

# How to control Raspberry Pi Pico via Wifi

## What is Wifi

WiFi, short for Wireless Fidelity, is a technology that enables devices like smartphones, laptops, and other electronic gadgets to connect to the internet or communicate with each other without the need for physical cables. It operates by using radio frequency signals to transmit data between devices and a wireless router. The router, connected to the internet via a wired connection, acts as a central hub that facilitates communication between devices within its range. WiFi has become a ubiquitous and convenient way for people to access the internet and share information wirelessly, contributing to the proliferation of wireless connectivity in homes, businesses, and public spaces.

You can create a local wifi network with just powering up a wifi router. When two or more devices (computers, microcontrollers...) are connected to the same wifi network, they can communicate with each other, without accessing the internet. Not all microcontrollers come with a built-in Wifi module so we will be using Pico W (with a built-in wifi module) in this tutorial. But there are plenty of add-ons options you can add to your microcontroller for wifi.

**External read** for fun:)



## What is IP address

An IP address, or Internet Protocol address, is a unique numerical label assigned to each device connected to a computer network that uses the Internet Protocol for communication. It consists of a series of numbers separated by periods, like **xxx.xxx.xxx.xxx**. Think of it as a digital address for your device on the internet. It allows devices to identify and communicate with each other on a network, much like a home address allows mail to be delivered to a specific location. IP addresses can be dynamic, meaning they change periodically, or static, where they remain constant. They play a crucial role in routing data packets across the internet, ensuring that information reaches the correct destination.

## How to find my IP address

1. Google Search - "What's my IP address?"
2. Go to Network Preferences (MAC), Wifi Properties (Windows)
3. Go to **Terminal** (MAC) or **Command Prompt** (Windows) and type in `ipconfig getifaddr en1`

## Get started

The example code will turn on/off the onboard LED on Pico via wifi without an external circuit. See **this tutorial** to learn how to set up your Pico. Before you go ahead, there are some things that you need to change for your wifi.

1. Change `ssid`, `password` in the code
2. Run your code in Thonny
3. Find your `IP address` in the Thonny Shell, something like `ip = 10.3.15.120`
4. open up a web browser
5. go to `http:// **IP address**/light/on` to turn the LED on
6. go to `http:// **IP address**/light/off` to turn the LED off

```
import network
import socket
import time

from machine import Pin

led = Pin("LED", Pin.OUT)

ssid = 'YOUR NETWORK NAME'
password = 'YOUR NETWORK PASSWORD'

wlan = network.WLAN(network.STA_IF)
wlan.active(True)
wlan.connect(ssid, password)

html = """<!DOCTYPE html>
<html>
  <head> <title>Pico W</title> </head>
  <body> <h1>Pico W</h1>
    <p>%s</p>
  </body>
</html>
"""

max_wait = 10
while max_wait > 0:
    if wlan.status() < 0 or wlan.status() >= 3:
        break
    max_wait -= 1
    print('waiting for connection...')
    time.sleep(1)
```

```
if wlan.status() != 3:
    raise RuntimeError('network connection failed')
else:
    print('connected')
    status = wlan.ifconfig()
    print( 'ip = ' + status[0] ) //print your address
```

```
addr = socket.getaddrinfo('0.0.0.0', 80)[0][-1]
```

```
s = socket.socket()
s.bind(addr)
s.listen(1)
```

```
print('listening on', addr)
```

```
# Listen for connections
```

```
while True:
```

```
    try:
        cl, addr = s.accept()
        print('client connected from', addr)
        request = cl.recv(1024)
        print(request)
```

```
        request = str(request)
        led_on = request.find('/light/on')
        led_off = request.find('/light/off')
        print( 'led on = ' + str(led_on))
        print( 'led off = ' + str(led_off))
```

```
        if led_on == 6:
            print("led on")
            led.value(1)
            stateis = "LED is ON"
```

```
        if led_off == 6:
            print("led off")
            led.value(0)
            stateis = "LED is OFF"
```

```
        response = html % stateis
```

```
cl.send('HTTP/1.0 200 OK\r\nContent-type: text/html\r\n\r\n')
cl.send(response)
cl.close()
```

```
except OSError as e:
```

```
    cl.close()
```

```
    print('connection closed')
```

---

Revision #2

Created 23 February 2024 12:34:57 by Joanne Leung

Updated 23 February 2024 13:21:18 by Joanne Leung