

# Sensores y actuadores en red inalámbrica

# Metodología

Para introducir los conceptos de integración de sensores y actuadores en red, vamos a emplear un acercamiento práctico.

# Material de referencia:

Biblioteca WiFi de arduino:

<https://www.arduino.cc/en/Reference/WiFi>

Tutoriales y bibliotecas:

<https://github.com/espressif/arduino-esp32/tree/master/libraries/WiFi>

<https://techtutorialsx.com/2017/04/24/esp32-connecting-to-a-wifi-network/>

<https://techtutorialsx.com/2017/06/29/esp32-arduino-getting-started-with-wifi/>

Para la documentación de algunas bibliotecas este enlace puede ser de mucha utilidad:

<https://arduino-esp8266.readthedocs.io/en/latest/libraries.html#wifi-esp8266wifi-library>

Ejercicio 1: vamos al tablero para  
explicar cómo funciona...

## Ejercicio 2: conectarse a una red WiFi

Para conectarnos a una red WiFi necesitamos saber:

- El ssid de la red (el nombre de la red)
- La clave de la red

Una vez nos conectemos el Access Point nos dará una dirección IP que identificará el dispositivo en la red y permitirá que otros dispositivos le envíen mensajes

## WiFiConnect\$

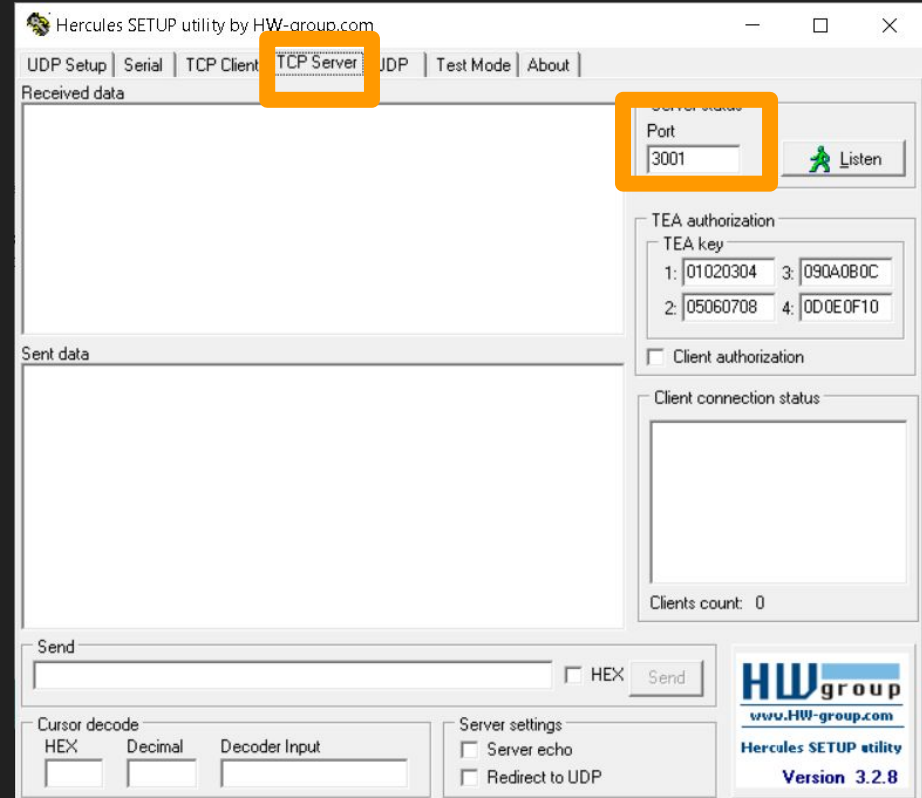
```
1 #include <WiFi.h>
2 #include <WiFiMulti.h>
3
4 WiFiMulti WiFiMulti;
5
6 void setup()
7 {
8     Serial.begin(115200);
9     WiFiMulti.addAP("your_ssid", "your_password");
10    Serial.println();
11    Serial.print("Waiting for WiFi... ");
12    while(WiFiMulti.run() != WL_CONNECTED) {
13        Serial.print(".");
14        delay(500);
15    }
16    Serial.println();
17    Serial.println("WiFi connected");
18    Serial.println("IP address: ");
19    Serial.println(WiFi.localIP());
20    delay(500);
21 }
22
23 void loop()
24 {
25 |
26 }
```

1. Para usar el radio WiFi del ESP32 incluimos los archivos de cabecera: WiFi y WiFiMulti
2. Creamos un objeto de tipo [WiFiMulti](#).
3. Usando el objeto WiFiMulti adicionamos las posibles redes WiFi a los cuales se conectará nuestro sensor.
4. Finalmente comenzamos el proceso de conexión con el método **run()**.
5. Note que el método run() se llama repetidamente hasta que se logre la conexión a la red.
6. Por último usamos el método localIP() para obtener la dirección IP asignada por el AP al sensor.

# Ejercicio 3

1. Descargue un terminal TCP/IP que tenga cliente y servidor. Por ejemplo, [Hercules](#).
2. Abra Hercules.
3. Seleccione la pestaña TCP Server y configure el puerto con el valor 3001.
4. Click en el botón Listen.

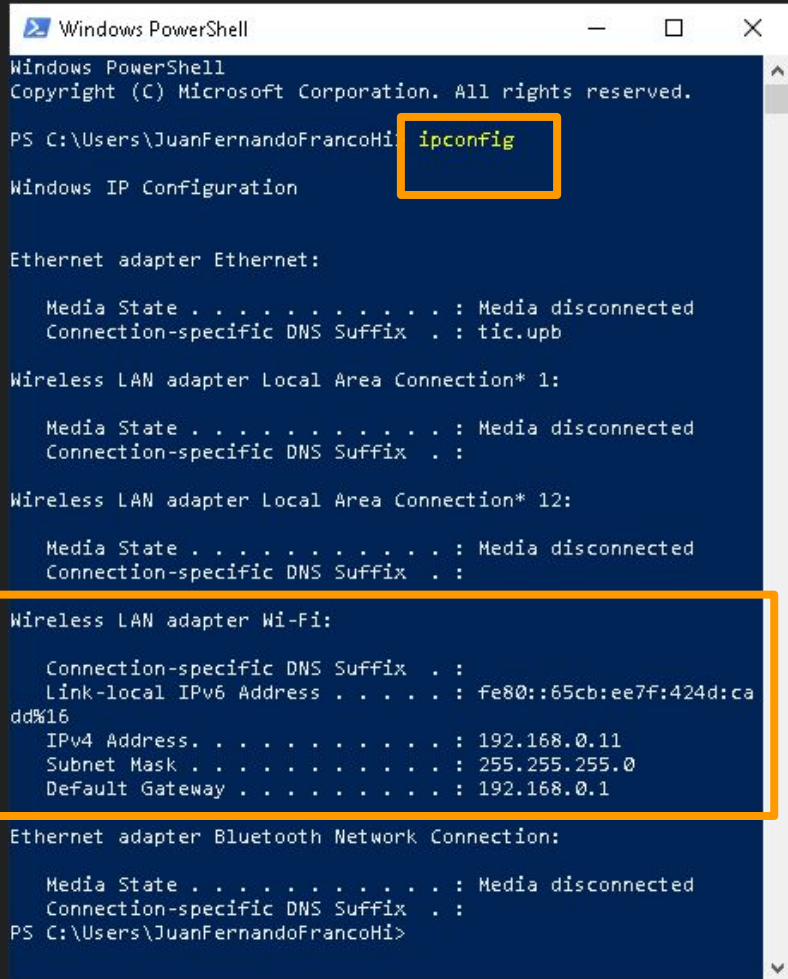
Este procedimiento permitirá tener un servidor escuchando conexiones en el puerto 3001.



# Ejercicio 3 (cont)

1. Ejecute la aplicación Windows PowerShell.
2. Escriba el comando **ipconfig**
3. Escriba la dirección IP que el AP de su casa le asignó a su computador.

NOTA: tenga en cuenta que esta dirección IP puede cambiar, por tanto siempre debe verificarla cuando haga experimentos.



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Users\JuanFernandoFrancoHí> ipconfig

Windows IP Configuration

Ethernet adapter Ethernet:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . : tic.upb

Wireless LAN adapter Local Area Connection* 1:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Local Area Connection* 12:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

Wireless LAN adapter Wi-Fi:

    Connection-specific DNS Suffix  . :
    Link-local IPv6 Address . . . . : fe80::65cb:ee7f:424d:ca
    dd%16
    IPv4 Address. . . . . : 192.168.0.11
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.0.1

Ethernet adapter Bluetooth Network Connection:

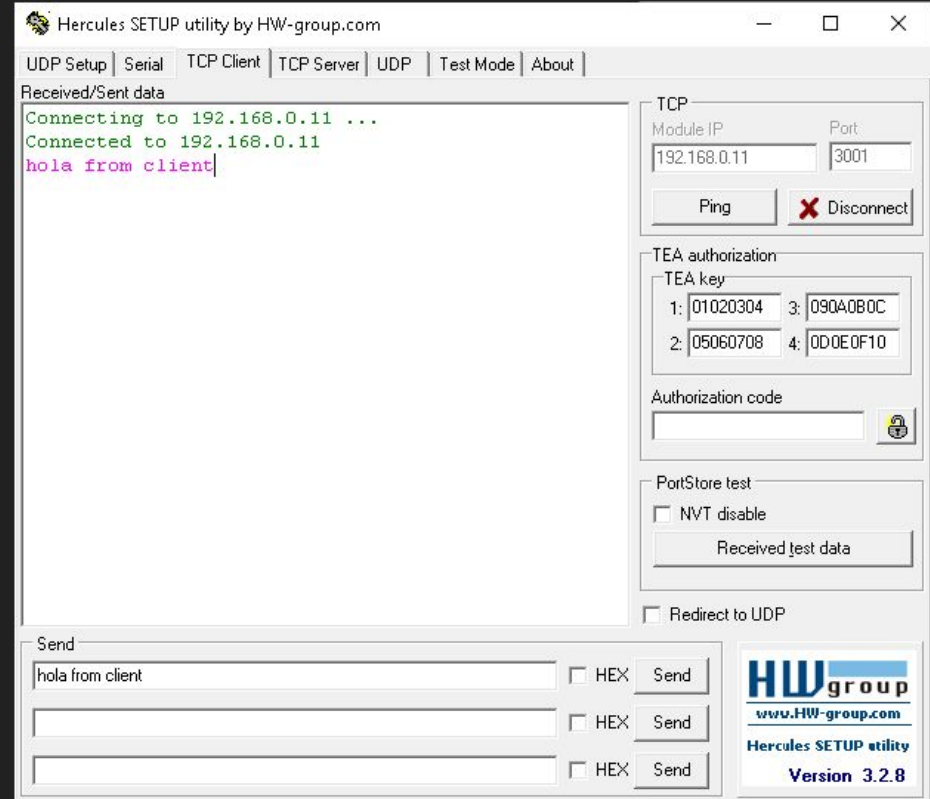
    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :

PS C:\Users\JuanFernandoFrancoHí>
```



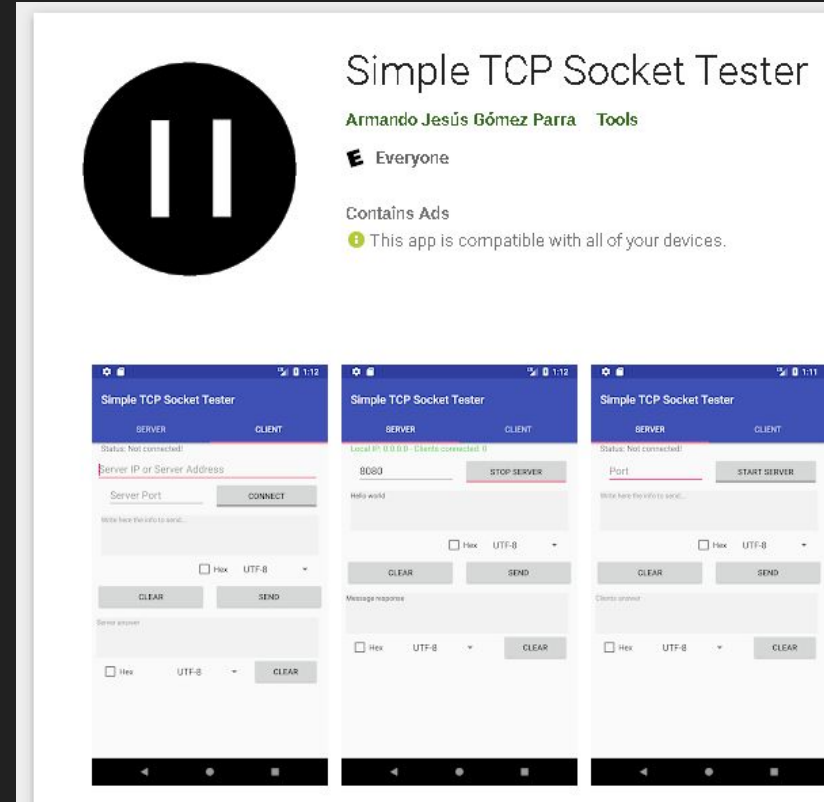
# Ejercicio 3 (cont)

1. Abra una segunda ventana de Hercules.
2. Seleccione la pestaña TCP Client.
3. En el campo Module IP y Port coloque las direcciones IP del servidor (slide anterior) y el puerto (3001).
4. Click en Connect
5. Escriba una cadena de caracteres en la primera caja de Texto de la sección Send de la ventana.
6. Click en Send.
7. ¿Qué sucede en la ventana del TCP Server?
8. Envíe una respuesta desde la ventana del TCP Server al TCP Client



# Ejercicio 4

1. Descargue un cliente TCP para su celular (Ejemplo: simple TCP socket Tester)
2. Conecte su celular a la misma Red Wifi.
3. Conecte la aplicación al TCP Server.
4. Envíe un mensaje al server y del server a los clientes conectados.
5. Si no funciona la conexión, mire el siguiente slide



# Ejercicio 4 (Cont)

Es posible que no pueda conectar su celular al TCP Server que corre en el computador. Sí este es el caso, deshabilitar temporalmente el antivirus apagar el firewall.

**OJO: NO OLVIDE HABILITAR EL FIREWALL UNA VEZ TERMINE LAS PRUEBAS**



# Ejercicio 5

1. En los Ejemplos del Arduino para el ESP32 busque en la carpeta Wifi el ejemplo **WiFiTelnetToSerial**.
2. Abra el ejemplo.
3. Salve una versión del ejemplo.
4. Modifique el ejemplo cambiando serial2 por serial y la red WiFi a la cual se conectará el servidor.
5. Programe en el arduino el ejemplo.
6. Conecte un cliente al servidor usando Hercules.
7. Intercambie mensajes entre el cliente y el servidor.

## NOTAS:

- Sí no funciona, revise de nuevo los ejercicios anteriores.
- Recuerde modificar serial2 por serial.

Este ejemplo es un servidor. Abra el monitor de arduino y tome nota de la dirección IP asignada a este servidor y lea en el programa el puerto en el cual está escuchando el server.

# Ejercicio 6

1. Salve una nueva versión del ejemplo **WiFiTelnetToSerial** que modificó en el ejercicio anterior. Debe tener cambiado serial2 por serial.
2. Ahora aumente la cantidad de clientes que puede aceptar de 1 a 3:  
MAX\_SRV\_CLIENTS
3. Conecte varios TCP Clients usando Hercules y/o el celular (Ejercicio 4).
4. Experimente enviando y recibiendo mensajes.

# Ejercicio 7

1. Ahora abra el ejemplo WiFiClientBasic
2. Modifique el programa para configurar la red WiFi a la cual se conectará.
3. Abra un TCP Server en Hercules. Tome nota de la dirección IP del server y el puerto donde estará escuchando. Recuerde usar ipconfig y apagar los demás TCP Server de Hercules que tenga previamente escuchando en el mismo puerto.
4. Modifique de nuevo el ejemplo pero esta vez para modificar el puerto y la dirección IP del host (server) al cual se conectará.
5. Comente la línea 59 y descomente la línea 57.
6. Programe el ESP32 con el ejemplo modificado.
7. Abra la terminal y experimente.
8. Experimente con este código.

# Ejercicio 8

Estudie detenidamente el código de los Ejercicios 5, 6.

¿Qué hace cada parte?

# Ejercicio 9

Estudie detenidamente el código del Ejercicio 7

¿Qué hace cada parte?



# Ejercicio 10

Ya estudió en detalle y al menos durante una hora los ejercicios 8 y 9?

Hágalo, lo necesitará para el siguiente reto

# Ejercicio 11

Se debe repetir lo realizado en el ejercicio 5 pero esta vez permita que el servidor pueda atender varios clientes simultáneamente.

Tenga en cuenta lo siguiente:

- El servidor deberá atender hasta tres clientes.
- Los mensaje escritos en la terminal del servidor serán enviados a cada cliente conectado.
- El servidor debe recibir los mensajes de cada cliente.

# RETO A

Utilizando 3 dispositivos: ESP32, computador (o celular) y computador (u otro celular) realizar una aplicación estilo grupo de “Whatsapp”:

- Uno de los dispositivos será el **servidor: el ESP32**
- Los otros serán clientes.
- La idea de la aplicación es que aparezca en la terminal de cada cliente los mensajes que colocan los demás.

# RETO B

- Usa como servidor Hercules u otro programa similar
- Define un protocolo ASCII
- Construye un programa **cliente** en el ESP32 que responda a diferentes comandos enviados desde el servidor:
  - Escribir un puerto
  - Leer un puerto

# RETO C

- Repite lo anterior pero esta vez define un protocolo binario.