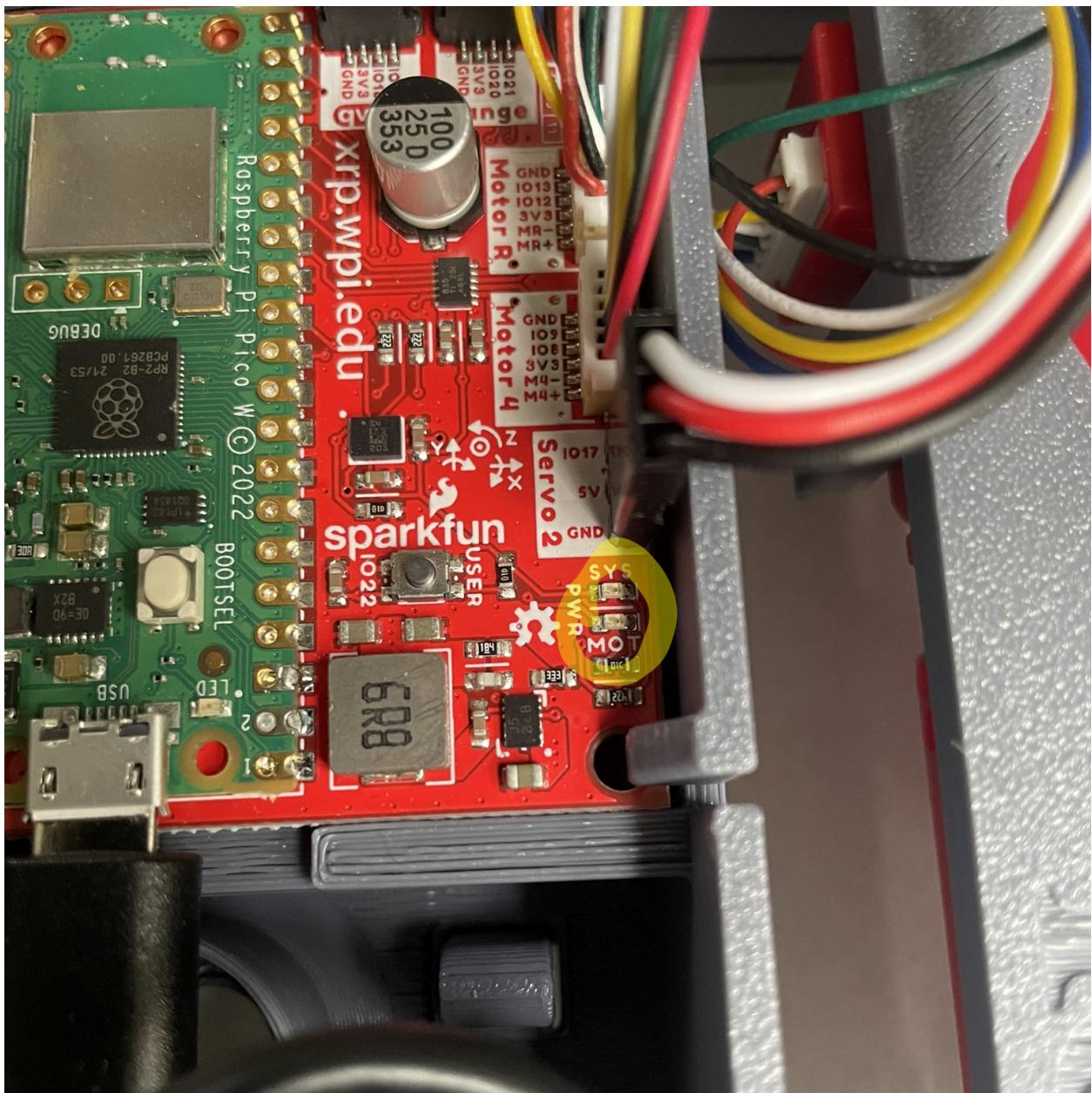


WIFI Module Quickstart

First make sure that the robot is fully powered off. Both power lights should be off, then the usb to your computer.



Once it's connected, run the example program. Then connect to the wifi network configured in example.

The terminal window shows the following Python code:

```
wifi nonblocking.py × wifi_testing.py × APpy × wifi.py × wifi_example.py ×
1 from XRPLib.defaults import *
2 from wifi import *
3 import time
4
5 ssid = 'ServerTest' #Replace with your AP's name
6 passkey = 'password' #Replace with your AP's password
7 ip = '192.168.4.30' #Replace with desired ip address, usually in the form 192.168.4.*** replacing *** with your desired number.
8
9 #####
10 # NOTES #
11 #####
12 #####
13 # Closing the remote session will restart your robot's wifi module requiring a full restart.
14 # Fully power down robot before starting program after running off battery
15 #####
16 # Default port is 50000
17 #####
18 #####
19 #####
20 ##### EXAMPLE FUNCTIONS #####
21 #####
22 #####
23 #####
24 # The following are example functions that can be run as called from the remote terminal.
25 # These functions can be modified, appended etc so long as they are passed to net.add.functions
26 # with the appropriate keyword. For instance, when "w" is sent from the remote terminal the function
27 # "forward" is called.
28 # The return value will be printed to the remote terminal.
29 #####
30 #####
31 def forward_fast():
32     drivetrain.set_speed(100, 100)
33     time.sleep(1)
34     drivetrain.set_speed(0,0)
35     return 'Moving Forward Fast'
36 #####
37 def forward():
38     drivetrain.set_effort(0.5, 0.5)
39     time.sleep(.6)
40     drivetrain.set_effort(0,0)
41     return 'Moving Forward'
```

Below the code, the terminal shows:

```
Shell
Hosting AP with:
ssid: ServerTest, password: robottest
Starting socket
-----
MicroPython v1.22.2 on 2024-02-22; Raspberry Pi Pico W with RP2040
Type "help()" for more information.
>>>
OK
Hosting AP with:
ssid: ServerTest, password: password
Starting socket
```

A yellow arrow points from the line "ssid: ServerTest, password: password" in the terminal to the "password" field in the WiFi settings dialog on the right. Another yellow arrow points from the "password" field in the dialog to the "password" field in the terminal.

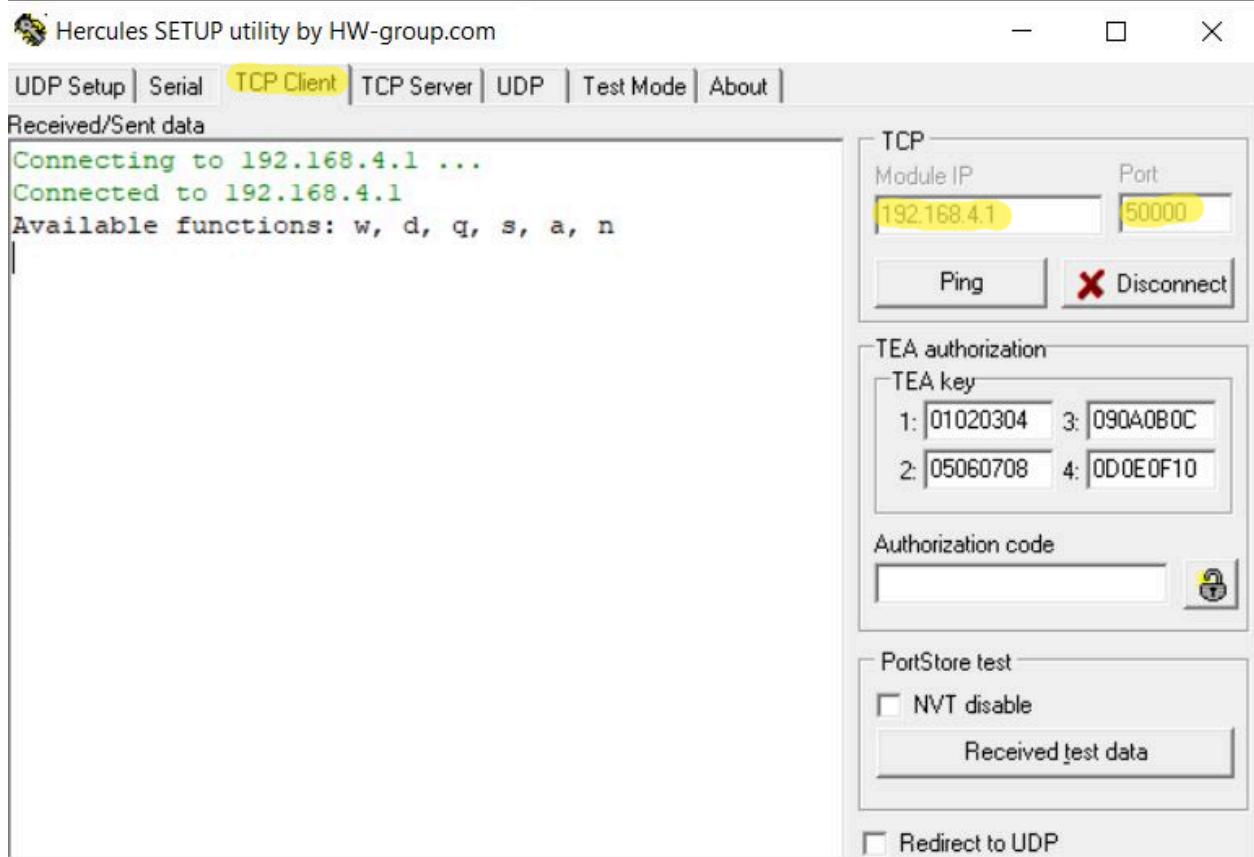
The WiFi settings dialog shows:

- SSID: ServerTest (Secured)
- Network security key: password
- Note: Some information changed since the last time you connected. We need additional information to complete the connection.
- Buttons: Next, Cancel

The Network & Internet settings screen shows:

- Wi-Fi: Bruin-Secure (Connected)
- DIRECT-DC-HP Smart Tank 5100
- Mobile hotspot: off
- Change settings, such as making a connection metered.

Once you're connected, open up hercules and use tcp client to connect to 192.168.4.1 on port 50000.



Now you should be all set! You can get a better idea of how the module works by playing around with the example. All of the functions have more outlined descriptions included in the example.