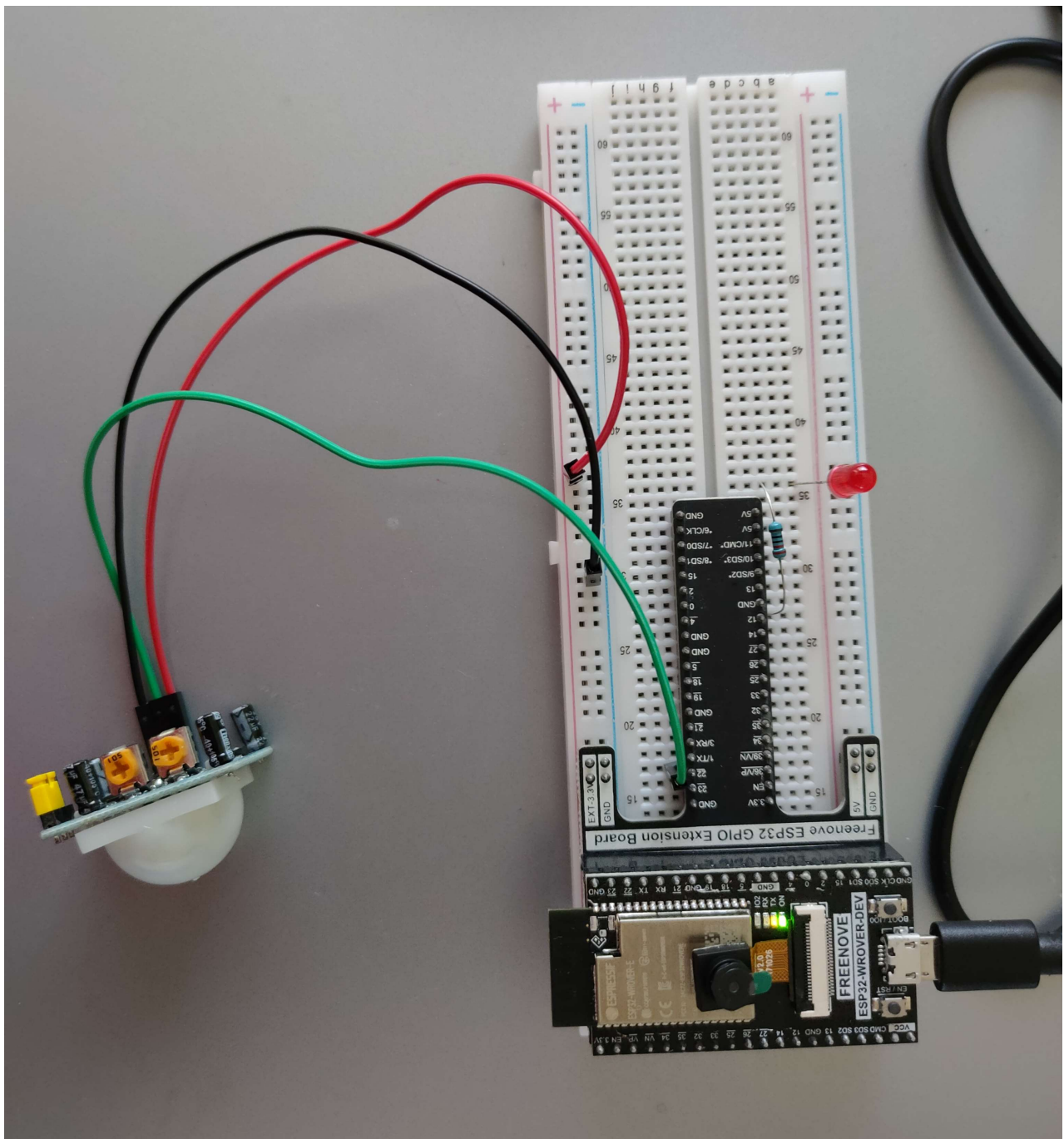


Sensor de presenza domótico

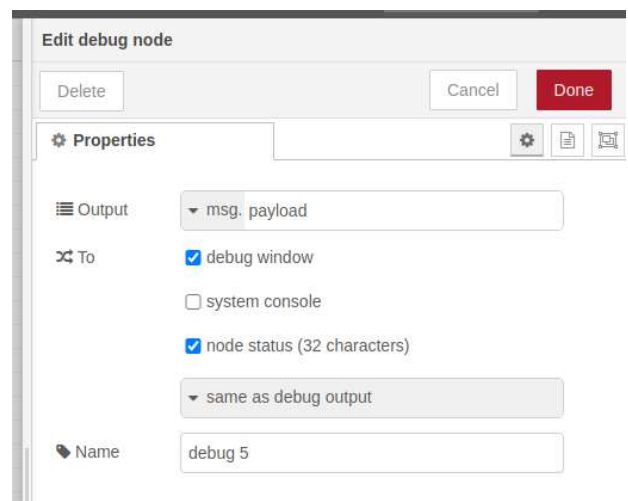
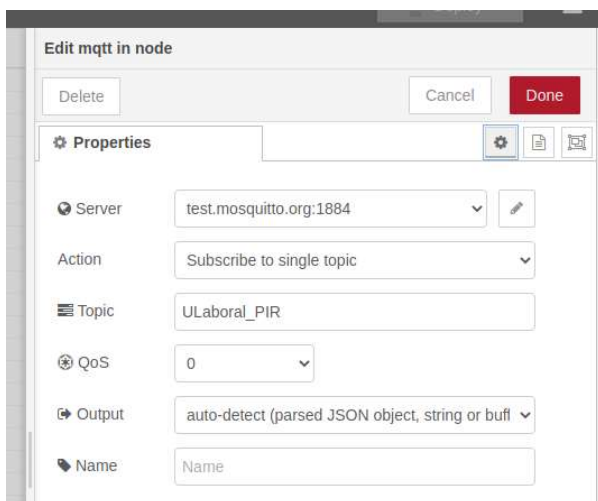
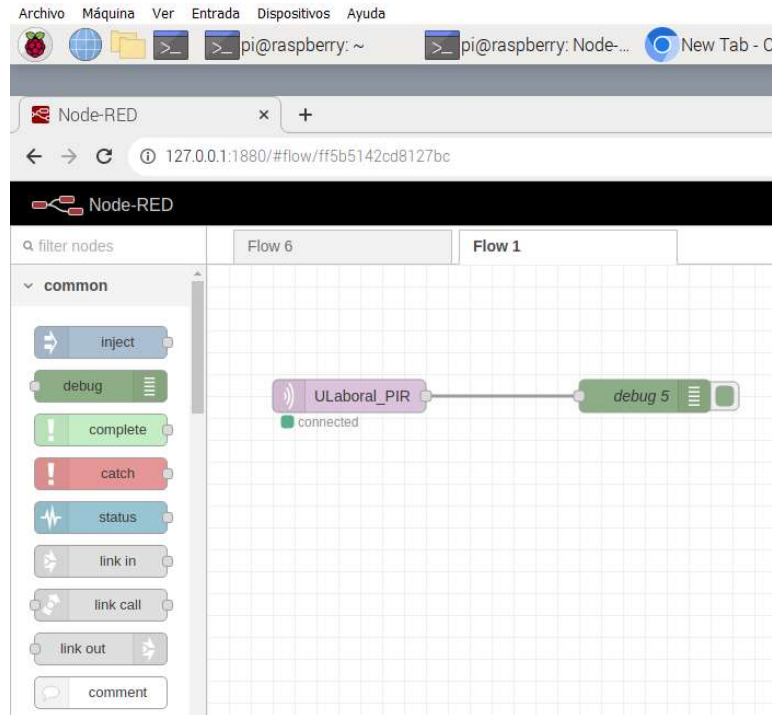
Conexionado:

Conexionado do ESP32, do sensor domótico no pin 22 e dun LED testemuña da detección de presenza no pin 12.



Creación do topic:

Créase en mosquitto.org empregando Node – RED:



Programación:

Programa creado no entorno Visual Code Studio:

```
#include <Arduino.h>
#include <WiFi.h>
#include <PubSubClient.h>

//SSID da WiFi onde queremos conectar
const char* ssid="IES ULaboral";
//contrasinal rede
const char* password="*****";
const char* mqtt_server="test.mosquitto.org";
const char* tema="ULaboral_PIR";
WiFiClient clienteWiFi;
PubSubClient clienteMQTT (clienteWiFi);

int ledPin = 12; // declaramos el led en el pin 12 del arduino
int sensorPin=22; // declaramos el sensor PIR en el pin 11
int val = 0; //variable para asignar la lectura del sensor PIR
boolean estadoAlarma= false;
boolean alarmaRearmada="true";

void setup(){ // put your setup code here, to run once:
  pinMode(ledPin, OUTPUT); //El pin 12 del arduino lo asignamos como salida para el led
  pinMode(sensorPin, INPUT); //El pin 11 lo asignamos como entrada para la señal del sensor
  Serial.begin(9600); //115200// 9600

  for(int i = 0; i > 30; i++) //Utilizamos un for para calibrar el sensor depende del tipo de sensor que utilicemos va a cambiar el tiempo de calibración
  {
    Serial.print("*");
    delay(1000);
  }
  delay(50);

  //WiFi-Inicia conexion WiFi
  Serial.print("Conectando a " + String(ssid) + " ");
  WiFi.begin(ssid,password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print (".");
  }
  Serial.println(" Conectado");
  Serial.println();
  //Fin WiFi

  //Broker-Establecer broker a utilizar
  clienteMQTT.setServer(mqtt_server,1883); //1884??
  //Fin broker
}

void loop() { // put your main code here, to run repeatedly:
  //Establece conexion con broker
  while (!clienteMQTT.connected()) {
```

```

Serial.print("Conectando al broker... ");
if (clienteMQTT.connect("monic")){
    Serial.println();
    Serial.println("Conectado al broker");
} else {
    delay (5000);
}
}

val = digitalRead(sensorPin); //Lee el valor de la variable (val)
if (val == HIGH) //Si detecta que hay movimiento manda activar el led que hay
conectado en el pin 12 del arduino
{
    Serial.print(" H- ");
    digitalWrite(ledPin, HIGH);
    estadoAlarma= true;
    alarmaRearmada= false;
    Serial.println("Alarma activada");
    clienteMQTT.publish (tema,"ON");
}
else //Si la condición anterior no se cumple manda apagar el led //no me gusta el
else if, refresca continuamente??
{
    Serial.print(" L- ");
    digitalWrite(ledPin, LOW);
    estadoAlarma= false;
    alarmaRearmada= true;
    Serial.println("Alarma desactivada");
    clienteMQTT.publish (tema,"OFF");
}

delay (10); //cliente escuchando

clienteMQTT.loop();
}

```

Bibliografía:

Tomás Domínguez Mínguez (2020).
Desarrollo de aplicaciones IoT en la nube para Arduino y ESP8266.
 Marcombo. ISBN:978-84-267-2845-6.