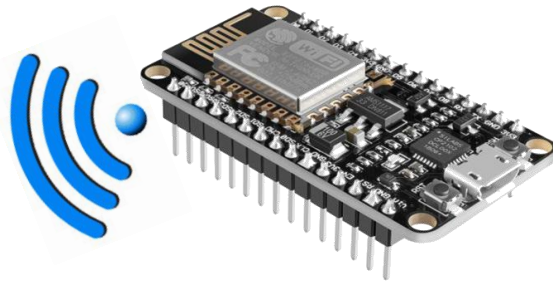




Efrén Juárez

NodeMCU ESP8266

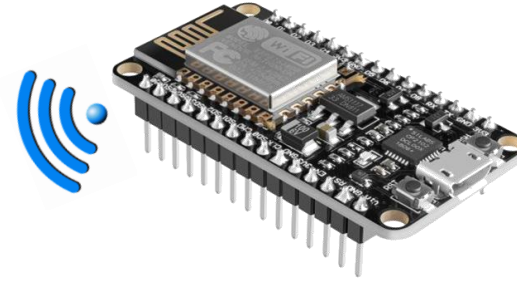
Conexión WiFi



Station (STA)



Station (STA)



Station (STA)



Punto de acceso inalámbrico (AP)

Nombre de Red

SSID (Service Set Identifier)

Código 1

```
#include <ESP8266WiFi.h>

String ssid      = "ejcuthh";
String password = "12345678";

byte cont = 0;
byte max_intentos = 50;

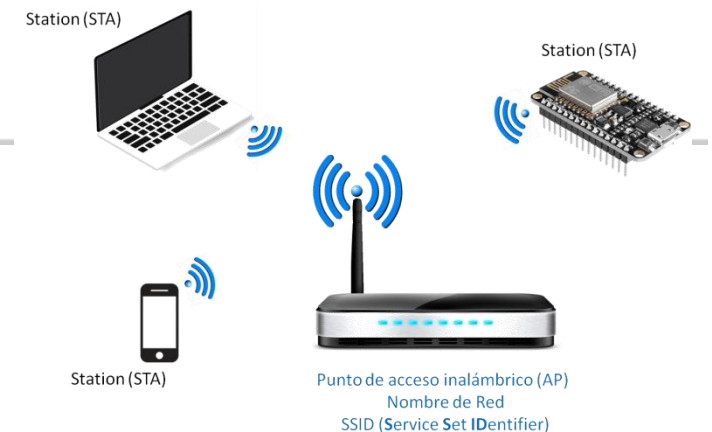
void setup() {
  // Inicia Serial
  Serial.begin(115200);
  Serial.println("\n");

  // Conexión WIFI
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED and cont < max_intentos) { //Cuenta hasta 50
    cont++;
    delay(500);
    Serial.print(".");
  }
}
```

```
Serial.println("");

if (cont < max_intentos) { //Si se conectó
  Serial.println("*****");
  Serial.print("Conectado a la red WiFi: ");
  Serial.println(WiFi.SSID());
  Serial.print("IP: ");
  Serial.println(WiFi.localIP());
  Serial.print("macAddress: ");
  Serial.println(WiFi.macAddress());
  Serial.println("*****");
}
else { //No se conectó
  Serial.println("-----");
  Serial.println("Error de conexion");
  Serial.println("-----");
}
}

void loop() {
}
```



WiFi.status()

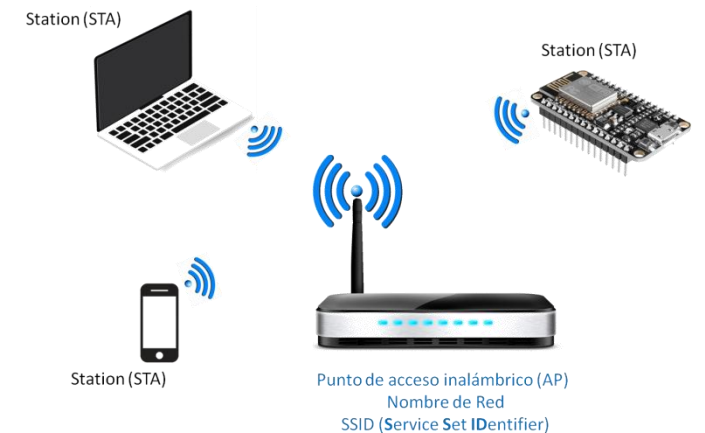
Esta función devuelve los siguientes códigos para describir lo que está sucediendo con la conexión WiFi:

- 0 : `WL_IDLE_STATUS` cuando WiFi está en proceso de cambio de estado
- 1 : `WL_NO_SSID_AVAIL` en caso de que no se pueda alcanzar el SSID configurado
- 3 : `WL_CONNECTED` después de establecida una conexión exitosa
- 4 : `WL_CONNECT_FAILED` si la contraseña es incorrecta
- 6 : `WL_DISCONNECTED` si el módulo no está configurado en modo estación



Parpadear Led

- `#include <Ticker.h>`
- `Ticker tic_WifiLed;`
- `void parpadeoLedWifi(){`
 - `byte estado = digitalRead(ledWifi);`
 - `digitalWrite(ledWifi,!estado);`
 - `}`
- `tic_WifiLed.attach(0.2,parpadeoLedWifi);`
- `tic_WifiLed.detach();`



```

#include <ESP8266WiFi.h>
#include <Ticker.h>

#define ledWifi D4

String ssid      = "ejcuthh";
String password = "12345678";

Ticker tic_WifiLed;

byte cont = 0;
byte max_intentos = 50;

void parpadeoLedWifi(){
    byte estado = digitalRead(ledWifi);
    digitalWrite(ledWifi,!estado);
}

void setup() {
    // Inicia Serial
    pinMode(ledWifi,OUTPUT);
    Serial.begin(115200);
    Serial.println("\n");

    tic_WifiLed.attach(0.2,parpadeoLedWifi);

    // Conexión WIFI
    WiFi.begin(ssid, password);
    while (WiFi.status() != WL_CONNECTED and cont < max_intentos) { //Cuenta hasta
        cont++;
        delay(500);
        Serial.print(".");
    }
}

```

Código 2



```

Serial.println("");
if (cont < max_intentos) { //Si se conectó
    Serial.println("*****");
    Serial.print("Conectado a la red WiFi: ");
    Serial.println(WiFi.SSID());
    Serial.print("IP: ");
    Serial.println(WiFi.localIP());
    Serial.print("macAdress: ");
    Serial.println(WiFi.macAddress());
    Serial.println("*****");
}
else { //No se conectó
    Serial.println("-----");
    Serial.println("Error de conexion");
    Serial.println("-----");
}

tic_WifiLed.detach();
digitalWrite(ledWifi,HIGH);
}

void loop() {
}

```



Efrén Juárez

NodeMCU ESP8266 Conexión WiFi

