

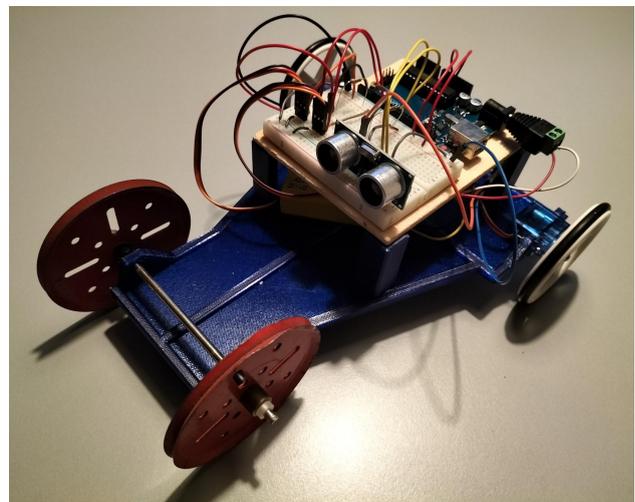


Attribuzione - Non commerciale -
Condividi allo stesso modo 4.0
Internazionale (CC BY-NC-SA 4.0)

Ar.Ve.Us (Arduino's Vehicle with Ultrasounds)

MATERIALE:

- Arduino
- Breadboard
- Led
- Fili
- 2 servomotori continui
- Sensore ad ultrasuoni (HC-SR04)
- Resistenze
- 4 ruote
- Jack per l'alimentazione
- Batteria da 9V
- Struttura base veicolo

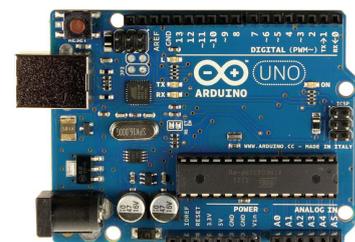


SCOPO:

Realizzare un veicolo in grado di muoversi da solo, di fermarsi davanti ad un oggetto ad una prefissata distanza, girare di 90° e riprendere a muoversi.

INTRODUZIONE AD ARDUINO:

Arduino è una scheda elettronica dotata di un microcontrollore ("cervello") in grado di controllare sensori, led ed altri componenti; il programma è compilato sul computer e poi trasferito sulla scheda.



SENSORE AD ULTRASUONI:

Il sensore ad ultrasuoni (HC-SR04) è il principale componente del progetto. Il suo funzionamento è molto semplice; è dotato di 2 "occhi", uno emette delle onde ad ultrasuoni mentre l'altro le riceve. Tali onde si propagano alla velocità del suono (343 m/s a temperatura ambiente) ed in caso ci sia un ostacolo, rimbalzano e tornano indietro; il tempo impiegato dalle onde a percorrere la distanza viene misurato. Dalla formula $V = S/t$ deriviamo quella inversa $S = V*t$. Conoscendo la velocità di propagazione delle onde



(V del suono) ed il tempo, Arduino calcola la distanza dal sensore all'oggetto (da dividere poi per 2 perchè compresa andata e ritorno).

SERVOMOTORE:

Il servomotore è un particolare motore che può funzionare a velocità variabile; è controllato da tre cavi; due servono per l'alimentazione a 5V ed un terzo per il controllo. Il livello di tensione che viene inviato è proporzionale alla velocità di rotazione.

Il controllo avviene secondo il metodo PWM (Pulse Width Modulation), ovvero Arduino fa variare la tensione del motore in base alla velocità richiesta.



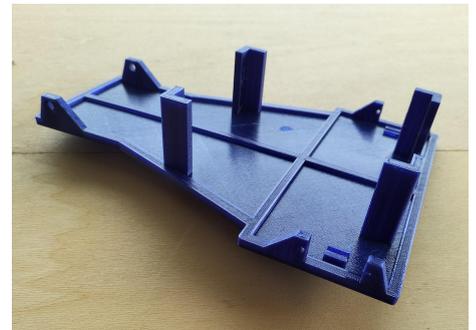
PROCEDIMENTO:

Parte elettronica:

1. Imparare ad utilizzare il sensore ad ultrasuoni con dei comandi semplici
2. Comprendere il funzionamento dei servomotori e PWM
3. Logica di controllo:
 - a. Impostare un ciclo IF (se): "se la distanza calcolata dalla funzione è minore di un dato valore → allora ferma i motori"
 - b. Far ruotare la macchinina bloccando una ruota (ovvero un motore)
 - c. Far fermare la macchinina bloccando entrambe le ruote (motori) una volta girata e farla ripartire
4. Si possono aggiungere anche altri componenti come dei led che si accendono quando la macchinina si ferma

Parte modellatore solido Freecad:

1. Definire su uno schizzo le misure della base e dei vari componenti
2. Modellare in 3D con Freecad la base e i vari supporti
3. Inserire i modelli dei componenti (motori, sensori, ...) per verificare che la base sia corretta e per avere un'idea dell'assieme
4. Stampare la base con una stampante 3D (PRUSA).



Una volta stampato il tutto si assembla e si rende la macchinina indipendente con un'alimentazione con una batteria da 9V.

PARTI IMPORTANTI DEL CODICE:

- La funzione per il calcolo della distanza:
Tutto si basa sulla formula $V=S/t$. Noi conosciamo infatti la velocità del suono e il tempo che passa da quando le onde sono emesse a quando tornano. Con questi dati siamo in grado di calcolare la distanza, $S= V*t$. E' importante ricordarsi che il tempo tra emissione e ritorno delle onde va diviso per due in quanto comprende andata e ritorno. La velocità inserita 0.034 cm/micro_s corrisponde a 343 m/s.

```

int calculateDistance() {
    digitalWrite(trigPin, LOW);

    digitalWrite(trigPin, HIGH); //emette onde per 10 microsecondi
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW); //smette di emettere onde
    durata = pulseIn(echoPin, HIGH); // calcola il tempo
    distanza= durata*0.034/2; //calcolo della distanza
    return distanza;
    delay(1000);
}

```

- Comando servomotori

```

motoredx.write(70); //questa scrittura ci permette di far
//girare i motori con diverse velocità (WPM)

```

- Ciclo if per farla fermare

```

if (distanza <= 30){
    digitalWrite(led, HIGH);
    digitalWrite(led2, HIGH);
    motoredx.write(90); //ferma il motore
    motorex.write(90);
    delay(1000);
    motoredx.write(90); // un motore rimane fermo mentre l'altro gira
    motorex.write(105);
    delay(2500);
    motoredx.write(90); //si fermano i motori
    motorex.write(90);
    delay(1000);
    digitalWrite(led, LOW);
    digitalWrite(led2, LOW);
}

```

Modello 3D del veicolo su Freecad

