



PARA CONFIGURAR EL PUERTO ANTES ESCRIBIR EN EL TERMINAL

```
sudo chmod 666 /dev/ttyACM0
```

```
0
```

```
sudo chmod 666 /dev/ttyUSB0
```

PRÁCTICA 1:ADELANTE ATRÁS.

```
// Configuramos los pines que vamos a usar
```

```
int motorDer1=2;//El pin 2 a In1 del L298N
```

```
int motorDer2=3;//El pin 3 a In2 del L298N
```

```
int motorIzq1=7;//El pin 7 a In3 del L298N
```

```
int motorIzq2=4;//El pin 4 a In4 del L298N
```

```
int derecho=5; //El pin 5 a EnA del L298N
```

```
int izquierdo=6;//El pin 6 aEnB del L298N
```

```
void setup()
```

```
{
```

```
  //Configuramos los pines como salida
```

```
  pinMode(motorDer1, OUTPUT);
```

```
  pinMode(motorDer2, OUTPUT);
```

```
  pinMode(motorIzq1, OUTPUT);
```

```
  pinMode(motorIzq2, OUTPUT);
```

```
  pinMode(derecho, OUTPUT);
```

```
  pinMode(izquierdo, OUTPUT);
```

```
}
```

```
void atras(){
```

```
  digitalWrite(motorDer1,HIGH);
```

```
  digitalWrite(motorDer2,LOW);
```

```
  digitalWrite(motorIzq1,HIGH);
```

```
digitalWrite(motorIzq2,LOW);
analogWrite(derecho,200);//Velocidad motor
analogWrite(izquierdo,200);
}
void adelante(){
digitalWrite(motorDer1,LOW);
digitalWrite(motorDer2,HIGH);
digitalWrite(motorIzq1,LOW);
digitalWrite(motorIzq2,HIGH);
analogWrite(derecho,200);
analogWrite(izquierdo,200);
}
void giraDerecha(){
digitalWrite(motorDer1,HIGH);
digitalWrite(motorDer2,LOW);
digitalWrite(motorIzq1,LOW);
digitalWrite(motorIzq2,HIGH);
analogWrite(derecho,150);
analogWrite(izquierdo,150);
}
void giraIzquierda(){
digitalWrite(motorDer1,LOW);
digitalWrite(motorDer2,HIGH);
digitalWrite(motorIzq1,HIGH);
digitalWrite(motorIzq2,LOW);
analogWrite(derecho,150);
analogWrite(izquierdo,150);
}
void parar(){
digitalWrite(motorDer1,LOW);
digitalWrite(motorDer2,LOW);
digitalWrite(motorIzq1,LOW);
```

```
digitalWrite(motorIzq2,LOW);
analogWrite(derecho,200);
analogWrite(izquierdo,200);
}
```

```
void loop() {
```

```
adelante();
delay (2000);
```

```
atras();
delay (2000);
}
```

PRÁCTICAS PARA HACER

PRÁCTICA 2 (AVANZA ESPERA Y RETROCEDE ESPERA)

///*Avanza a 200 con los dos motores durante un segundo y medio , luego para durante 1 segundo, va hacia atrás durante otro segundo y medio, espera un segundo antes de repetir el proceso.*

PRÁCTICA 3 (SERPIENTE)

///*Avanza a 200 con uno de los motores durante un segundo y medio , luego con el otro y así sucesivamente describiendo el movimiento de una serpiente.*

PRÁCTICA 4 (AVANZAR GIRO Y VOLVER)

///*Avanza a 200 con los dos motores un segundo y medio, se para un segundo, se gira 180 grados (probar con los tiempos y velocidades para que de la vuelta) y vuelve sobre sus pasos.*

PRÁCTICA 5 (GIRO EN CÍRCULOS)

///*Gira en círculos para lo cual el motor derecho a 200 revoluciones y el izquierdo a 80 revoluciones, durante 5 segundos, espera uno y se hace girar con los mismos datos pero en sentido contrario*

PRÁCTICA 6 (RECORRER UN CUADRADO)

///*Avanza los dos motores a 200 un segundo y medio, para un segundo, gira a 100 revoluciones 1,2 segundos para hacer más o menos 90 grados, para un segundo y repite el proceso. (variar los tiempos y velocidades hasta conseguir describir el cuadrado.*

PRÁCTICA MANEJO BLUETOOTH

```
// Configuramos los pines que vamos a usar
#include <SoftwareSerial.h>
#define rxPin 12
#define txPin 13
int motorDer1=2;//El pin 2 a In1 del L298N
int motorDer2=3;//El pin 3 a In2 del L298N
int motorIzq1=7;//El pin 7 a In3 del L298N
int motorIzq2=4;//El pin 4 a In4 del L298N
int derecho=5; //El pin 5 a EnA del L298N
int izquierdo=6;//El pin 6 aEnB del L298N

int velocidad=150;

// configura un nuevo puerto
SoftwareSerial miPuertoSerie = SoftwareSerial(rxPin, txPin);

int unChar;

void setup()
{
miPuertoSerie.begin(9600);

//Configuramos los pines como salida
pinMode(motorDer1, OUTPUT);
pinMode(motorDer2, OUTPUT);
pinMode(motorIzq1, OUTPUT);
pinMode(motorIzq2, OUTPUT);
pinMode(derecho, OUTPUT);
pinMode(izquierdo, OUTPUT);
```

```
}  
void atras(){  
    digitalWrite(motorDer1,HIGH);  
    digitalWrite(motorDer2,LOW);  
    digitalWrite(motorIzq1,HIGH);  
    digitalWrite(motorIzq2,LOW);  
    analogWrite(derecho,200);//Velocidad motor  
    analogWrite(izquierdo,200);  
}  
void adelante(){  
    digitalWrite(motorDer1,LOW);  
    digitalWrite(motorDer2,HIGH);  
    digitalWrite(motorIzq1,LOW);  
    digitalWrite(motorIzq2,HIGH);  
    analogWrite(derecho,200);  
    analogWrite(izquierdo,200);  
}  
void giraDerecha(){  
    digitalWrite(motorDer1,HIGH);  
    digitalWrite(motorDer2,LOW);  
    digitalWrite(motorIzq1,LOW);  
    digitalWrite(motorIzq2,HIGH);  
    analogWrite(derecho,150);  
    analogWrite(izquierdo,150);  
}  
void giraIzquierda(){  
    digitalWrite(motorDer1,LOW);  
    digitalWrite(motorDer2,HIGH);  
    digitalWrite(motorIzq1,HIGH);  
    digitalWrite(motorIzq2,LOW);  
    analogWrite(derecho,150);  
    analogWrite(izquierdo,150);
```

```
}  
void parar(){  
    digitalWrite(motorDer1,LOW);  
    digitalWrite(motorDer2,LOW);  
    digitalWrite(motorIzq1,LOW);  
    digitalWrite(motorIzq2,LOW);  
    analogWrite(derecho,200);  
    analogWrite(izquierdo,200);  
}  
  
void loop() {  
    // escucha la llegada de nuevos datos serie:  
    char unChar = miPuertoSerie.read();  
    // imprime el caracter:  
  
    if (unChar > 0) {  
        switch (unChar) {  
            case '2':  
                adelante();  
                delay (1000);  
                break;  
            case '3':  
                giraIzquierda();  
                delay (300);  
                break;  
            case '1':  
                giraDerecha();  
                delay (300);  
                break;  
            case '0':  
                parar();  
                delay (1000);  
        }  
    }  
}
```

```
break;  
case '9':  
  atras();  
  delay (1000);  
  break;  
}  
}  
}
```


PRÁCTICA ESQUIVA-OBSTÁCULOS

```
#include <Servo.h> //incluye la librería servo
Servo servoX; //llamamos al servo 1 servo X
```

```
int motorDer1=2;//El pin 2 a In1 del L298N
int motorDer2=3;//El pin 3 a In2 del L298N
int motorIzq1=7;//El pin 7 a In3 del L298N
int motorIzq2=4;//El pin 4 a In4 del L298N
int derecho=5; //El pin 5 a EnA del L298N
int izquierdo=6;//El pin 6 aEnB del L298N
```

```
int velocidad=150;
```

```
long distancia0;
long tiempo;
long distancia1;
long distancia2;
```

```
void setup(){
```

```
Serial.begin(9600);
pinMode(13, OUTPUT); /*activación del pin 13 como salida: para el pulso ultrasónico*/
pinMode(12, INPUT); /*activación del pin 12 como entrada: tiempo del rebote del ultrasonido*/
servoX.attach(8); //le damos señal al servo X con el pin 8
servoX.write(45); //le damos orden de reinicio a 45 grados
```

```
//Configuramos los pines como salida
pinMode(motorDer1, OUTPUT);
```

```
pinMode(motorDer2, OUTPUT);
pinMode(motorIzq1, OUTPUT);
pinMode(motorIzq2, OUTPUT);
pinMode(derecho, OUTPUT);
pinMode(izquierdo, OUTPUT);

}
void atras(){
  digitalWrite(motorDer1,HIGH);
  digitalWrite(motorDer2,LOW);
  digitalWrite(motorIzq1,HIGH);
  digitalWrite(motorIzq2,LOW);
  analogWrite(derecho,200);//Velocidad motor
  analogWrite(izquierdo,200);
}
void adelante(){
  digitalWrite(motorDer1,LOW);
  digitalWrite(motorDer2,HIGH);
  digitalWrite(motorIzq1,LOW);
  digitalWrite(motorIzq2,HIGH);
  analogWrite(derecho,200);
  analogWrite(izquierdo,200);
}
void giraDerecha(){
  digitalWrite(motorDer1,HIGH);
  digitalWrite(motorDer2,LOW);
  digitalWrite(motorIzq1,LOW);
  digitalWrite(motorIzq2,HIGH);
  analogWrite(derecho,150);
  analogWrite(izquierdo,150);
}
void giraIzquierda(){
```

```

digitalWrite(motorDer1,LOW);
digitalWrite(motorDer2,HIGH);
digitalWrite(motorIzq1,HIGH);
digitalWrite(motorIzq2,LOW);
analogWrite(derecho,150);
analogWrite(izquierdo,150);
}
void parar(){
digitalWrite(motorDer1,LOW);
digitalWrite(motorDer2,LOW);
digitalWrite(motorIzq1,LOW);
digitalWrite(motorIzq2,LOW);
analogWrite(derecho,200);
analogWrite(izquierdo,200);
}
void calculo0(){
servoX.write(45); //le decimos que el servo X vaya a la posición de 90 grados
delay(500);
digitalWrite(13,LOW); /* Por cuestión de estabilización del sensor*/
delayMicroseconds(5);
digitalWrite(13, HIGH); /* envío del pulso ultrasónico*/
delayMicroseconds(10);
tiempo=pulseIn(12, HIGH); /* Función para medir la longitud del pulso entrante. Mide el tiempo que transcurrido entre el envío
del pulso ultrasónico y cuando el sensor recibe el rebote, es decir: desde que el pin 12 empieza a recibir el rebote, HIGH, hasta que
deja de hacerlo, LOW, la longitud del pulso entrante*/
distancia0= int(0.017*tiempo);

}

void calculo1(){
servoX.write(90); //le decimos que el servo X vaya a la posición de 90 grados

```

```
delay (500);
digitalWrite(13,LOW); /* Por cuestión de estabilización del sensor*/
delayMicroseconds(5);
digitalWrite(13, HIGH); /* envío del pulso ultrasónico*/
delayMicroseconds(10);
tiempo=pulseIn(12, HIGH); /* Función para medir la longitud del pulso entrante. Mide el tiempo que transcurrido entre el envío
del pulso ultrasónico y cuando el sensor recibe el rebote, es decir: desde que el pin 12 empieza a recibir el rebote, HIGH, hasta que
deja de hacerlo, LOW, la longitud del pulso entrante*/
distancia1= int(0.016669*tiempo);

}
void calculo2(){
servoX.write(0); //le decimos que el servo X vaya a la posición de 90 grados
delay (500);
digitalWrite(13,LOW); /* Por cuestión de estabilización del sensor*/
delayMicroseconds(5);
digitalWrite(13, HIGH); /* envío del pulso ultrasónico*/
delayMicroseconds(10);
tiempo=pulseIn(12, HIGH); /* Función para medir la longitud del pulso entrante. Mide el tiempo que transcurrido entre el envío
del pulso ultrasónico y cuando el sensor recibe el rebote, es decir: desde que el pin 12 empieza a recibir el rebote, HIGH, hasta que
deja de hacerlo, LOW, la longitud del pulso entrante*/
distancia2= int(0.017001*tiempo);

}

void loop(){

calculo0();
calculo1();
calculo2();
```

```
servoX.write(45);

Serial.println("Distancia ");
Serial.println(distancia0);
Serial.println(distancia1);
Serial.println(distancia2);
Serial.println(" cm");

if (distancia0<=30 || distancia1<30 || distancia2<30)
{

parar();
delay(100);
atras();
delay(300);
parar();
delay(200);
calculo1();
delay(200);
calculo2();
delay(200);

if (distancia1>distancia2)
{
giraDerecha();
delay(500);
parar();
delay(200);
}else
{
```

```
giraIzquierda();  
delay(500);  
parar();  
delay(200);  
}  
}else  
{  
adelante();  
}  
  
}
```