



TECNICATURA SUPERIOR EN
Telecomunicaciones

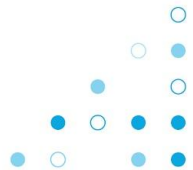
Arquitectura y conectividad

Trabajo Práctico N°6

Grupo N°1

INTEGRANTES

- Fernando Gimenez Coria
- Nicolás Nahuel Barrionuevo
- Macarena Aylen Carballo
 - Raul Antonio Jara
 - Diego Ezequiel Ares
- Juan Diego González Antoniazzi

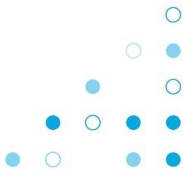


Proyecto 1

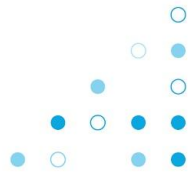
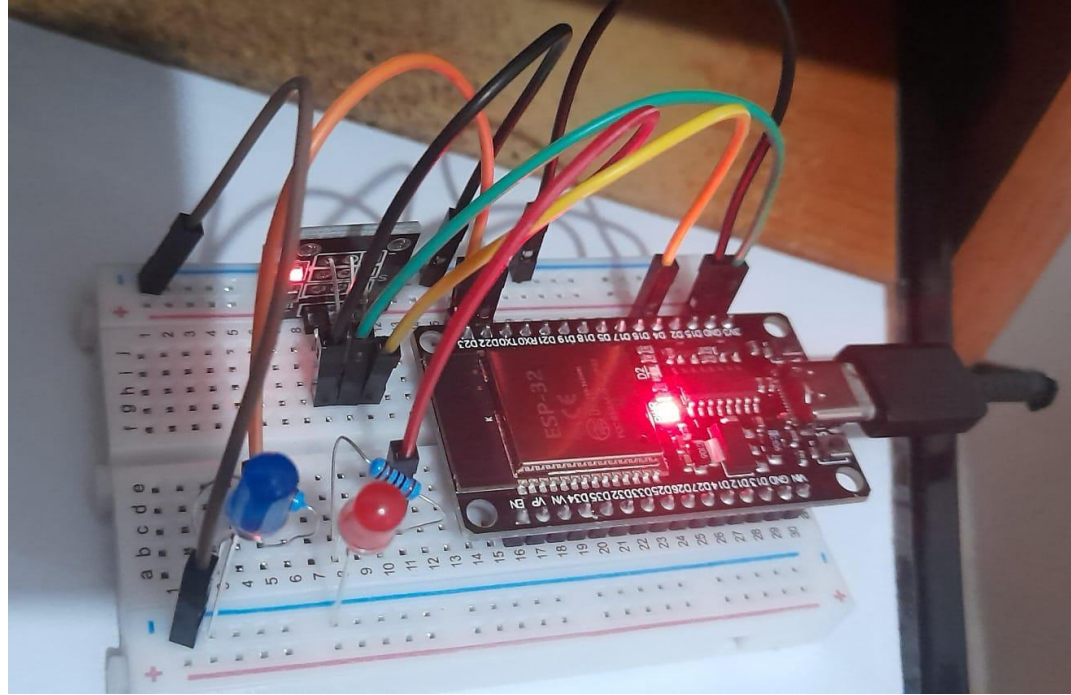
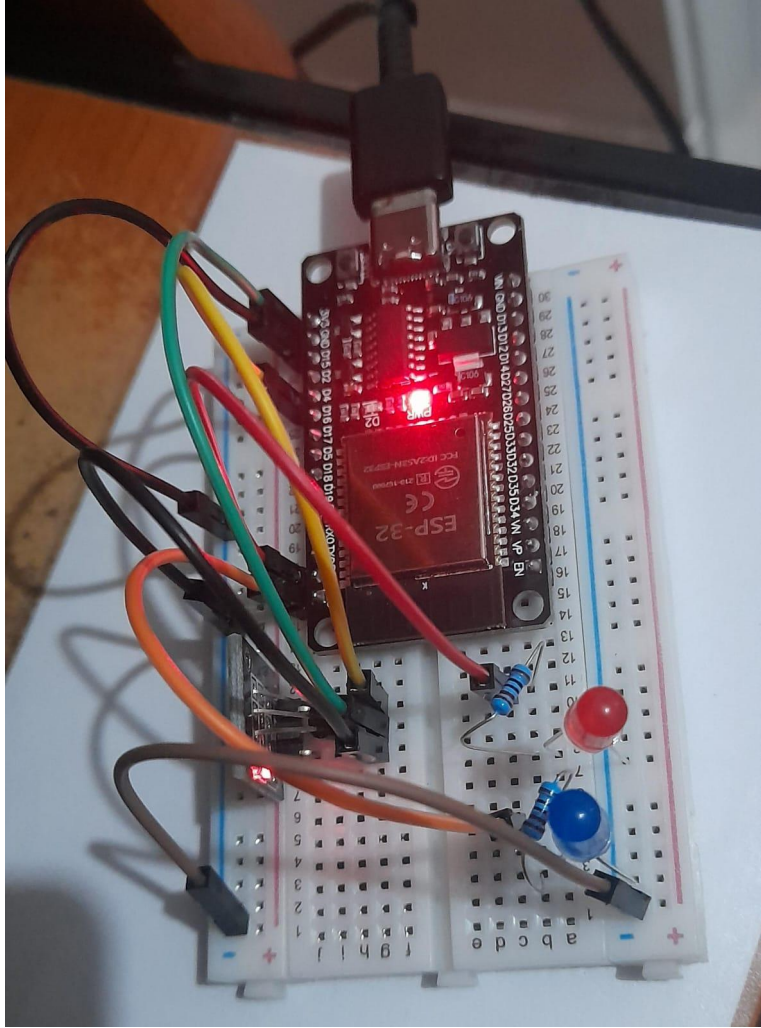
Implementar un proyecto, mediante el cual puedan monitorear un sensor de temperatura y humedad estableciendo una conexión con ESP-32 y Smartphone.

La consigna es comunicarse usando protocolo de comunicaciones Bluetooth.

La comunicación es entre el Smartphone o entorno virtual (PC) y el ESP-32.



Conexiones



Codigo

```
1  #include <OneWire.h>
2  #include <DallasTemperature.h>
3  #include <BluetoothSerial.h>
4
5
6  // Here the input pin is declared to which the sensor module is connected
7  #define sensor_temperatura_PIN 4
8  #define LED_TEMPERATURA 22
9  #define LED_EXTRACTOR 23
10 BluetoothSerial SerialBT; // Objeto para la comunicación Bluetooth
11
12 // Libraries are configured
13 OneWire oneWire(sensor_temperatura_PIN);
14 DallasTemperature sensor_t(&oneWire);
15
16
```

```
17 void setup() {
18
19     // Initialize serial output
20     Serial.begin(115200);
21     SerialBT.begin("ESP32_Sensor"); // Nombre del dispositivo Bluetooth
22     pinMode(LED_EXTRACTOR, OUTPUT); // Configuración del pin del LED de extractor
23     pinMode(LED_TEMPERATURA, OUTPUT); // Configuración del pin del LED de temperatura
24     Serial.println("Sensor DS18B20"); // nombre del dispositivo sensor
25
26     sensor_t.begin(); // Se inicializa el sensor.
27 }
28
```

```
29 //main program loop
30 void loop()
31 {
32     // Inicio de medición de temperatura.
33     sensor_t.requestTemperatures();
34     // ... se asigna la medición a una variable.
35     float temperatura = sensor_t.getTempCByIndex(0);
36     Serial.print("Temperatura: ");
37     Serial.print(temperatura);
38     Serial.println(" °C");
39     // Se manda dato por bluetooth.
40     SerialBT.print("T:");
41     SerialBT.println(temperatura);
42     // Control del LED de temperatura
43     if (temperatura <= 24) {
44         digitalWrite(LED_TEMPERATURA, LOW);
45     } else if (temperatura <= 30) {
46         digitalWrite(LED_TEMPERATURA, HIGH);
47     } else {
48         digitalWrite(LED_TEMPERATURA, HIGH);
49         delay(200);
50         digitalWrite(LED_TEMPERATURA, LOW);
51         delay(200);
52     }
53     if (SerialBT.available() > 0){
54         String mensaje = SerialBT.readStringUntil('\n'); // lee el strig proveniente del telefono hasta el caracter de nueva linea
55         mensaje.trim(); // recorta espacios y saltos
56         Serial.println("Comando Recibido");
57         Serial.println(mensaje);
58
59         if (mensaje.equalsIgnoreCase("encender extractor")){
60             digitalWrite(LED_EXTRACTOR, HIGH);
61             Serial.println("Extractor encendido");
62         }
63         else if (mensaje.equalsIgnoreCase("apagar extractor")){
64             digitalWrite(LED_EXTRACTOR, LOW);
65             Serial.println("Extractor apagado");
66         }
67         else {
68             Serial.println("Comando incorrecto");
69         }
70         delay(1000); // 1s pause until next measurement
71     }
72 }
```





¡Muchas gracias!