

LABORATORIO 6 – CONFIGURACIÓN DE ACCESS POINT

Material:

NodeMCU y cable USB

Ejercicio 1. El Nodemcu como Acces Point (Punto de Acceso - AP)

En este laboratorio se configura un punto de acceso inalámbrico con el NodeMCU para lo cual se escribirá el SSID y PASSWORD con el fin de que solo los host que conozcan el nombre de la red y password se conecten a el. La configuración de stos dos parámetros se realiza a través de: **WiFi.softAP("PRUEBA", "ricardo901");**

Recuerde verificar la velocidad de transmisión entre el NodMCU y el IDE instalado en el PC ya que es útil verificar la dirección de nuestro AP a través de **Serial.println(WiFi.softAPIP());**

En este primer ejemplo se visualizará el número de conexiones a nuestro Access point a través de: **WiFi.softAPgetStationNum());**

La conexión a Wi-Fi es proporcionada por un punto de acceso (AP), que actúa como hub para una o más estaciones.

Un punto de acceso suele estar integrado con un enrutador para proporcionar acceso desde la red Wi-Fi a Internet. Cada punto de acceso es reconocido por un SSID (Service Set Identifier), que esencialmente es el nombre de red que se selecciona cuando se conecta un dispositivo (estación) a la red Wi-Fi.

El módulo ESP8266 puede funcionar como una estación, por lo que podemos conectarlo a la red Wi-Fi. También puede funcionar como un punto de acceso (soft-AP), para establecer su propia red Wi-Fi. Por lo tanto, podemos conectar otras estaciones a dicho módulo ESP. El ESP8266 también es capaz de operar como estación o como de punto de acceso. Esto proporciona la posibilidad de construir, p.e. redes en malla.

En el NodeMCU con el ESP8266 el número máximo de estaciones conectadas al soft-AP es de cinco.

```
#include <ESP8266WiFi.h>
```

```
void setup(){  
  Serial.begin(9600);  
  WiFi.softAP("PRUEBA", "ricardo901");  
  Serial.println(WiFi.softAPIP());  
}
```

```
void loop (){  
  Serial.println("Estaciones conectadas:");  
  Serial.println(WiFi.softAPgetStationNum());  
  delay(3000);  
}
```

Verifique la configuración conectando un dispositivo al NodeMCU por WiFi. Puede conectar el PC o su smatphone, para ello haga una escaneo de las redes WiFi, identifique el SSID de la red que usted creo, escriba el password cuando el dispositivo quiera conectarse a la red.

Si todo está bien, el dispositivo deberá conectarse a nuestra red creada y podrá realizar ping a la dirección por defecto del NodeMCU, 192.168.4.1

Como trabajo en clase, repita el procedimiento pero cambie la dirección IP por una dirección cualquiera clase

B. Recuerde hacer el cambio de IP en el pc o Smartphone con el que quieras probar el dispositivo. Realice ping de nuevo a la dirección IP especificada en el código para el NodeMCU.

Ejercicio 2. Servidor WEB con librería **ESP8266WebServer.h**

Un servidor Web es un programa que utiliza el protocolo de transferencia de hiper texto, HTTP (Hypertext Transfer Protocol), para servir los archivos que forman páginas Web a los usuarios, en respuesta a sus solicitudes, que son reenviados por los clientes HTTP de sus computadoras. Las computadoras y los dispositivos dedicados como en el caso del NodeMCU también pueden denominarse servidores Web.

Generalmente se usa el protocolo HTTP para estas comunicaciones, perteneciente a la capa de aplicación del modelo OSI.

Un servidor web es un programa que sirve datos en forma de Páginas Web, hipertextos o páginas HTML (HyperText Markup Language): textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de sonidos. La comunicación de estos datos entre cliente y servidor se hace por medio un protocolo, concretamente del protocolo Http. Con esto, un servidor Web se mantiene a la espera de peticiones HTTP, que son ejecutadas por un cliente HTTP; lo que solemos conocer como un Navegador Web. A modo de ejemplo: al teclear (<http://www.cnice.mec.es>) en un navegador, éste realizará una petición HTTP al servidor que tiene asociada dicha URL.

El servidor responde al cliente enviando el código HTML de la página; el navegador cuando recibe el código, lo interpreta y lo muestra en pantalla. El Cliente es el encargado de interpretar el código HTML, es decir, de mostrar las fuentes, los colores y la disposición de los textos y objetos de la página. El servidor se encarga de transferir el código de la página sin llevar a cabo ninguna interpretación de la misma.

La librería ESP8266WebServer.h se utiliza para simplificar la creación de un servidor web. En el siguiente ejemplo tenemos un código donde el Node MCU está configurado como AP y a su vez es un servidor WEB.

```
#include <ESP8266WiFi.h>
#include <WiFiClient.h>
#include <ESP8266WebServer.h>
```

```
ESP8266WebServer server(80);
```

```
void setup() {
  delay(1000);
```

```
  WiFi.softAP("PRUEBA", "ricardo901");
```

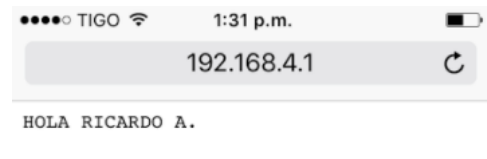
```
  server.on("/", [](){
    server.send(200, "text/plain", "HOLA RICARDO A.");
  });
```

```
  server.begin();
}
```

```
void loop() {
  server.handleClient();
}
```

Después de subir el sketch al NodeMCU podemos verificar la conectividad a través de ping o podemos abrir nuestro Web Browser para ver el texto en html **HOLA RICARDOA.** a través de la dirección por defecto 192.168.4.1.

Observe en la siguiente figura como sería el resultado:



Como trabajo de clase en el sketch anterior incluya los comandos necesarios que nuestro servidor IP tenga una dirección fija clase A y también que se visualice en el monitor serial la dirección IP de nuestro servidor IP y el número de clientes conectados.

TRABAJO PARA LA CASA:

Realice un servidor WEB utilizando la librería ESP8266WebServer.h configurando el NodeMCU como estación que reciba direccionamiento IP a través de DHCP y que el cliente vea en el web browser su nombre completo.

De igual forma que se pueda visualizar en el monitor serial la IP del NodeMCU y la potencia de la señal RSSI (Received Signal Strength Indication).