Servo Motor+ Ultrasonic Sensor

Huaishu Peng | UMD CS | Fall 2024



DC Motor









https://www.etonm.com/About/5408770.html

3PPR, 7PPR 12PPR

Encoder available



Servo Motor

Low-speed, high-torque motor with precise positional control

A control signal is sent to the servo to position the shaft to desired angle.

When to use it?



What is a servo motor?

a motor with a built-in "servomechanism".

Consist of:

An electric motor (e.g., DC motor)

A feedback device

An electronic controller



What is a servo motor?

Motors can be DC, or AC

Feedback device can be encoder or other sensors for position sensing















Servo Motor

What we have today is a RC servo motor (because it is designed for RC car at the beginning)

Rotational range: $0 \sim 180$ degree

Wire connection: Power (Red) | GND (Brown) | Signal (Orange)

Operating Voltage: 5V

Controlling the position of a servo motor



Using a Servo lib

1. Install the ESP32servo lib from the library manager

PLATFORMIO	G ⁺ main.cp	p 🤯 PIO Home ×	く、 4 田 …	
> PROJECT TASKS				
\sim QUICK ACCESS			X	
✓ PIO Home	\wedge			
Open	ٹما ا	💾 Registry 🖹 Installed 💿 Built-in 🗘 Updates		
PIO Account	Home			
Inspect	_			
Projects & Configuration	₽	servo	Q	
Libraries	Projects			
Boards		C tit display C dnt [*] C header:RH_ASK.n C keyword:mqtt C framework:mbed C platform:espressif8266 more		
Platforms	Q			
Devices	Inspect	Libraries 301		
✓ Debug				
Start Debugging	ГЩ	ESP32Servo by Kevin Harrington	17,245 🗍 6 墩 Arduino	
Toggle Debug Console	Librariaa			
\sim Miscellaneous	Libraries	Allows ESD22 beards to control carve, topo and analog Write motors using Arduine comantics. This library can control a many types of carves <	of conversion the lateral land of	
Serial & UDP Plotter	E-	the ESP32 PWM timers: the library can control up to 16 servos on individual channels No attempt has been made to support multiple servos per channel. 		
PlatformIO Core CLI	×=			
Clone Git Project	Boards	I device, control		
New Terminal		器 Espressif 32		
Upgrade PlatformIO Core				

Using a Servo lib

2. Wiring



Wire connection: Power (Red) | GND (Brown) | Signal (Orange)5VGNDGPIO 13



TOUCH 6

TOUCH 5

Using a Servo lib

3. Coding

Servo Motors Pulse Width Modulation					
2.6ms		180°			
	20 ms Diagram not to scale				

#include <Arduino.h>
#include <ESP32Servo.h>

```
int pos = 0;
int servoPin = 13;
Servo myservo;
```

```
void setup() {
  myservo.attach(servoPin, 500, 2600);
}
```

```
void loop() {
   for (pos = 0; pos<180; pos+=1) {
     myservo.write(pos);
     delay(15);
   }
   delay(500);</pre>
```

```
for (pos = 180; pos>0; pos-=1) {
  myservo.write(pos);
  delay(15);
```

```
delay(500);
```

Pulse Width Modulation (PWM) Can you control the servo motor without using Lib but just PWM?

Hint:





Supporting functions you might need:

map(x, fromA, toA, fromB, toB);

delayMicroseconds(x);

For our servo motor: Pulse width: 20ms 0.5ms -> 0 degree 2.6ms -> 180 degree

```
#include <Arduino.h>
#include <ESP32Servo.h>
int pos = 0;
int servoPin = 13;
Servo myservo;
void setup() {
 myservo.attach(servoPin, 500, 2600);
void loop() {
  for (pos = 0; pos<180; pos+=1) {</pre>
    myservo.write(pos);
    delay(15);
  delay(500);
    for (pos = 180; pos>0; pos-=1) {
    myservo.write(pos);
    delay(15);
  delay(500);
```

```
#include <Arduino.h>
 1
 2
      int pos = 0;
      int servoPin = 13;
 4
 5
      void angle(int a);
 6
      void setup() {
        pinMode(servoPin, OUTPUT);
 9
10
11
12
      void loop() {
13
        for (pos = 0; pos<180; pos+=1) {</pre>
          angle(pos);
14
15
          delay(15);
16
        }
17
18
        delay(500);
19
20
          for (pos = 180; pos>0; pos-=1) {
21
          angle(pos);
22
          delay(15);
23
24
25
26
        delay(500);
27
28
      void angle(int a) {
29
30
```

Supporting functions you might need: Map(x, fromA, toA, fromB, toB); delayMicroseconds(x);



```
void angle(int a) {
    int pulseWidth = map(a, 0, 180, 500, 2600);
```

digitalWrite(servoPin, HIGH); delayMicroseconds(pulseWidth); digitalWrite(servoPin,LOW); delayMicroseconds(20000-pulseWidth); Supporting functions you might need: Map(x, fromA, toA, fromB, toB); delayMicroseconds(x);



Ultrasonic Sensor HC – SR04







The time between the transmission and reception of the signal allows us to calculate the distance to an object. This is possible because we know the sound's velocity in the air



VCC: +5VDC Trig : Trigger (**OUTPUT**) -> GPI05 Echo: Echo (**INPUT**) -> GPI018 GND: GND

Arduino has library for it, but Can you do it without library?

a. The sensor is triggered by a HIGH pulse of 10 or more microseconds.

b. To read the sensing signal: a HIGH pulse whose duration is the time (in microseconds) from the sending of the ping to the reception of its echo off of an object.

The pulseIn() function in Arduino is used to measure the duration of a pulse on a specified pin. It listens for either a HIGH or LOW pulse on that pin and returns the time in microseconds that the pulse lasted.



https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/

VCC: +5VDC Trig : Trigger (**OUTPUT**) Echo: Echo (**INPUT**) GND: GND

VCC: +5VDC Trig : Trigger (**OUTPUT**) Echo: Echo (INPUT) GND: GND



void distance() { delayMicroseconds(2);

digitalWrite(triggerPin, HIGH); delayMicroseconds(10); digitalWrite(triggerPin, LOW);

duration = pulseIn(echoPin, HIGH); distance = (duration/2) * 0.0343;

```
if(distance >= 20) {
  Serial.println("Out of range");
else {
  Serial.print(distance);
 Serial.println(" cm");
```

```
long duration=0, distance = -1;
digitalWrite(triggerPin, LOW);
```

Assignment:

Make a simple radar with the servo motor and the ultrasonic sensor Code should not use the servo library.

Submission: Code + Video

In the video, please put some obstacles in front of your radar and show the distance reading changes from the serial port.

Optional: write a GUI.



Radar

using Processing 3