CAPITULO III

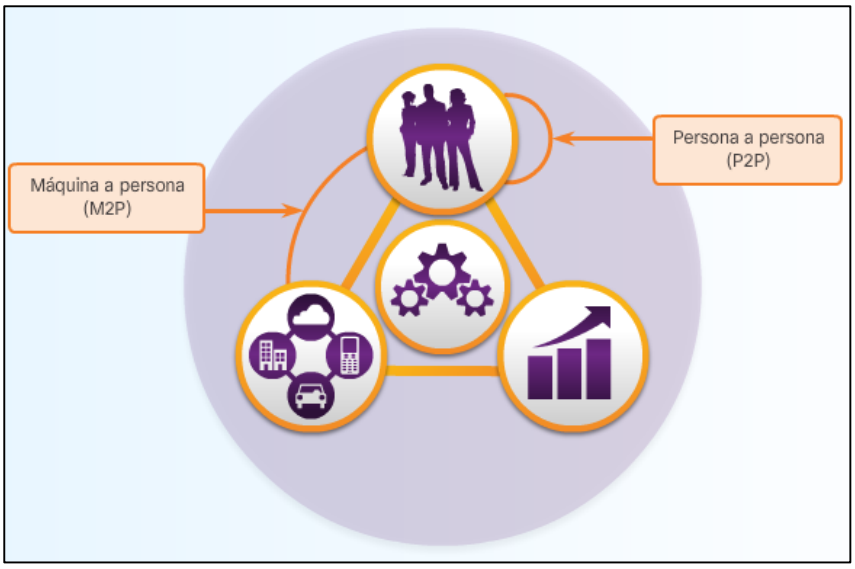
3.1 LAS PERSONAS DEBEN ESTAR CONECTADAS

Las personas son la figura central en cualquier sistema económico: Interactúan como productores y consumidores en un entorno cuyo propósito es mejorar el bienestar satisfaciendo las necesidades humanas. Ya sean las conexiones de persona a persona (P2P), de máquina a persona (M2P) o de máquina a máquina (M2M), todas las conexiones y los datos generados a partir de ellas se utilizan para aumentar el valor para las personas.

3.2 LOS PROCESOS COMO PILARES

Los procesos desempeñan una función fundamental en la manera en que los otros pilares —los objetos, los datos y las personas— operan juntos para ofrecer valor en el mundo conectado de IdT.

Internet revolucionó la manera en que las empresas administran sus cadenas de suministros y la forma en que compran los consumidores. Muy pronto, podremos acceder a detalles de procesos que nunca antes habíamos podido ver. Esto proporcionará oportunidades para hacer que estas interacciones sean más rápidas y simples.



Los procesos facilitan las interacciones entre las personas, los objetos y los datos. En la actualidad, IdT los une mediante la combinación de conexiones de máquina a máquina (M2M), de máquina a persona (M2P) y de persona a persona (P2P)

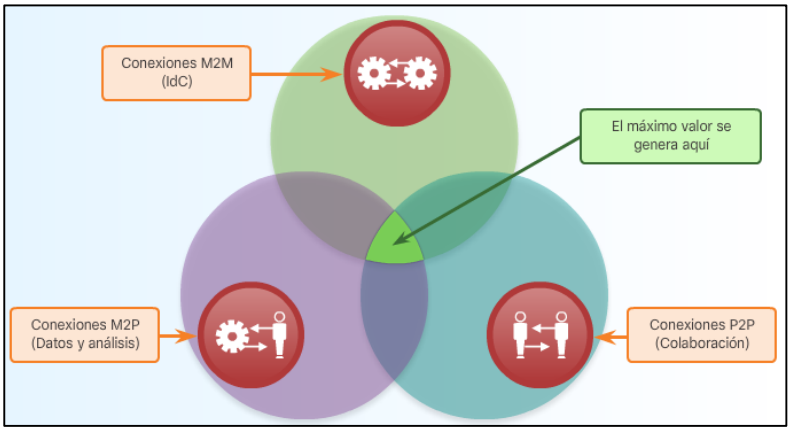
Las conexiones de máquina a máquina (M2M) tienen lugar cuando se transfieren datos de una máquina u “objeto” a otro a través de una red. Las máquinas incluyen sensores, robots, computadoras y dispositivos móviles. Estas conexiones M2M a menudo se denominan “Internet de las cosas”.

Un ejemplo de M2M es un automóvil conectado que emite una señal para informar que un conductor ya casi llega a casa, lo que le indica a la red doméstica que ajuste la temperatura y la iluminación del hogar.

Las conexiones de máquina a persona (M2P) tienen lugar cuando la información se transfiere entre una máquina (como una computadora, un dispositivo móvil o un letrero digital) y una persona, como se muestra en la figura. Cuando una persona obtiene información de una base de datos o realiza un análisis complejo, tiene lugar una conexión M2P. Estas conexiones M2P facilitan el movimiento, la manipulación y la información de datos de máquinas para ayudar a las personas a que tomen decisiones fundadas. Las acciones que las personas realizan según sus razonamientos fundados completan un ciclo de realimentación de IdT.

3.3 CONEXIONES P2P

Las conexiones de persona a persona (P2P) tienen lugar cuando la información se transfiere de una persona a otra. Las conexiones P2P se producen cada vez más a través de video, dispositivos móviles y redes sociales. Con frecuencia, estas conexiones P2P se denominan “colaboración”.



Como se muestra en la ilustración, el valor más alto de IdT se obtiene cuando el proceso facilita la integración de las conexiones M2M, M2P y P2P.

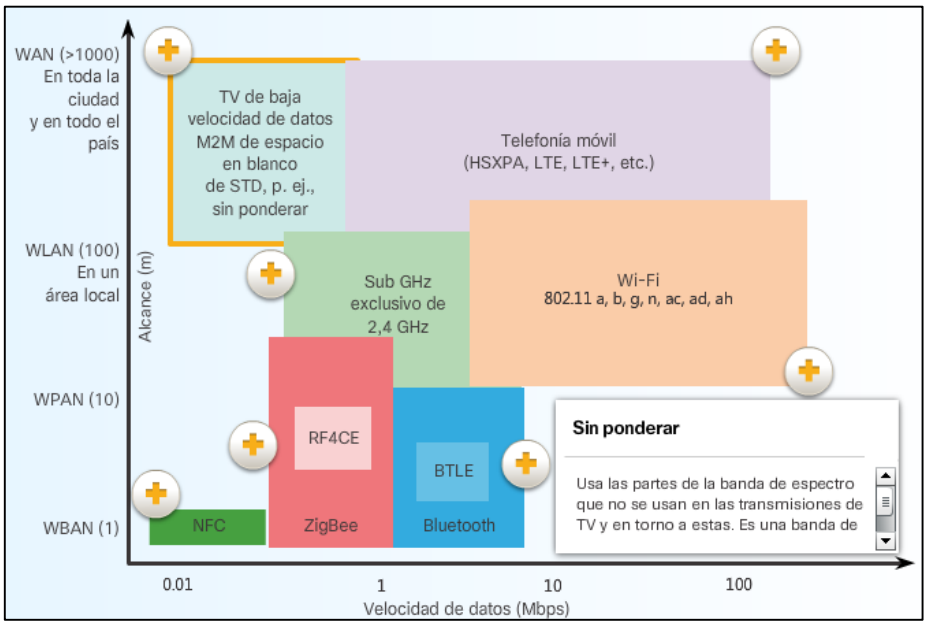
3.4 CONECTIVIDAD ENTRE REDES

En la capa de acceso de red, los dispositivos se pueden conectar a la red en una de dos formas: por cable o de manera inalámbrica.

El protocolo cableado más implementado es el protocolo Ethernet. Ethernet utiliza un paquete de protocolo que permite que los dispositivos de red se comuniquen a través de una conexión LAN cableada. Una LAN Ethernet puede conectar dispositivos con diferentes tipos de medios de cableado.

Actualmente, existen varios protocolos de red inalámbrica disponibles. Las características de estos protocolos varían en gran medida. En la figura siguiente, se proporcionan algunos protocolos inalámbricos comunes y se muestra una representación visual de la ubicación de estos protocolos en el espectro de clasificación. Observe que un protocolo puede abarcar varias clasificaciones.

Además de estos protocolos, hay otros protocolos de capa de acceso de red disponibles en forma inalámbrica y por cable.



3.5 Acceso de red para los objetos actualmente no conectados

Para que los objetos con muy pocos requisitos de energía envíen información a través de la red, existen varios protocolos de comunicación inalámbrica de corto alcance. En algunos casos, estos protocolos no tienen IP habilitado y deben reenviar información a un dispositivo conectado con IP habilitado, como un controlador o una gateway. Por ejemplo, un dispositivo que no usa TCP/IP se puede comunicar con otro dispositivo que no usa este estándar, como el estándar 802.15 del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE). En este estándar se encuentra el conocido Bluetooh y otros como ZigBee.

Bluetooth

El protocolo Bluetooth se suele usar entre dispositivos que están a distancias cortas, como la conexión de un smartphone a auriculares con Bluetooth habilitado, o un teclado inalámbrico con Bluetooth habilitado conectado a un dispositivo informático.

ZigBee

ZigBee es otro ejemplo de un paquete de protocolo 802.15 que usa el emparejamiento entre un origen y un destino específicos. Un ejemplo de esto es entre un sensor para puerta y un sistema de seguridad que envía una alerta cuando se abre la puerta.

NFC

La transmisión de datos en proximidad (NFC) es un estándar para la comunicación entre objetos que están a muy poca distancia, generalmente a pocos centímetros. Por ejemplo, NFC funciona en el punto de venta entre una etiqueta RFID y el lector.

6LoWPAN

6LoWPAN surgió de la necesidad de incluir dispositivos de extremadamente baja energía con capacidades de procesamiento limitadas como parte de IdC; por ejemplo, los medidores inteligentes en una red pequeña.

3.6 WIFI

El wifi es un mecanismo de conexión de dispositivos electrónicos de forma inalámbrica. Dentro de estos dispositivos se encuentra la placa o tarjeta de desarrollo NodeMCU.

Wi-Fi es una marca de la Alianza Wi-Fi, la organización comercial que adopta, prueba y certifica que los equipos cumplen con los estándares 802.11 de la IEEE.

Los estándares IEEE 802.11b, IEEE 802.11g e IEEE 802.11n disfrutan de una aceptación internacional trabajando a frecuencias de 2,4 GHz y 5GHz y con velocidades de hasta 11 Mbit/s, 54 Mbit/s y 300 Mbit/s, respectivamente.

En la actualidad ya se maneja también el estándar IEEE 802.11ac, conocido como WIFI 5, que opera en la banda de 5 GHz. La banda de 5 GHz ha sido recientemente habilitada y, además, no existen otras tecnologías (Bluetooth, microondas, ZigBee) que la estén utilizando, por lo tanto existen muy pocas interferencias. Su alcance es algo menor que el de los estándares que trabajan a 2,4 GHz (aproximadamente un 10 %).

Las redes WiFi utilizan la banda ISM (Industrial, Scientific and Medical) que son bandas reservadas internacionalmente para uso no comercial de radiofrecuencia electromagnética en áreas industrial, científica y médica. En la actualidad estas bandas han sido popularizadas por su uso en comunicaciones WLAN (e.g. Wi-Fi) o WPAN (e.g. Bluetooth).

IEEE 802.11ac (también conocido como WiFi 5G o WiFi Gigabit) es una mejora a la norma IEEE 802.11n, se ha desarrollado entre el año 2011 y el 2013, y finalmente aprobada en enero de 2014. El estándar consiste en mejorar las tasas de transferencia hasta 433 Mbit/s por flujo de datos, consiguiendo teóricamente tasas de 1.3 Gbit/s empleando 3 antenas. Opera dentro de la banda de 5 GHz, amplía el ancho de banda hasta 160 MHz (40 MHz en las redes 802.11n), utiliza hasta 8 flujos MIMO e incluye modulación de alta densidad (256 QAM).

Cada red inalámbrica tiene un nombre o SSID (Service Set Identifier) que es una secuencia de 0-32 octetos incluida en todos los paquetes de la red inalámbrica para identificarlos como parte de esa red. El SSID se configura dentro de los dispositivos que se consideran parte de la red, y se transmite en los paquetes. Los receptores ignoran paquetes inalámbricos de redes con un SSID diferente.

Desde junio de 2014 las empresas de microcontroladores iniciaron a implementar sistemas de interconexión WiFi de esta forma Texas Instruments presentó el primer microcontrolador ARM Cortex-M4 con una MCU dedicada Wi-Fi embebida.

Otro ejemplo es el del Arduino MKR1000 que usa el microcontrolador ATSAMW25 (http://www.atmel.com/devices/ATSAMW25.aspx) con un módulo wifi y otro de criptoauteticación incluido.

El ESP8266 es un chip Wi-Fi de bajo costo producida por el fabricante chino Espressif Systems, con sede en Shanghai. El chip primero llegó a la atención de los fabricantes occidentales en agosto de 2014 con el módulo ESP-01. Este pequeño módulo permite a los microcontroladores conectarse a una red Wi-Fi y realizar conexiones TCP/IP sencillas.

A finales de octubre de 2014, Espressif lanzó un kit de desarrollo de software (SDK) que permite programar el chip, eliminando la necesidad de un microcontrolador por separado. Desde entonces, ha habido muchos lanzamientos oficiales de SDK.

Otros SDK de código abierto para el ESP8266:

- NodeMCU: un firmware basado en Lua.
- Arduino: un firmware basado en C ++. Este núcleo permite que la CPU ESP8266 y sus componentes Wi-Fi sean programados como cualquier otro dispositivo Arduino. El Arduino Core ESP8266 está disponible a través de GitHub: https://github.com/esp8266/Arduino y cuyo reference es https://github.com/esp8266/Arduino/blob/master/doc/reference.md
- MicroPython: una implementación de Python para dispositivos embebidos a la plataforma ESP8266.
- ESP8266 BASIC: Un intérprete básico de código abierto específicamente diseñado para el Internet de las cosas.
- Mongoose Firmware: Un firmware de código abierto con servicio gratuito en la nube: https://github.com/cesanta/mongoose-iot

3.7 El módulo ESP8266

El módulo WIFI ESP8266 incluye toda la electrónica necesaria para la comunicación Radio Frecuencia en la banda WiFi, así como la pila TCP/IP y que se comunica con nosotros a través de un puerto serie.

Dentro de la gran cantidad de usos para este módulo cabe destacar los siguientes:

- Electrodomésticos conectados.
- Automatización del hogar.
- Automatización de la industria.
- Cámaras IP.
- Redes de sensores.
- Wereables.
- IoT (Internet of Things o Internet de las Cosas)
- IIoT (Industrial Internet of Things o Internet de las Cosas para el sector Industrial)

EL ESP8266 no tiene ROM y usa una ROM externa SPI y soporta hasta 16MB y los podemos encontrar en diferentes encapsulados y placas como los de la siguiente figura:

Existen varias placas NodeMCU. Para empezar a comparar las placas NodeMCU que hay en el mercado, en primer lugar, debemos saber que actualmente se está comercializando la SEGUNDA GENERACIÓN (1.0), conocida como V2 ó V3 en función del fabricante. Estas placas integran el procesador ESP8266-12E. La PRIMERA GENERACIÓN (0.9), conocida como V1, integraba el modelo ESP8266-12.

En la actualidad ya está en el mercado la TERCERA GENERACIÓN de placas NodeMcu que integran el nuevo procesador ESP32. El ESP32 no solo es más rápido sino también está diseñado pensando en que sea un microcontrolador para el IoT. En este caso, utiliza un procesador Xtensa Dual-Core LX6 de 32 bits a 160 ó 240 MHz. El usar dos núcleos permite dedicar uno de ellos a la comunicación IP y WiFi y el otro al resto de procesos. Se resuelve así una de las dificultades más importantes que imponía la arquitectura del ESP8266.

Tiene una memoria RAM de 520 kB, accesible por ambos procesadores y puede utilizar memoria RAM externa adicional de hasta 8 MB.

Otras diferencias las podemos apreciar en el siguiente cuadro:

Característica	ESP8266	ESP32
Procesador	Tensilica LX106 32 bit a 80 MHz (hasta 160 MHz)	Tensilica Xtensa LX6 32 bit Dual-Core a 160 MHz (hasta 240 MHz)
Memoria RAM	80 kB (40 kB disponibles)	520 kB
Memoria Flash	Hasta 4 MB	Hasta 16 MB
ROM	No	448 kB
Alimentación	3.0 a 3.6 V	2.2 a 3.6 V
Rango de temperaturas	-40°C a 125°C	-40°C a 125°C
Consumo de corriente	80 mA (promedio). 225 mA máximo	80 mA (promedio). 225 mA máximo
Consumo en modo sueño profundo	20 uA (RTC + memoria RTC)	2.5 uA (10 uA RTC + memoria RTC)
Coprocesador de bajo consumo	No	Sí. Consumo inferior a 150 uA
WiFi	802.11 b/g/n (hasta +20 dBm) WEP, WPA	802.11 b/g/n (hasta +20 dBm) WEP, WPA
Soft-AP	Sí	Sí

3.8 SEGURIDAD EN WIFI

Existen varias alternativas para garantizar la seguridad de estas redes. Las más comunes son la utilización de protocolos de cifrado de datos para los estándares wifi como el WEP, el WPA, o el WPA2 que se encargan de codificar la información transmitida para proteger su confidencialidad, proporcionados por los propios dispositivos inalámbricos. La mayoría de las formas son las siguientes:

- WEP, cifra los datos en su red de forma que sólo el destinatario deseado pueda acceder a ellos. Los cifrados de 64 y 128 bits son dos niveles de seguridad WEP. WEP codifica los datos mediante una “clave” de cifrado antes de enviarlo al aire. Este tipo de cifrado no está recomendado debido a las grandes vulnerabilidades que presenta ya que cualquier cracker puede conseguir sacar la clave, incluso aunque esté bien configurado y la clave utilizada sea compleja.
- WPA: presenta mejoras como generación dinámica de la clave de acceso. Las claves se insertan como dígitos alfanuméricos.
- WPA2 (estándar 802.11i): que es una mejora relativa a WPA. En principio es el protocolo de seguridad más seguro para Wi-Fi en este momento. Sin embargo requieren hardware y software compatibles, ya que los antiguos no lo son. Utiliza el algoritmo de cifrado AES (Advanced Encryption Standard).
- IPSEC (túneles IP) en el caso de las VPN y el conjunto de estándares IEEE 802.1X, que permite la autenticación y autorización de usuarios.

- Filtrado de MAC, de manera que solo se permite acceso a la red a aquellos dispositivos autorizados teniendo en cuenta su dirección MAC. Es lo más recomendable si solo se va a usar con los mismos equipos, y si son pocos.
- Ocultación del punto de acceso: se puede ocultar el el SSID de manera que sea invisible a otros usuarios.

3.9 Potencia en redes WIFI

El dBm (a veces también dBmW o decibelio-milivatio) es una unidad de medida de potencia expresada en decibelios (dB) relativa a un milivatio (mW). Se utiliza en redes WiFi, de radio, microondas y fibra óptica como una medida conveniente de la potencia absoluta a causa de su capacidad para expresar tanto valores muy grandes como muy pequeñas en forma corta.

Los valores aproximados de potencia en una red WiFi en dBm son:

- -40 a -60: señal idónea con tasas de transferencia estables. dependiendo.
- -60: enlace bueno; ajustando TX y basic rates se puede lograr una conexión estable al 80%.
- -70: enlace normal -bajo; es una señal medianamente buena, aunque se pueden sufrir problemas con lluvia y viento.
- -80: es la señal mínima aceptable para establecer la conexión; puede ocurrir caídas, que se traducen en corte de comunicación (pérdida de llamada, perdida de datos, mensajes (sms) corruptos (ilegibles).

3.10 TUNIOT

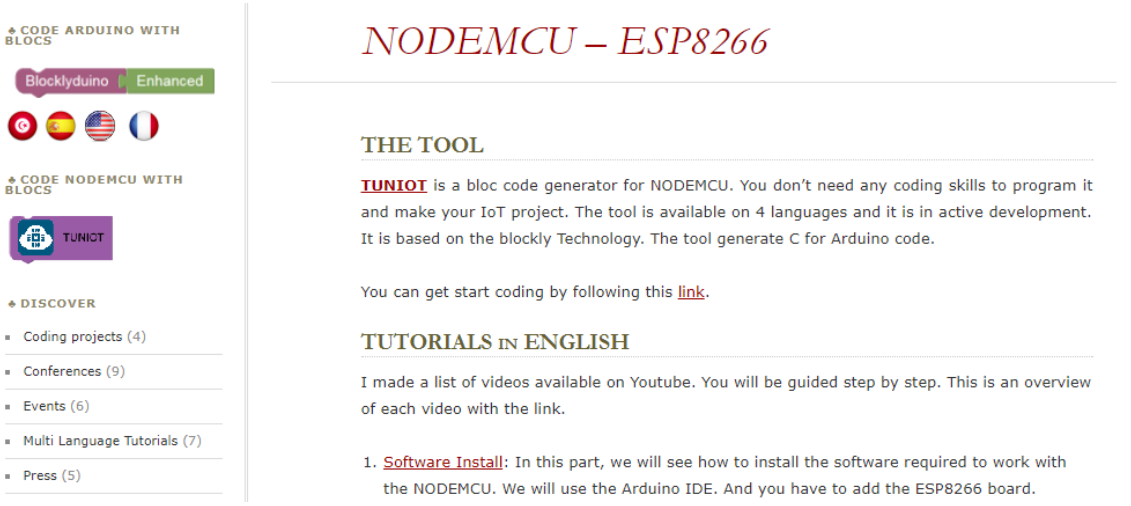
TUNIOT es un generador de código de bloque para NODEMCU. Usted no necesita ninguna habilidad de codificación para programarlo y hacer su proyecto IoT. La herramienta está disponible en 4 idiomas y está en desarrollo activo. Se basa en la tecnología en bloque. La herramienta genera su código para ser llevado automáticamente al IDE de Arduino

Esta herramienta cuenta con tutoriales desde como instala el IDE de arduino hasta como realizar códigos para poner a trabajar el NodeMCU con proyectos IoT.

En esta página encuentras el tutorial paso a paso para manejarlo:

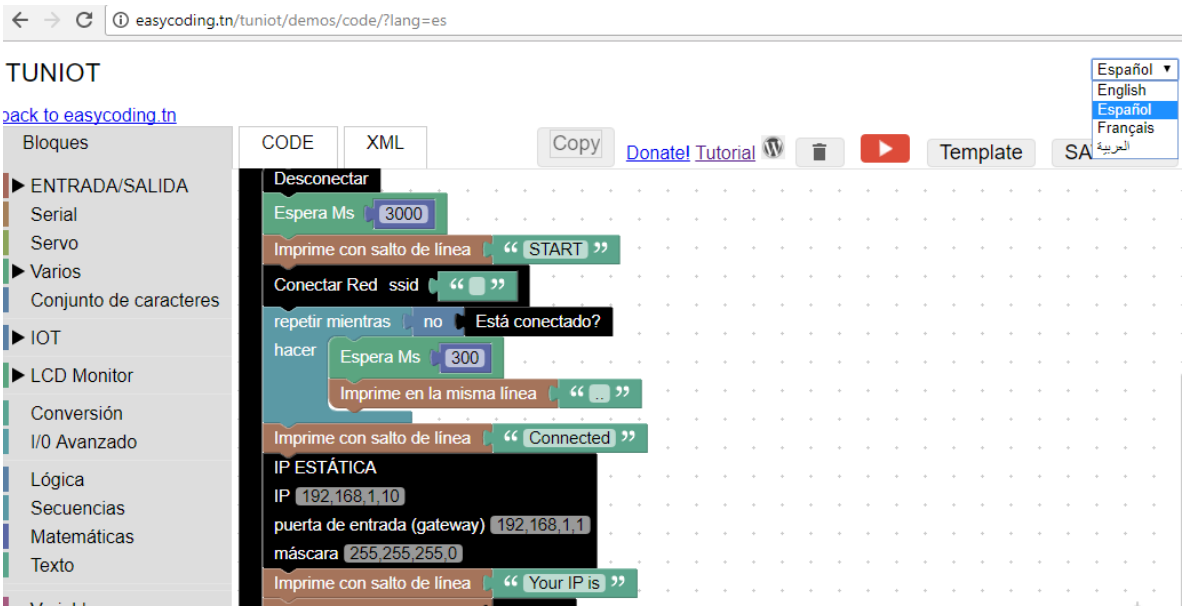
http://easycoding.tn/index.php/nodemcu/

https://www.youtube.com/watch?v=HBSH_x6J1IY&index=2&list=PLfPtpZzK2Z_Qy2ZbbzvWa58cKKOisMUZ1



Adel Kassah, un profesor de informática en la escuela secundaria en Túnez buscando promover la cultura de programación en su país ha creado el TUNIOT, el cual facilita la programación de la placa NODEMCU o cualquier ESP8266.

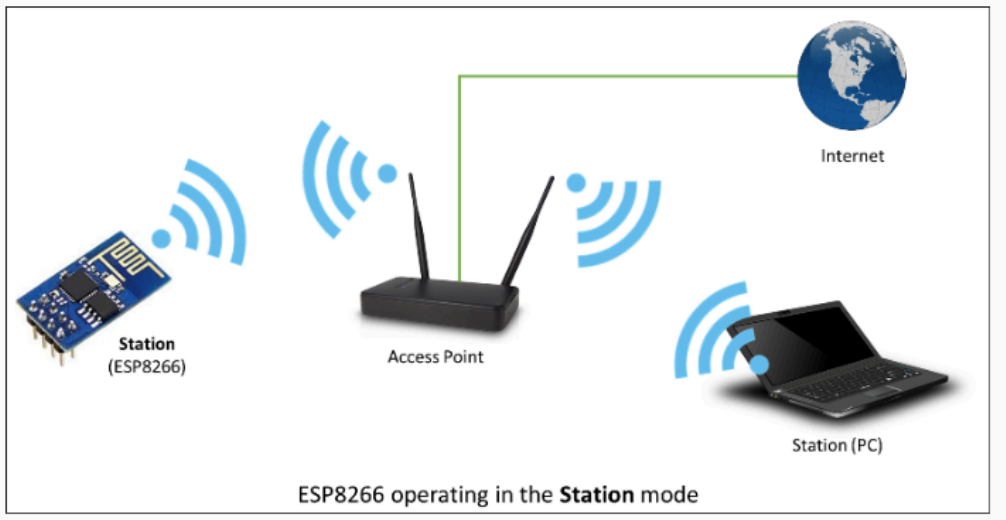
El TUNIOT es una plataforma que sirve para programar en bloques de una forma muy sencilla para después pasarlo al IDE de arduino. Recuerda que este generador de código por bloque es On-line y no necesitas descargar nada pero si debes tener instalado el IDE de arduino con las librerías actualizadas y que reconozca el NodeMCU o ESP8266.



3.11 CONECTANDOSE A UNA RED WIFI MODO ESTACIÓN.

Modo estación (STA): El modo STATION es aquel en el que el dispositivo se conecta a un router para obtener conexión a internet, captando del router los parámetros necesarios para conectarse. El modo Estación (STA) se utiliza para conectar el módulo ESP a una red Wi-Fi establecida por un punto de acceso en el que previamente debemos seleccionar una red visible de nuestro entorno y proporcionar la clave de seguridad.

El comando utilizado en este caso es : **WiFi.begin(ssid, password)**



ESP8266 operating in the **Station** mode

Recuerde que si configuramos el modulo o placa NodeMCU como cliente tenemos que especificar el identificador de red SSID y la clave o password del Acces point a donde nos deseemos conectar.

WiFi.status() es una función que nos devuelve un entero y que nos muestra el estado de la conexión. Los valores que se pueden obtener son:

- 0 : **WL_IDLE_STATUS** cuando el Wi-Fi está en proceso de cambiar de estado
- 1 : **WL_NO_SSID_AVAIL** en caso de que el SSID configurado no pueda ser alcanzado
- 3 : **WL_CONNECTED** después de establecer una conexión satisfactoriamente
- 4 : **WL_CONNECT_FAILED** si la contraseña es incorrecta
- 6 : **WL_DISCONNECTED** si el módulo no está configurado en el modo de estación
-

Una vez conectados, se puede adquirir por defecto la dirección DHCP que nos coloca el Gateway.

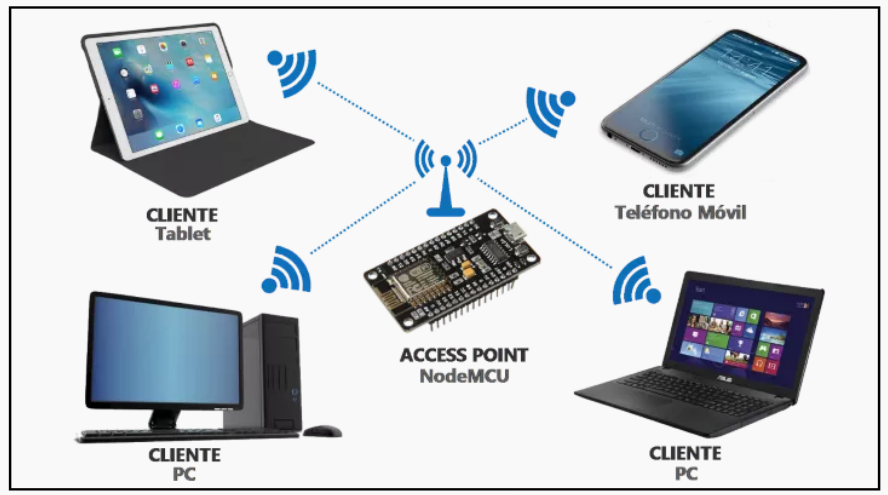
En caso de que se pierda la conexión, ESP8266 se reconectará automáticamente al último punto de acceso utilizado, una vez que esté de nuevo disponible. Lo mismo ocurre en el reinicio del módulo. Esto es posible porque ESP guarda las credenciales del último punto de acceso utilizado en la memoria flash (no volátil).

Para mas información:

http://arduino-esp8266.readthedocs.io/en/latest/esp8266wifi/station-class.html

3.12 NODEMCU ESP8266 COMO ACCESSS POINT O PUNTO DE ACCESO

Modo AP: En este modo de operación el modulo NodeMCU puede servir como un punto de acceso y permitir/negar el acceso a la red de datos. Este modo trabaja muy bien incluso tiene varias formas de seguridad de conexión (WPA2 PSK por ejemplo).



En el modo Access Point el NodeMCU difunde un SSID (Service Set Identifier), es decir, el "nombre de red" que se visualiza desde los dispositivos WiFi clientes (salvo que lo ocultemos, por supuesto). La conexión se realiza cuando el NodeMCU autoriza las peticiones de conexión de los clientes.

Por defecto la dirección IP del NodeMCU, que nos sirve para conectar con él desde un navegador Web, es la 192.168.4.1 y asigna a los clientes las siguientes direcciones IP disponibles, 192.168.4.2, 192.168.4.3, etc.

3.13 NODEMCU ESP8266 COMO ESCANER DE REDES

En este caso nuestra placa NodeMCU busca redes WiFi que sea capaz de detectar. De igual forma puedes programar directamente desde el IDE de arduino o hacerlo desde TUNIOT.

Para obtener el número de redes wifi detectadas utilizamos la función en el IDE de arduino **WiFi.scanNetworks()** que devuelve el número de redes encontradas. De igual forma existen comando para visualizar la MAC, la potencia de la red, la MAC del dispositivo detectado y el tipo de seguridad que maneja.

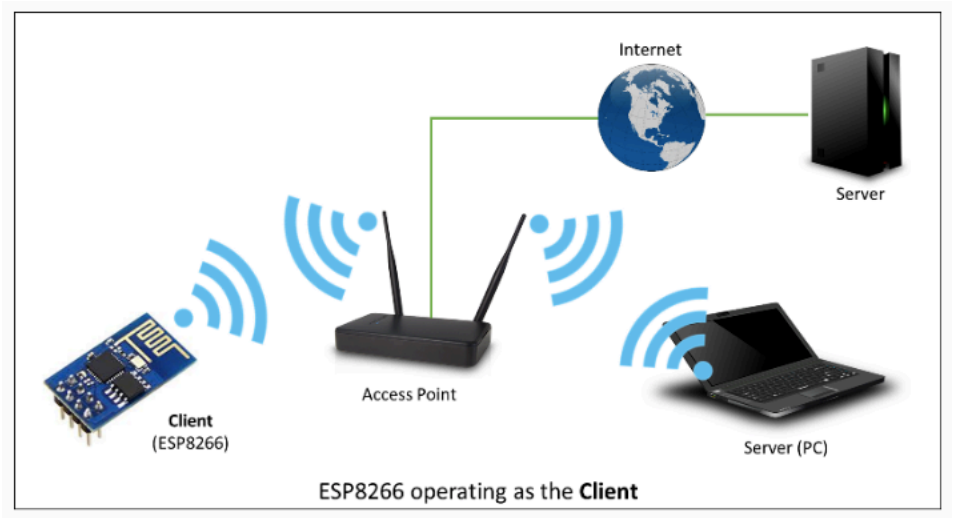
Para obtener información adicional: http://arduino-esp8266.readthedocs.io/en/latest/esp8266wifi/scan-class.html

3.14 NodeMCU como Cliente

La clase Client crea clientes que pueden acceder a los servicios proporcionados por los servidores para enviar, recibir y procesar datos. La clase client crea clientes que pueden conectarse a los servidores y enviar y recibir datos.

Debemos declarar un cliente que se pondrá en contacto con el host (servidor):

WiFiClient client;



Si la conexión es exitosa, debemos enviar solicitud al host para proporcionar la información específica que necesitamos. Esto se hace utilizando la solicitud HTTP GET como en las líneas siguientes:

```
client.print(String("GET /") + " HTTP/1.1\r\n" +
             "Host: " + host + "\r\n" +
             "Connection: close\r\n" +
             "\r\n"
            );
```

En el siguiente capítulo se explicará con más claridad este proceso de conexión entre cliente y servidor.

3.15 Comandos WiFi con el IDE de Arduino

WiFi.scanNetworks()
Analiza las redes WiFi disponibles y devuelve el número descubierto. (Modo escaner).

WiFi.begin()
Inicializa la configuración WiFi para conectarse a una red proporciona el estado actual. (Modo estación).

WiFi.softAP()
Para configurar una red como AP (la contraseña debe tener al menos 8 caracteres) WPA2-PSK. (Modo AP).

WiFi.config()
Permite configurar direcciones IP estáticas, cambiar las direcciones DNS y puerta de enlace.

WiFi.SSID()
Obtiene el SSID de la red actual

WiFi.RSSI()
Toma el nivel de seña en potencia dbm en conexión a un Gateway.

WiFi.BSSID()
Toma la MAC del equipo al que está conectado.

WiFi.encryptionType()
Toma el tipo de encriptación utilizado

WiFi.disconnect()
Desconecta de una red WiFi.

WiFi.localIP() si es STA**, WiFi.softAPIP()** si es AP.
Toma la IP utilizada localmente.

3.14 COMPARACIÓN DE CÓDIGOS PARA IMPLEMENTACIÓN DE WIFI

ESCANER DE REDES	ACCESS POINT	ESTACIÓN
#include <ESP8266WiFi.h>	#include <ESP8266WiFi.h>	#include <ESP8266WiFi.h>

voidSetup(){
 	
Serial.begin(9600);
Serial.println();
 	
WiFi.mode(WIFI_STA);
WiFi.disconnect();
delay(100);
 	
{
	voidSetup(){
	
	Serial.begin(9600);
	WiFi.mode(WIFI_AP);
	
	WiFi.softAP("SSID", "PASSWORD);
	Serial.println((WiFi.softAPIP()));
	
	{
		voidSetup(){
		
		Serial.begin(9600);
		WiFi.mode(WIFI_STA);
		
		delay(1000);
		WiFi.disconnect();
		Serial.println("INICIANDO");
		WiFi.begin("SSID","PASSWORD");
		while((!(WiFi.status() == WL_CONNECTED))){
		delay(300);
		Serial.println("....");}
		Serial.println("CONECTADO");
		Serial.println((WiFi.localIP()));
		
		{

Serial.print("Iniciando escaneo ...");
int n = WiFi.scanNetworks();
Serial.println(n);
Serial.println("Redes encontradas:");
for (int i = 0; i < n; i++)
{
Serial.println(WiFi.SSID(i));
Serial.println(WiFi.RSSI(i));
delay(500);
}
Serial.println();
delay(9000);
	Serial.println("Estaciones conectadas:");
	Serial.println((WiFi.softAPgetStationNum()));
	delay(3000);	

3.14 PRACTICAS DE WIFI CON EL NODEMCU.

En este capítulo nos enfocaremos a prácticas configuradas para conectarse de alguna forma a WiFi.

Laboratorio 5. Escáner y Estación WIFI

Laboratorio 6. Access Point y servidor WEB