

## Coche teledirigido

Rubén García y Joan Gallardo





#### Introducción

El proyecto consiste en la **construcción y programación** de un coche teledirigido controlado mediante el **procesador ESP-32** y una **aplicación móvil vía Bluetooth**.

Vamos a ver:

Componentes
Programación
Aplicación mòvil
Pruebas y resultados
Conclusión y posibles mejoras futuras





#### Componentes

**Procesador ESP-32** 



Kit del coche



Controlador de motor de corriente continua



Aplicación mòvil









### Programación

```
#include <Arduino.h>
#include "BluetoothSerial.h"

int Pin1 = 15; //P
int Pin2 = 2; //N
int Pin3 = 4; //P
int Pin4 = 0; //N

BluetoothSerial SerialBT;
char t;
```

```
/oid setup()
   pinMode(Pin1, OUTPUT);
   pinMode(Pin2, OUTPUT);
   pinMode(Pin3, OUTPUT);
  pinMode(Pin4, OUTPUT);
  Serial.begin(115200);
  SerialBT.begin("ESP32RubenJoan");
  delay(500);
  Serial.println("ESP32 en busqueda");
   ledcSetup(0, 5000, 8); //Canal, frecuencia y resolución de 8 bits
   ledcSetup(1, 5000, 8);
  ledcSetup(2, 5000, 8);
  ledcSetup(3, 5000, 8);
  ledcAttachPin(Pin1, 0);
  ledcAttachPin(Pin2, 1);
  ledcAttachPin(Pin3, 2);
   ledcAttachPin(Pin4, 3);
```



#### Programación

```
void loop()
   if(SerialBT.available())
     t = SerialBT.read();
     Serial.println(t);
           ledcWrite(0, 0);
           ledcWrite(1, 200);
           ledcWrite(2, 0);
           ledcWrite(3, 200);
           Serial.println("Front"); //Front
       else if (t == 'P')
           ledcWrite(1, 0);
           ledcWrite(0, 0);
           ledcWrite(2, 0);
           ledcWrite(3, 0);
           Serial.println("Stop"); //Stop
           ledcWrite(1, 210);
           ledcWrite(2, 0);
           ledcWrite(3, 135);
           Serial.println("Forward Right"); //Forward Righ
```

```
else if (t == 'Q')
    ledcWrite(0, 0);
    ledcWrite(1, 135);
    ledcWrite(2, 0);
   ledcWrite(3, 210);
   Serial.println("Forward Left"); //Forward Left
else if (t == 'R')
   ledcWrite(0, 0);
   ledcWrite(1, 200);
   ledcWrite(2, 200);
   ledcWrite(3, 0);
   Serial.println("Right"); //Right
else if (t == 'L')
    ledcWrite(0, 200);
    ledcWrite(1, 0);
    ledcWrite(2, 0);
    ledcWrite(3, 200);
    Serial.println("Left"); //Left
else if (t == 'B')
    ledcWrite(0, 200);
   ledcWrite(1, 0);
   ledcWrite(2, 200);
    ledcWrite(3, 0);
   Serial.println("Backward"); //Backward
```



# Control del coche mediante la aplicación móvil



- Interfaz de diseño visual
- Programación por bloques

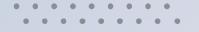




. . . . . . . . . .

```
when ListPicker1 AfterPicking
 do (if call BluetoothClient1 .Connect
                                  address | ListPicker1 | Selection |
     then set ListPicker1 . Elements to BluetoothClient1 . AddressesAndNames
when ListPicker1 BeforePicking
    set ListPicker1 . Elements to BluetoothClient1 . AddressesAndNames .
 when Clock1 .Timer
 do (3) if BluetoothClient1 = 1 IsConnected =
    then set State . Text to Connected
    else set State . Text to Disconnected
 when Forward TouchDown
                                               when Forward_Left . TouchDown
 do call BluetoothClient1 .SendText
                                              do call BluetoothClient1 . SendText
                                                                           text • • • • •
 when Stop ... TouchDown
                                               when Forward_Right ... TouchDown
   call BluetoothClient1 .SendText
                                                  call BluetoothClient1 . SendText
                            text P
                                                                           text "E"
 when Backward TouchDown
                                               when Backward_Left .TouchDown
 do call BluetoothClient1 SendText
                                                 call BluetoothClient1 . SendText
                             text B
                                                                           text A
when Left .TouchDown
                                              when Backward_Right . TouchDown
do call BluetoothClient1 . SendText
                                              do call BluetoothClient1 : SendText
                            text " L "
                                                                           text D
when Right ... TouchDown
do call BluetoothClient1 v SendText
                            text R
```





#### Pruebas y resultados

**PWM:** Cambiar 'digitalWrite' por un PWM para permitir modificar la velocidad a la que giran los motores

Ciclos de PWM: Ciclo mínimo a 125 (Le falta fuerza al motor para girar)

**Sobretensión:** Problemas con el reset del microcontrolador al hacer cambios en el sentido de los motores





Excelente oportunidad para aplicar los conocimientos aprendidos en clase, cómo **programación**, **electrónica** y **protocolos Bluetooth**, entre otros.

#### Posibles mejoras

Mejorar la precisión del control Implementar sensores Mejorar la aplicación móvil

