vi è un ostacolo, e riprenda il movimento solo se l'ostacolo è rimosso. Suggerimento: è sufficiente una istruzione if/else che testi la distanza dell'ostacolo e, se essa è minore di 25 cm, esegua mStop();, altrimenti esegua mForward(int vA, int vB);
 - Al punto precedente, aggiungere l'accensione del LED13 quando vi è l'ostacolo