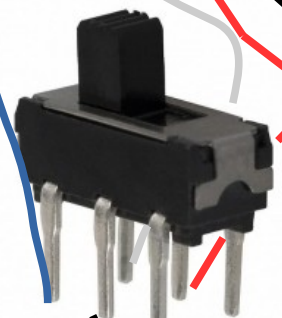
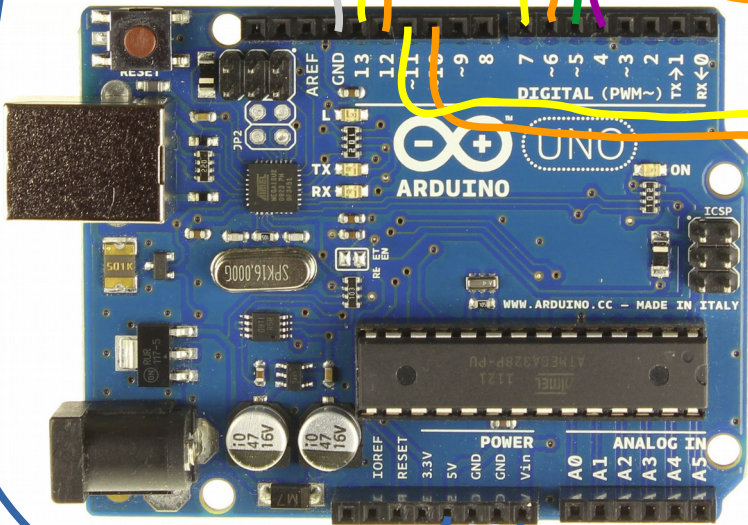
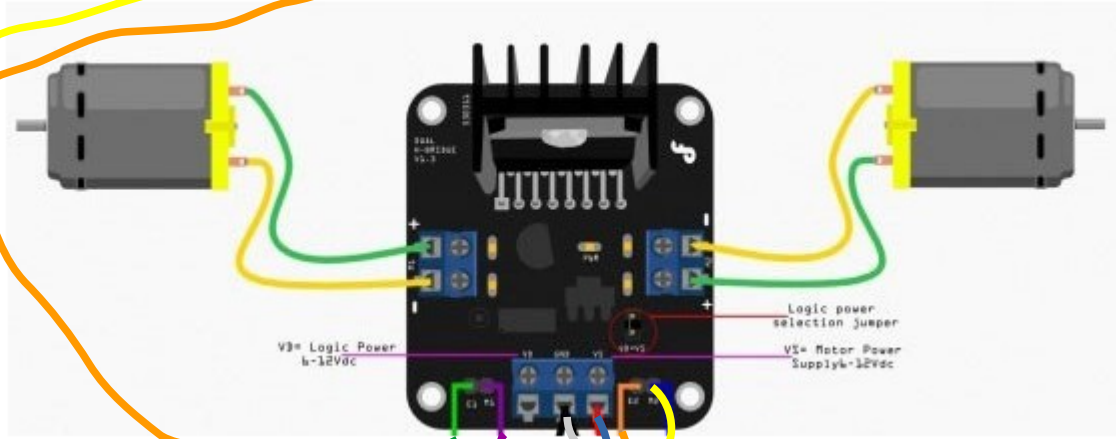


# PRÁCTICAS ROBÓTICA



## **PROGRAMA PARA CONFIGURAR EL PUERTO EN UBUNTU**

Probando el puerto USB

LOS COMANDOS LOS EJECUTAMOS TODOS DESDE EL TERMINAL

A continuación probaremos que nuestro Arduino está conectado correctamente a nuestro GNU/Linux.

```
dmesg | grep ttyACM
```

Que tendría que devolver lo siguiente (pero no exacto!):

```
[25714.468493] cdc_acm 3-1:1.0: ttyACM0: USB ACM device
```

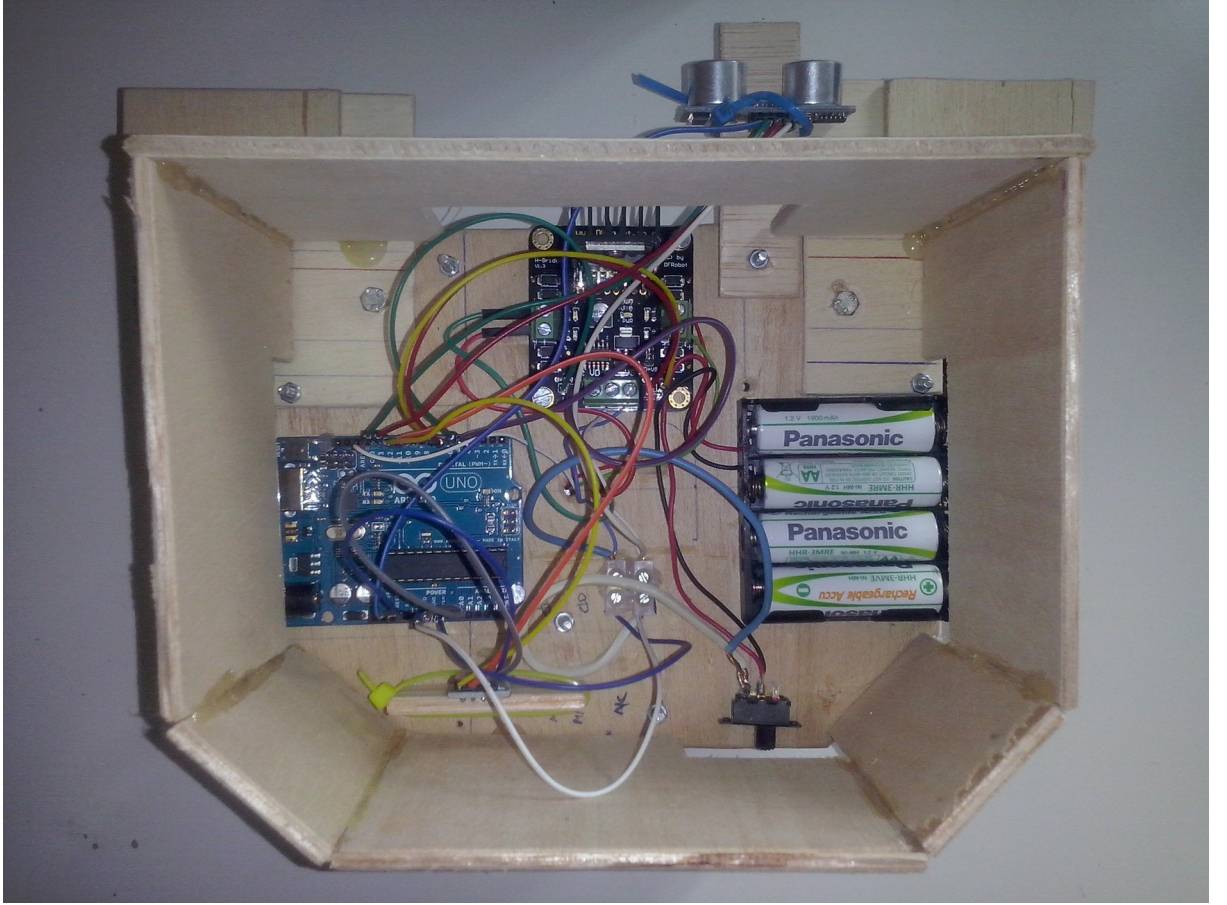
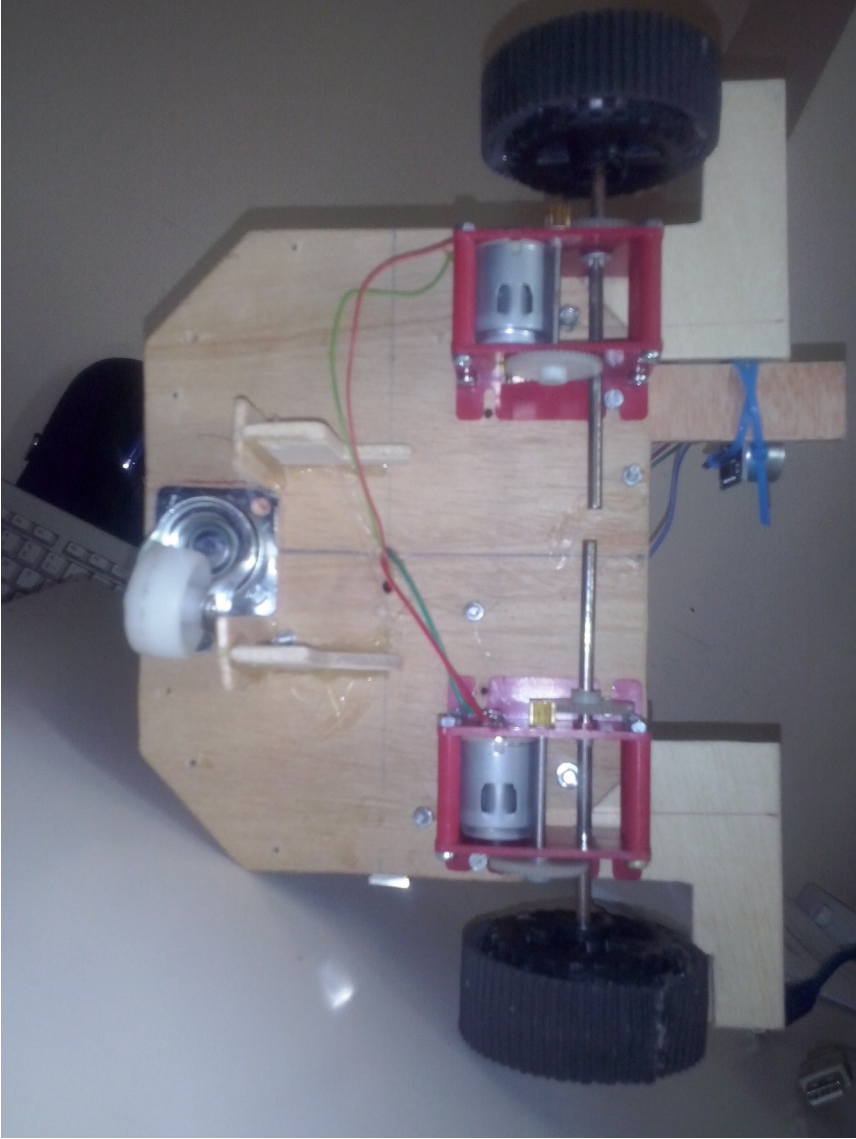
No todo está listo... ahora tendríamos que darle permisos suficientes al puerto ttyACM0 para que nuestro usuario pueda acceder a el ( a través del IDE ):  
Esto hay que hacerlo con el arduino conectado al ordenador

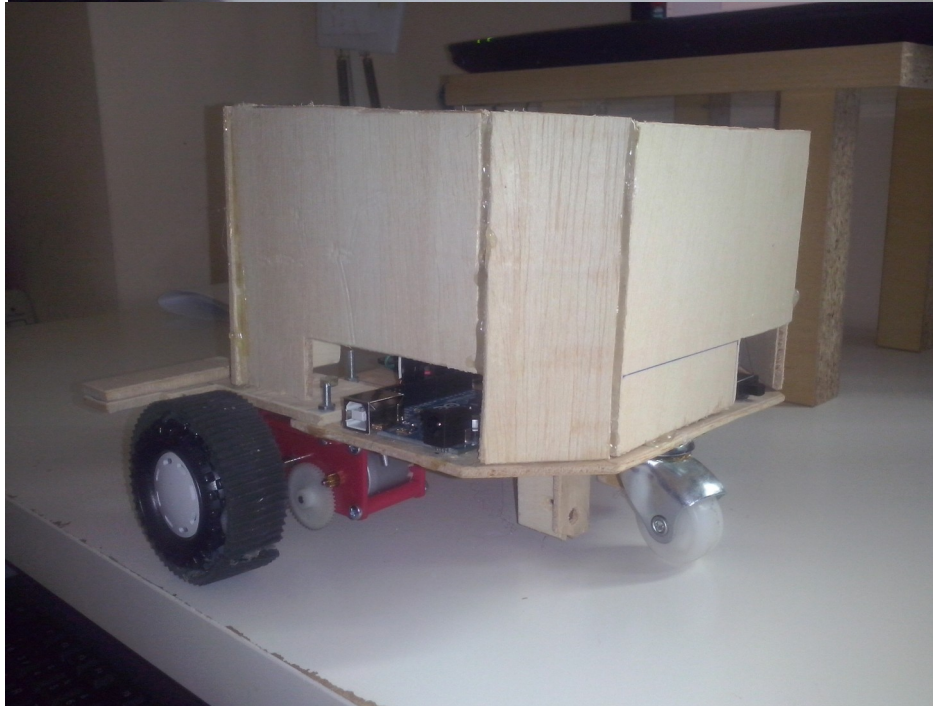
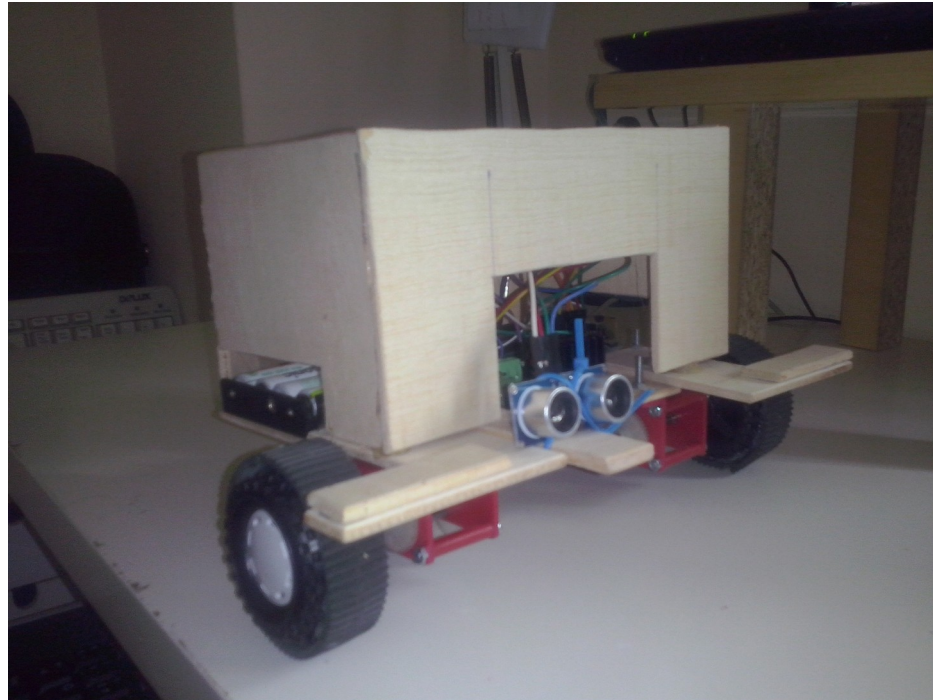
```
sudo chmod 666 /dev/ttyACM0
```

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

Ahora ya lo tenemos todo listo para poder ejecutar el IDE...

**NOTA: utilizar en lugar de motores de corriente continua normales, unos que tengan acoplada una reductora para que transmitan mayor fuerza o par.**





## PRÁCTICA 1. (HACIA DELANTE Y ATRÁS)

**/// **Avanza a 150 con los dos motores durante un segundo y medio , luego va hacia atrás durante otro segundo y medio.****

```
int E1 = 6;           // definimos el pin 6 con el nombre de E1
int M1 = 7;           // definimos el pin 7 con el nombre de M1
int E2 = 5;           // definimos el pin 5 con el nombre de E2
int M2 = 4;           // definimos el pin 4 con el nombre de M2

void setup()          // con la instrucción void setup configuramos los pines como entrada o salida
{
  pinMode(M1, OUTPUT); //le decimos que M1 es de salida
  pinMode(M2, OUTPUT); // le decimos que M2 es de salida
}

void loop()           //con void loop definimos las instrucciones de funcionamiento del programa
{
  {
    digitalWrite(M1,HIGH); // de decimos que el primer paso es que el motor M1 gire hacia delante
    digitalWrite(M2,HIGH); // los mismo que el motor M2 para delante
    analogWrite(E1, 150); //con la orden analogWrite establecemos la velocidad del motor 150 rpm
    analogWrite(E2, 150); // hacemos lo mismo con el motor 2 , E1 y E2 son sus variables para velocidad
    delay(1500);          // le decimos que espere un segundo y medio
    digitalWrite(M1,LOW); // ahora ordenamos que M1 gire hacia atrás
    digitalWrite(M2,LOW); // ahora los mismo pero con M2
    analogWrite(E1, 150); // definimos de nuevo la velocidad de cada motor con las variables E1 y E2
    analogWrite(E2, 150);
    delay(1500); // esperamos un segundo y medio

  }
}
```

## PRÁCTICA 2 (AVANZA ESPERA Y RETROCEDE ESPERA)

**///*# Avanza a 150 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.***

```
int E1 = 6;
int M1 = 7;
int E2 = 5;
int M2 = 4;

void setup()
{
  pinMode(M1, OUTPUT);
  pinMode(M2, OUTPUT);
}

void loop()
{
  {
    digitalWrite(M1,HIGH);
    digitalWrite(M2,HIGH);
    analogWrite(E1, 150);
    analogWrite(E2, 150);
    delay(1500);
    digitalWrite(M1,HIGH);
    digitalWrite(M2,HIGH);
    analogWrite(E1, 0);
    analogWrite(E2, 0);
    delay(1000);
    digitalWrite(M1,LOW);
    digitalWrite(M2,LOW);
    analogWrite(E1, 150);
    analogWrite(E2, 150);
    delay(1500);
    digitalWrite(M1,HIGH);
    digitalWrite(M2,HIGH);
    analogWrite(E1, 0);
    analogWrite(E2, 0);
    delay(1000);
  }
}
```

### PRÁCTICA 3 (SERPIENTE)

///**# Avanza a 150 con uno de los motores durante un segundo y medio , luego con el otro y asi sucesivamente describiendo el movimiento de una serpiente.**

```
int E1 = 6;
int M1 = 7;
int E2 = 5;
int M2 = 4;

void setup()
{
  pinMode(M1, OUTPUT);
  pinMode(M2, OUTPUT);
}

void loop()
{
  {
    digitalWrite(M1,HIGH);
    digitalWrite(M2,HIGH);
    analogWrite(E1, 150);
    analogWrite(E2, 0);
    delay(1500);

    digitalWrite(M1,HIGH);
    digitalWrite(M2, HIGH);
    analogWrite(E1, 0);
    analogWrite(E2, 150);
    delay(1500);
  }
}
```

## PRÁCTICA 4 (AVANZAR GIRO Y VOLVER)

///**# Avanza a 150 con los dos motores un segundo y medio, se para un segundo, se gira 180 grados (motor derecho a 100 y izquierdo -100 durante dos segundos) y vuelve sobre sus pasos.**

```
int E1 = 6;  
int M1 = 7;  
int E2 = 5;  
int M2 = 4;
```

```
void setup()  
{  
  pinMode(M1, OUTPUT);  
  pinMode(M2, OUTPUT);  
}
```

```
void loop()  
{  
  {  
    digitalWrite(M1,HIGH);  
    digitalWrite(M2,HIGH);  
    analogWrite(E1, 150);  
    analogWrite(E2, 150);  
    delay(1500);  
    digitalWrite(M1,HIGH);  
    digitalWrite(M2,LOW);  
    analogWrite(E1, 0);  
    analogWrite(E2, 0);  
    delay(1000);
```

```
    digitalWrite(M1,HIGH);  
    digitalWrite(M2, LOW);  
    analogWrite(E1, 100);  
    analogWrite(E2, 100);  
    delay(2000);  
    digitalWrite(M1,HIGH);  
    digitalWrite(M2,HIGH);  
    analogWrite(E1, 0);  
    analogWrite(E2, 0);  
    delay(1000);  
  }  
}
```



## PRÁCTICA 5 (GIRO EN CÍRCULOS)

**/// **Gira en círculos para lo cual el motor derecho a 150 revoluciones y el izquierdo a 80 revoluciones, durante 5 segundos, espera uno y se hace girar con los mismos datos pero en sentido contrario.****

```
int E1 = 6;
int M1 = 7;
int E2 = 5;
int M2 = 4;

void setup()
{
  pinMode(M1, OUTPUT);
  pinMode(M2, OUTPUT);
}

void loop()
{
  {
    digitalWrite(M1,HIGH);
    digitalWrite(M2,HIGH);
    analogWrite(E1, 150);
    analogWrite(E2, 80);
    delay(5000);
    digitalWrite(M1,HIGH);
    digitalWrite(M2,LOW);
    analogWrite(E1, 0);
    analogWrite(E2, 0);
    delay(1000);

    digitalWrite(M1,HIGH);
    digitalWrite(M2, HIGH);
    analogWrite(E1, 80);
    analogWrite(E2, 150);
    delay(5000);
    digitalWrite(M1,HIGH);
    digitalWrite(M2,HIGH);
    analogWrite(E1, 0);
    analogWrite(E2, 0);
    delay(1000);
  }
}
```

## PRÁCTICA 6 (RECORRER UN CUADRADO)

**///  
// Avanza los dos motores a 150 un segundo y medio, para un segundo, gira a 100 revoluciones 1,2 segundos para hacer mas o menos 90 grados, para un segundo y repite el proceso.**

```
int E1 = 6;
int M1 = 7;
int E2 = 5;
int M2 = 4;

void setup()
{
  pinMode(M1, OUTPUT);
  pinMode(M2, OUTPUT);
}

void loop()
{
  {
    digitalWrite(M1,HIGH);
    digitalWrite(M2,HIGH);
    analogWrite(E1, 150);
    analogWrite(E2, 150);
    delay(1500);
    digitalWrite(M1,HIGH);
    digitalWrite(M2,LOW);
    analogWrite(E1, 0);
    analogWrite(E2, 0);
    delay(1000);

    digitalWrite(M1,HIGH);
    digitalWrite(M2, LOW);
    analogWrite(E1, 100);
    analogWrite(E2, 100);
    delay(1200);
    digitalWrite(M1,HIGH);
    digitalWrite(M2,HIGH);
    analogWrite(E1, 0);
    analogWrite(E2, 0);
    delay(1000);
  }
}
```

## PRÁCTICA 7 (AVANZANDO DE MENOS A MÁS)

**//# Avanza con intervalo de 0 a 150 luego se para 200 milisegundos y se vuelve a repetir el proceso.**

```
int E1 = 6;
int M1 = 7;
int E2 = 5;
int M2 = 4;

void setup()
{
  pinMode(M1, OUTPUT);
  pinMode(M2, OUTPUT);
}

void loop()
{
  int value;          // definimos el valor de las revoluciones de cada motor con el nombre value
  for(value = 0 ; value <= 150; value+=5) // aumentamos el valor value de 5 en 5 de 0 hasta 150 revol.
  {
    digitalWrite(M1,HIGH);
    digitalWrite(M2,HIGH);
    analogWrite(E1, value);
    analogWrite(E2, value);
    delay(200);
  }
}
```

**//PRÁCTICA 8: colocamos un sensor de distancias por ultrasonido y le decimos que cuando el //robot detecte algún objeto a uno 50 cm de distancia rote y siga avanzando.**

```
int E1 = 6;
int M1 = 7;
int E2 = 5;
int M2 = 4;

long distancia;
long tiempo;
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*/
  pinMode(M1, OUTPUT);
  pinMode(M2, OUTPUT);
}

void loop(){
  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*/
  distancia= int(0.017*tiempo); /*fórmula para calcular la distancia obteniendo un valor entero*/
  /*Monitorización en centímetros por el monitor serial*/
  Serial.println("Distancia ");
  Serial.println(distancia);
  Serial.println(" cm");
```

```
if (distancia>50)           //50 cm es la distancia de emergencia
{
digitalWrite(M1,HIGH);
  digitalWrite(M2,HIGH);
  analogWrite(E1, 200); //a uno de los motores más velocidad que al otro porque el robot se desviaba hacia un lado.
  analogWrite(E2, 175);
  delay(1000);
digitalWrite(M1,HIGH);
  digitalWrite(M2,HIGH);
  analogWrite(E1, 0);
  analogWrite(E2, 0);
  delay(1000);
}
else if (distancia<50)
{
digitalWrite(M1,HIGH);
  digitalWrite(M2,LOW);
  analogWrite(E1, 150);
  analogWrite(E2, 150);
  delay(1200);
digitalWrite(M1,HIGH);
  digitalWrite(M2,HIGH);
  analogWrite(E1, 0);
  analogWrite(E2, 0);
  delay(5000);
}
}
```

**PRÁCTICA 9: mediante un sensor bluetooth controlamos los movimientos del robot, asignamos números a los movimientos que queremos que haga el robot., con el programa Bluetooth Controller asociamos a varios botones los números determinados para controlar el desplazamiento del robot mediante dicha app.**

GIADO CON VARIOS BOTONES 1 IZQ, 3 DER, 2 AVANCE, 9 RETROCESO Y O PARO.(a uno de los motores se le hace girar menos revoluciones porque el robot se desviaba hacia un lado y no iba del todo en línea recta, supongo que uno de los motores recibía más corriente, ya que al modificar la alineación de los motores seguía habiendo fallo, el delay de un segundo sirve para que no se quedará el micro bloqueado en un proceso determinado).

```
#include <SoftwareSerial.h>

#define rxPin 10
#define txPin 11
int E1 = 6;
int M1 = 7;
int E2 = 5;
int M2 = 4;

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

int unChar;

void setup() {
  miPuertoSerie.begin(9600);
  pinMode(M1, OUTPUT);
  pinMode(M2, OUTPUT);
}

void loop() {
  // escucha la llegada de nuevos datos serie:
  char unChar = miPuertoSerie.read();
  // imprime el caracter:
```

```
if (unChar > 0) {
  switch (unChar) {
    case '2':
      digitalWrite(M1,HIGH);
      digitalWrite(M2,HIGH);
      analogWrite(E1, 200);
      analogWrite(E2, 175);
      delay (1000);
      break;
    case '3':
      digitalWrite(M1,LOW);
      digitalWrite(M2,HIGH);
      analogWrite(E1, 150);
      analogWrite(E2, 150);
      delay (500);
      break;
    case '1':
      digitalWrite(M1,HIGH);
      digitalWrite(M2,LOW);
      analogWrite(E1, 150);
      analogWrite(E2, 150);
      delay (500);
      break;
    case '0':
      digitalWrite(M1,HIGH);
      digitalWrite(M2,HIGH);
      analogWrite(E1, 0);
      analogWrite(E2, 0);
      delay (1000);
      break;
    case '9':
      digitalWrite(M1,LOW);
      digitalWrite(M2,LOW);
      analogWrite(E1, 150);
      analogWrite(E2, 150);
      delay (1000);
      break;
  }
}
```