

ARDUINO SONAR

COMPONENTI

Arduino Uno

Servo Motore

Sensore di distanza a ultrasuoni HC RS04

PC con IDE Arduino e Processing



Come Funziona

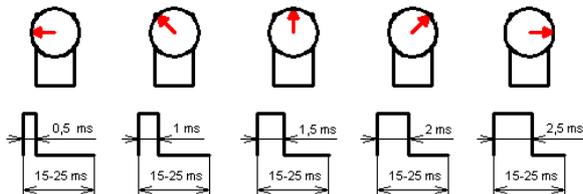
Viene programmato uno sketch che muove il servo motore su cui è montato il sensore a ultrasuoni e che rileva la distanza da eventuali oggetti che si trovano davanti. Il valore della distanza viene scritto sulla porta seriale e letto da un programma Processing che disegna il classico diagramma a raggi del sonar.

Servo Motore

Un servo motore permette il posizionamento del proprio albero su un angolo di circa 180°, tramite impulsi di comando di ampiezza variabile tra 0,5 ms (-90°) e 2,5 ms (+90°).

I servo motori presentano 3 connessioni: Vcc, Ground, Vin

Per generare gli impulsi che comandano il movimento normalmente si utilizza un PWM con frequenza 50 Hz ($T=20\text{ms}$) modulando il duty cycle.



Esempio di codice per muovere far oscillare il Servo Motore tra 10 e 170 gradi, si suppone Vin collegato al Pin 9 di Arduino:

```
#include <Servo.h>
Servo myservo;
int segno=1;
int angle=10;
void setup() {
  myservo.attach(9);
}
void loop(){
  angle+=segno;//per stabilizzare servo
  if (angle >= 170) segno=-1;
  if (angle <= 10) segno=1;
  myservo.write(angle);
}
```

Sensore di distanza a ultrasuoni HCSR 04

Emette un impulso a ultrasuoni e ne riceve l'eco.

Ha 4 Pin: Vcc, GND, Trigger, Echo.

Si comanda con un impulso di 10µs sul Trigger.

Sul Pin Echo parte un impulso che termina quando viene ricevuto il rimbalzo. La durata di questo impulso è proporzionale alla distanza (doppia) secondo la velocità del suono.

La massima distanza rilevabile è circa 2m.

Codice per utilizzare il sensore, si suppone trigger sul Pin 7 e echo sul pin 8 di arduino.

```
//HC RS04 Sensore ultrasuoni
int triggerPort = 7;
int echoPort = 8;
void setup() {
  pinMode( triggerPort, OUTPUT );
  pinMode( echoPort, INPUT );
  Serial.begin( 9600 );
  Serial.println( "Sensore ultrasuoni: " );
}
void loop() {
  //porta bassa l'uscita del trigger
  digitalWrite( triggerPort, LOW );

  //invia un impulso di 10microsec su trigger
  digitalWrite( triggerPort, HIGH );
  delayMicroseconds( 10 );
  digitalWrite( triggerPort, LOW );
  long duration = pulseIn( echoPort, HIGH ); //microsec
  long r = 0.034 * duration / 2; //cm

  Serial.print( "durata: " );
  Serial.print( duration );
  Serial.print( " , " );
  Serial.print( "distanza: " );

  //dopo 38ms è fuori dalla portata del sensore
  if( duration > 38000 ) Serial.println( "fuori portata");
  else { Serial.print( r ); Serial.println( "cm" );}
  //aspetta 1.5 secondi
  delay( 1500 );
}
```

Sonar

Arduino

Si deve costruire un supporto per montare il sensore HCSR04 sul Servo Motore. Una volta montato si collegano Vcc, Gnd di Servo Motore e HCSR04, Vin del Servo Motore al pin 9, Trigger al pin7 e echo al pin 8.

Il codice che muove il Servo, legge la distanza e la scrive sulla porta seriale è il seguente:

```
#include <Servo.h>

//HC RS04 Sensore ultrasuoni
int triggerPort = 7;
int echoPort = 8;

Servo myservo;
int segno=2;
int angle=10;
long distanza;
void setup(){
  pinMode( triggerPort, OUTPUT );
  pinMode( echoPort, INPUT );
  Serial.begin(9600);
  myservo.attach(9);
}
```

```

void loop(){
  angle+=segno;//per stabilizzare servo
  if (angle >= 170) segno=-2;
  if (angle <= 10) segno=2;

  myservo.write(angle);
  delay(100);
  //porta bassa l'uscita del trigger
  digitalWrite( triggerPort, LOW );

  //invia un impulso di 10microsec su trigger
  digitalWrite( triggerPort, HIGH );
  delayMicroseconds( 10 );
  digitalWrite( triggerPort, LOW );
  long duration = pulseIn( echoPort, HIGH ); //microsec
  if( duration > 38000 ) distanza=0;
  else distanza = 0.034 * duration / 2; //cm

  Serial.print ( "Angolo: " );
  Serial.print (angle);
  Serial.print( " durata: " );
  Serial.print( duration );
  Serial.print( " " );
  Serial.print( "distanza: " );
  Serial.print( distanza );
  Serial.println( " " );

  //aspetta 0.4 secondi
  delay( 400 );
}

```

Processing

Utilizzando la IDE di Processing si utilizza un programma che legge dalla porta seriale angolo e distanza e li rappresenta graficamente. Il codice è il seguente:

```

import processing.serial.*;
Serial porta;
int r;
Integer Colore=0;
void setup(){
  porta=new Serial(this,Serial.list()[0],9600);
  size(600, 600);
  background(19, 64, 180);
  stroke(19, 64, 180);
  delay(500);
}
void draw(){
  String val;
  Integer a;
  Integer d;
  String s[];
  if (porta.available()>0){
    val = porta.readStringUntil('\n');
    if (val != null){
      println(val);
      s=val.split(" ");
      d=Integer.valueOf(s[5]);
      a=Integer.valueOf(s[1]);
      fill(255, 64, 180);
      arc(300, 580,500, 500, 2*PI-(a+1)*PI/180, 2*PI-(a-1)*PI/180);
      fill(0);
      arc(300, 580, d*1.7, d*1.7, 2*PI-(a+1)*PI/180, 2*PI-(a-1)*PI/180);
    }
  }
}

```