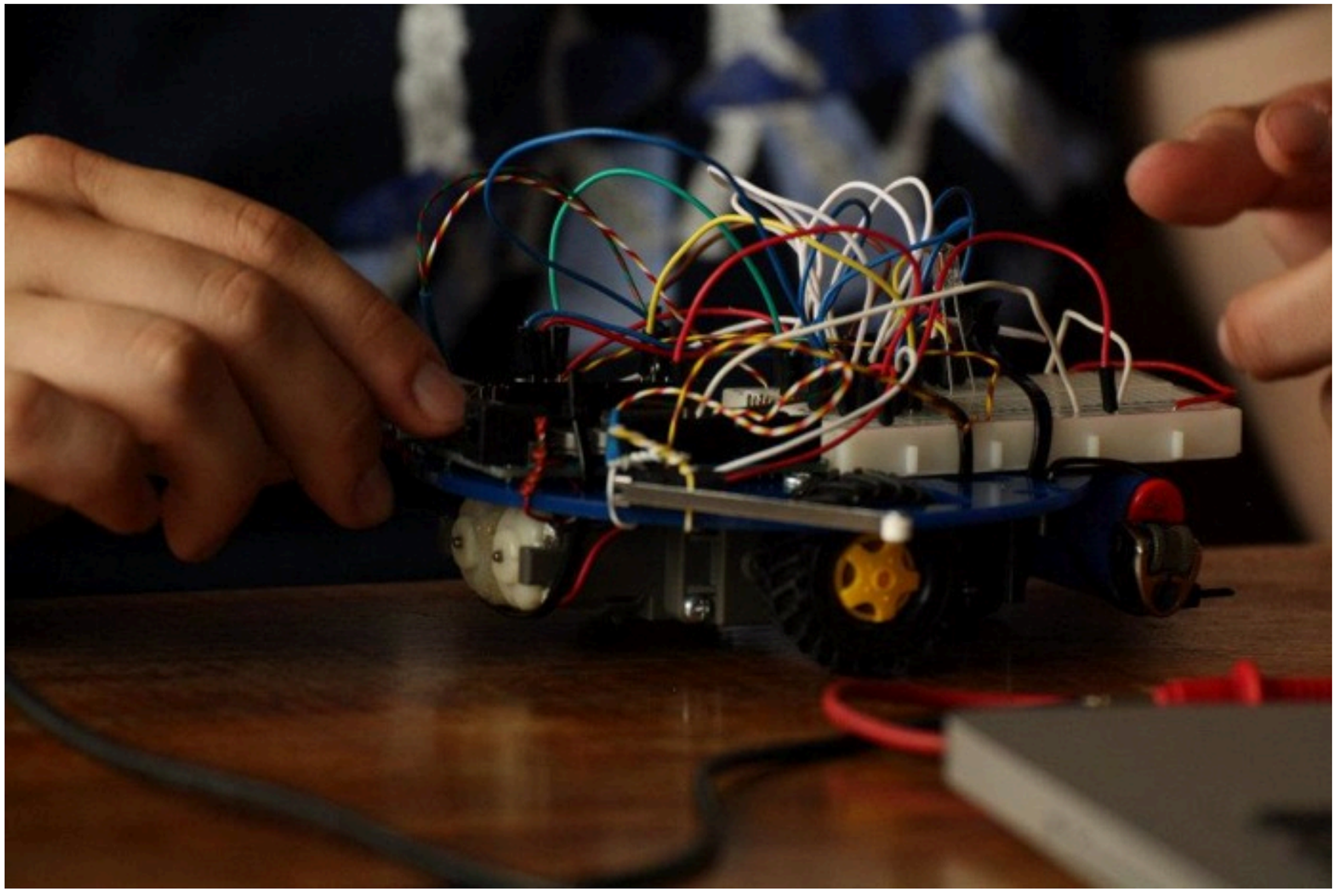
Immagina
progetta
crea

Robot con Arduino

Arduino e la robotica

Scuola di Robotica



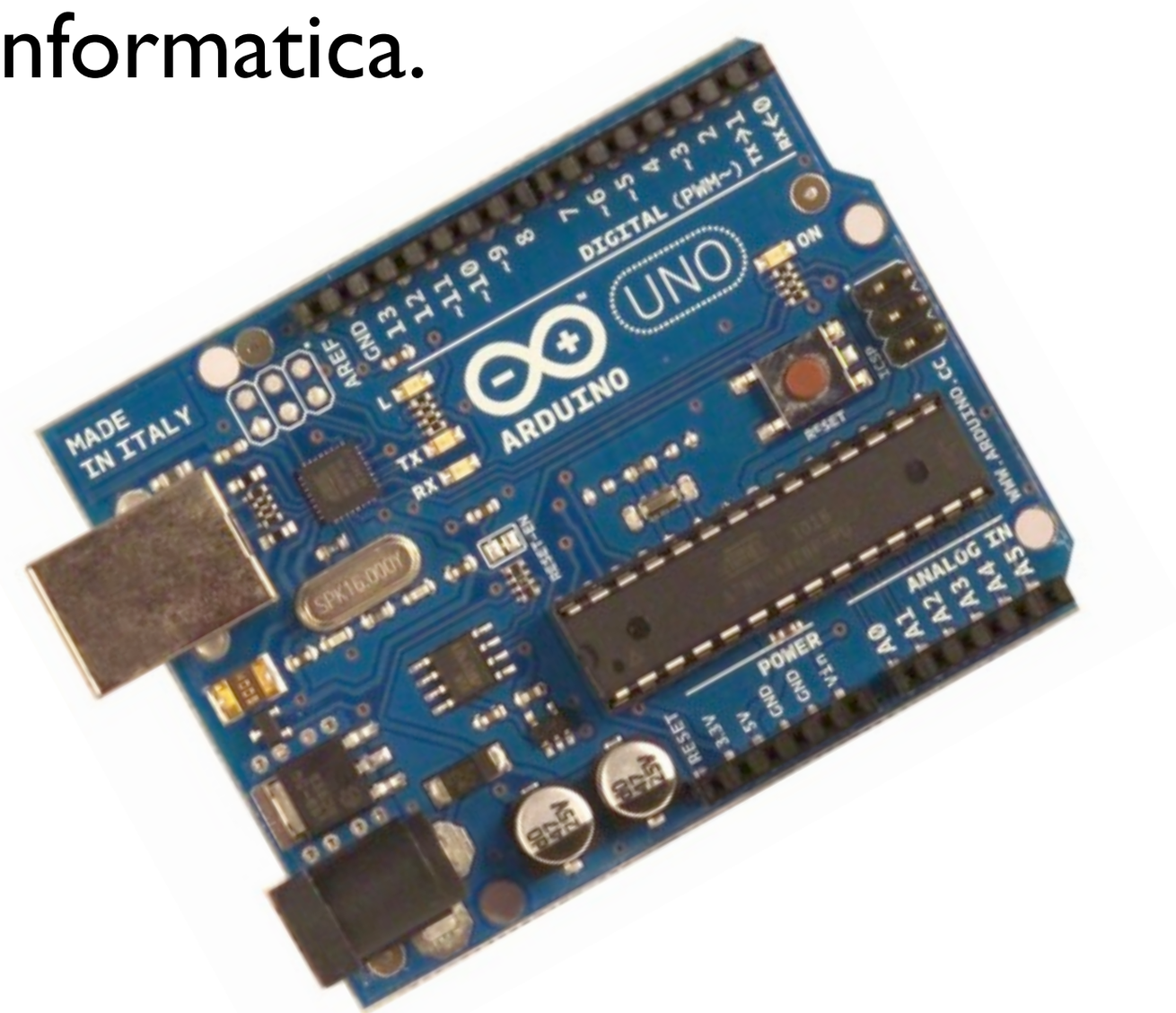
Arduino

Scuola di Robotica

Arduino è una piattaforma multifunzione low cost che facilita l'apprendimento, la progettazione e la creazione di un progetto. Comprende sia la parte tecnica che informatica.

Caratteristiche tecniche:

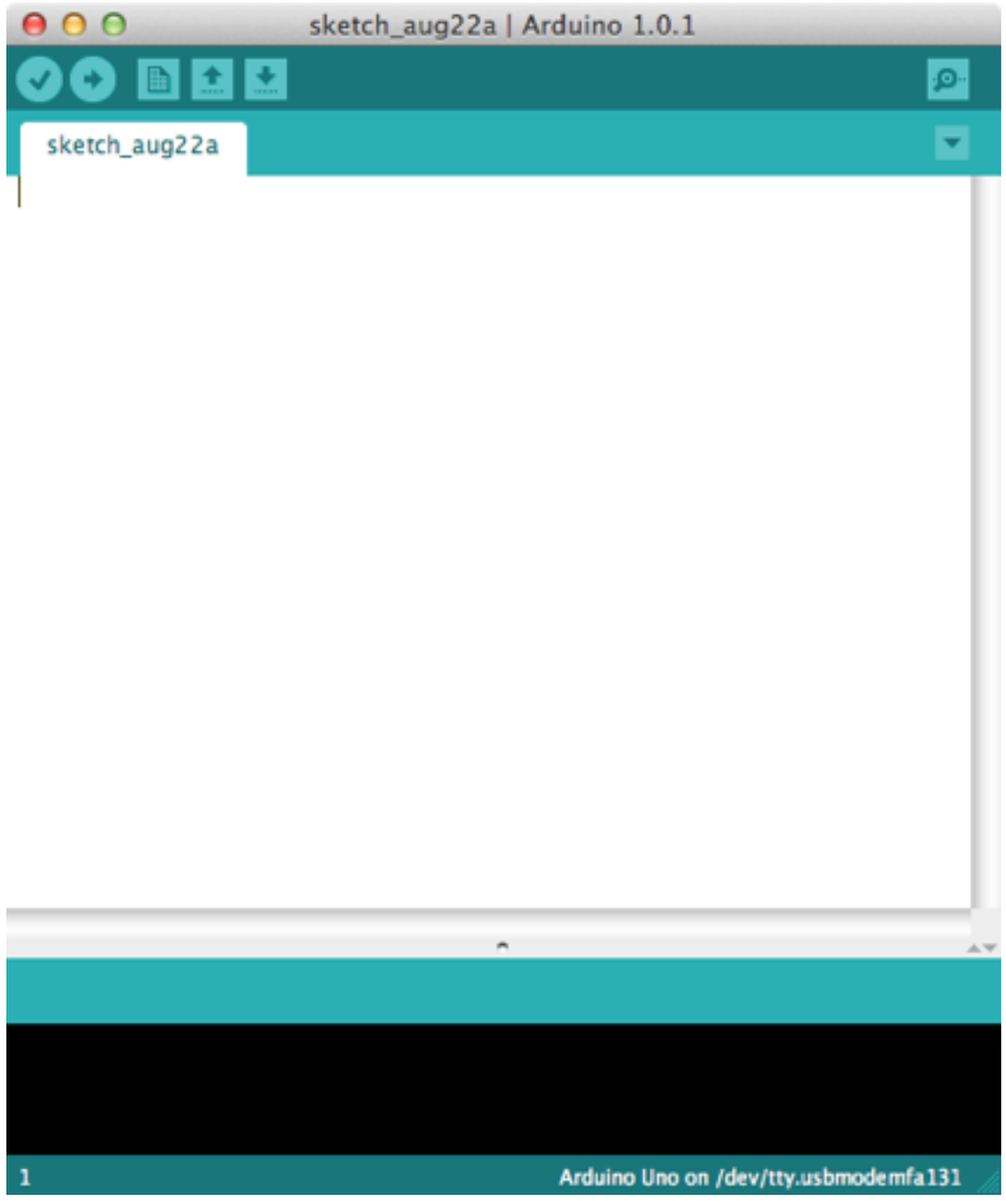
- AtMega 328
- Volt: 7-12v
- 14 I/O digitali (6PWM)
- M.Flash 32K
- Clock 16MHz
- Prezzo ~23€



Software

Scuola di Robotica

Arduino si programma con il software apposito open source scaricabile direttamente dal sito di arduino.



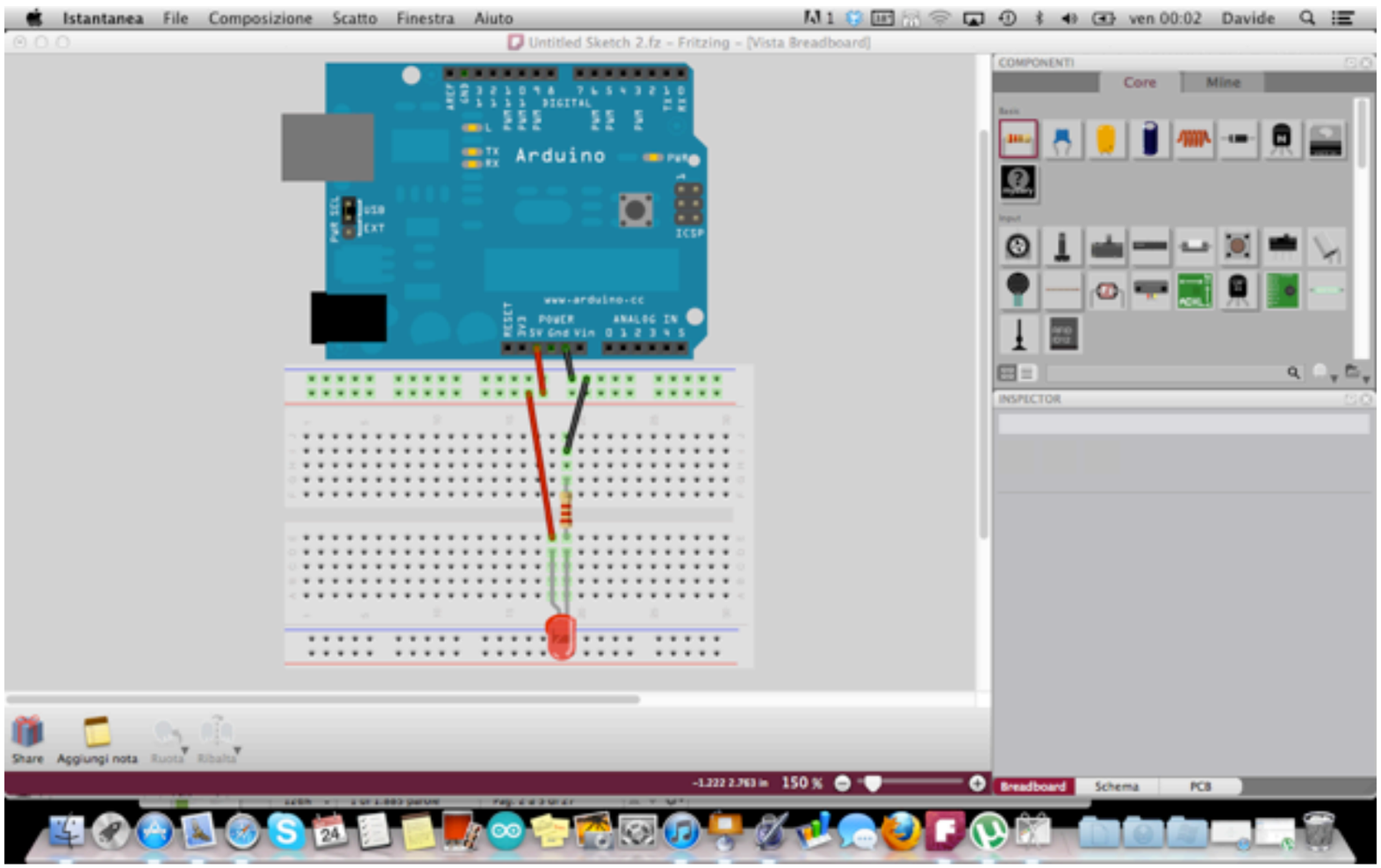
http://arduino.cc/en/Main/Software

Software utile

Per creare schemi e circuiti semplici da capire si può usare Fritzing.

Tutti gli schemi del manuale sono fatti con questo software open source.

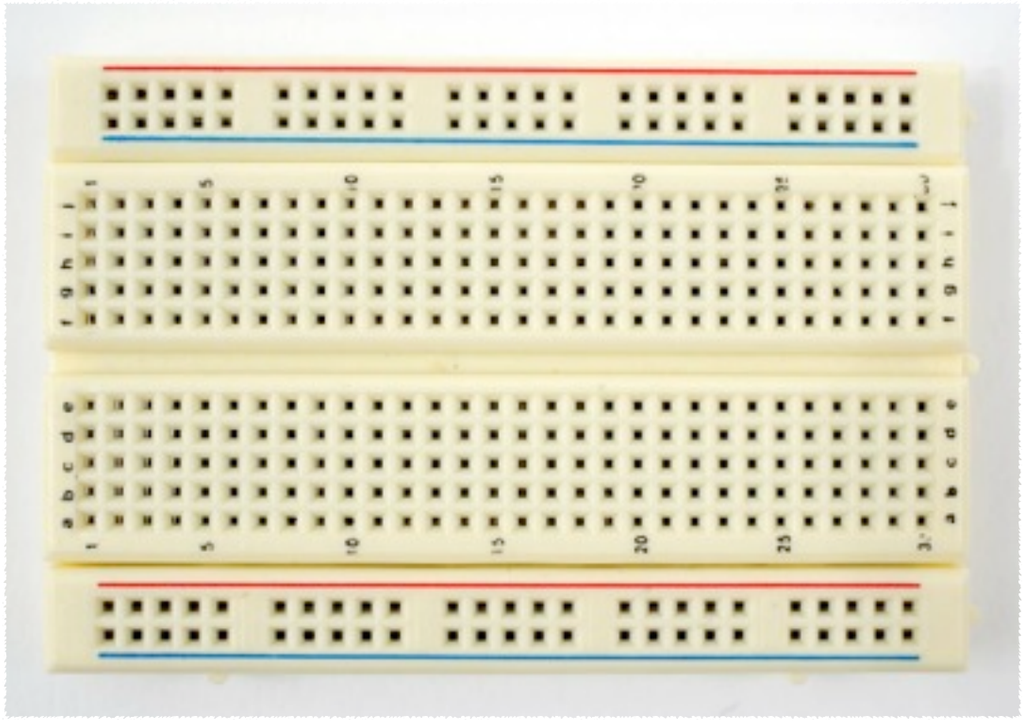
http://fritzing.org

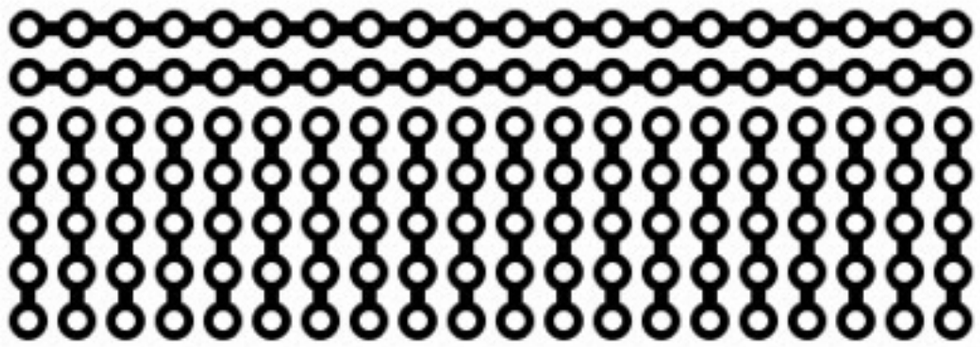


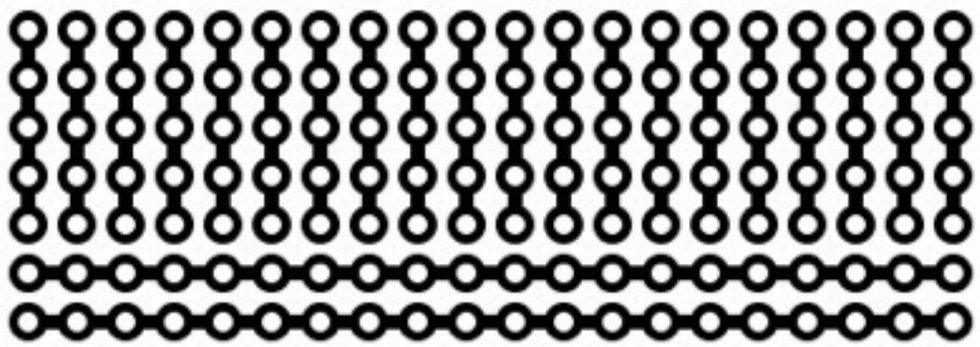
Scuola di Robotica

Le basi

Scuola di Robotica





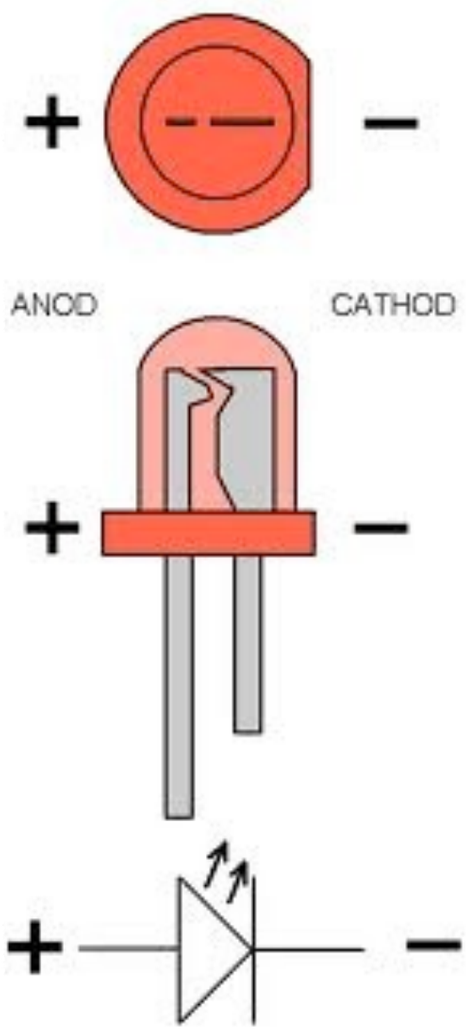


La breadbord

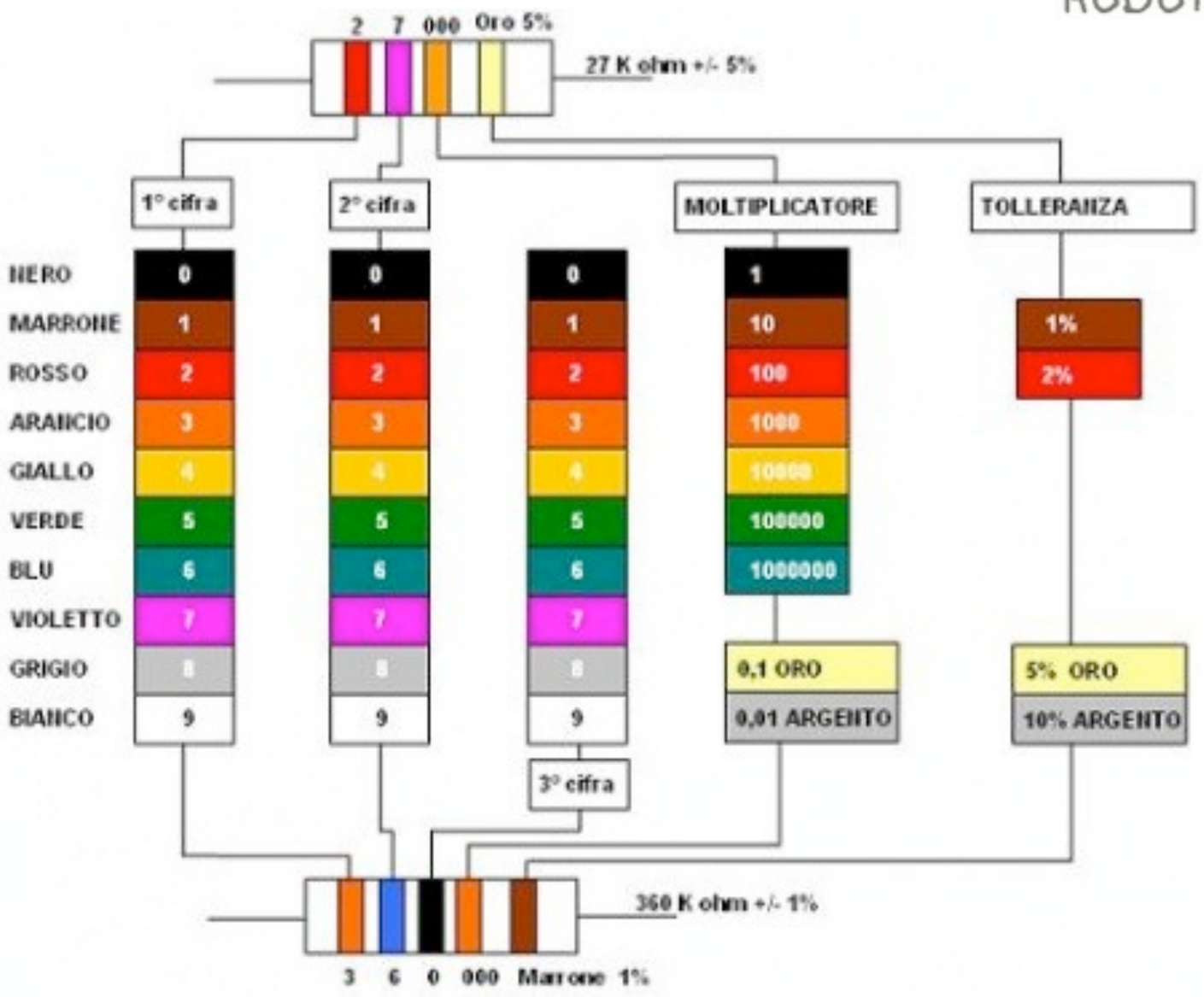
La breadbord è una scheda che ci permette di creare circuiti elettrici “volanti”, senza saldare possiamo provare un qualsiasi circuito.

Le basi

Scuola di Robotica



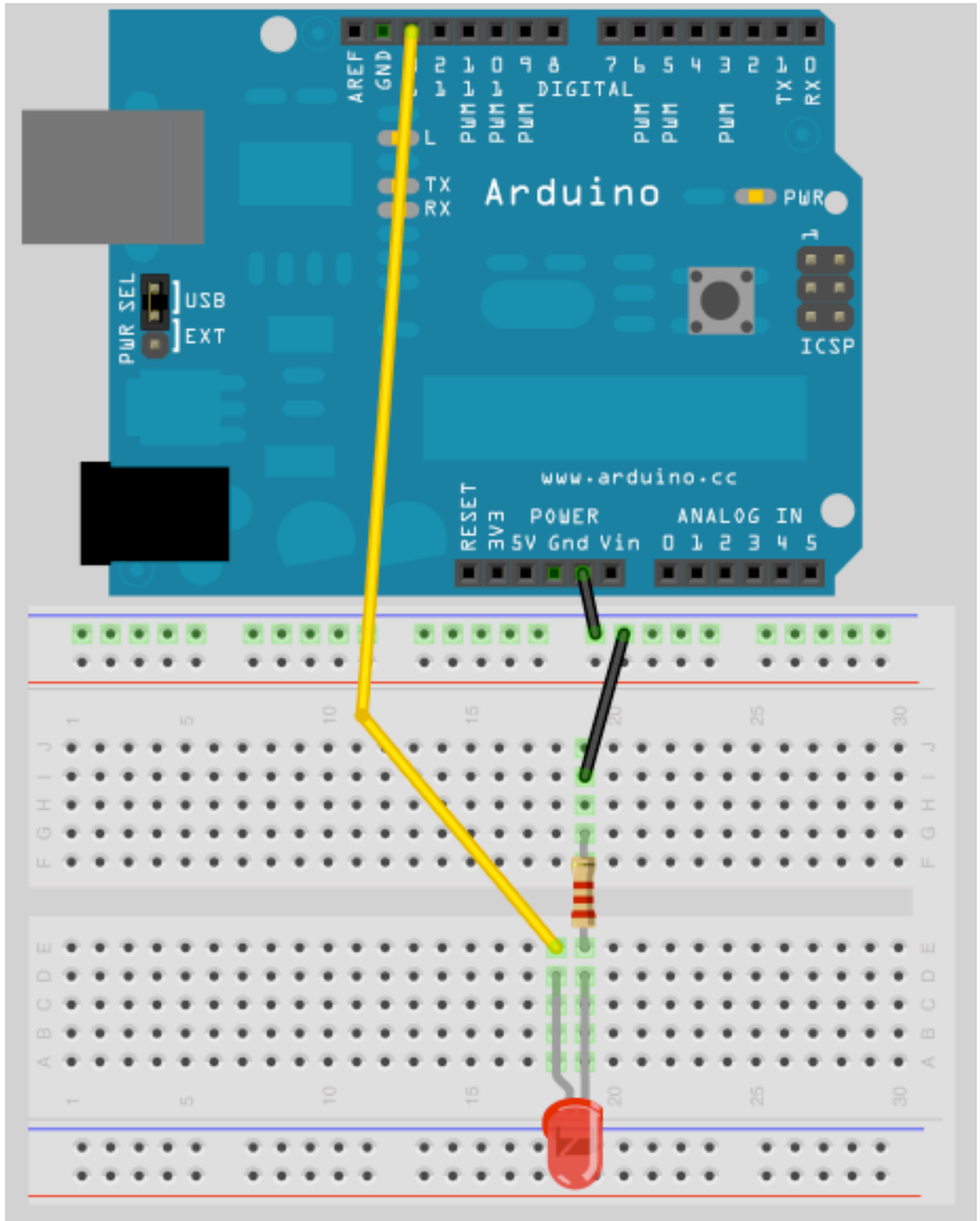
Led



Resistenze

I° esercizio

Scuola di Robotica



Es 1

```
void setup()
{
  pinMode(13, OUTPUT);
}
void loop()
{
  digitalWrite(13, HIGH);   // set the LED on
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);    // set the LED off
  delay(1000);              // wait for a second
}
```

I° esercizio

Scuola di Robotica

Es 1

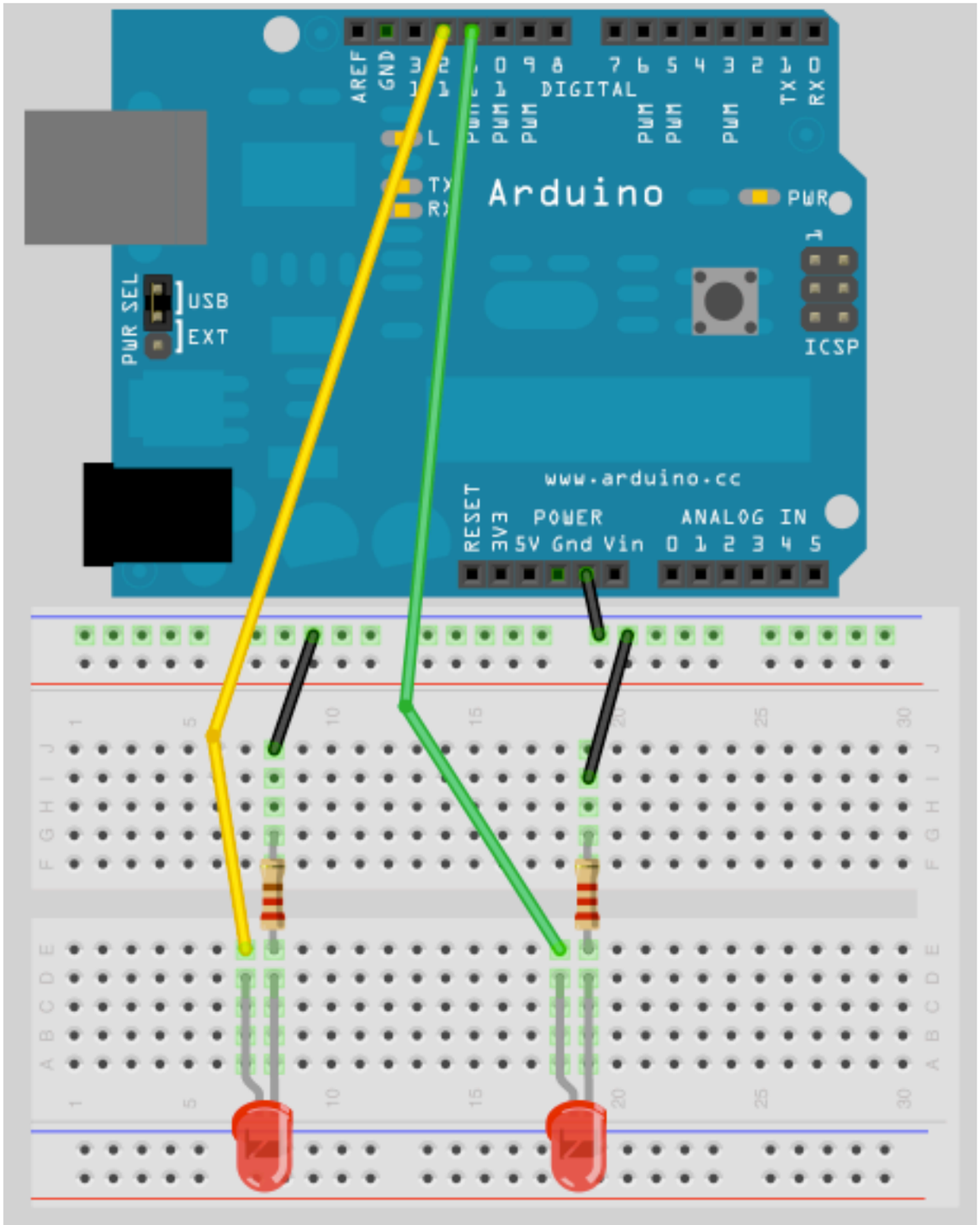
```
void setup()
{
  pinMode(13, OUTPUT);
}
void loop()
{
  digitalWrite(13, HIGH);   // set the LED on
  delay(1000);              // wait for a second
  digitalWrite(13, LOW);    // set the LED off
  delay(1000);              // wait for a second
}
```

Es 1.1

```
#define   LED   13
void setup()
{
  pinMode(LED, OUTPUT);
}
void loop()
{
  digitalWrite(LED, HIGH);   // set the LED on
  delay(1000);               // wait for a second
  digitalWrite(LED, LOW);    // set the LED off
  delay(1000);               // wait for a second
}
```

2° esercizio

Scuola di Robotica

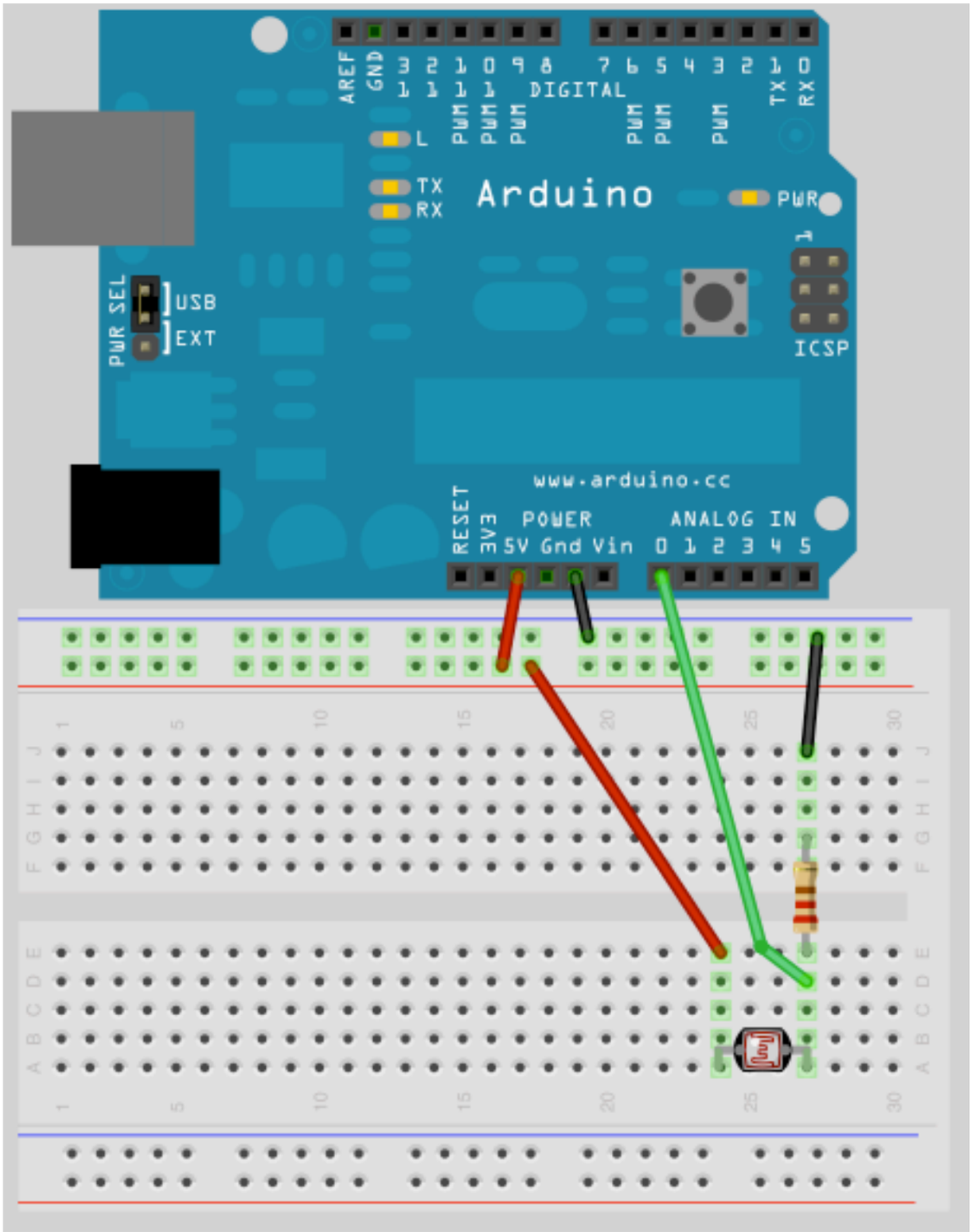


Es 2

```
void setup()
{
   //imposto le uscite
  pinMode(12, OUTPUT);
  pinMode(11, OUTPUT);
}
void loop()
{
  digitalWrite(12, HIGH);   // set the LED1 on
  digitalWrite(11, LOW);    // set the LED2 off
  delay(500);               // aspetto mezzo secondo
  digitalWrite(12, LOW);    // set the LED1 off
  digitalWrite(11, HIGH);   // set the LED2 on
  delay(500);               // aspetto mezzo secondo
}
```

3° esercizio

Scuola di Robotica



Es 3

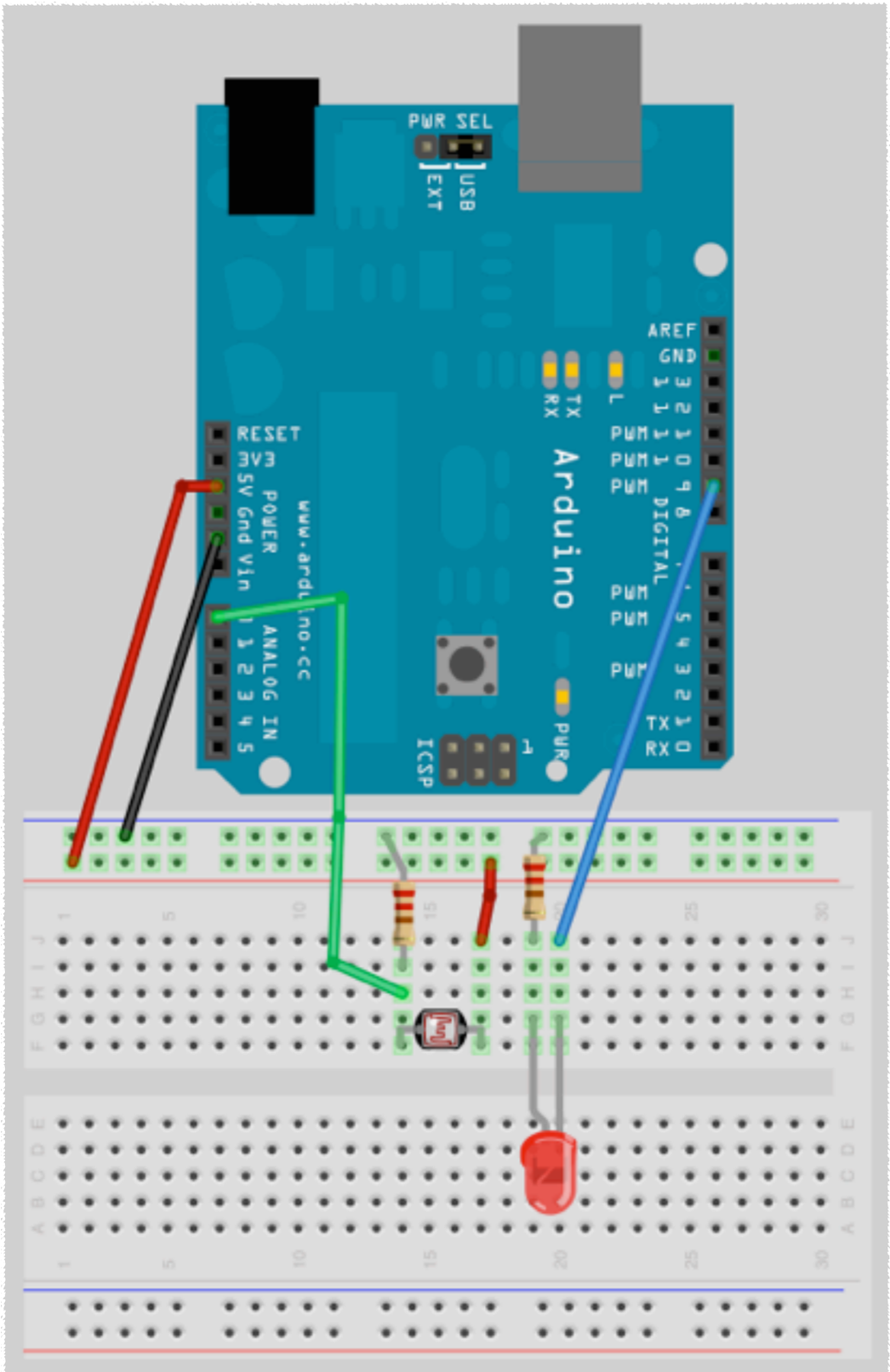
Uso della seriale.

Rappresentare il valore del sensore letto in una finestra di testo seriale.

```
void setup()
{
  Serial.begin(9600);
}

void loop()
{
  int sensorValue = analogRead(A0);
  Serial.println(sensorValue, DEC);
}
```

4° esercizio

Scuola di Robotica



Es 4

Al variare della luce varia il tempo di lampeggio del led.

```
#define LED 9
int Pin_sensore = A0;
int sensore = 0;
void setup()
{
 pinMode(LED, OUTPUT);
}
void loop()
{
  sensore = analogRead(Pin_sensore);
  digitalWrite(LED, HIGH);
  delay(sensore);
  digitalWrite(LED, LOW);
  delay(sensore);
}
```

5° esercizio

Scuola di Robotica

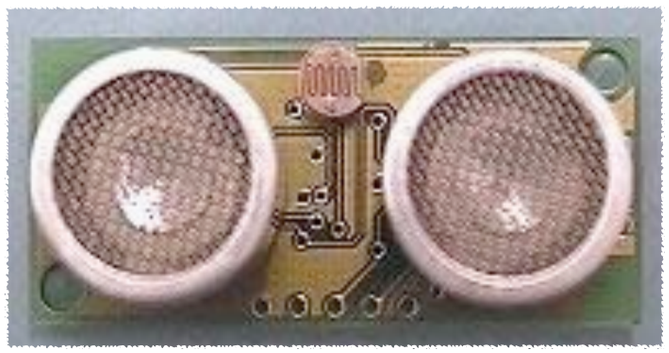
Gli ultrasuoni

Devantech

SRF02 I2C

Tensione Operativa	5V
Corrente Operativa Tipica	4mA
Frequenza	40 Khz
Portata	15cm - 6mt
Modalita' di funzionamento	Seriale o I2C
Dimensioni	24 x 20 x H 17 mm

SRF08



Tensione Operativa	5V
Corrente Operativa Tipica	15mA - 3mA Standby
Frequenza	40 Khz
Portata	3cm - 6mt
Max Analogue Gain	Variabile da 94 a 1025 in 32 steps
Unita'	Distanza in uS, mm o pollici
Comunicazione	Protocollo I2C
Dimensioni	43 x 20 x H 17 mm

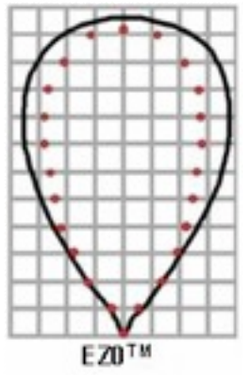
5° esercizio

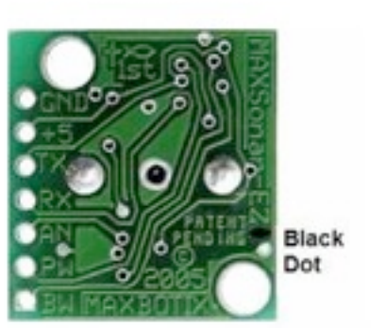
Scuola di Robotica

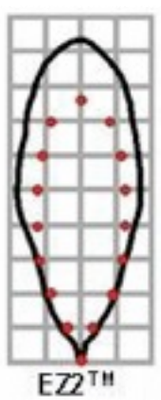
Gli ultrasuoni

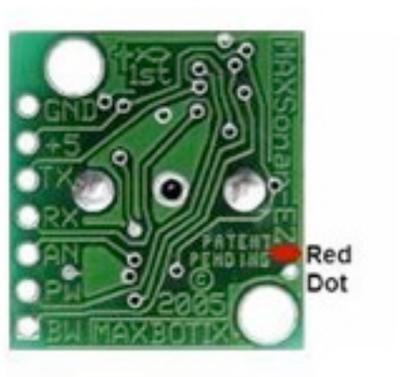
Maxbotix xxx

Uso terrestre (da 25 150€)

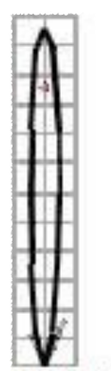








Uso marino (>100€)



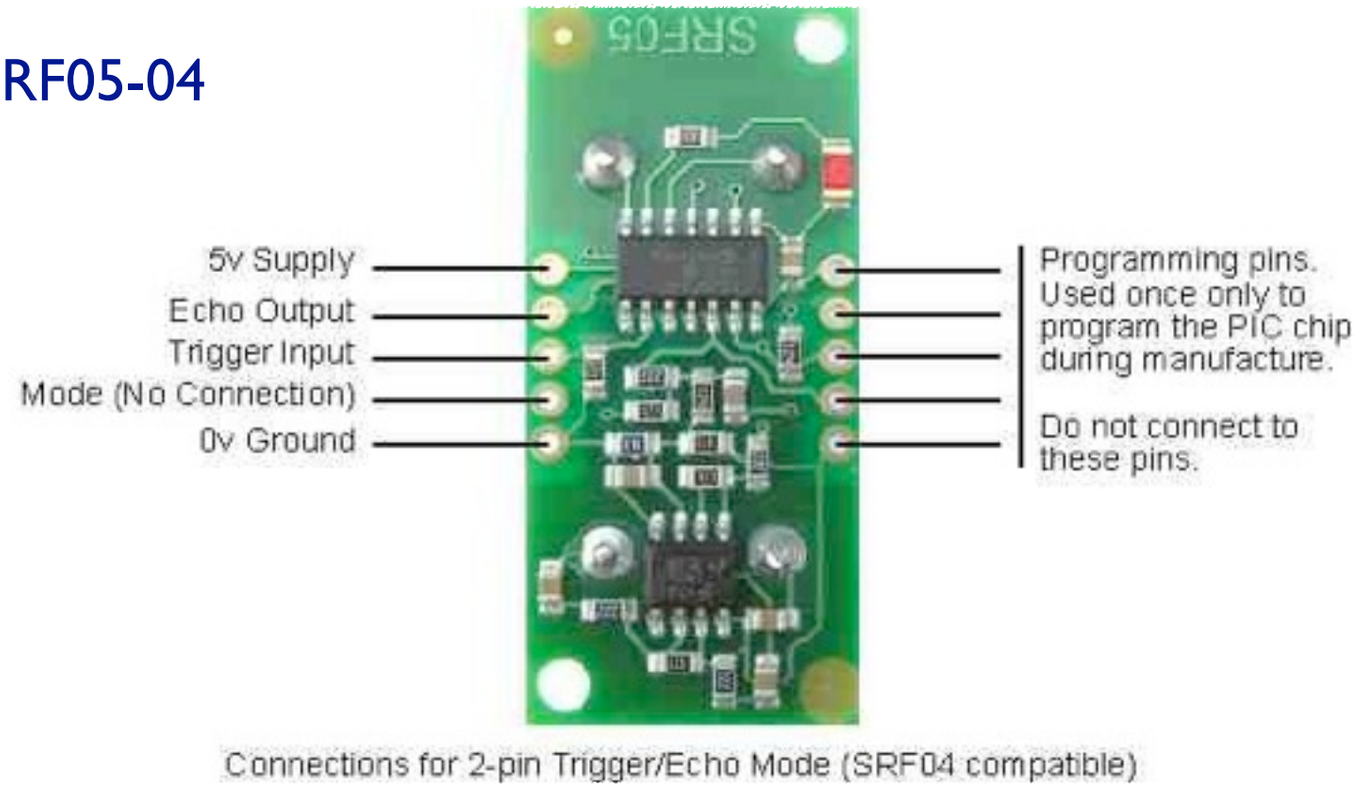
5° esercizio

Scuola di Robotica

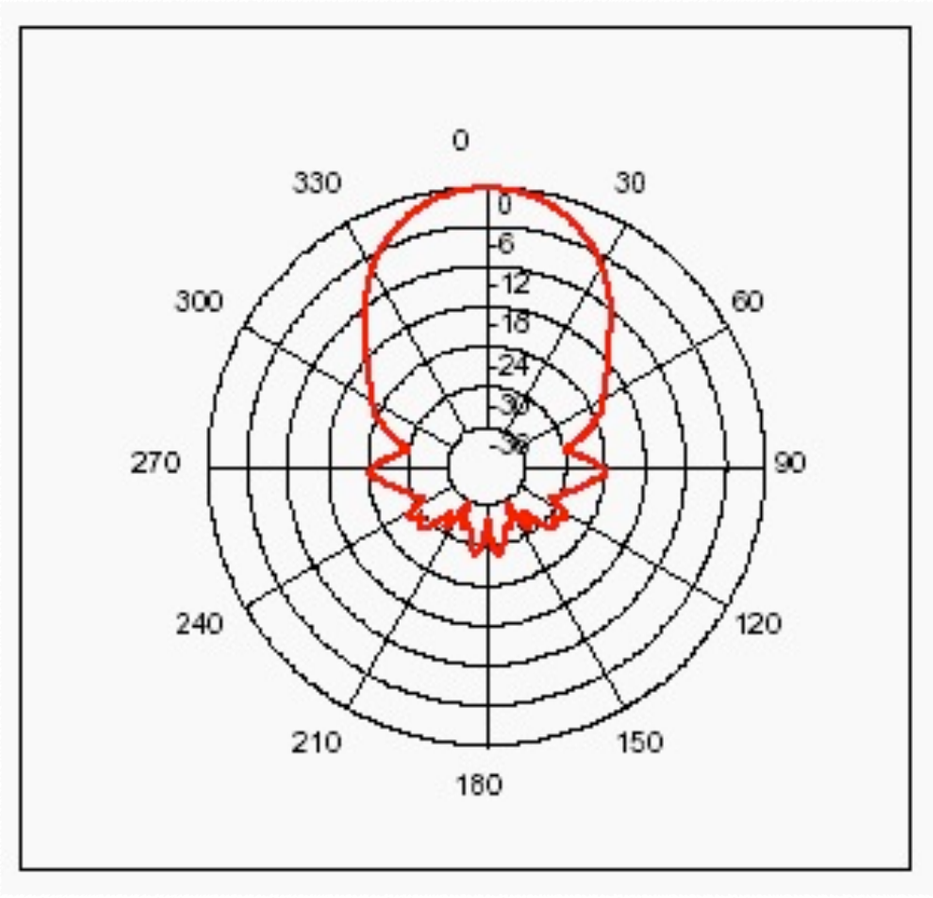
Gli ultrasuoni

Devantech

SRF05-04

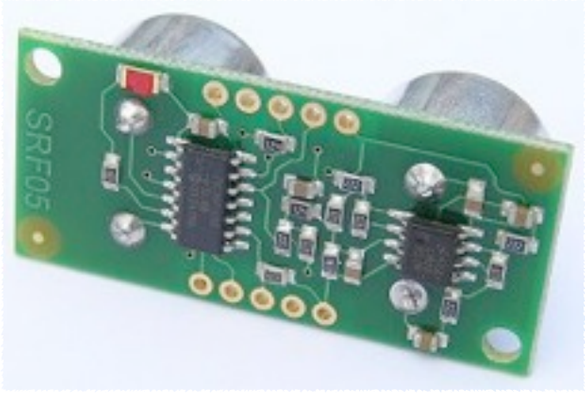


Connections for 2-pin Trigger/Echo Mode (SRF04 compatible)



Caratteristiche Tecniche:

Tensione Operativa	5V
Corrente Operativa Tipica	4mA
Frequenza	40 Khz
Portata	1cm - 4mt
Impulso di ritorno	Segnale TTL positivo
Trigger di Input	Impulso TTL di durata minima di 10 uS.
Dimensioni	43 x 20 x H 17 mm



5° esercizio

Es 5

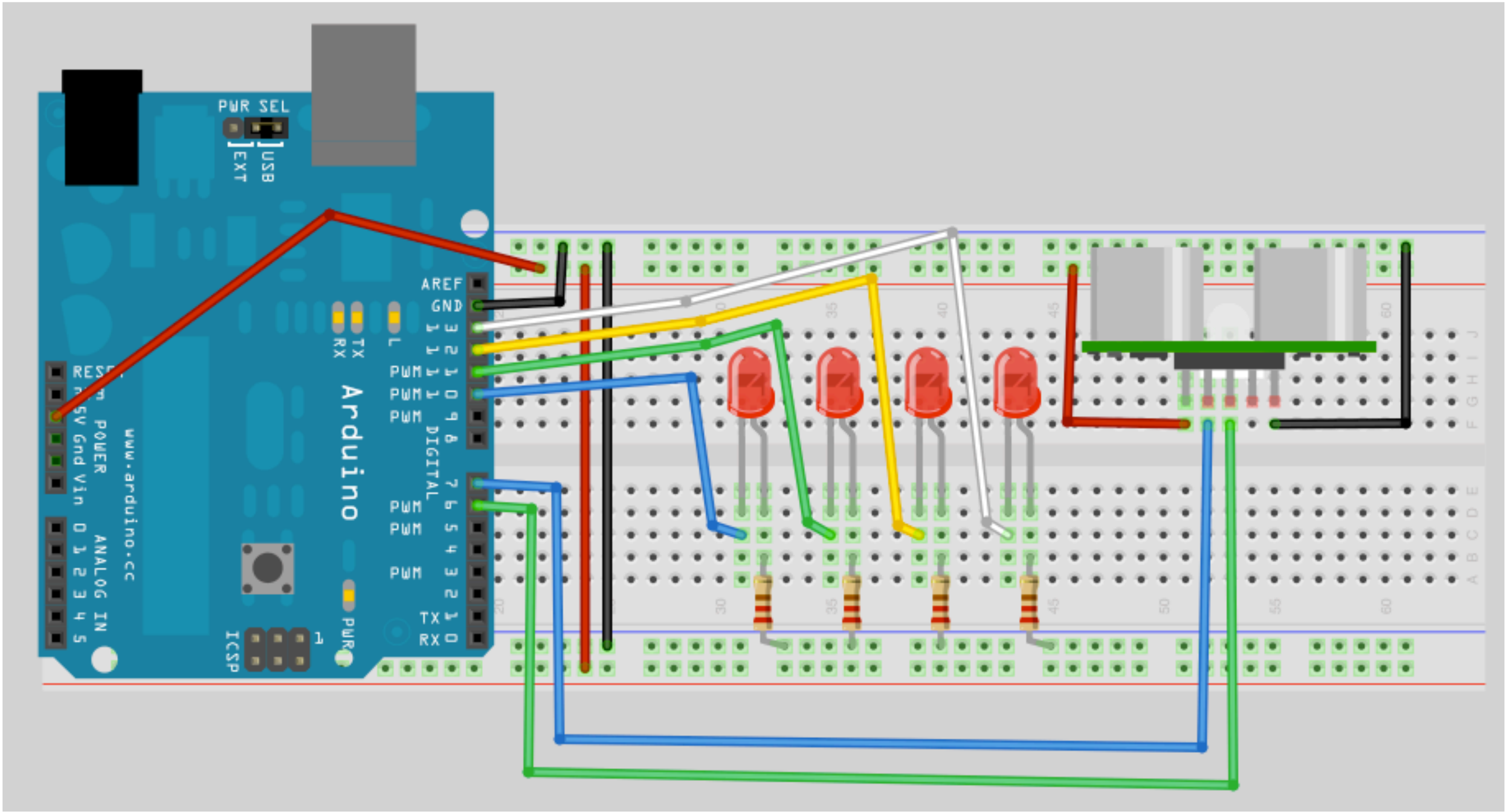
Leggere gli ultrasuoni

```
#define SONAR_TRIGGER_PIN   6
#define SONAR_ECHO_PIN      7
#define LED 13
// creiamo una subroutine per la gestione degli ultrasuoni
unsigned int measure_distance()
{
   digitalWrite(SONAR_TRIGGER_PIN, HIGH);
   delayMicroseconds(10);
   digitalWrite(SONAR_TRIGGER_PIN, LOW);
   unsigned long pulse_length = pulseIn(SONAR_ECHO_PIN, HIGH);

   //  pulseIn(,)   legge quando il pin va a livello richiesto
   //  ( nel nostro caso a livello alto) e ci
   //  restituisce il tempo in ms che rimane a quel livello

   delay(50);
   return( (unsigned int) (pulse_length / 58) );
   //  uS/58=cm or uS/148=inches
}
```

```
void setup ()
{
  Serial.begin(9600);
  pinMode(SONAR_TRIGGER_PIN, OUTPUT);
  pinMode(SONAR_ECHO_PIN, INPUT);
  pinMode(LED,OUTPUT);
}
void loop()
{
  unsigned int ostacolo = measure_distance();
  Serial.println(ostacolo, DEC);

  if(ostacolo < 13) //sotto i 13cm si accende il led
     {
     digitalWrite(LED, HIGH);
     }
     else
     {
     digitalWrite(LED,LOW);
     }
}
```

6° esercizio

Scuola di Robotica



6° esercizio

es 6

```
#define SONAR_TRIGGER_PIN   6
#define SONAR_ECHO_PIN      7
#define LED1 13
#define LED2 12
#define LED3 11
#define LED4 10

unsigned int measure_distance()
{
   digitalWrite(SONAR_TRIGGER_PIN, HIGH);
   delayMicroseconds(10);
   digitalWrite(SONAR_TRIGGER_PIN, LOW);
   unsigned long pulse_length = pulseIn(SONAR_ECHO_PIN, HIGH);
   delay(50);
   return( (unsigned int) (pulse_length / 58) );
//uS/58=cm or uS/148=inches
}

void setup ()
{
  Serial.begin(9600);
  pinMode(SONAR_TRIGGER_PIN, OUTPUT);
  pinMode(SONAR_ECHO_PIN, INPUT);
  pinMode(LED1,OUTPUT);
  pinMode(LED2,OUTPUT);
  pinMode(LED3,OUTPUT);
  pinMode(LED4,OUTPUT);
}
```

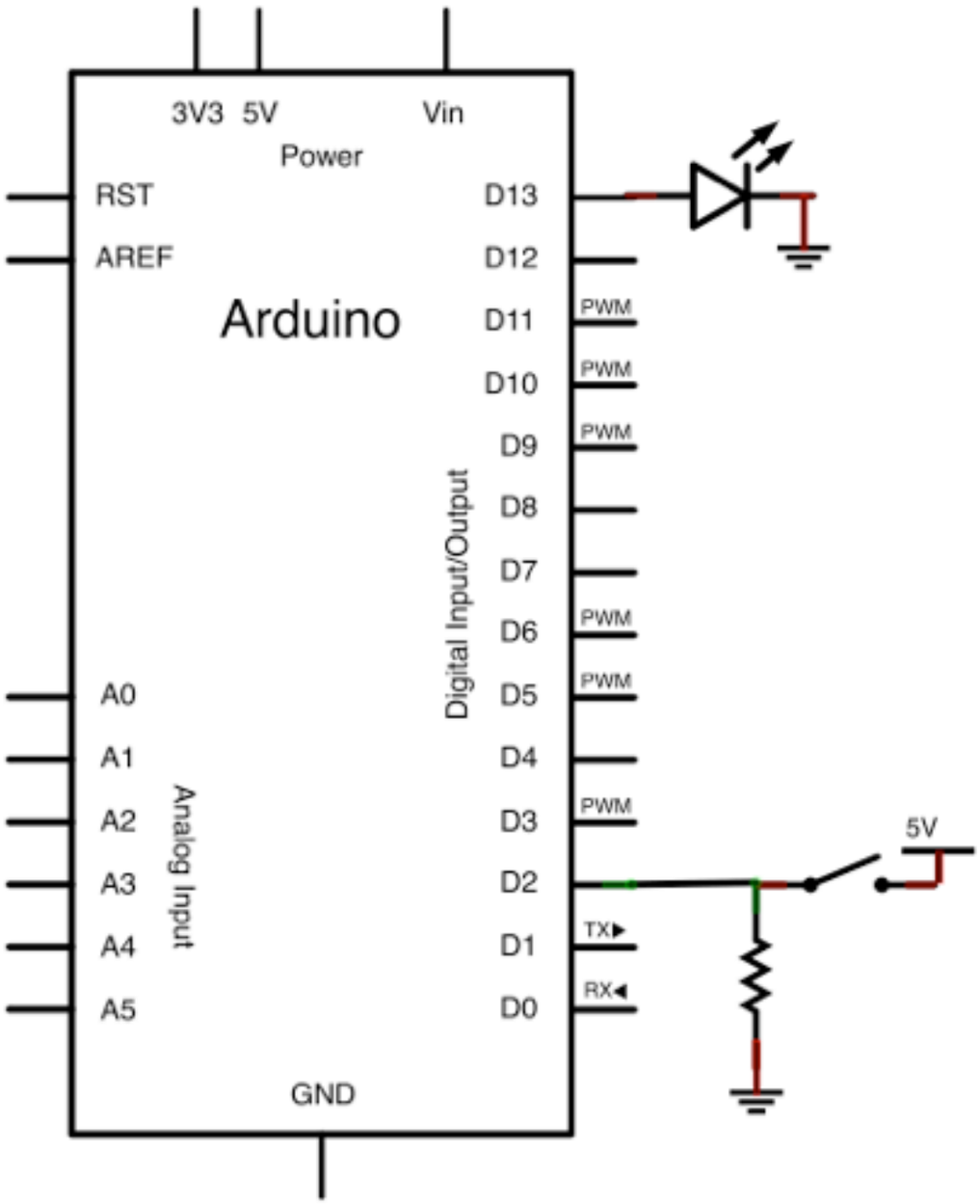
```
void loop()
{
  unsigned int ostacolo = measure_distance();
  Serial.println(ostacolo, DEC);

  if(ostacolo > 10)
       {  digitalWrite(LED1, HIGH);  }
     else
       {  digitalWrite(LED1,LOW);   }

  if(ostacolo > 20)
       {  digitalWrite(LED2,HIGH);  }
     else
       {  digitalWrite(LED2,LOW);   }

   if(ostacolo > 30)
       {  digitalWrite(LED3, HIGH);  }
     else
       {  digitalWrite(LED3,LOW);   }

  if(ostacolo > 40)
       {  digitalWrite(LED4,HIGH);  }
     else
       {  digitalWrite(LED4,LOW);   }
}
```

7° esercizio

Scuola di Robotica

Es 7

Sensori di tatto

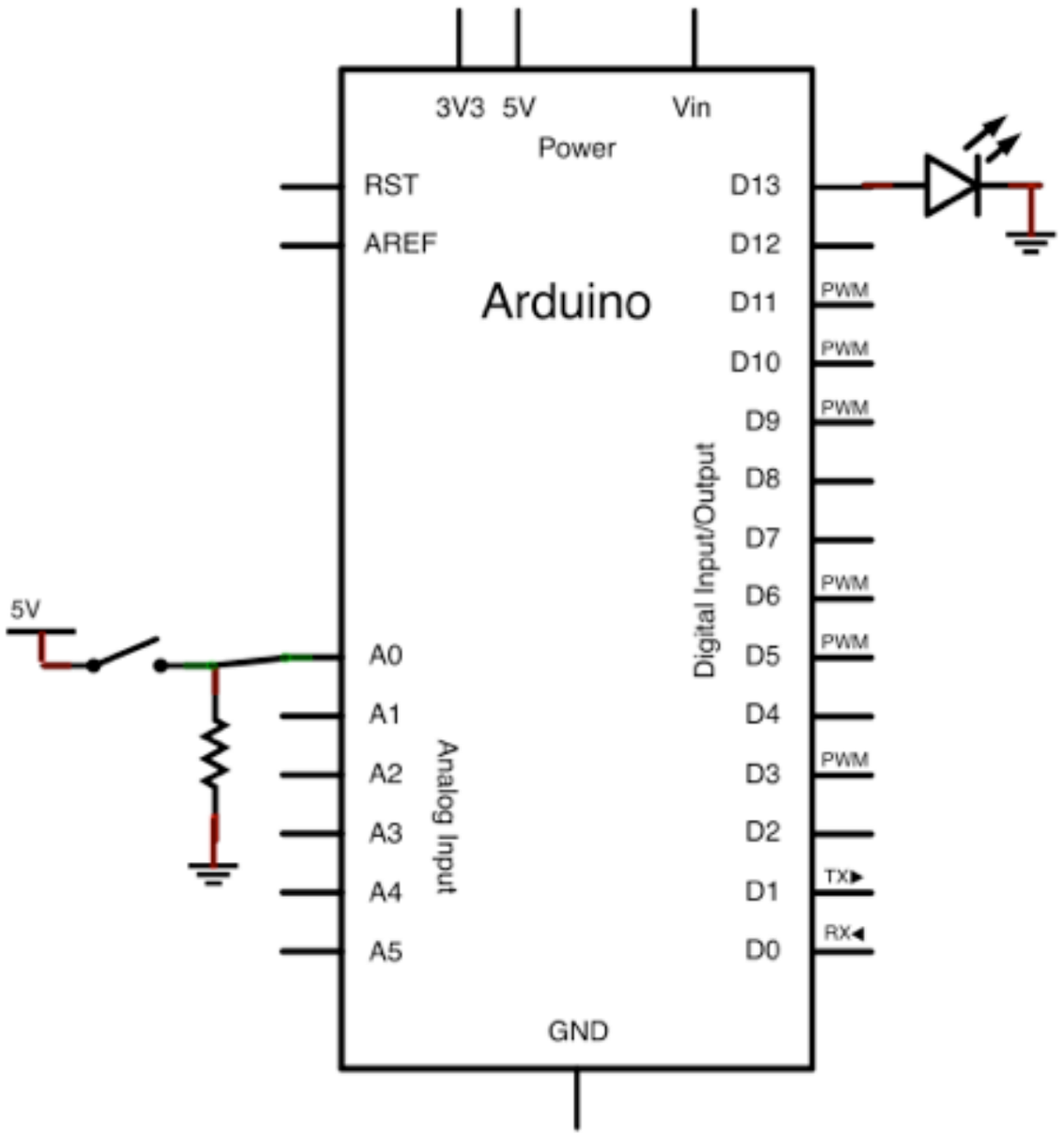


Es 7

```
void setup()
{
  pinMode(13, OUTPUT);
  pinMode(2,INPUT);
}
void loop()
{
  int sensorValue = digitalRead(2);
  digitalWrite(13, sensorValue);
}
```

7° esercizio

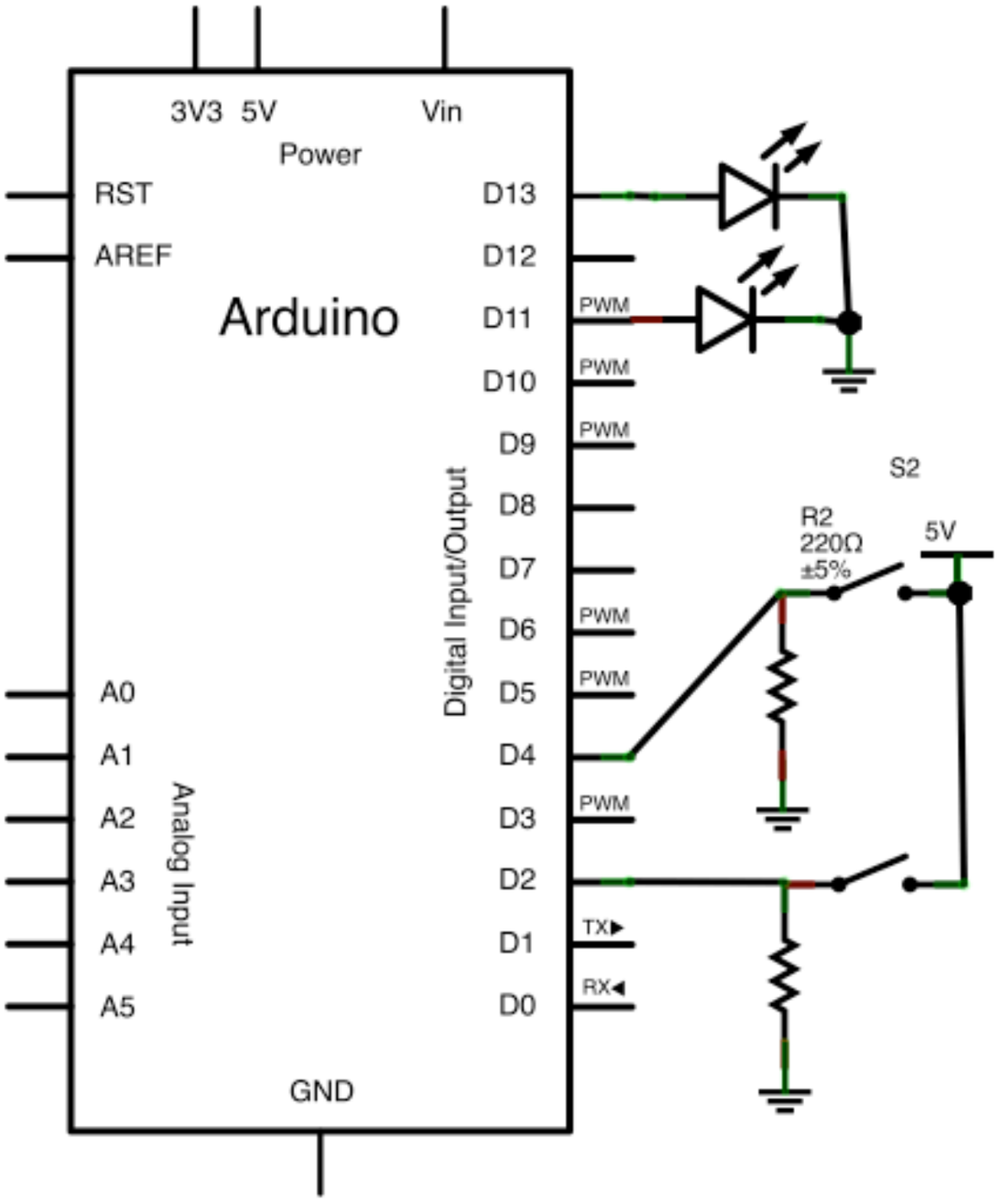
Scuola di Robotica



Es 7.1

```
void setup()
{
  pinMode(13, OUTPUT);
}

void loop()
{
  int sensorValue = analogRead(A0);
  digitalWrite(13, sensorValue);
}
```

8° esercizio



Es 8

```
#define SensD 2
#define SensS 4
#define LedD 13
#define LedS 11
void setup()
{
  pinMode(LedD, OUTPUT);
  pinMode(LedS, OUTPUT);
  pinMode(SensD,INPUT);
  pinMode(SensS,INPUT);
}
void loop()
{
  int ValSensD = digitalRead(SensD);
  int ValSensS = digitalRead(SensS);

  digitalWrite(LedD,ValSensD);
  digitalWrite(LedS,ValSensS);
}
```

8.1° esercizio

Es 8.1

```
#define SensD 2
#define SensS 4

#define LedD 13
#define LedS 11

void setup()
{
  pinMode(13, OUTPUT);
  pinMode(11, OUTPUT);

  pinMode(2,INPUT);
  pinMode(4,INPUT);

 Serial.begin(9600);
}
```

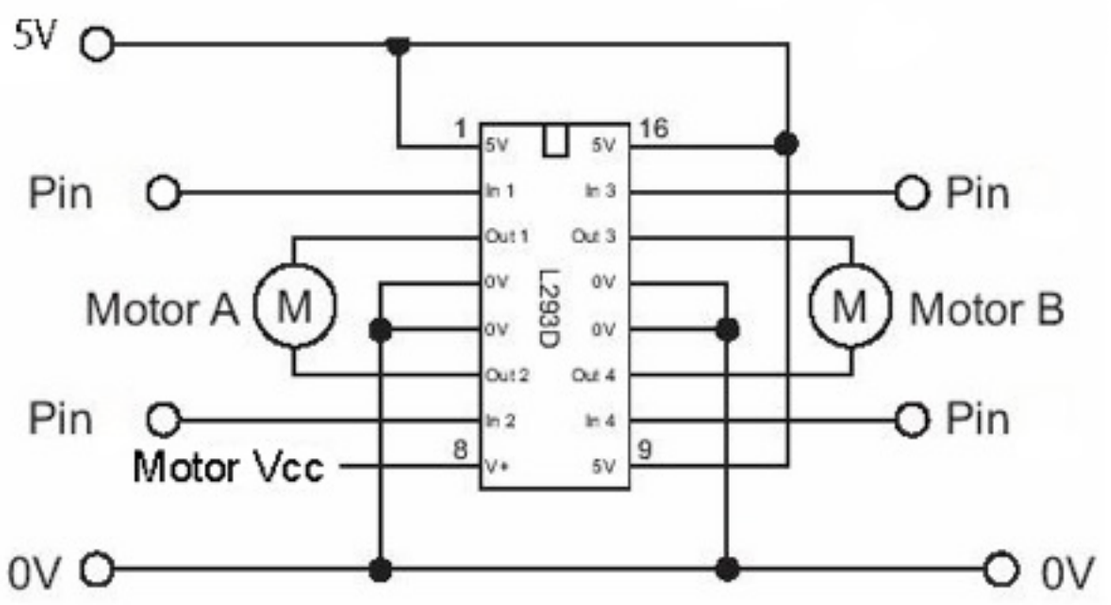
```
void loop()
{
  int ValSensD = digitalRead(SensD);
  int ValSensS = digitalRead(SensS);

  digitalWrite(13,ValSensD);
  digitalWrite(11,ValSensS);

  Serial.print(ValSensS, DEC);
  Serial.print("--");
  Serial.println(ValSensD, DEC);
}
```

9° esercizio

Scuola di Robotica

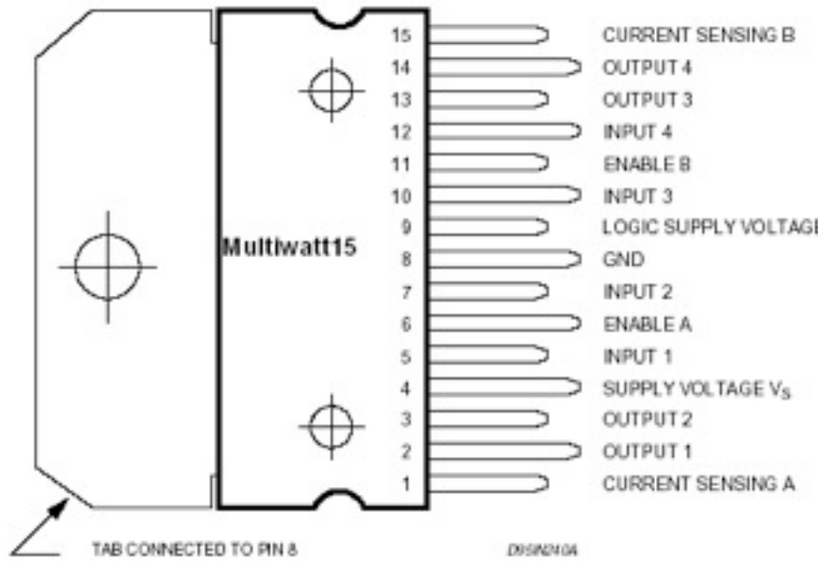
L293

Puo' pilotare indipendentemente una coppia di motori CC da 0.6 A (max) con voltaggio da 4.5 a 36 volts.

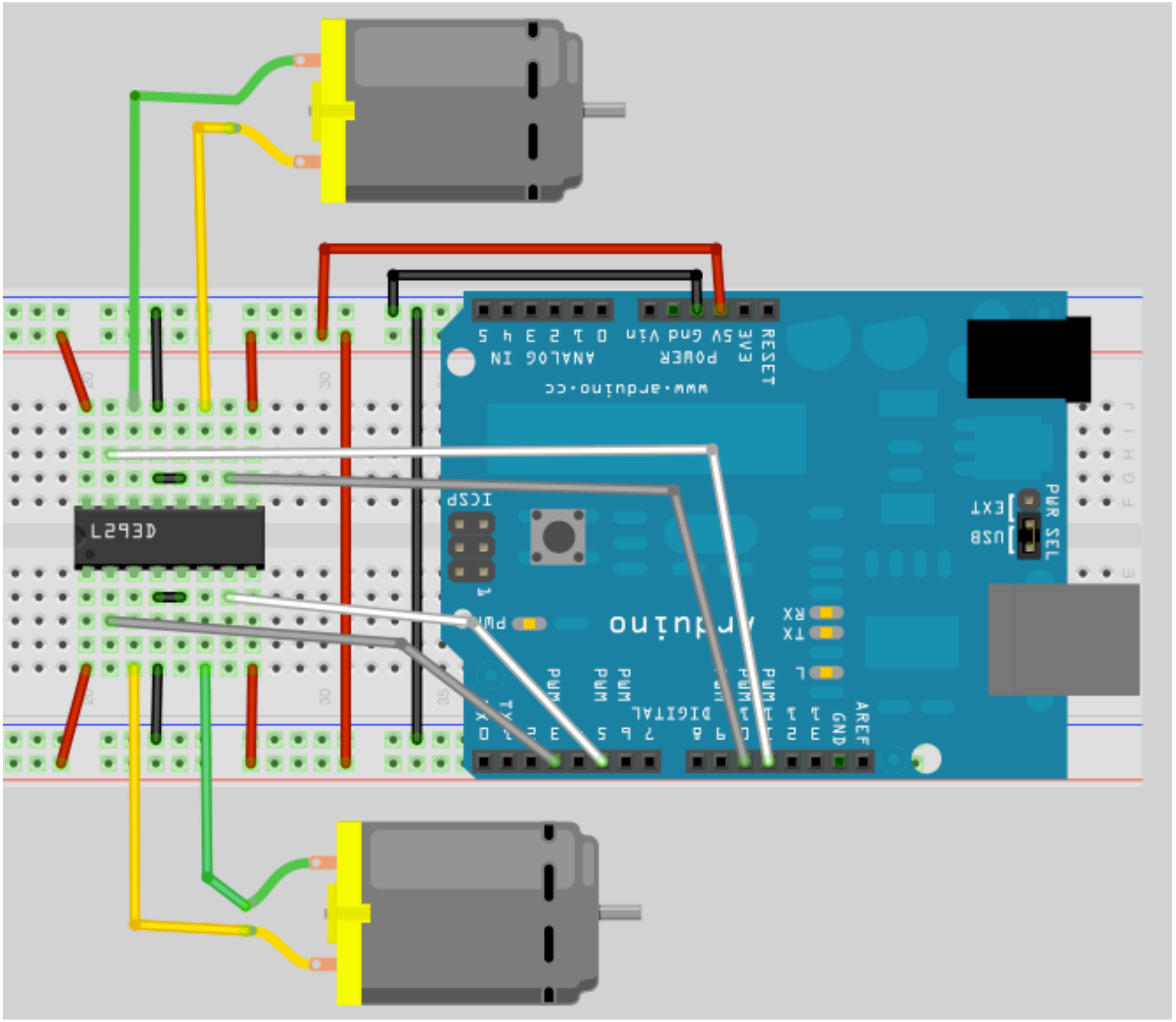


L298

Pilota 2 motori supportando una corrente di picco di 6A e un carico continuo di 2.5 Ampere ogni stadio. Accetta tensioni di carico fino a 50V.



9° esercizio


L293D
ICSP
PWR SEL
EXT
USB
RESET
POWER
ANALOG IN
www.arduino.cc
DIGITAL
AREF
GND

9° esercizio

Es 9

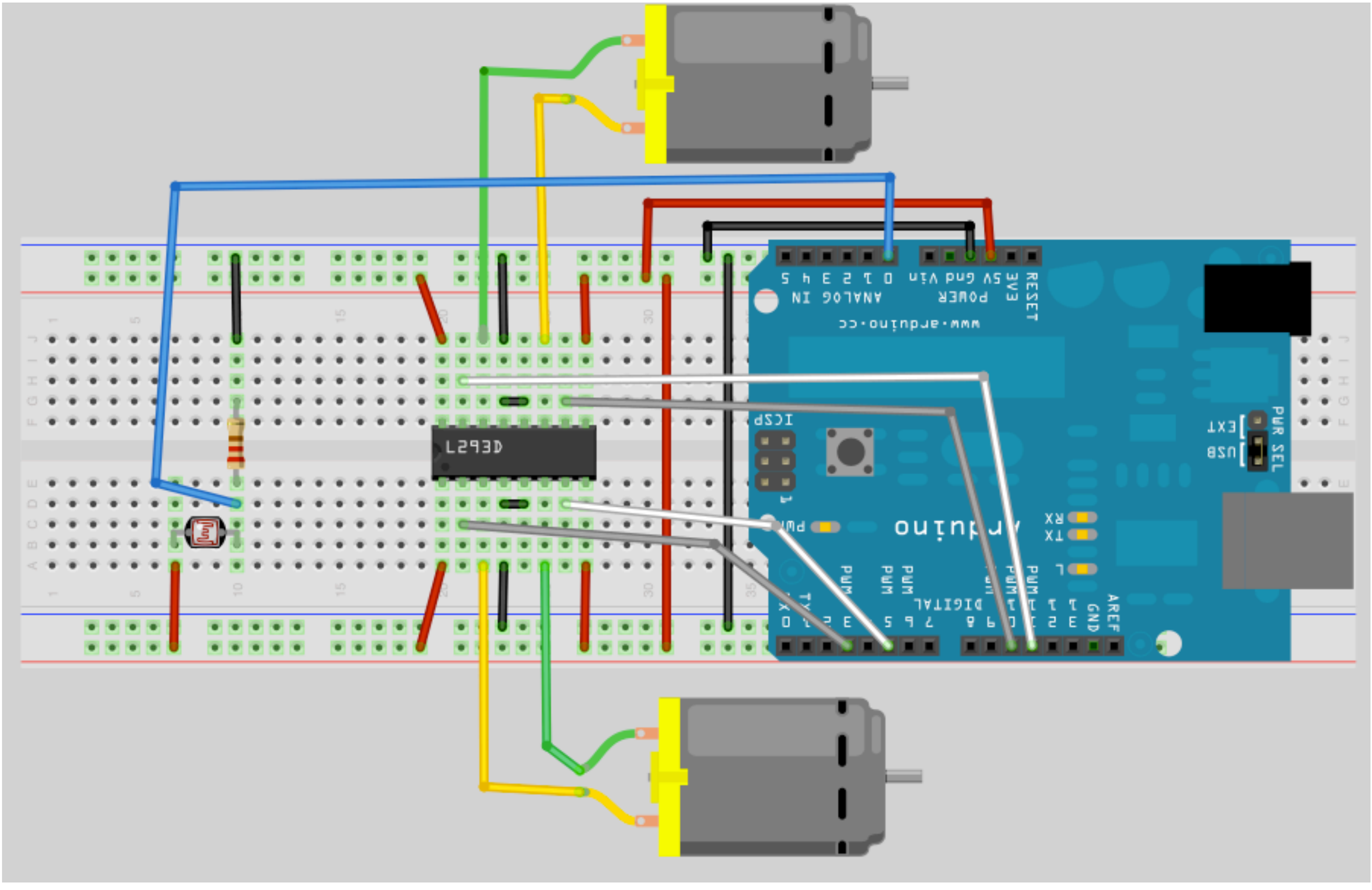
Far avanzare il robot :

```
#define M1r     3      //motore destro
#define M1b     5
#define M2r     10      //motore sinistro
#define M2b     11

void setup()
{
  pinMode(M1r,OUTPUT);
  pinMode(M1b,OUTPUT);
  pinMode(M2r,OUTPUT);
  pinMode(M2b,OUTPUT);
}
```

```
void loop()
{
  digitalWrite(M1r,HIGH);  //faccio avanzare il robot
  digitalWrite(M1b,LOW);
  digitalWrite(M2r,HIGH);
  digitalWrite(M2b,LOW);
}
```

10° esercizio

Scuola di Robotica



10° esercizio

Es 10

Far avanzare il robot con velocità variabile a seconda dell'intensità luminosa.

Scuola di Robotica

```
#define M1r     3      //motore destro
#define M1b     5

#define M2r     10      //motore sinistro
#define M2b     11


void setup()
{
  Serial.begin(9600);
  pinMode(M1r,OUTPUT);
  pinMode(M1b,OUTPUT);
  pinMode(M2r,OUTPUT);
  pinMode(M2b,OUTPUT);
}
```

```
void loop()
{
  int Sens = analogRead(A0); //leggo il pin 0analogico
  Serial.println(Sens);      //scrivo in seriale il valore letto

  Sens = map(Sens, 0, 1023, 0, 255); /* Questa funzione permette di
                                adattare uniformemente il segnale
                                d'ingresso (Sens) a una scala di valori
                                da noi scelta*/

  analogWrite(M1r,Sens);  //faccio avanzare il robot
  digitalWrite(M1b,LOW);
  analogWrite(M2r,Sens);
  digitalWrite(M2b,LOW);
}
```

11° esercizio

Scuola di Robotica

Sharp Distance Sensor 2D120X (4-30cm)

Characteristic	Value
Operating Supply Voltage	4.5V to 5.5V
Minimum Measuring Distance	4cm
Maximum Measuring Distance	30cm
Average Supply Current - Typical	33mA
Response Time	38 ± 10ms
Weight	3.5g

Sharp Distance Sensor 2Y0A21 (10-80cm)

Characteristic	Value
Operating Supply Voltage	4.5V to 5.5V
Minimum Measuring Distance	10cm
Maximum Measuring Distance	80cm
Average Supply Current - Typical	30mA
Response Time	38 ± 10ms
Weight	3.5g

11° esercizio

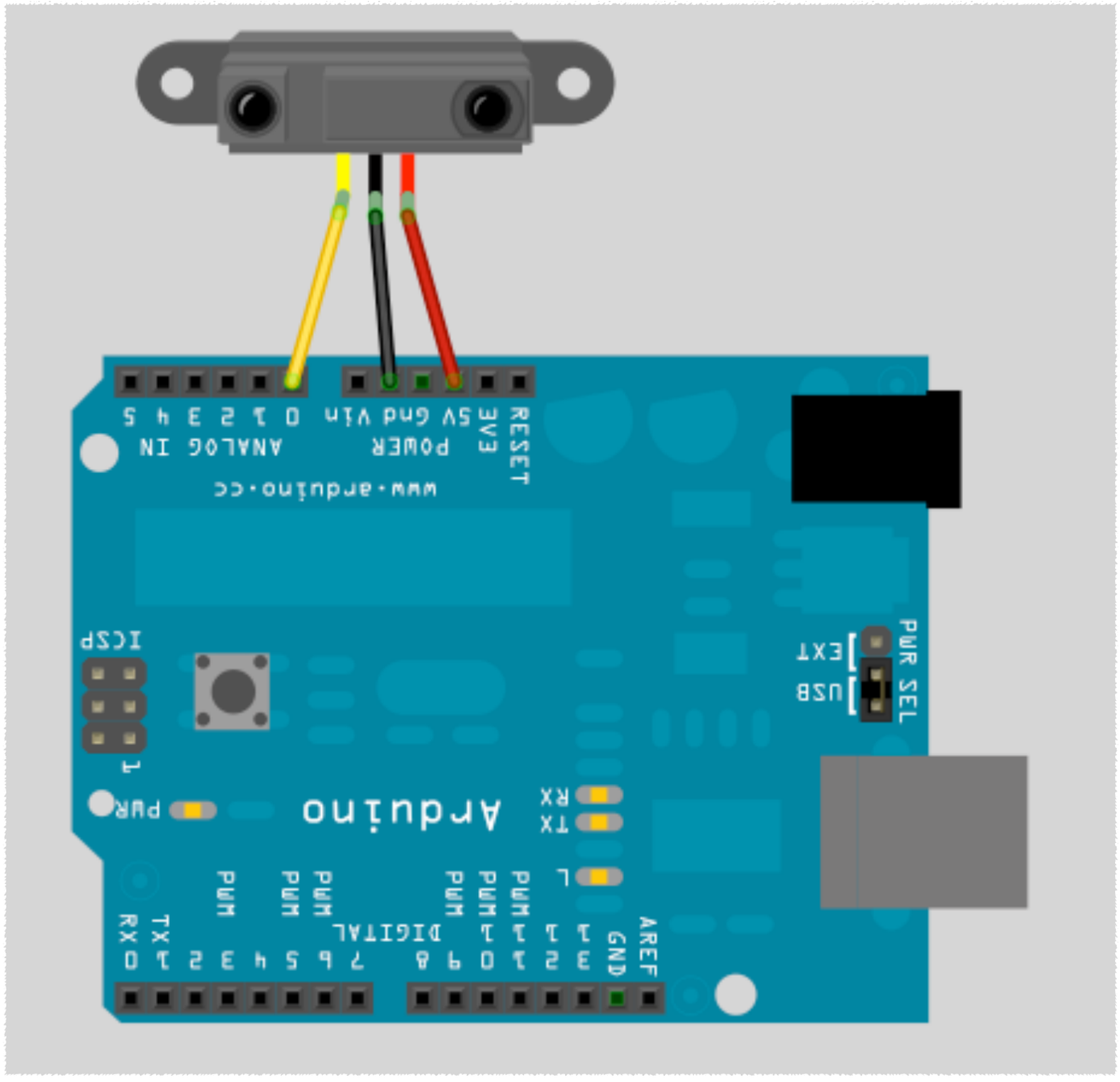
Scuola di Robotica

Sharp Distance Sensor 2Y0A02 (20-150cm)

Characteristic	Value
Operating Supply Voltage	4.5V to 5.5V
Minimum Measuring Distance	20cm
Maximum Measuring Distance	150cm
Average Supply Current - Typical	30mA
Response Time	38 ± 10ms
Weight	4.5g

11° esercizio

Scuola di Robotica



Per sapere la distanza letta basta fare:

Distance (cm) = 4800/(SensorValue - 20)

Es 11

Leggere il sensore di distanza

```
void setup()
{
  Serial.begin(9600);
}

void loop()
{
    double Valsens = analogRead(A0); //lettura sensore
    int Dist=4800/(Valsens-20);

    Serial.println(Dist, DEC);      //leggo seriale
}
```

12° esercizio

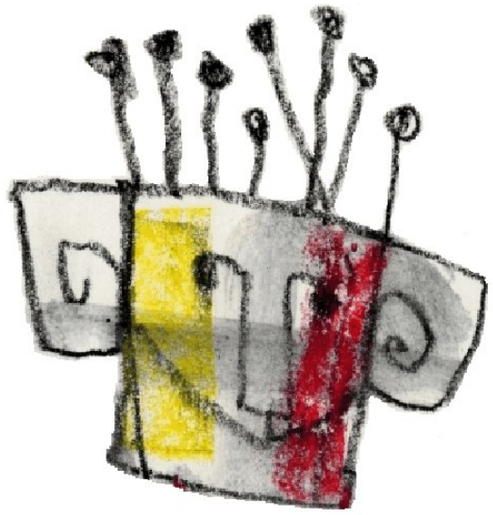
Scuola di Robotica

Facciamo un robot che vada sempre dritto evitando gli ostacoli con i sensori di tatto.

Materiale

Il materiale si può reperire su più siti on-line o 2 principali sono RobotItaly e Futurashop. I codici elencati sotto sono di RobotItaly, i codici e i rezzi sono aggiornati ma molto spesso cambiano, attraverso i nomi comunque la ricerca non dovrebbe essere difficile. Comprando dalla cina su ebay si riesce a risparmiare ma si allungano di molto i tempo di spedizione.

1x	500043	Arduino Duemilanove - USB ATMega328
1x	L293D	Motor Driver L293D
1x	POCHS	Telaio circolare in acrilico per mini Robot
1x	70097	Tamiya Twin Motor Gearbox
1x	70144	Tamiya Ball Caster
1x	70101	Tamiya Truck Tire Set
1x	SRF05	Sensore di distanza ad Ultrasuoni SRF05
6x	301011	Resistenza a strato metallico 2,21 KOhm 1/4W 1%
2x	MSW	Microswitch a levetta
1x	709088	Fotoresistenza
1x	390028	LED 5mm - ROSSO/VERDE - 5pz
1x	301012	Resistenza a strato metallico 221 Ohm 1/4W 1%
1x	115013	Cavetti Jumper M/M - lunghezze varie - 70pz
1x	333368	Cavo USB 2.0 tip A-B 1.8mt
1x	BB830	Bread Board con 830 punti di connessione
1x	301032	Potenziometro ALPHA 5KOhm Lineare - 16mm
1x	315002	Strip Maschio Passo 2,54 - 40 pin



Scuola di Robotica

www.scuoladirobotica.it