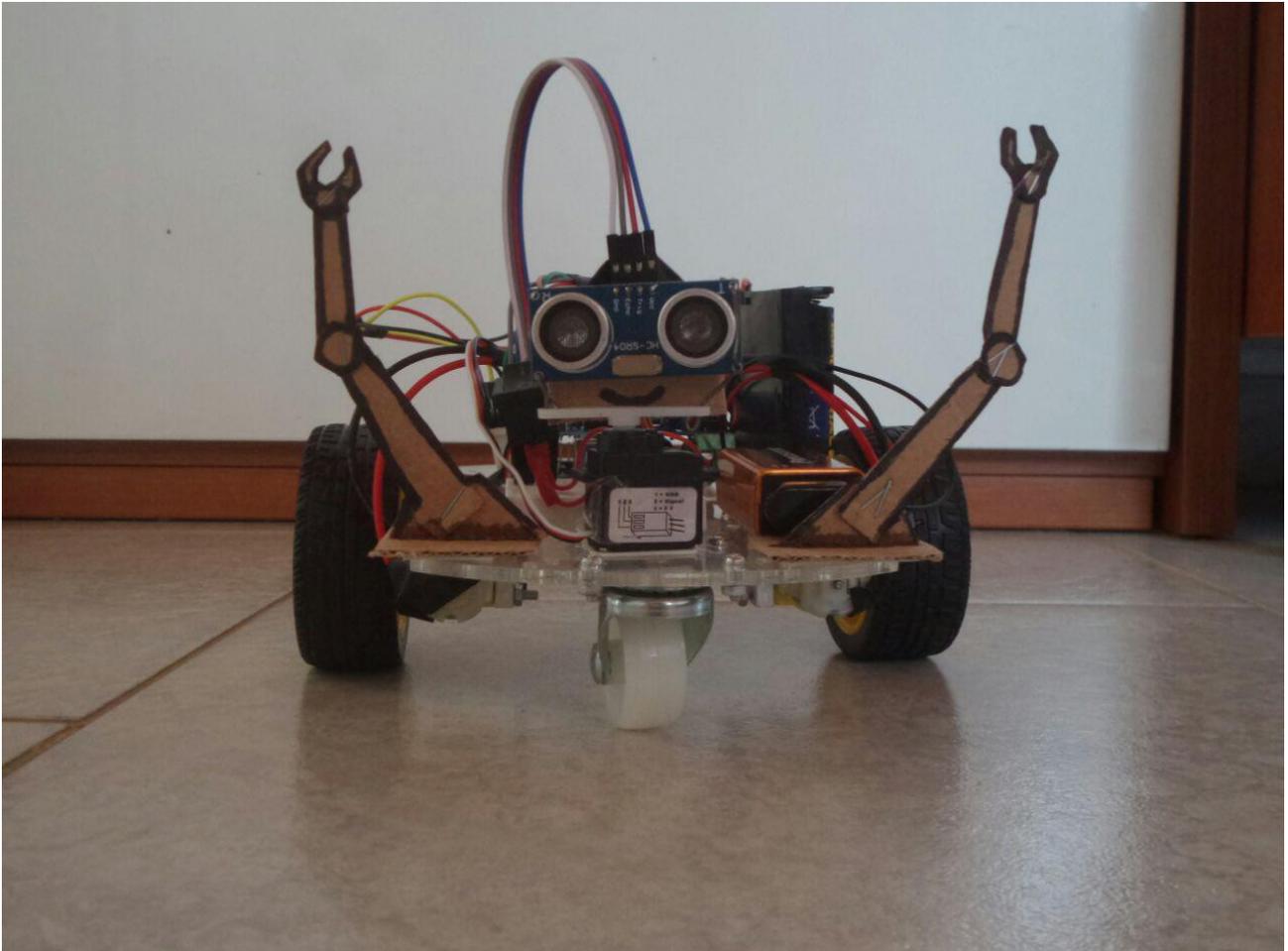


# Macchinina evita ostacoli Arduino



Sviluppo di una macchina autonoma in grado di evitare gli ostacoli sul suo percorso.

## Introduzione

In questo progetto abbiamo realizzato una macchinina in grado di muoversi autonomamente, rilevare ostacoli, guardarsi attorno e scegliere il percorso migliore.

Per fare ciò abbiamo sfruttato un sensore ad ultrasuoni, che manda un segnale all'Arduino quando rileva un ostacolo e una motor shield, che permette di gestire i due motori DC.

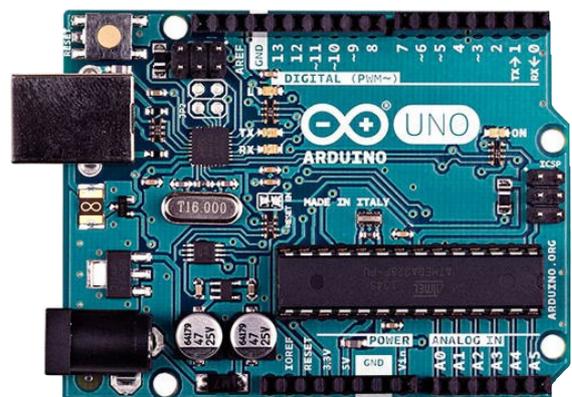
## Componenti:

Quantità	Descrizione	Costo
1	Arduino uno r3	20.00 €
1	Motor Shield	7.35 €
1	Sensore Ultrasuoni	4.59 €
1	Mini breadboard	2.60 €
1	Chassis	15.90 €
1	Servo motore	3.00 €
1	Batteria 9V	3.00 €

## Arduino Uno r3

Caratteristiche:

- Microcontrollore: Atmel ATmega328
- Tensione operativa: 5V
- Input Voltage: (consigliata) 7-12V
- Input Voltage: (limiti) 6-20V
- Pin di I/O Digitali: 14
- Pin di Input Analogici: 6
- Memoria flash: 32KB



- Memoria SRAM: 2KB
- Memoria: 1KB
- Frequenza di Clock: 16MHz

La piattaforma Arduino è costituita da una serie di schede elettroniche con diverse caratteristiche tecniche ed innumerevoli ambiti di utilizzo.

Per questo progetto abbiamo scelto di utilizzare un Arduino Uno, che pur mantenendo un numero di pin adatto allo scopo d'uso, presenta delle dimensioni abbastanza ridotte. Ciò è molto importante per la costruzione della nostra macchinina per la mancanza di spazio sullo chassis.

## Sensore ad ultrasuoni

### Caratteristiche:

- Alimentazione: 5V DC
- Angolo di misura:  $< 30^\circ$
- Distanza di rilevamento: 2- 400 cm
- Risoluzione: 1 cm
- Frequenza: 4kHz

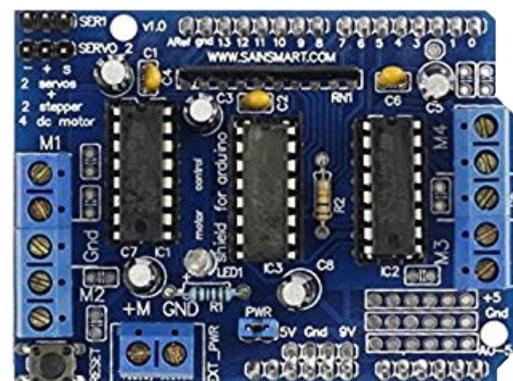


Il sensore ad ultrasuoni dispone di 4 pin: VCC, Trigger, Echo e GND. Il sistema emette delle onde sonore dal pin Trigger e riceve sul pin Echo la misura del tempo di rimbalzo contro un

qualsunque oggetto posto di fronte. In questo modo il sensore riesce a capire a quale distanza si trovi da un eventuale ostacolo. Questo modulo è capace di misurare in un range compreso tra 2 a 450 cm con una precisione di 3 mm.

## Motor Shield

Dato il limite di corrente generato dall'Arduino Uno, è stato necessario utilizzare una Motor Shield per la gestione dei motori della macchinina. Con un'alimentazione esterna la shield può fornire fino a 12V e 2A per canale motore o 4A su un singolo canale.



## Mini Breadboard



Piccola breadboard per connettere circuiti senza saldatura e di piccola dimensione.

## Chassis

Chassis della macchinina, che comprende due ruote, due motori, contenitore per 4 batterie AA, viti e cacciavite.



## Servomotore

In grado di ruotare di un angolo compreso tra 0 e 180°, utilizzato per far ruotare la testa del robot e riuscire così a rilevare la distanza degli ostacoli anche alla sua destra e sinistra.



Infine abbiamo utilizzato dei cavi per collegare il sensore all'Arduino, una batteria da 9V, più 4 batterie AA da 1.5V (non indispensabili)

## Realizzazione:

### Hardware

Come prima cosa abbiamo montato lo chassis avvitando i motori alle due ruote, a loro volta avvitate ai due lati della struttura. Poi abbiamo collegato i pin del sensore ad ultrasuoni: Vcc al pin 5V di Arduino; gnd del sensore al pin gnd di Arduino; il pin Trig al pin analogico A4; il pin Echo al pin analogico A5. In seguito abbiamo montato la motor shield al di sopra di Arduino e vi abbiamo collegato i due motori con gli appositi cavi e il servomotore. Infine abbiamo alimentato l'Arduino con una batteria da 9V e la motor shield con le 4 batterie AA da 1.5V. Poteva essere sufficiente alimentare solamente la motor shield con la batteria da 9V, ma constatando che i motori non avevano abbastanza potenza, abbiamo preferito aumentare l'alimentazione.

Piccoli accorgimenti:

- abbiamo deciso di saldare i fili che collegano i motori alla motor shield per fissarli meglio;
- per riuscire a far entrare i cavi del sensore insieme ai diodi della motor shield all'interno dei pin di Arduino, abbiamo tagliato e spellato le estremità dei cavi e infilato solo il rame, in modo che occupassero meno spazio;
- per non dover sempre togliere e rimettere le batterie, abbiamo ricavato un piccolo interruttore da una vecchia lampada e vi abbiamo collegato i cavi della corrente dalle batterie all'alimentazione.

### Software

Il codice di questo progetto è relativamente semplice.

Abbiamo utilizzato la libreria standard per il servomotore "Servo.h", la libreria per il sensore ad ultrasuoni "NewPing.h" e la libreria per la motor shield "AFMotor.h".

Funzionamento del codice:

La macchinina avanza autonomamente e rileva la distanza da eventuali ostacoli; quando questa è minore di 30 cm il robot si ferma, si guarda attorno e sceglie se andare a destra o a sinistra, a seconda di quale delle due vie abbia più spazio di

manovra. Se invece anche lo spazio laterale è inferiore a 30 cm, torna un po' indietro.

Per rilevare la distanza si utilizza la funzione "sonar.ping\_cm()", la quale restituisce la distanza in cm da un eventuale ostacolo, oppure zero, se non trova ostacoli. Per questo motivo abbiamo scritto una funzione "leggi\_cm" in cui, se la funzione "sonar.ping\_cm()" restituisce zero, la distanza acquisisce il valore massimo.

Il meccanismo di movimento delle ruote è molto banale:

per andare dritto si impostano entrambi i motori in modalità "FORWARD" sfruttando la funzione di libreria "nome\_motore.run(FORWARD)"; per andare indietro si impostano entrambi i motori in modalità "BACKWARD"; per girare a destra si imposta il motore sinistro in modalità "FORWARD" e quello destro in modalità "BACKWARD"; per girare a sinistra si imposta il motore destro in modalità "FORWARD" e quello sinistro in modalità "BACKWARD"; per fermare un motore si imposta in modalità "RELEASE".

link del codice:

<https://pastebin.com/WmaP2J5r>