**********Default**********

pico_w or
nano-rp2040-connect
access point
for pico_w
11/14/22

**********Default**********

This needs to be appended to the end of .bashrc file.
export PICO_SDK_PATH=/home/devel/sdk/pico-sdk
The pico-sdk is requited in this folder with its sub modules.
The submodules are tinyusb and the wifi lib
The FreeRTOS-Kernel is needed in **/home/devel/FreeRTOS-Kernel**
Several of files need modifications when you want to change the WIFI_SSID and WIFI_PASSWORD if the first character WIFI_PASSWORD is not a number it can be put in the pw_ssid.h file

modified: pico_w/freertos/iperf/picow_freertos_iperf.c
modified: pico_w/freertos/iperf/pw_ssid.h
modified: pico_w/freertos/ping/picow_freertos_ping.c
modified: pico_w/freertos/ping/pw_ssid.h
modified: pico_w/iperf/picow_iperf.c
modified: pico_w/iperf/pw_ssid.h
modified: pico_w/ntp_client/picow_ntp_client.c
modified: pico_w/ntp_client/pw_ssid.h
modified: pico_w/tcp_client/picow_tcp_client.c
modified: pico_w/tcp_client/pw_ssid.h
modified: pico_w/tcp_server/picow_tcp_server.c
modified: pico_w/tcp_server/pw_ssid.h

Either the pico_w or the Arduino Nano-RP2040-connect can be be used as the access -point.

This will require minor changes in the file pico-examples/pico_w/access_point
line 130 const char *ap_name = "picow_test"; –> const char *ap_name = "nanotest";
line 132 const char *password = "password"; –> const char *password = "12345678";
line 140 IP4_ADDR(&gw, 192, 168, 4, 1); –> IP4_ADDR(&gw, 10, 0, 1, 10);

Pico_W

https://www.amazon.com/Vis-Viva-Raspberry-Basic-PICO-W/dp/B0BDSPXP2R/ref=mp_s_a_1_2?crid=2RQUZH6WT21DV&keywords=pico_w&qid=1666627291&sprefix=pico_w%2Caps%2C269&sr=8-2

$17,95

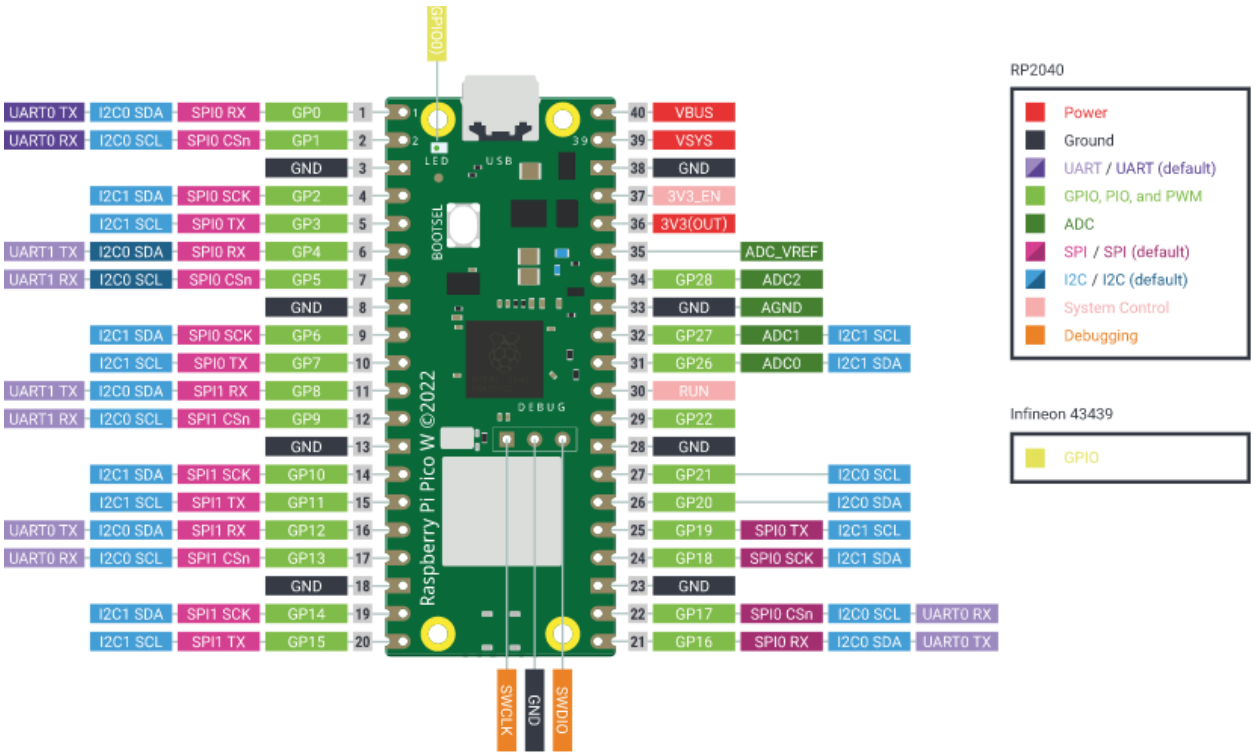
pico_w pins

git clone https://github.com/develone/FreeRTOS-Kernel.git

git clone https://github.com/develone/pico_w-remotes.git

cd pico_w-remotes

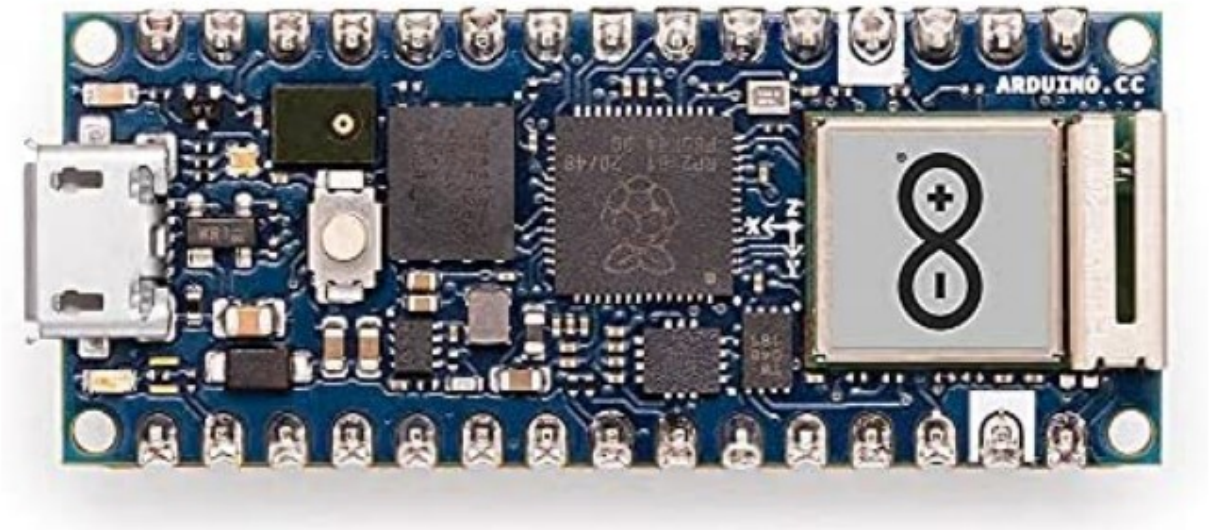
./5remotes.sh



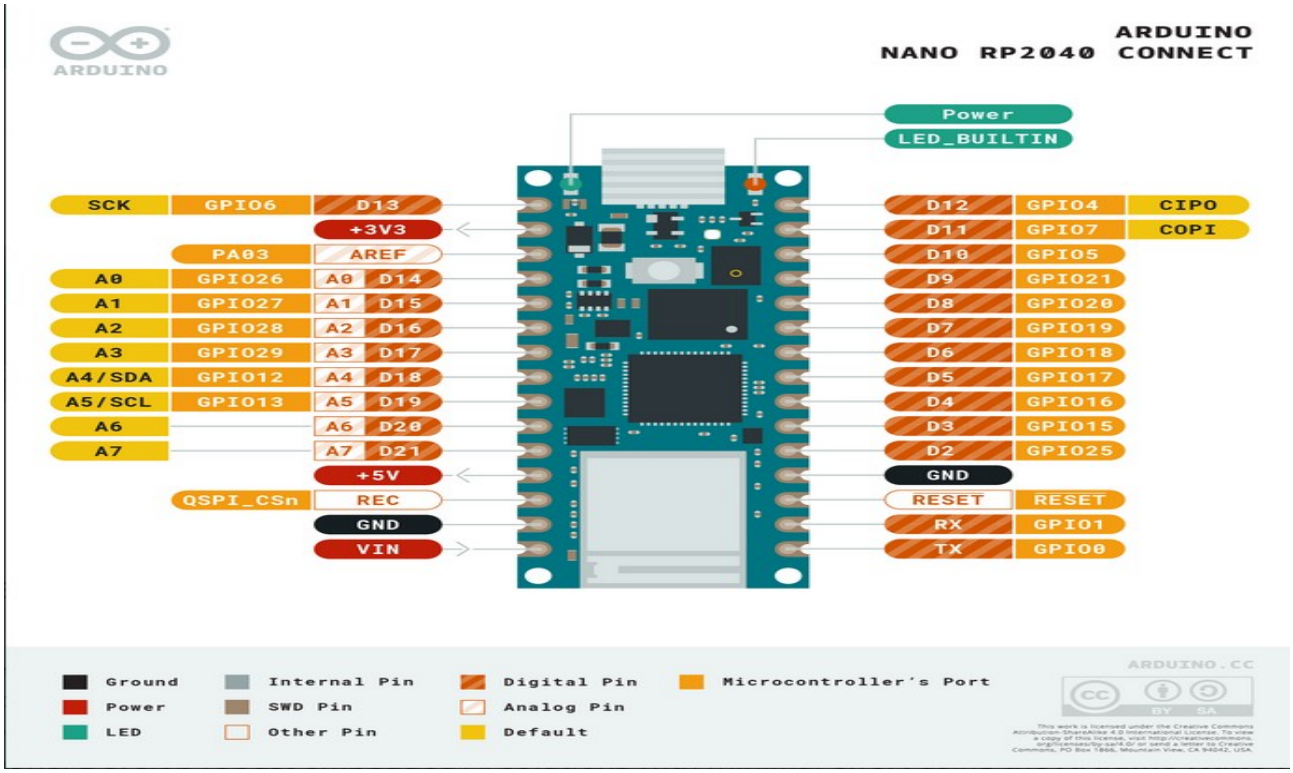
xx
Arduino Nano-RP2040-Connect

https://www.amazon.com/Arduino-Nano-RP2040-Connect-Headers/dp/B095J4KFVT/ref=asc_df_B095J4KFVT/?tag=hyprod-20&linkCode=df0&hvadid=533458241275&hvpos=&hvnetw=g&hvrand=11004597131761763700&hvpone=&hvptwo=&hvqmt=&hvdev=m&hvdvcmdl=&hvlocint=&hvlocphy=9028705&hvtargid=pla-1588131359952&psc=1

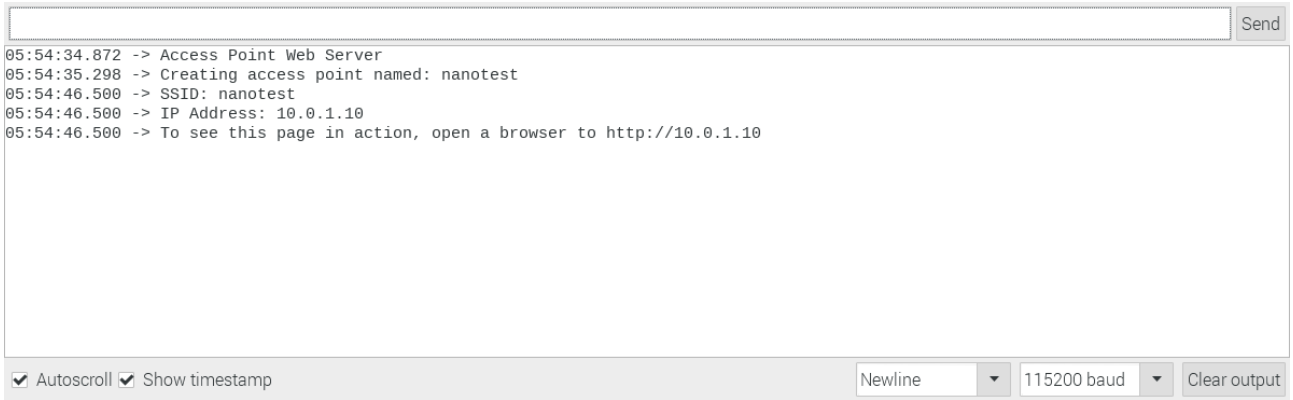
$34.99



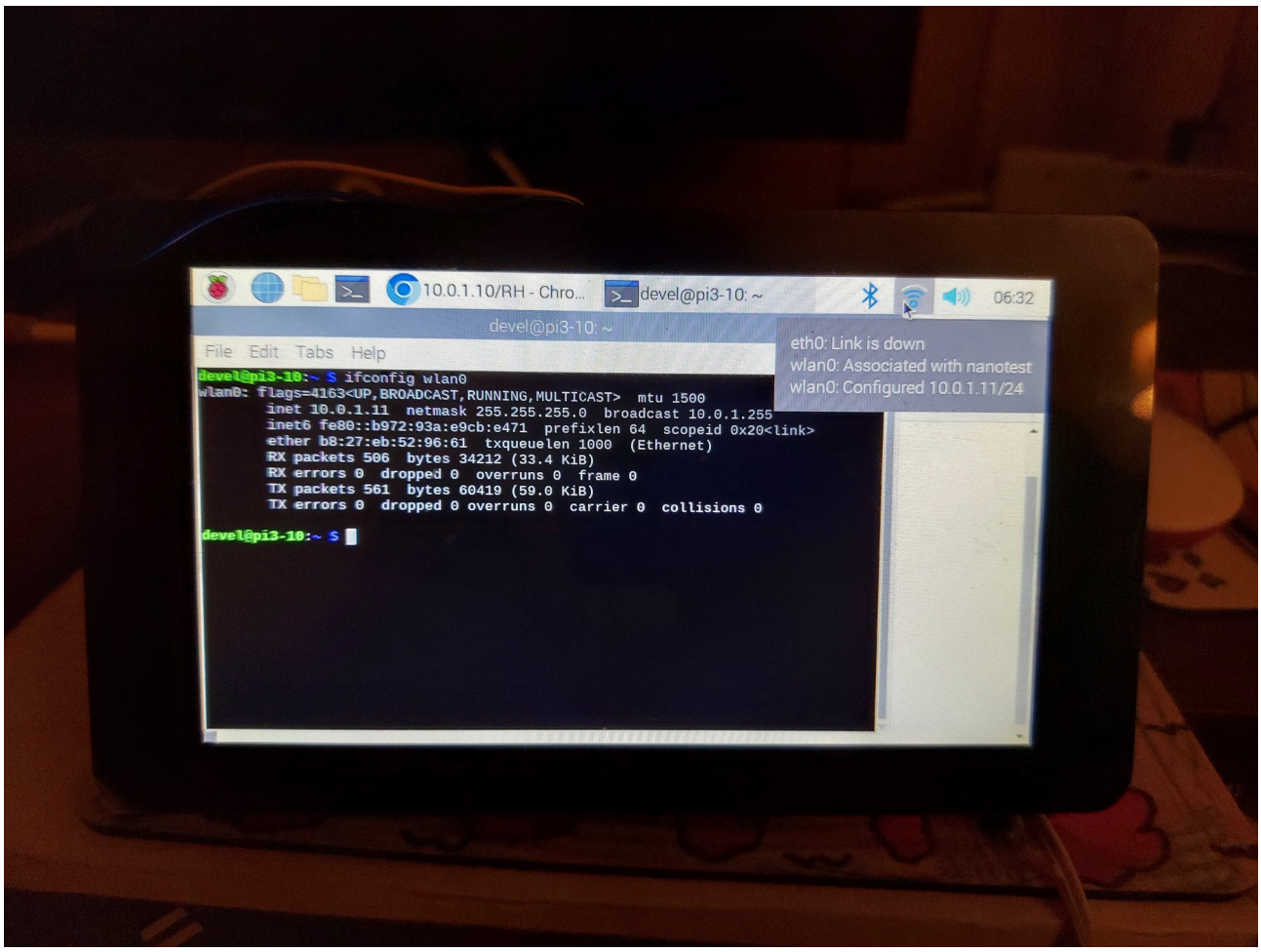
xx



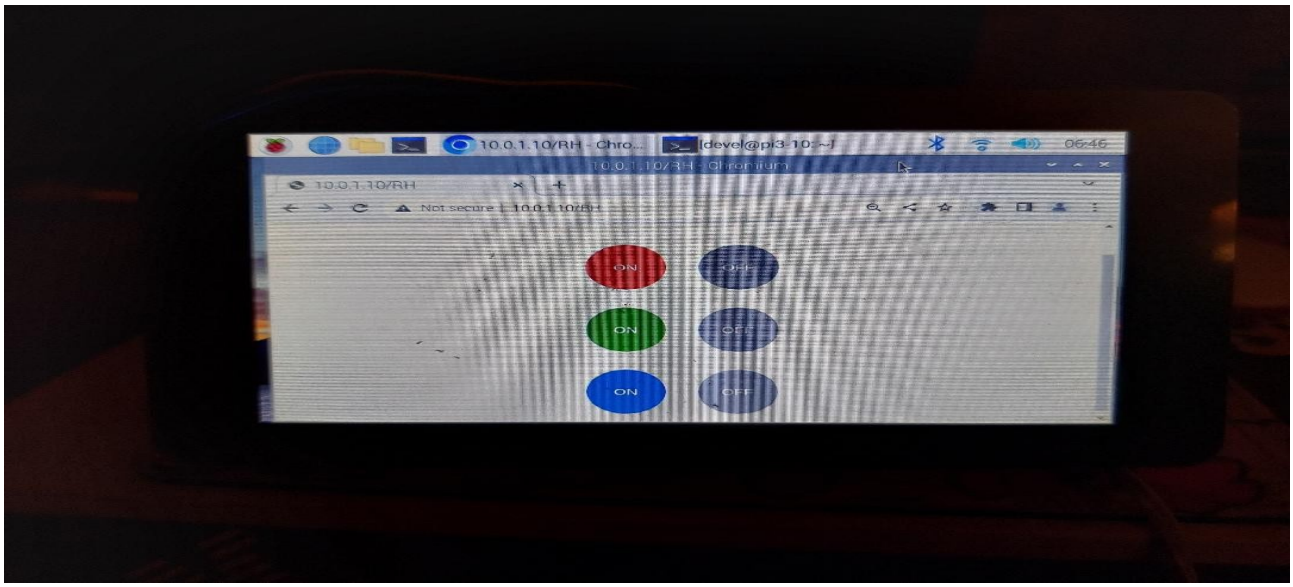
Programmed the Arduino Nano-RP2040-Connect as WIFI access point. This provides a DHCP server.



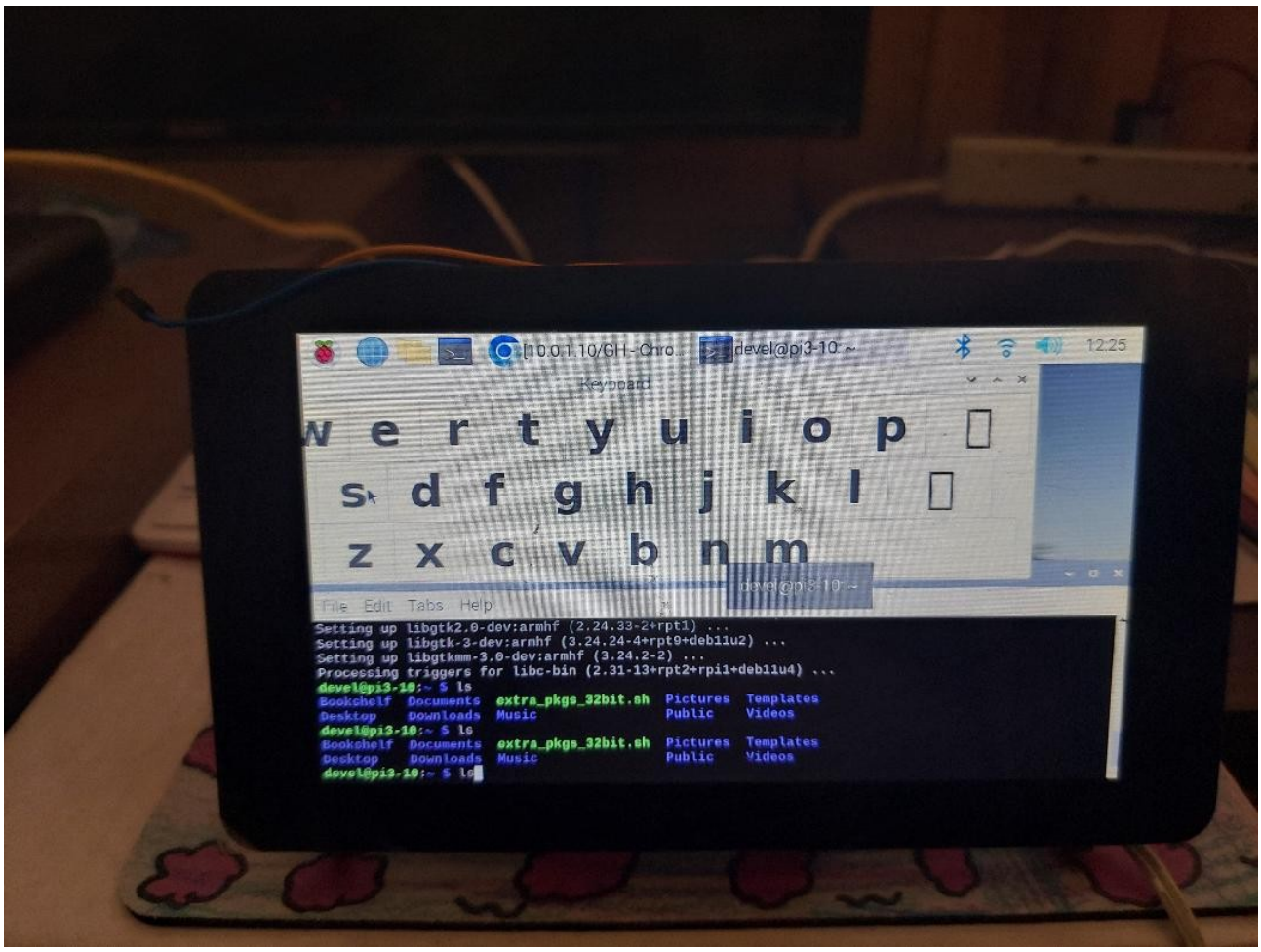
Two Raspberry Pi3B with 7 in displats were used as remotes on this network. The IP assigned to pi3-10 was 10.0.1.11. The IP assigned to pi3-11 was 10.0.1.12.



The access point also provides a web server to control the RGB Led on the Nano-RP2040-Connect.



Adding a virtual keyboard will make it easier to do the field tests.



xx

Press CTRL-A Z for help on special keys

Connecting to WiFi...
Connected.

Ready, running iperf server at 10.0.1.13
Completed iperf transfer of 10 MBytes @ 8.3 Mbits/sec
Total iperf megabytes since start 10 Mbytes

iperf -c 10.0.1.13

Client connecting to 10.0.1.13, TCP port 5001
TCP window size: 43.8 KByte (default)
---
[3] local 10.0.1.11 port 43808 connected with 10.0.1.13 port 5001
[ID] Interval Transfer Bandwidth
[3] 0.0000-10.0892 sec 10.0 MBytes 8.31 Mbits/sec

Client/Server
In the command below **note: -DTEST_TCP_SERVER_IP="10.0.1.13" -DWIFI_SSID="nanotest" -DWIFI_PASSWORD="12345678" are dependent on your WiFi.**

```
 cmake -DPICO_BOARD=pico_w -DTEST_TCP_SERVER_IP="10.0.1.13" -
DWIFI_SSID="nanotest" -DWIFI_PASSWORD="12345678" -
DFREERTOS_KERNEL_PATH="/home/devel/FreeRTOS-Kernel" ..



openocd -f interface/raspberrypi-swd.cfg -f target/rp2040.cfg -c "program
pico_w/wifi_scan/picow_wifi_scan_poll.elf verify reset exit"

Press CTRL-A Z for help on special keys


Performing wifi scan
ssid: nanotest                 rssi:  -50 chan:   3 mac: 30:c6:f7:01:8f7
ssid: ATTtpHTfPi                  rssi:  -11 chan:   1 mac: cc:ab:2c:c7:5e5
ssid: ATT3TV6WQs                     rssi:  -70 chan:   1 mac: c8:52:61:4e:d25
ssid: ATTtpHTfPi                  rssi:  -12 chan:   1 mac: cc:ab:2c:c7:5e5
ssid: nanotest                 rssi:  -51 chan:   3 mac: 30:c6:f7:01:8f7
ssid: ATT3TV6WQs                     rssi:  -71 chan:   1 mac: c8:52:61:4e:d25
ssid: House                    rssi:  -74 chan:   3 mac: c4:41:1e:4e:c35
ssid: nanotest                 rssi:  -53 chan:   3 mac: 30:c6:f7:01:8f7
ssid: nanotest                 rssi:  -51 chan:   3 mac: 30:c6:f7:01:8f7
ssid:                       rssi:  -71 chan:   3 mac: ca:41:1e:4e:c35
ssid:                       rssi:  -76 chan:   3 mac: ca:41:1e:4e:c35
ssid: ATTjw8tqXi                  rssi:  -82 chan:   6 mac: f4:17:b8:de:a65
ssid: ATTtpHTfPi                  rssi:  -46 chan:   1 mac: cc:ab:2c:c7:5e5
ssid: ATT47CJH5z_EXT                  rssi:  -70 chan:  11 mac: 3c:84:6a:46:987
ssid: ATT47CJH5z_EXT                  rssi:  -73 chan:  11 mac: 3c:84:6a:46:987
ssid: ATT47CJH5z_EXT                  rssi:  -73 chan:  11 mac: 3c:84:6a:46:987
ssid: ATT47CJH5z_EXT                  rssi:  -71 chan:  11 mac: 3c:84:6a:46:987
ssid: ATT47CJH5z_EXT                  rssi:  -74 chan:  11 mac: 3c:84:6a:46:987

openocd -f interface/raspberrypi-swd.cfg -f target/rp2040.cfg -c "program
pico_w/iperf/picow_iperf_server_background.elf verify reset exit"


openocd -f interface/raspberrypi-swd.cfg -f target/rp2040.cfg -c "program
pico_w/tcp_server/picow_tcpip_server_background.elf verify reset exit"

Back from buildCRCTable
0xd3 0x1 0x2e
Connecting to WiFi...
Connected.
Starting server at 10.0.1.13 on port 4242
Client connected
Writing 2048 bytes to client
tcp_server_sent 1460
tcp_server_sent 588
Waiting for buffer from client
tcp_server_recv 1460/0 err 0
tcp_server_recv 588/1460 err 0
tcp_server_recv buffer ok
```

Writing 2048 bytes to client
tcp_server_sent 1460
tcp_server_sent 588
Waiting for buffer from client
tcp_server_recv 1460/0 err 0
tcp_server_recv 588/1460 err 0
tcp_server_recv buffer ok
Writing 2048 bytes to client
tcp_server_sent 1460
tcp_server_sent 588
Waiting for buffer from client
tcp_server_recv 1460/0 err 0
tcp_server_recv 588/1460 err 0
tcp_server_recv buffer ok
Writing 2048 bytes to client
tcp_server_sent 1460
tcp_server_sent 588
Waiting for buffer from client
tcp_server_recv 1460/0 err 0
tcp_server_recv 588/1460 err 0
tcp_server_recv buffer ok
Writing 2048 bytes to client
tcp_server_sent 1460
tcp_server_sent 588
Waiting for buffer from client
tcp_server_recv 1460/0 err 0
tcp_server_recv 588/1460 err 0
tcp_server_recv buffer ok
Writing 2048 bytes to client
tcp_server_sent 1460
tcp_server_sent 588
Waiting for buffer from client
tcp_server_recv 1460/0 err 0
tcp_server_recv 588/1460 err 0
tcp_server_recv buffer ok
Writing 2048 bytes to client
tcp_server_sent 1460
tcp_server_sent 588
Waiting for buffer from client
tcp_server_recv 1460/0 err 0
tcp_server_recv 588/1460 err 0
tcp_server_recv buffer ok
Writing 2048 bytes to client
tcp_server_sent 1460
tcp_server_sent 588
Waiting for buffer from client
tcp_server_recv 1460/0 err 0
tcp_server_recv 588/1460 err 0
tcp_server_recv buffer ok
Writing 2048 bytes to client
tcp_server_sent 1460
tcp_server_sent 588

Waiting for buffer from client
tcp_server_recv 1460/0 err 0
tcp_server_recv 588/1460 err 0
tcp_server_recv buffer ok
Writing 2048 bytes to client
tcp_server_sent 1460
tcp_server_sent 588
Waiting for buffer from client
tcp_server_recv 1460/0 err 0
tcp_server_recv 588/1460 err 0
tcp_server_recv buffer ok
test success

openocd -f interface/raspberrypi-swd.cfg -f target/rp2040.cfg -c "program pico_w/tcp_client/picow_tcpip_client_background.elf verify reset exit"

Connecting to WiFi...
Connected.
Connecting to 10.0.1.13 port 4242
Waiting for buffer from server
recv 1460 err 0
recv 588 err 0
Writing 2048 bytes to server
tcp_client_sent 1460
tcp_client_sent 588
Waiting for buffer from server
recv 1460 err 0
recv 588 err 0
Writing 2048 bytes to server
tcp_client_sent 1460
tcp_client_sent 588
Waiting for buffer from server
recv 1460 err 0
recv 588 err 0
Writing 2048 bytes to server
tcp_client_sent 1460
tcp_client_sent 588
Waiting for buffer from server
recv 1460 err 0
recv 588 err 0
Writing 2048 bytes to server
tcp_client_sent 1460
tcp_client_sent 588
Waiting for buffer from server
recv 1460 err 0
recv 588 err 0
Writing 2048 bytes to server
tcp_client_sent 1460
tcp_client_sent 588
Waiting for buffer from server
recv 1460 err 0
recv 588 err 0

```
Writing 2048 bytes to server
tcp_client_sent 1460
tcp_client_sent 588
Waiting for buffer from server
recv 1460 err 0
recv 588 err 0
Writing 2048 bytes to server
tcp_client_sent 1460
tcp_client_sent 588
Waiting for buffer from server
recv 1460 err 0
recv 588 err 0
Writing 2048 bytes to server
tcp_client_sent 1460
tcp_client_sent 588
Waiting for buffer from server
recv 1460 err 0
recv 588 err 0
Writing 2048 bytes to server
tcp_client_sent 1460
tcp_client_sent 588
Waiting for buffer from server
recv 1460 err 0
recv 588 err 0
Writing 2048 bytes to server
tcp_client_sent 1460
tcp_client_sent 588
test success

openocd -f interface/raspberrypi-swd.cfg -f target/rp2040.cfg -c "program
pico_w/freertos/iperf/picow_freertos_iperf_server_sys.elf verify reset exit"

Welcome to minicom 2.8

OPTIONS: I18n
Port /dev/ttyUSB0, 05:23:50

Press CTRL-A Z for help on special keys

Back from buildCRCTable
0xd3 0x1 0x2e
Starting FreeRTOS on core 0:
Connecting to WiFi...
Connected.

Ready, running iperf server at 10.0.1.13
blink_task starts

iperf -c 10.0.1.13
------------------------------------------------------------
Client connecting to 10.0.1.13, TCP port 5001
TCP window size: 43.8 KByte (default)
```

```
------------------------------------------------------------
[  3] local 10.0.1.12 port 47696 connected with 10.0.1.13 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3] 0.0000-10.3064 sec  7.13 MBytes  5.80 Mbits/sec
devel@pi3-11:~ $ iperf -c 10.0.1.13
------------------------------------------------------------
Client connecting to 10.0.1.13, TCP port 5001
TCP window size: 43.8 KByte (default)
------------------------------------------------------------
[  3] local 10.0.1.11 port 50480 connected with 10.0.1.13 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3] 0.0000-10.1188 sec  7.50 MBytes  6.22 Mbits/sec


Completed iperf transfer of 7 MBytes @ 6.2 Mbits/sec
Total iperf megabytes since start 7 Mbytes

openocd -f interface/raspberrypi-swd.cfg -f target/rp2040.cfg -c "program
pico_w/freertos/iperf/picow_freertos_iperf_server_nosys.elf verify reset exit"

Welcome to minicom 2.8

OPTIONS: I18n
Port /dev/ttyUSB0, 05:23:50

Press CTRL-A Z for help on special keys

Back from buildCRCTable
0xd3 0x1 0x2e
Starting FreeRTOS on core 0:
Connecting to WiFi...
Connected.

Ready, running iperf server at 10.0.1.13
blink_task starts

iperf -c 10.0.1.13
------------------------------------------------------------
Client connecting to 10.0.1.13, TCP port 5001
TCP window size: 43.8 KByte (default)
------------------------------------------------------------
[  3] local 10.0.1.11 port 50480 connected with 10.0.1.13 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3] 0.0000-10.1188 sec  7.50 MBytes  6.22 Mbits/sec
devel@pi3-11:~ $ iperf -c 10.0.1.13
------------------------------------------------------------
Client connecting to 10.0.1.13, TCP port 5001
TCP window size: 43.8 KByte (default)
------------------------------------------------------------
[  3] local 10.0.1.11 port 60174 connected with 10.0.1.13 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3] 0.0000-10.2288 sec  7.88 MBytes  6.46 Mbits/sec
```

Completed iperf transfer of 7 MBytes @ 6.4 Mbits/sec
Total iperf megabytes since start 7 Mbytes

openocd -f interface/raspberrypi-swd.cfg -f target/rp2040.cfg -c "program pico_w/freertos/ping/picow_freertos_ping_nosys.elf verify reset exit"

Starting FreeRTOS on core 0:
Connecting to WiFi...
Connected.
ping 10.0.1.10 10.0.1.10
ping 10.0.1.10 10.0.1.10
ping 10.0.1.10 10.0.1.10
ping 10.0.1.10 10.0.1.10
ping 10.0.1.10 10.0.1.10
ping 10.0.1.10 10.0.1.10