

Compilando o MQTT:

Primeiro passo, instalar a biblioteca "PubSubClient". Vá para a aba Sketch, na sua IDE do Arduino, clique em incluir biblioteca, gerenciar bibliotecas, digite o nome dela e por fim, clique em instalar. Inclua no programa as seguintes bibliotecas: `#include <ESP8266WiFi.h> #include <PubSubClient.h> #include <ESP8266mDNS.h> #include <ArduinoOTA.h>`.

O programa em questão serve para se conectar ao MQTT com um servidor e tópicos específicos. O MQTT é um protocolo de comunicação e troca de mensagens fundamentado no modelo Publicador (publica a mensagem)-Subscritor (recebe a mensagem).

Na prática, pudemos instalar um aplicativo de MQTT dashboard e nos conectar ao tópico passado em aula para entrarmos no mesmo canal de comunicação e nos inscrever ou publicar alguma palavra. Já no código, após incluir as bibliotecas, definimos o tópico, o clientId, que deve ser único para cada usuário, os pinos do NodeMCU e o usuário e senha do MQTT, que podem ficar em branco, uma vez que o servidor é gratuito.

Além disso, o programador deve pôr o nome e senha da sua rede Wi-fi e definir o broker (servidor) a ser utilizado, assim como a porta (no caso, a 1883). Em seguida, inicia-se a serialização com a velocidade padrão de 115200 e a função "initWifi()" é chamada, o programa conecta-se ao Wi-fi ou fica em looping até conseguir a conexão, caso não consiga de primeira.

Depois, é chamada a função "initOTA()", na qual é necessário setar a senha do seu wi-fi no programa e o nome da porta de rede, a qual será criada. Na sequência, o programa se conecta com o ArduinoOTA e, caso surja algum erro de autenticação, o erro será printado na tela, assim como o progresso do OTA.

Adiante, inicia-se o MQTT, setando o broker e chamando a função "mqtt_callback", a qual permite exibir na tela a mensagem digitada e, caso a mensagem seja "ON" é escrito "HIGH" na variável D1. Caso seja "OFF", é escrito "LOW" nessa mesma variável.

A seguir, é chamada a última função, InitOutput() que envia o conteúdo da variável D1 para o Output, que controla o acender e apagar do led no NodeMCU, através dos comandos HIGH (acende) e LOW (apaga). Abaixo, é possível conferir o monitor serial, em 2 momentos diferentes.

Prints do monitor serial, após o programa ter sido compilado:

