Loading the picow_tcpip_server_background.elf using openocd.

```
devel@pi4-27:~/pico-examples/build $ openocd -f interface/raspberrypi-swd.cfg -f target/rp2040.cfg -c
"program pico_w/tcp_server/picow_tcpip_server_background.elf verify reset exit"
Open On-Chip Debugger 0.11.0-g610f137-dirty (2022-05-03-08:59)
Licensed under GNU GPL v2
For bug reports, read
	http://openocd.org/doc/doxygen/bugs.html
adapter speed: 1000 kHz

Info : Hardware thread awareness created
Info : Hardware thread awareness created
Info : RP2040 Flash Bank Command
Info : BCM2835 GPIO JTAG/SWD bitbang driver
Info : clock speed 1001 kHz
Info : SWD DPIDR 0x0bc12477
Info : SWD DLPIDR 0x00000001
Info : SWD DPIDR 0x0bc12477
Info : SWD DLPIDR 0x10000001
Info : rp2040.core0: hardware has 4 breakpoints, 2 watchpoints
Info : rp2040.core1: hardware has 4 breakpoints, 2 watchpoints
Info : starting gdb server for rp2040.core0 on 3333
Info : Listening on port 3333 for gdb connections
target halted due to debug-request, current mode: Thread
xPSR: 0xf1000000 pc: 0x000000ea msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0xf1000000 pc: 0x000000ea msp: 0x20041f00
** Programming Started **
Info : RP2040 B0 Flash Probe: 2097152 bytes @10000000, in 512 sectors

target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
Info : Writing 323584 bytes starting at 0x0
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
```

```
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
** Programming Finished **
** Verify Started **
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
```

```
target halted due to debug-request, current mode: Thread
xPSR: 0x01000000 pc: 0x00000138 msp: 0x20041f00
** Verified OK **
** Resetting Target **
shutdown command invoked
```

This is what you see when the pico-w sends after the program is loaded.

```
Connecting to WiFi...
Connected.
Starting server at 192.168.1.159 on port 4242
[vidal@laptop ~]$ ./client '192.168.1.159'
After the send function
�/���h��<�5�*��@��Rqjr)#�[=: -�F)�x�h��?+���z�
          ,�Wtdį(��W�^ �8�N��%V$D�Y�C{����~w�/&�����0�4�:
fter the send function
Received in pid=3476, text=: �*��D0�k�����IɥB�9�s�����!
�u�Ƈ�X�$�]D@�Z#^����?
��zAr���l���4�4#z3p��R�~Ҕ�f#Y�o�����
After the send function
Received in pid=3476, text=:
��9���_����$�a�xE�<Gx9Gə�J�a;�j��D[��_u_A��8��Z�/F��5*8�*
��b�uUtp�         �_��z'X'k8�ہ��
�Uz�u5�>
After the send function
Received in pid=3476, text=: ql�'w�������e dmt
                                          ]�`6��[□□����
�ZH�ʼniÇ2�q��Q���������Wg�����S��Dـ
ɋL�����$t����A�OB��s��T�
After the send function
Received in pid=3476, text=: ���Vc{`��������8Mf:�P�g8��f�G��d�
)�o� r*%�����-4�h�vr����+Z/&����8��ٲ                              �`H�f��'s\
��I0�!�
After the send function
Received in pid=3476, text=: x��
r�*L�wNJ�}
```

```
Client connected
Writing 2048 bytes to client
tcp_server_recv 14/0 err 0
tcp_server_sent 1460
tcp_server_sent 588
Waiting for buffer from client
tcp_server_recv 14/0 err 0
tcp_server_recv 14/14 err 0
tcp_server_recv 14/28 err 0
tcp_server_recv 14/42 err 0
tcp_server_poll_fn
test failed -1
```

Need to modify the client.c which I found on the Internet.

```
/************************************************************************/
/*                                                                      */
/* Client program which gets as parameter the server name or            */
/*    address and tries to send the data into non-blocking server.      */
/*                                                                      */
/* The message is sent after 5 seconds of wait                          */
/*                                                                      */
/*                                                                      */
/* based on Beej's program - look in the simple TCp client for further doc.*/
/*                                                                      */
/*                                                                      */
/************************************************************************/
   #include <stdio.h>
   #include <stdlib.h>
   #include <errno.h>
   #include <string.h>
   #include <netdb.h>
   #include <sys/types.h>
   #include <netinet/in.h>
   #include <sys/socket.h>
   #include <unistd.h>

   #define PORT 4242    /* the port client will be connecting to */

   #define MAXDATASIZE 100 /* max number of bytes we can get at once */

   int main(int argc, char *argv[])
   {
      int sockfd, numbytes;
      char buf[MAXDATASIZE];
      struct hostent *he;
      struct sockaddr_in their_addr; /* connector's address information */

      if (argc != 2) {
         fprintf(stderr,"usage: client hostname\n");
         exit(1);
      }

      if ((he=gethostbyname(argv[1])) == NULL) {  /* get the host info */
         herror("gethostbyname");
         exit(1);
      }
```

```
    if ((sockfd = socket(AF_INET, SOCK_STREAM, 0)) == -1) {
        perror("socket");
        exit(1);
    }

    their_addr.sin_family = AF_INET;      /* host byte order */
    their_addr.sin_port = htons(PORT);    /* short, network byte order */
    their_addr.sin_addr = *((struct in_addr *)he->h_addr);
    bzero(&(their_addr.sin_zero), 8);     /* zero the rest of the struct */

    if (connect(sockfd, (struct sockaddr *)&their_addr, \
                                          sizeof(struct sockaddr)) == -1) {
        perror("connect");
        exit(1);
    }
        while (1) {
                    if (send(sockfd, "Hello, world!\n", 14, 0) == -1){
                  perror("send");
                          exit (1);
                    }
                    printf("After the send function \n");

        if ((numbytes=recv(sockfd, buf, MAXDATASIZE, 0)) == -1) {
                                perror("recv");
                                exit(1);
                    }

                buf[numbytes] = '\0';

        printf("Received in pid=%d, text=: %s \n",getpid(), buf);
                    sleep(1);

        }

    close(sockfd);

    return 0;
}
```