

# ANÁLISIS DE REDES CON RASPBERRY PI PICO 2W

CARABALLO CARDENAS ANDRES FELIPE, UNIVERSIDAD MILITAR NUEVA GRANADA  
(est.andres.caraballo@unimilitar.edu.co)

<https://github.com/ANDRES1Q647/COMUNICACION-DIGITAL.git>

**Resumen** La presente práctica de laboratorio tuvo como objetivo analizar las redes Wi-Fi disponibles en el entorno utilizando el microcontrolador Raspberry Pi Pico 2W. A través de los códigos suministrados por el docente, se identificó la dirección MAC del controlador y se realizó un escaneo de redes Wi-Fi, registrando parámetros como el SSID (nombre de la red), la dirección MAC y el canal asociado. Durante el proceso, se implementó un sistema de registro de la potencia de señal (RSSI), tomando tres mediciones por red y organizándolas de mayor a menor intensidad en una tabla. Posteriormente, se creó un punto de acceso inalámbrico (Access Point) mediante un código proporcionado, configurando el nombre del SSID como “PicoW-AP-<GRUPO2>”. Este punto de acceso permitía ejecutar un código con interfaz HTML, donde se podía acceder desde el navegador mediante la dirección generada por el programa. En la página web resultante, se incluyeron botones interactivos para encender y apagar un LED. Además, se modificó el canal del punto de acceso (estableciéndolo en el canal 8) y se verificó el cambio a través de un escáner de redes, comprobando que la red aparecía correctamente con su nueva configuración.

Finalmente, se realizaron modificaciones tanto en el código HTML como en el programa APWifipico para permitir la lectura del valor ADC a través de un potenciómetro, mostrando estos valores en tiempo real dentro de la página web. También se desarrolló un código adicional para registrar cómo variaba la potencia de la señal Wi-Fi a medida que aumentaba la distancia, tomando diez mediciones y organizándolas en una tabla para su posterior representación gráfica. En conjunto, estas actividades permitieron comprender mejor el funcionamiento de las redes inalámbricas, la configuración de puntos de acceso y la integración de sensores analógicos con comunicación Wi-Fi, cumpliendo así los objetivos propuestos en la práctica de laboratorio.

## I. INTRODUCCIÓN

El desarrollo de este laboratorio se llevó a cabo utilizando los códigos suministrados por el docente y siguiendo las indicaciones establecidas en la guía. Cada ítem

fue desarrollado con el propósito de obtener los resultados correspondientes a cada punto planteado. De esta manera, fue posible realizar los análisis y elaborar las conclusiones basadas en los datos obtenidos durante la práctica.

## II. PROCEDIMIENTO

#### *A. Dirección MAC del dispositivo*

En esta sección se utilizó el código `mac_wifi.py`, cuyo propósito es configurar la interfaz Wi-Fi del microcontrolador en modo estación, también conocido como modo cliente. Una vez activada la interfaz, el programa ejecuta una serie de instrucciones para obtener la dirección MAC, almacenando los bytes correspondientes en una variable. Esta configuración permite que el sistema identifique y muestre en la consola la dirección MAC asociada al microcontrolador. A continuación, se presenta la evidencia del proceso descrito anteriormente.

```
File Theory - C:\Users\anjal\Downloads\mac_wRPy ② 6:1
File Editor Visual Studio Code Homecoming Applet

[uncamedednspay] mac_wRPy.py

# Read the 6-byte MAC address from the ML-FI interface in colon-separated hexdecimal.
# If the MAC address is None, then the ML-FI interface does not have one assigned yet.
# Prints the result, and (optionally) disables the interface.

import network
wlan = network.WLAN(network.STA_IF)
wlan.active(True)

# Read the 6-byte MAC address and Format as colon-separated hex
mac_bytes = wlan.config('mac')
mac_hex = " ".join(["{:02X}:".format(b) for b in mac_bytes])
mac_hex = mac_hex[:-1]

print("Raspberry Pi: Mac w MAC (STA):", mac_hex)

# Turn ML-FI OFF to save power if no further ML-FI actions are needed
wlan.active(False)

# 

#
# Console : 
#>>> wlan < SEDITOR_CONTENT

MLFI soft reboot
Raspberry Pi: Mac w MAC (STA): 3C:CF:E7:ED:92:5F
#>
```

MicroPython (Raspberry Pi Pico) - Board CDC © COM-1

Al registrar la dirección MAC y reiniciar el programa en varias ocasiones, se observa que la dirección mostrada permanece igual. Esto ocurre porque cada microcontrolador posee una dirección MAC única asignada por el fabricante, la cual actúa como un identificador exclusivo del dispositivo. Por esta razón, incluso al reiniciar el programa múltiples veces, la misma dirección MAC se mantiene visible de forma constante.

### **B. Escaneo de redes Wi-Fi**

En el segundo apartado se emplea el código scanner\_wifi.py, cuyo propósito es ejecutar un escaneo de las redes Wi-Fi disponibles en el entorno. Para ello, se dejó correr el programa durante un tiempo mínimo de 60 segundos, con el fin de observar en la consola los resultados del escaneo. Durante la prueba, se realizaron mediciones en intervalos de 0, 10, 20 y 30 segundos, utilizando estos tiempos como método de muestreo para obtener los tres valores con mayor RSSI, es decir, la potencia de la señal recibida desde los distintos puntos de acceso.

Además, el escáner implementado mostró información complementaria como el SSID (nombre de la red), el canal, y la dirección MAC de cada punto de acceso detectado. A estos datos se añadió una columna adicional con la frecuencia calculada según el valor del canal, con el propósito de obtener una visión más completa de las características de las redes analizadas.

A continuación, se presenta la tabla elaborada en Excel, junto con las evidencias de cada medición realizada con los diferentes intervalos de tiempo mencionados anteriormente.

Tiempo (s)	RSSI (potencia recibida dBm)	Nombre de Red	Canal	Frecuencia (MHz)	MAC
0	-56	OneScreen_7288	3	2422	DE:74:E5:58:B0:39
0	-54	<Hidden>	8	2447	C0:25:2F:44:F5:38
0	-50	Laboratorios	4	2427	10:F0:68:77:21:22
10	-47	Umgng-pubR-cll100	4	2427	10:F0:68:77:21:20
10	-46	Umgng-prvR-cll100	4	2427	10:F0:68:77:21:21
10	-46	Laboratorios	4	2427	10:F0:68:77:21:22
20	-50	Laboratorios	4	2427	10:F0:68:77:21:22
20	-50	Umgng-prvR-cll100	4	2427	10:F0:68:77:21:21
20	-47	Recover_Me-372120	4	2427	10:F0:68:77:21:2F
30	-57	A55_of_danish	6	2437	DE:52:01:24:60:AE
30	-56	Andrey	1	2412	6:E6:C4:AE:A2:7F:3D
30	-54	<Hidden>	8	2447	C0:25:2F:44:F5:38

**Tabla 1.** Resumen de datos obtenidos en cada scanner

## Evidencias

**t = 0 seg**

MicroPython (Raspberry Pi Pico)

**t = 10 seg**

**Figura 3. Resultado del scanner en tiempo 10 seg**

**t = 20 seg**

MicroPython (Raspberry Pi Pico) • 8

t = 30 seg

**Figura 5. Resultado del scanner en tiempo 30 seg**

Al analizar estos datos, se evidencia que la potencia de la señal (RSSI) varía conforme transcurre el tiempo de medición. En general, se puede afirmar que a medida que aumenta el tiempo, la potencia tiende a disminuirse, mostrando así una relación directamente inversa proporcional entre el tiempo y la intensidad de la señal recibida.

## Solución a preguntas

**Compare los canales de los tres AP con mayor RSSI. ¿Hay solapamiento (1,6,11, etc. en 2.4GHz)? ¿Qué implicaciones tiene?**

Al comparar los canales, se observa que el canal 4 presenta solapamiento con el canal 6, lo que genera interferencia entre ambos canales. Esta superposición de frecuencias puede ocasionar una disminución en el rendimiento de la conexión Wi-Fi, afectando directamente la velocidad de transmisión de datos, la estabilidad de la señal y provocando pérdida de paquetes. En consecuencia, las redes que operan en el canal 4 pueden experimentar una calidad de conexión inferior en comparación con aquellas configuradas en canales no solapados, como los canales 1, 6 o 11, que son los más recomendados para minimizar interferencias en la banda de 2.4 GHz.

**¿El RSSI fluctúa entre iteraciones? ¿Cuáles podrían ser las causas?**

Se evidenció que el RSSI presenta fluctuaciones entre las diferentes iteraciones de tiempo. Al analizar las mediciones, la red “Laboratorios” muestra valores que varían entre -46 dBm y -50 dBm, mientras que la red “Umng-pubR-chl100” oscila entre -47 dBm y -49 dBm. Estas variaciones se deben principalmente a cambios en la propagación de la señal, influenciados por el entorno físico del lugar de medición en este caso, el sótano donde la señal tiende a dispersarse, generando inestabilidad. Además, la presencia de otras redes Wi-Fi en el mismo entorno provoca interferencias electromagnéticas, las cuales contribuyen a la pérdida de paquetes y a la reducción de la calidad de la conexión, dificultando en algunos casos el acceso estable a la red.

#### **C. Creación de un punto de acceso inalámbrico**

Para este segmento del proyecto, utilizando el microcontrolador Raspberry Pi Pico 2W, se configuró un punto de acceso WiFi. En él se cargaron dos códigos que trabajan de forma conjunta:

1. APWifipico.py: encargado de la configuración del punto de acceso. Este programa permite que el dispositivo cree su propia red WiFi y genere un enlace de navegación para acceder a una página web. En dicha página se incluyen dos botones que

permiten encender y apagar un LED conectado al microcontrolador.

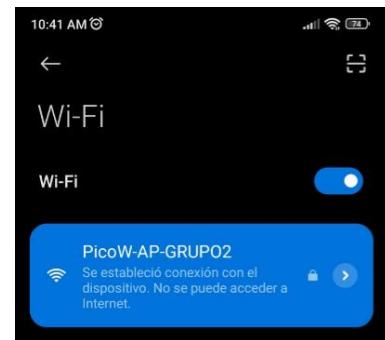
2. Código HTML: corresponde a la estructura y diseño de la página web mencionada. Este archivo está vinculado al código principal y define la interfaz que el usuario visualiza al conectarse a la red creada por el Pico.

Cabe destacar que se realizó una pequeña modificación al código: se asignó un nombre personalizado a la red WiFi. En este caso, se utilizó simplemente el número del grupo (2). Todos los demás parámetros del programa se mantuvieron sin cambios.

A continuación, en la figura mostrada, se presenta la evidencia del funcionamiento de la configuración.

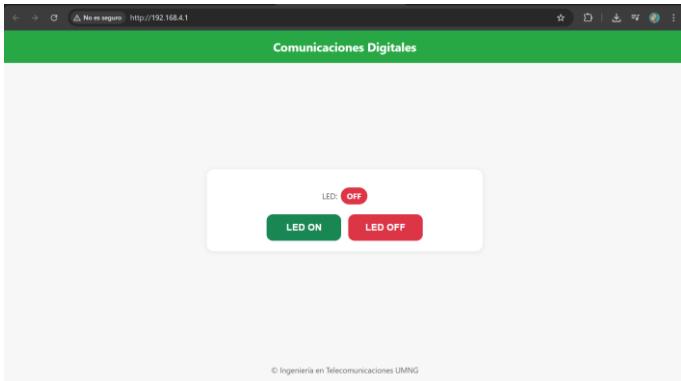
## **Figura 6. Configuración del punto de acceso**

Después de cargar y ejecutar el código, se procede a comprobar su funcionamiento. Para ello, desde un dispositivo en este caso, un teléfono se revisan las redes WiFi disponibles y se verifica que el punto de acceso creado por la Raspberry Pi Pico 2W aparezca correctamente. En la siguiente figura se muestra la evidencia de esta verificación.



**Figura 7. Visualización de la red creada**

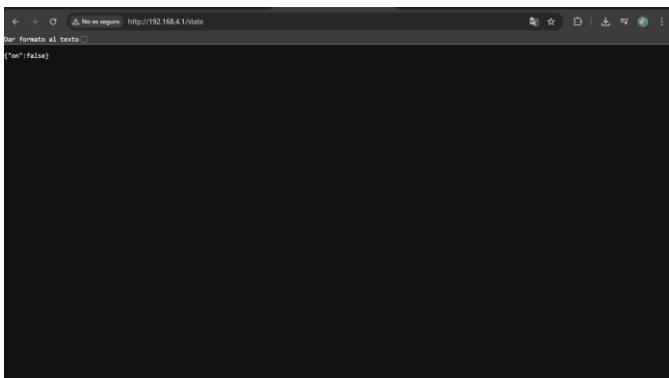
Al observar la figura, se confirma que el acceso al punto de acceso incluyendo el ingreso de la contraseña fue exitoso. Como siguiente paso, se abrió un navegador y se ingresó la dirección IP mostrada previamente en la consola del programa. Al digitar dicha dirección, se cargó correctamente la interfaz de la página web, como se muestra a continuación.



**Figura 8.** Visualización del entorno de la página web

Una vez establecida la conexión con la página web, se procedió a probar los botones de encendido y apagado del LED. En este caso, se utilizó el LED integrado en el microcontrolador. Al presionar cada botón, se verificó que la respuesta fuera inmediata y correcta, confirmando que el sistema funciona de manera óptima.

Luego de haber comprobado se dirige luego a la dirección <http://192.168.4.1/state> en el cual se visualiza el JSON como se muestra a continuación.

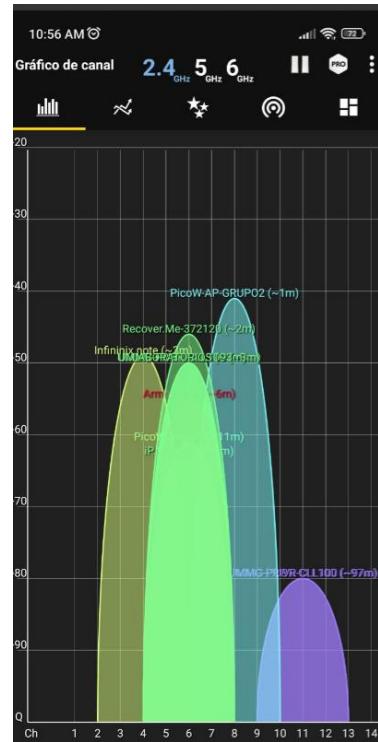


**Figura 9.** Visualización del entorno JSON

Como se aprecia en la figura anterior, se observa el funcionamiento interno de los botones. En el código, el valor false indica que el LED está apagado, mientras que true señala que está encendido. Esta representación permite comprender de manera más clara cómo actúan los botones a nivel interno dentro del programa.

#### D. Cambio de canal

En este cuarto apartado se realiza el cambio de canal el cual el elegido es el canal 8, donde al volver a ejecutar el código por medio de un scanner wifi, en donde ese scanner fue por medio de una aplicación se corrobora si el cambio del canal se visualiza de manera correcta por lo cual se muestra la siguiente figura.



**Figura 10.** Visualización de los canales

Como se observa en la figura anterior, la red de color azul aguamarina identificada como PicoW-AP-GRUPO2 aparece ubicada en el canal 8, lo cual confirma que el cambio de canal se realizó correctamente. Además, su potencia de señal es de aproximadamente -41 dBm, un valor cercano a 0 que indica una conexión estable.

En contraste, la red marcada en color morado presenta una potencia cercana a -80 dBm, lo que refleja una señal mucho más débil. Este valor suele estar asociado a interferencias e inestabilidad en el entorno, lo que afecta la calidad general de la conexión.

#### E. Actividades complementarias

Como primera actividad complementaria, se realizaron modificaciones en los códigos APWifipico.py y HTML con el fin de mostrar en la página web la lectura proveniente del conversor ADC 0. Para poder visualizar estos valores en tiempo real, se conectó un potenciómetro al microcontrolador, permitiendo variar manualmente la señal de entrada. A continuación, se presentan las lecturas obtenidas en la interfaz web a medida que se ajusta el potenciómetro, evidenciando el funcionamiento del conversor ADC.

Como se observa en las figuras adjuntas se observa la interfaz web generada por el Raspberry Pi Pico 2W después de modificar los códigos APWifipico.py y HTML para mostrar la lectura del conversor analógico-digital (ADC 0). Estas lecturas provienen directamente de un potenciómetro conectado al microcontrolador, el cual permite variar manualmente el voltaje de entrada.

A medida que se gira el potenciómetro, el ADC del Pico convierte ese voltaje en un número digital. Es decir, el ADC toma un voltaje entre 0 V y 3.3 V y lo transforma a un valor entre 0 y 65535, ya que es un ADC de 16 bits.

- **Potenciómetro al mínimo (ADC = 96 / Voltaje ≈ 0.005 V)**



**Figura 11. Visuliazcion de conversor ADC con volatje 0.005 V**

El valor que representa el ADC es de 96 donde este valor es bajo muy cercano a cero lo que indica que casi no está entrando voltaje al pin ADC así mismo el valor de voltaje marca 0.0005 V lo que es valor nulo.

- **Potenciómetro a la mitad (ADC = 29975 / Voltaje ≈ 1.509 V)**



**Figura 12. Visuliazcion de conversor ADC con volatje 1.509 V**

El valor que representa el ADC es de 29975 lo cual este valor representa la mitad del rango total de ADC por consiguiente el voltaje de entrada tambien esta alrededor de 1.509 V .

- **Potenciómetro casi al máximo (ADC = 65199 / Voltaje ≈ 3.238 V)**

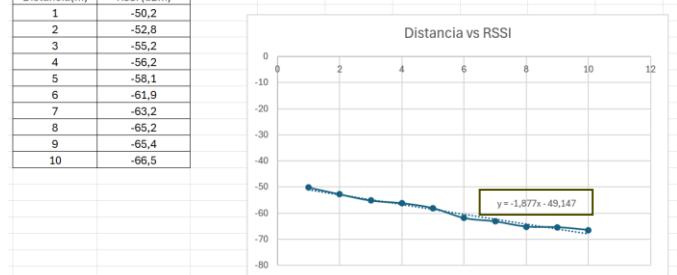


**Figura 13. Visuliazcion de conversor ADC con volatje 3.238 V**

El valor que representa el ADC es de 65199 dando muy cerca al valor máximo que es de 65535 lo que significa con esto es que el potenciómetro esta suministrando un voltaje casi completo al pin ADC.

Como segunda tarea complementaria se diseña un código en el cual se mide la potencia de salida de un punto de acceso a partir de la distancia que se tome, en este caso lo que se hizo fue tomar 10 distancias y de cada una se obtiene 10 mediciones en la cual se promedia y se registra en una tabla donde al completar la tabla con cada distancia y medida promediada se realiza una gráfica en el programa excel la cual se visuliza acontinuación.

Distancia(m)	RSSI (dBm)
1	-50,2
2	-52,8
3	-55,2
4	-56,2
5	-58,1
6	-61,9
7	-63,2
8	-65,2
9	-65,4
10	-66,5



**Figura 14. Gráfica de Distancia vs RSSI**

Como se evidencia en la gráfica anterior se observa una tendencia descendente, lo que significa que la señal disminuye a medida que se va aumentando la distancia dando así que la línea trazada sobre los puntos de la tabla corresponde a una regresión lineal la cual esta representada por la ecuación encerrada en el rectángulo de color verde.

Esta ecuación lo que representa es la relación de distancia en función de x y el RSSI en función de y dando así que el valor -1.877x indica la pendiente lo que permite que, por cada valor de distancia en metros, el RSSI disminuya a un valor de -49.147.

#### F. Diagrama UML de los protocolos HTTP y UDP

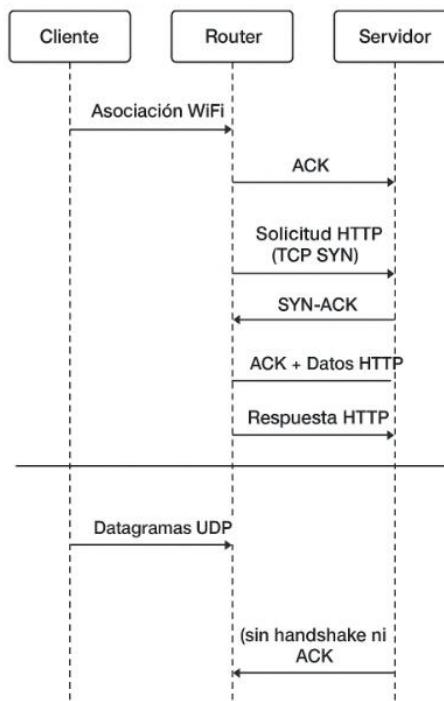


Figura 15. Diagrama UML de los protocolos HTTP y UDP

### III. ANÁLISIS DE RESULTADOS

- Al revisar las mediciones realizadas en diferentes intervalos de tiempo dentro del sótano, se observa que la intensidad de la señal Wi-Fi presenta variaciones importantes entre una lectura y otra. Por ejemplo, la red *Laboratorios* fluctúa entre -46 dBm y -50 dBm, mientras que la red *UMNG-pubR-cll100* varía entre -47 dBm y -49 dBm.
- Las fluctuaciones no son errores de medición, sino que reflejan las condiciones reales del entorno. El sótano es un espacio cerrado, con paredes gruesas y estructuras que dificultan la propagación de la señal. Además, la presencia de múltiples redes Wi-Fi en el mismo espacio genera interferencia entre canales, lo que contribuye a la inestabilidad.
- El solapamiento de canales se identificó que el canal que presenta esta anomalía es el canal 4 que presenta interferencia con el canal 6, lo cual afecta directamente el rendimiento de la red asociada al canal 4. Esto puede traducirse en problemas como pérdida de paquetes, menor velocidad, latencia alta y desconexiones momentáneas.

Los datos muestran que el ambiente presenta factores físicos y electromagnéticos que influyen en la calidad del Wi-Fi, y que la selección inadecuada de canales aumenta aún más esta afectación.

### IV. CONCLUSIONES

- La relación entre distancia y RSSI es inversamente proporcional, es decir, la intensidad de la señal disminuye conforme aumenta la separación entre el punto de acceso y el dispositivo receptor.
- La disminución del RSSI fue progresiva y estable, lo cual indica que las mediciones se realizaron en un ambiente con pocas interferencias.
- Esta práctica de laboratorio permitió comprender de manera práctica cómo se atenúa una señal WiFi y cómo influyen factores como la distancia en la calidad del enlace inalámbrico.
- La ecuación de tendencia lineal obtenida permite estimar cómo se comportará la señal a diferentes distancias.
- Para mejorar la calidad de la comunicación Wi-Fi en entornos como el sótano, sería recomendable reorganizar los canales, reducir el número de redes en competencia o instalar puntos de acceso adicionales que distribuyan mejor la señal.

### V. REFERENCIAS

- [1]. Cisco. (2023). *Understanding the 2.4 GHz channels*. Cisco Support Documentation. <https://www.cisco.com/c/en/us/support/docs/wireless-mobility/wireless-lan-wlan/23633-2.4ghz-channel.html>
- [2]. Cisco. (2023). *Understanding the 2.4 GHz channels*. Cisco Support Documentation. <https://www.cisco.com/c/en/us/support/docs/wireless-mobility/wireless-lan-wlan/23633-2.4ghz-channel.html>
- [3]. Raspberry Pi Foundation. (2023). *Raspberry Pi Pico W – Networking guide*. Raspberry Pi Documentation. <https://www.raspberrypi.com/documentation/microcontrollers/pico-w/>
- [4]. Ubiquiti. (2022). *Understanding WiFi interference and channel overlap*. Ubiquiti Help Center. <https://help.ui.com/hc/en-us/articles/115005212507-Wi-Fi-Interference-and-Channel-Overlap>