Lo primero es instalar la librería siguiendo.

Abre el  la opción del menú

Programa>Incluir Librería>Gestor de librerías

busca PubSubClient

Graphical user interface, text, application

Description automatically generated

Archivo>Ejemplos>PubSubClient>mqtt\_esp8266.

#include <WiFi.h>

#include <PubSubClient.h>

const char\* ssid = "YOUR\_SSID";

const char\* password = "YOUR\_PASSWORD";

const char\* mqtt\_server = "192.168.1.144 MQTT\_BROKER\_IP";

const string TOPIC\_READ = "**input/LED1**";

const string TOPIC\_SEND = "**output/Sensor1**";

long lastMsg = 0;

int ledPin = 4;

WiFiClient espClient;

PubSubClient client(espClient);

void setup() {

Serial.begin(115200);

randomSeed(42);

setup\_wifi();

* TCP Port: **1883**
* Websocket Port: **8083**

client.setServer(mqtt\_server, 1883);

client.setCallback(callback);

}

void setup\_wifi() {

delay(10);

Serial.println(ssid);

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED) {

delay(500);

Serial.print(".");

}

Serial.println("WiFi connected");

Serial.println("IP address: ");

Serial.println(WiFi.localIP());

}

// RECEIVING A MESSAGE

void callback(char\* topic, byte\* message, unsigned int length) {

Serial.print("Message arrived on topic: ");

Serial.print(topic);

Serial.print(". Message: ");

String messageTemp;

for (int i = 0; i < length; i++) {

Serial.print((char)message[i]);

messageTemp += (char)message[i];

}

Serial.println();

// If MSG received is on topic ‘INPUT/LED1’

// check if the msg is either "on" or "off".

if (String(topic) == TOPIC\_READ) {

int valor = LOW;

if(messageTemp == "on") valor = HIGH;

digitalWrite(ledPin, valor);

}

void reconnect() {

// Loop until we're reconnected

while (!client.connected()) {

Serial.print("Attempting MQTT connection...");

if (client.connect("**ESP8266Client**")) {

Serial.println("connected");

**// Subscribe**

client.subscribe(TOPIC\_SEND);

} else {

Serial.print("failed, rc=");

Serial.print(client.state());

Serial.println(" try again in 5 seconds");

delay(5000);

}

}

}

void loop() {

if (!client.connected()) {

reconnect();

}

client.loop();

long now = millis();

if (now - lastMsg > 5000) {

lastMsg = now;

string mensaje = strSensor();

client.publish(TOPIC\_SEND, mensaje);

}

}

string strConvert()

{

int valor = random(240); // float

char msg[8];

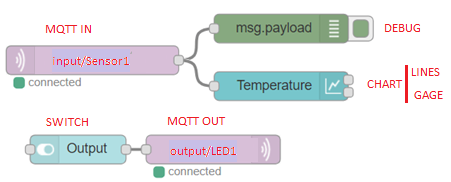
dtostrf(valor, 1, 2, msg);

Serial.println(msg);

return msg;

}

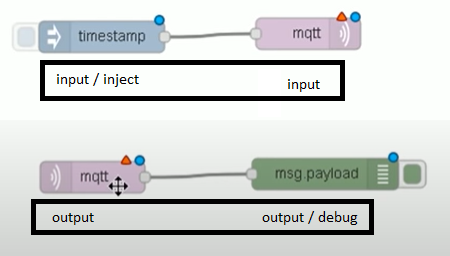
NODE-RED



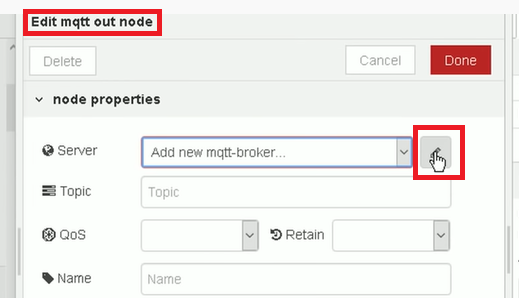
|  |  |
| --- | --- |
| MQTT IN | DEBUG |
| Chart Line Chart | GAGE |

|  |  |
| --- | --- |
| SWITCH | MQTT OUT |

<https://www.youtube.com/watch?v=XDrwgMSQrEY>



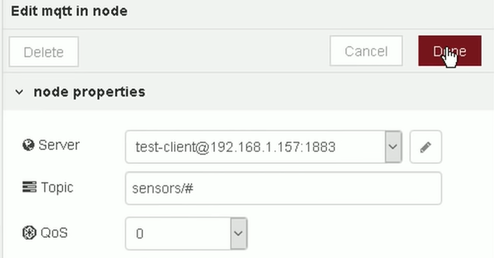
|  |  |
| --- | --- |
| En el timestamp  Payload : se escoge del dropdown “String”  El topico que lee el sensor es  “output/Sensor1”  client.publish("output/sensor1", mensaje); |  |



Asigne la IP de la RPi



El MQTT output



You want to publish a message to an MQTT topic on a broker.

Use the MQTT Output node to publish messages to a topic.

A picture containing text

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated