

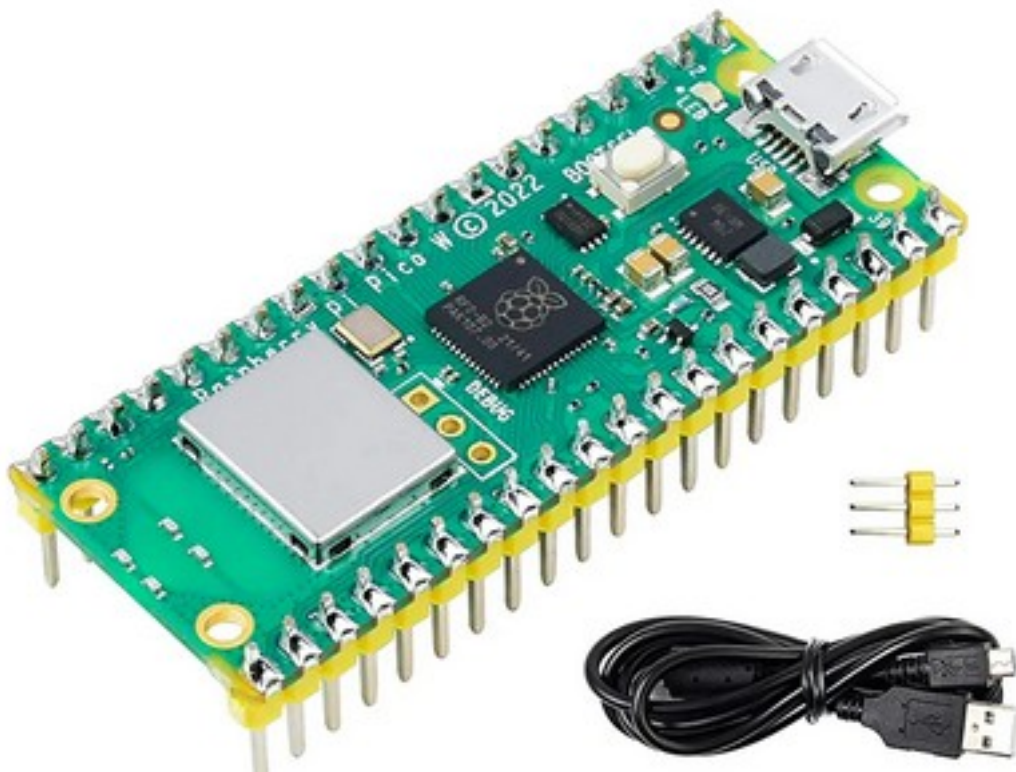
*****Draft*****

Adding a 2nd socket to pico_w freertos iperf
To provide Debug information previously provided by hard wired connection to UART
03/30/23

*****Draft*****

In the process of converting my “https://github.com/develone/pico_w-remotes.git”.

I have several pico_w connected to my home Wifi. Currently a 384 byte debug is sent to RPi4-4GB using (cli1, cli2, cli3, cli4, cli5, and cli6).



```
“git clone https://github.com/develone/pico_w-mqtt.git -b dev”  
“cd pico_w-mqtt”
```

Modify the script “6remotes.sh” WIFI_SSID with your SSID and WIFI_PASSWORD with your PASSWORD.

Modify the file “pico_w/wifi/freertos/iperf/picow_freertos_iperf.c” WIFI_PASSWORD with your PASSWORD.

“./6remotes.sh” creates 6 copies of the program

“remotex/pico_w/wifi/freertos/iperf/picow_freertos_iperf_server_nosys.elf” each with a different hostname. In addition copies “exe-ocd.sh” to each of the six folders remotex.

It also runs the script “build_cli.sh”.

The script “build_cli.sh” creates 6 programs (cli1, cli2, cli3, cli4, cli5, and cli6) in the folder pi_tcp_tests.

```
#!/bin/bash
```

```
cd pi_tcp_tests
```

```
rm -f cli1 cli2 cli5 cli6
```

```
gcc -v client.c -Drem1 -o cli1
```

```
gcc -v client.c -Drem2 -o cli2
```

```
gcc -v client.c -Drem3 -o cli3
```

```
gcc -v client.c -Drem4 -o cli4
```

```
gcc -v client.c -Drem5 -o cli5
```

```
gcc -v client.c -Drem6 -o cli6
```

The USB to UART is currently used to see the debug from pico_w. This will be remove and debug will be available using programs (cli1, cli2, cli3, cli4, cli5, and cli6).



and connected
to the RPi4B
4Gb USB to
see the debug output.

Now this can be done with the programs (cli1, cli2, cli3, cli4, cli5, and cli6).

Examples of the programming & debug are found
“https://github.com/develone/pico_w-mqtt/blob/dev/doc/info.txt”.

Modified output “https://github.com/develone/pico_w-mqtt/blob/dev/doc/info_1.txt”.

The buffer now is 512 bytes. The first 256 is used for booting information and the next 256 are used following the connection to WiFi. **Note: mqtt_connected 0 then mqtt_connected 1 which is when the connection to the Mosquitto Broker.**

```
devel@pi4-30:~/pico_w-mqtt/remote5 $ ../pi_tcp_tests/cli1
Socket created successfully
Connected with server successfully
Starting FreeRTOS on core 0: ver 0.0.01 remote1
Connecting to Wi-Fi...
Connected. iperf server 192.168.1.176 4001
starting watchdog timer task
mqtt_ip = 0xd401a8c0 mqtt_port = 1883
saved_mqtt_client 0x20021fcc check_mqtt_connected 0
saved_mqtt_client 0x_connected 1
saved_mqtt_client 0x20021fcc check_mqtt_connected 1
saved_mqtt_client 0x20021fcc check_mqtt_connected 1
saved_mq??
```