How to Make a J-LINK by Raspberry Pi Zero

1. Set up Raspberry Pi 4
   1. Download Raspberry Pi Imager from <https://www.raspberrypi.com/software/>
   2. Download the image from https://raspi.debian.net/daily/raspi\_4\_bullseye.img.xz
   3. Flash Debian TF card
   4. Edit /boot/config.txt and add ‘enable\_uart=1’ and ‘gpu\_freq=360 ’to the end of the file, and save.
   5. Login by user: root, no password
   6. apt-get update
   7. apt-get install net-tools
   8. apt-get install nano vim screen
   9. apt-get install openssh-server telnetd telnet
   10. Edit /etc/ssh/sshd\_config

nano /etc/ssh/sshd\_config

edit 'PermitRootLogin AUTH' to 'PermitRootLogin yes'

* 1. service ssh restart
  2. apt-get install samba
  3. smbpasswd -a (def:123456)
  4. mkdir /home/shared
  5. nano /etc/samba/smb.conf

#add this to the very end of the file:

[shared]

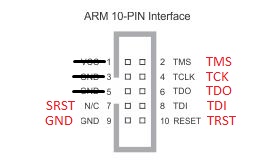
path = /home/shared

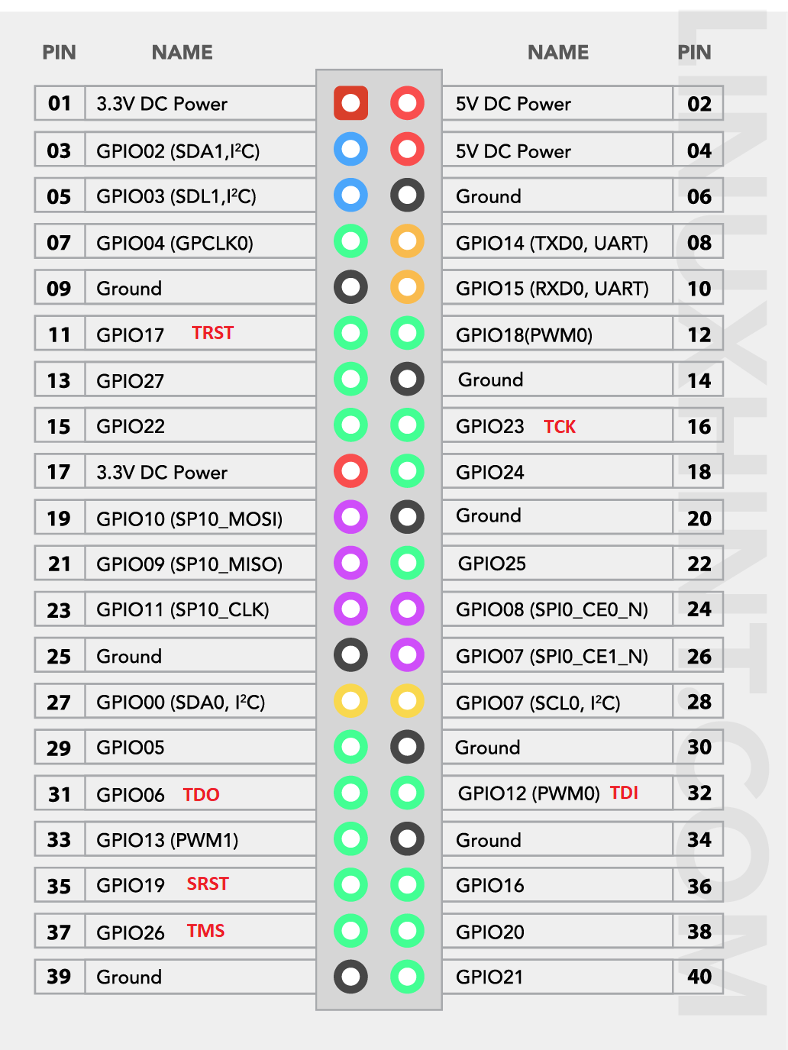
valid users = root

read only = no

* 1. service smbd restart
  2. testparm
  3. passwd root (def: 123456)

1. JTAG Pinout





1. Install and config OpenOCD (Also see: <https://neucrack.com/p/142>)
   1. sudo apt-get update
   2. sudo apt-get install -y autoconf libtool libftdi-dev libusb-1.0-0 libusb-1.0-0-dev git
   3. sudo apt-get install gpiod libgpiod-dev build-essential pkg-config libtool
   4. git clone --recursive git://git.code.sf.net/p/openocd/code openocd-code
   5. cd openocd-code
   6. ./bootstrap
   7. ./configure --enable-sysfsgpio\

--enable-linuxgpiod \

--enable-maintainer-mode \

--disable-werror \

--enable-ft2232\_libftdi \

--enable-ep93xx \

--enable-at91rm9200 \

--enable-usbprog \

--enable-presto\_libftdi \

--enable-jlink \

--enable-vsllink \

--enable-rlink \

--enable-arm-jtag-ew \

--enable-dummy \

--enable-buspirate \

--enable-ulink \

--enable-presto\_libftdi \

--enable-usb\_blaster\_libftdi \

--enable-ft2232\_libftdi\

--prefix=/usr

--disable-dependency-tracking

* 1. make
  2. make install
  3. git clone <https://github.com/Neutree/MT7688-OpenOCD> (download the cfg file from Angelic47)
  4. Edit jlink\_gpio.cfg for JTAG I/Os

adapter driver linuxgpiod

transport select jtag

linuxgpiod gpiochip 0

linuxgpiod jtag\_nums 23 26 12 3

linuxgpiod trst\_num 17

linuxgpiod srst\_num 19

adapter srst delay 100

jtag\_ntrst\_delay 100

reset\_config trst\_and\_srst

* 1. openocd -f jlink\_gpio.cfg
  2. telnet localhost 4444
  3. load\_image cc.mp4 0x88000000

1. **Reference:** GPIO commands example

gpiodetect # detect GPIO devices

gpioinfo 0 # check the configuration of GPIO device 0

gpioset 0 6=1 # set the GPIO line 6 of the gpio controller 0

gpioget 0 6 # get the GPIO line 6 of the gpio controller 0

gpiomon 0 6 # print out the behavior of GPIO line 6 of the gpio controller 0

1. Use Bus Priate to program SPI flash

apt-get install flashrom

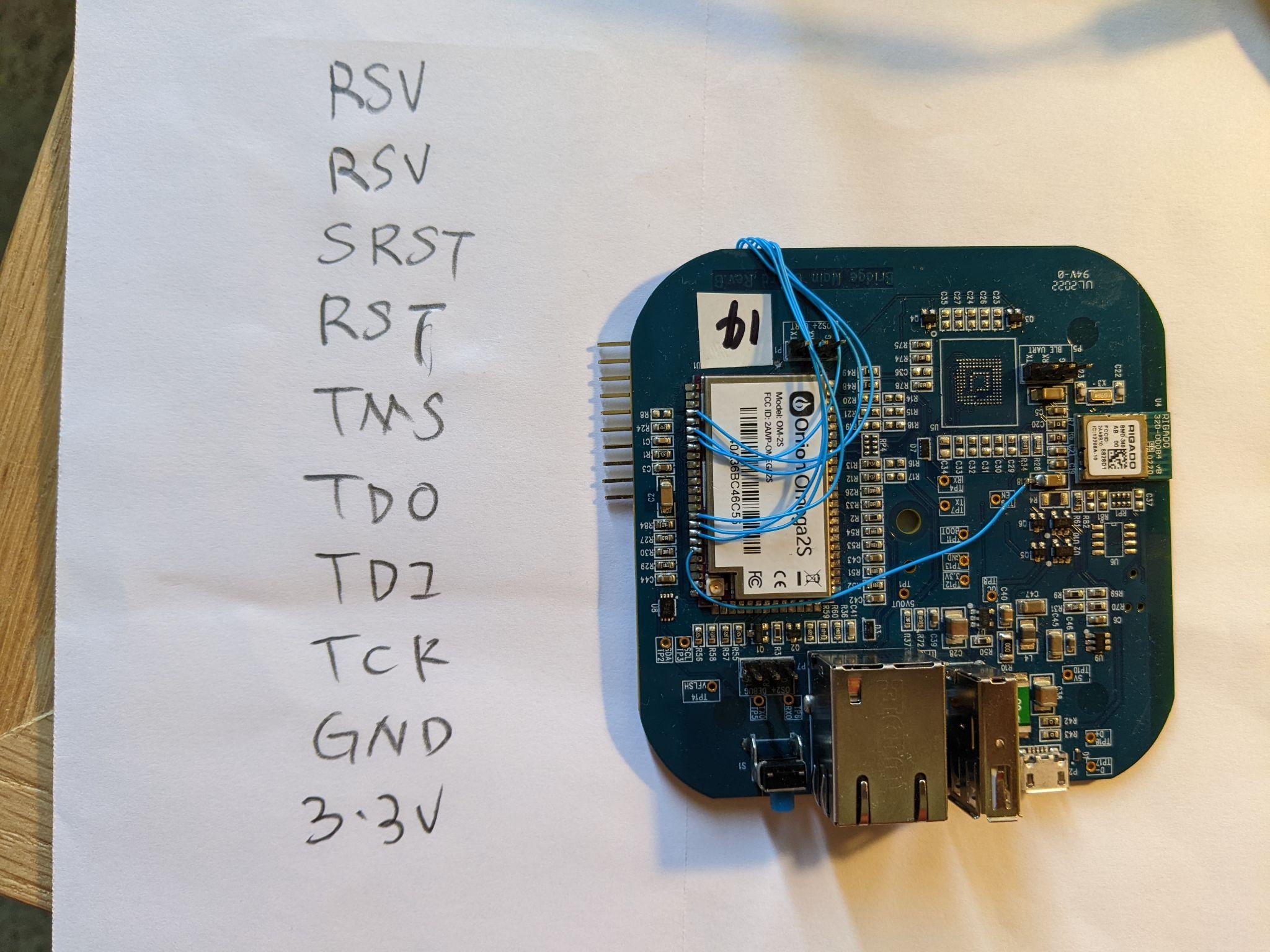
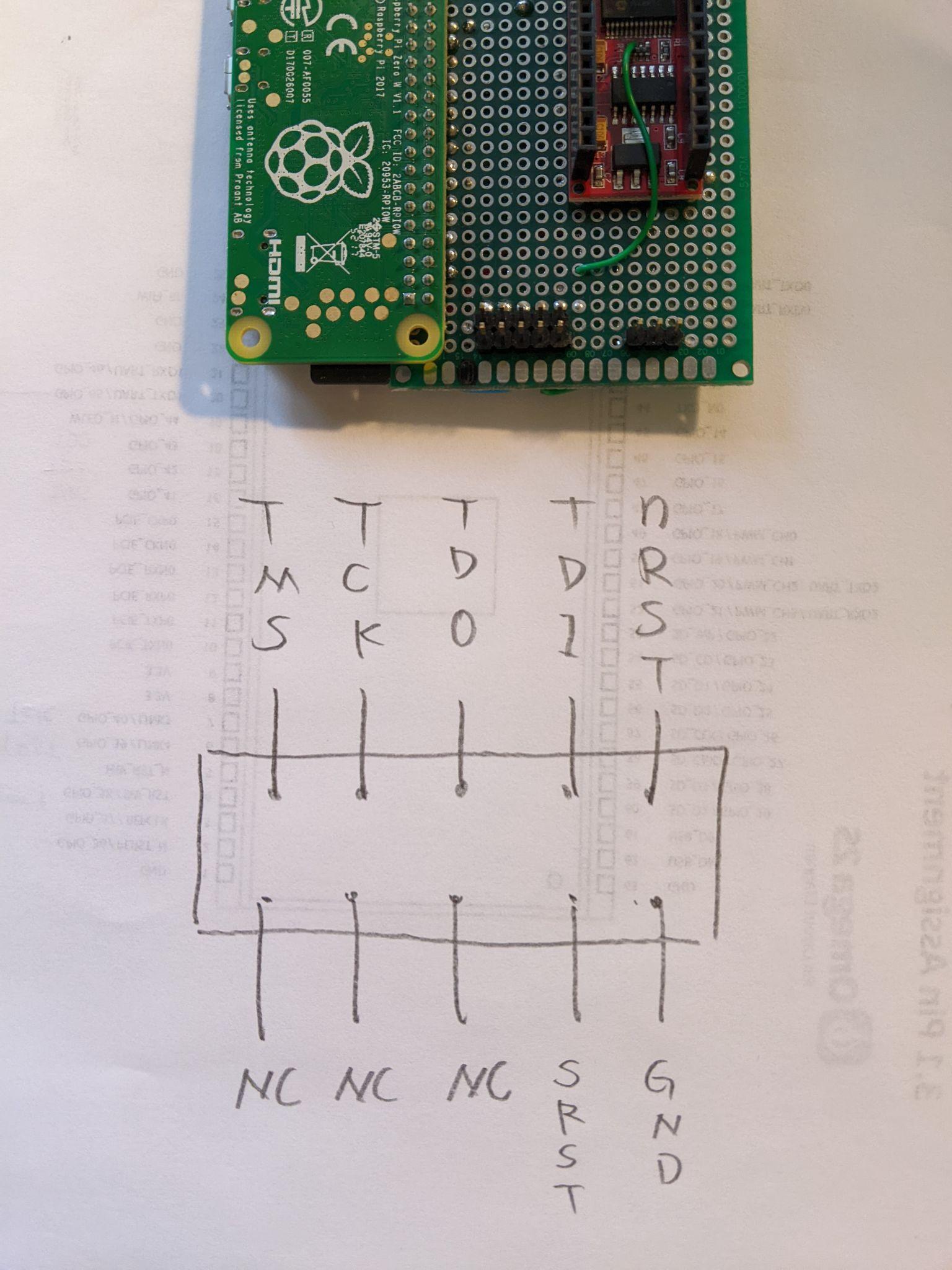
flashrom -p buspirate\_spi:dev=/dev/ttyUSB0,spispeed=8M -r /home/shared/MT7688-OpenOCD/backup.bin

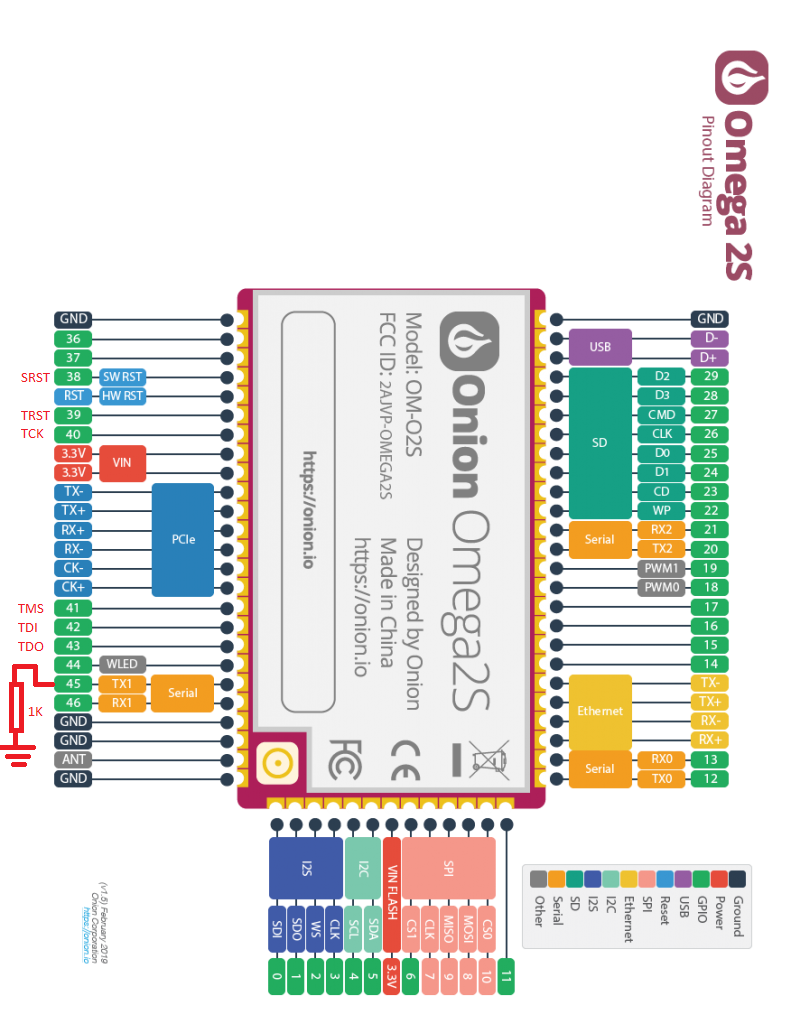
flashrom -p buspirate\_spi:dev=/dev/ttyUSB0,spispeed=8M -w /home/shared/MT7688-OpenOCD/O2S\_entire\_img.bin

flashrom -p buspirate\_spi:dev=/dev/ttyACM0,spispeed=8M -r /home/shared/MT7688-OpenOCD/backup.bin

flashrom -p buspirate\_spi:dev=/dev/ttyACM0,spispeed=8M -w /home/shared/MT7688-OpenOCD/O2S\_entire\_img.bin

1. Wiring of hardware





|  |  |  |
| --- | --- | --- |
| Signal Name | MT7688 Pin Name | O2S Pin Name |
| TDO | EPHY\_LED0, 143 |  |
| TDI | EPHY\_LED1, 142 |  |
| TMS | EPHY\_LED2, 141 |  |
| TCK | EPHY\_LED3, 140 |  |
| TRST | EPHY\_LED4, 139 |  |

1. rsv