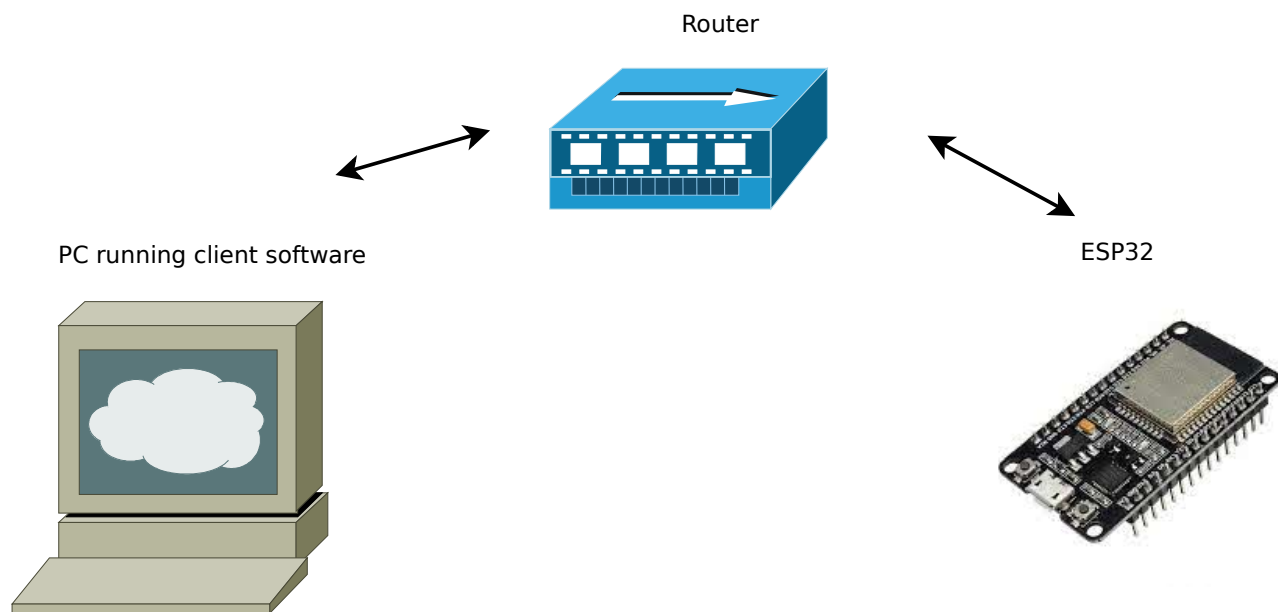


MicroPython and WiFi part 1: connect

By jean-claude.feltes@education.lu

1. Station mode and Access point mode

1.1. Connect over an existing network: Station mode



This connects to the network and shows the IP address of the ESP:

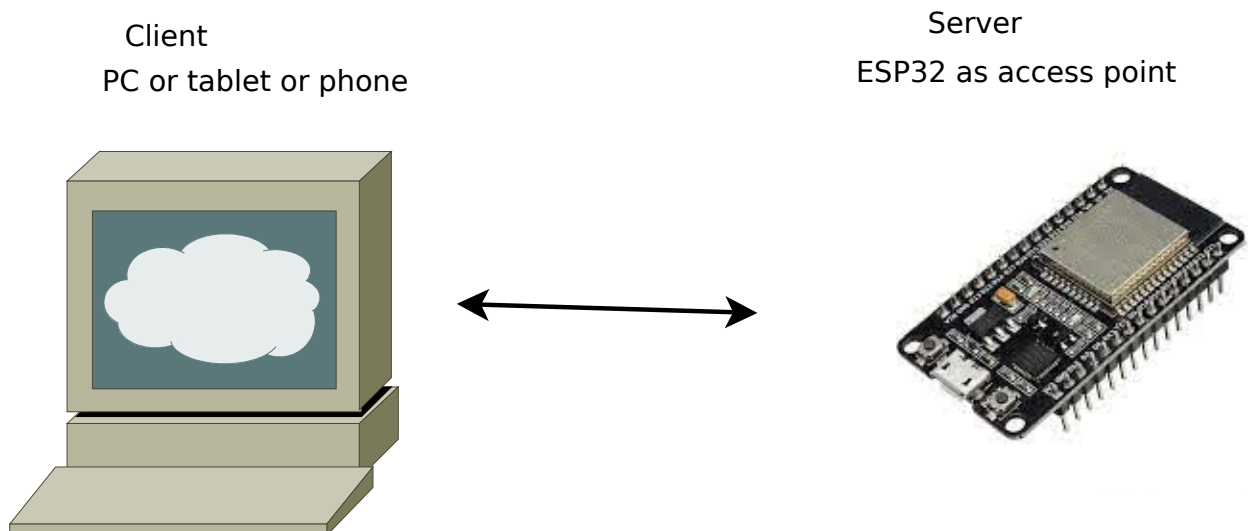
```
ssid = '*****'  
pwd = '*****'  
  
def connect_wifi():  
    import network  
    mynet = network.WLAN(network.STA_IF)  
    if not mynet.isconnected():  
        mynet.active(True)  
        mynet.connect(ssid, pwd)  
    ip = mynet.ifconfig()[0]  
    return ip  
  
ip = connect_wifi()  
print (ip)
```

(replace ssid and pwd with your credentials)

We can test the connection from the PC with ping:

```
jcf@jcf-Labo:~$ ping 192.168.179.41
PING 192.168.179.41 (192.168.179.41) 56(84) bytes of data.
64 bytes from 192.168.179.41: icmp_seq=1 ttl=255 time=171 ms
64 bytes from 192.168.179.41: icmp_seq=2 ttl=255 time=39.2 ms
...
```

1.2. The ESP32 as access point: Access point mode



The accesspoint on the ESP32 can be created by this code:

```
import network

ssid = 'MicroPython-AP'
password = '123456789'

def create_accesspoint(ssid, password):
    ap = network.WLAN(network.AP_IF)
    ap.active(True)
    ap.config(essid=ssid, password=password)

    while ap.active() == False:
        pass

    print('Access point ready!')
    ipinfo = ap.ifconfig()
    print(ipinfo)
    return ipinfo[0]
```

```
ip = create_accesspoint(ssid, password)
```

After this, an accesspoint with the name 'MicroPython-AP' is ready.

1.3. Setting a fixed IP address for the Access point

When there is no serial connection to a PC, e.g. when a smartphone or tablet is used, the IP address cannot be read from the output of the server program. So it is better to set it in the program as fixed IP address.

It seems that the default IP address is 192.168.4.1, but how can you be sure?

To set the address, our `create_accesspoint` function can be modified:

```
ipinfo = ('192.168.4.2', '255.255.255.0', '192.168.4.1', '8.8.8.8')
#         IP address, subnet mask, gateway and DNS server

def create_accesspoint(ssid, password, ipinfo):
    ap = network.WLAN(network.AP_IF)
    ap.active(True)
    ap.config(essid=ssid, password=password)
    if ipinfo:
        ap.ifconfig(ipinfo)

    while ap.active() == False:
        pass

    print('Access point ready!')
    ipinfo = ap.ifconfig()
    print(ipinfo)
    return ipinfo[0]

ip = create_accesspoint(ssid, password, ipinfo)
```

The 8.8.8.8 address of the DNS server is the address of the Google server. As we have no connection to Internet, it is irrelevant.

This method also works for the station mode.

1.4. A small WiFi library for connection or accesspoint

wifi2.py:

```
import network

'''Accesspoint'''
def create_accesspoint(ssid, password, ipinfo=''):
    ap = network.WLAN(network.AP_IF)
    ap.active(True)
    ap.config(essid=ssid, password=password)
```

```

    if ip:
        ipinfo = create_ipinfo(ip)
        print('Setting ipinfo ', ipinfo)
        ap.ifconfig(ipinfo)

    while ap.active() == False:
        pass

    print('Access point ready!')
    ipinfo = ap.ifconfig()
    print(ipinfo)
    return ipinfo[0]
#-----
'''Connect to existing WiFi'''
def connect_wifi(ssid, pwd, ip = ''):

    mynet = network.WLAN(network.STA_IF)
    if not mynet.isconnected():
        mynet.active(True)
        mynet.connect(ssid, pwd)
    if ip:
        ipinfo = create_ipinfo(ip)
        print('Setting ipinfo ', ipinfo )
        mynet.ifconfig(ipinfo)
    ip = mynet.ifconfig()[0]
    return ip
#-----
def create_ipinfo(ip):
    '''in: ip
       out: ipinfo needed by wifi, assuming 24bit IP subnet mask'''
    ipinfo = (ip, '255.255.255.0', ip, '8.8.8.8')
    return ipinfo

```

Here the helper function `create_ipinfo` creates the tuple (ipaddress, subnet mask, gateway, DHCP) needed by `ifconfig`, assuming a subnet mask of 255.255.255.0.

Use it like this:

```

ssid = 'MicroPython-AP'
password = '123456789'
ip = '192.168.179.41'          # optional

from wifi2 import *

# 4 possible modes:
# ip = connect_wifi(ssid, password)
# ip = connect_wifi(ssid, password, ip)
# ip = create_accesspoint(ssid, password)
ip = create_accesspoint(ssid, password, ip)
print(ip)

```

If you want to create an accesspoint or connect to an existing WiFi at boot time, put the above code in a file `main.py`.