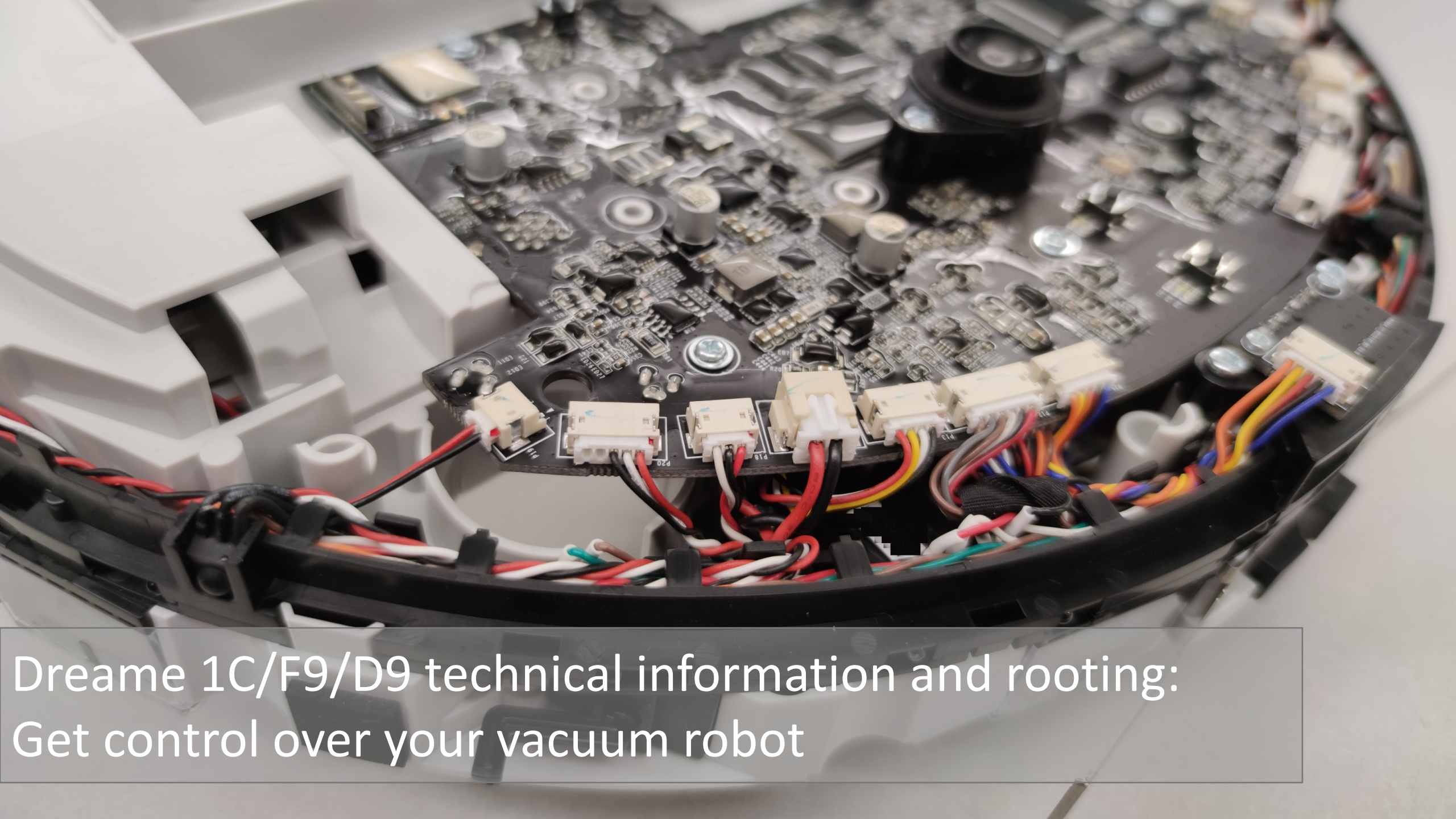


Dreame 1C/F9/D9 technical information and rooting: Get control over your vacuum robot

Before you continue: Please watch the whole video before you start your adventure

All commands and links are in the description

You might want to join the Telegram group

Good news: No soldering or teardown required

Why get root access?

- Use Valetudo (https://valetudo.cloud/)
  - Replace the cloud functionality with an open-source software
  - Integrate the device into your home automation
- Install your own soundfiles/voices

Risks

- A failed flash can leave the device in an undefined state
- Requires reflashing
- Problem:
  - All partitions need to be recreated
  - Device identity (Device ID, Cloud keys, MAC) gets lost
  - Calibration data lost
- Observations:
  - Device identity does not matter if you use Valetudo
  - Unknown consequences for lost calibration data

Risks

- Experiences so far:
  - >10 successfully rooted robots using „optimized“ method
  - 1 erased device (Dennis‘s 1T) without backups
    - Lost calibration and device identity
    - Assumption: wrong software and connection issues
  - 1 erased device (Sören‘s D9) with a backup
    - Some data lost, but we had backups due to a prior root
    - Assumption: wrong software and connection issues
  - 1 fried device (Dennis‘s 1T) with magic blue smoke
    - Mainboard damaged due mixing of VCC (5V) and GND
    - First iteration of custom board
    - Takeaway lession: double check your cables!!!

More risks

- Problem: Hardware differs even for the same „model“
- The root method cannot verify your exact model
- Flashing an incorrect firmware will perma-damage your device
  - Other, not obvious side effects might occur
  - Recovery might be tricky and can cause problems
- Important: Make sure that you have the correct firmware
  - Do not try to use the same generated firmware on multiple devices
  - If you are unsure, ask us first!

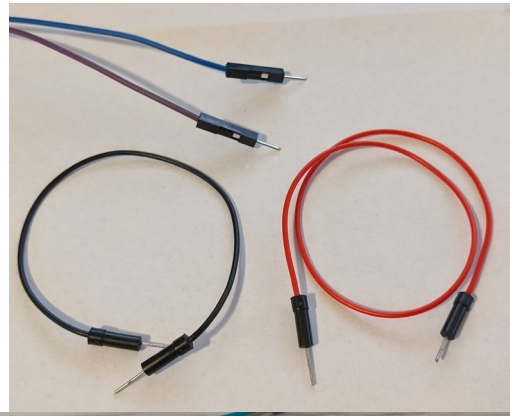
Sidenotes

- Rooting will permanently change files on your device*
  - Cloud connection / App usage should be still possible
- By installing a custom firmware, you cannot update your device with official firmwares anymore
  - That is the price of rooting
  - You can update your device with custom firmwares
- Use USB2 connections whenever possible
- Connect to your robot and write down the MAC address

* we needed to reconstruct parts from the firmware by extracting contents from flash and guessing unknown parts.

Tools required for root

- Breadboard Jumper Wires
- 2mm pitch headers
- USB cable
  - (e.g. from a broken USB mouse)
- Alternative: custom PCBs



Opening the device

Opening the device

Opening the device

Opening the device

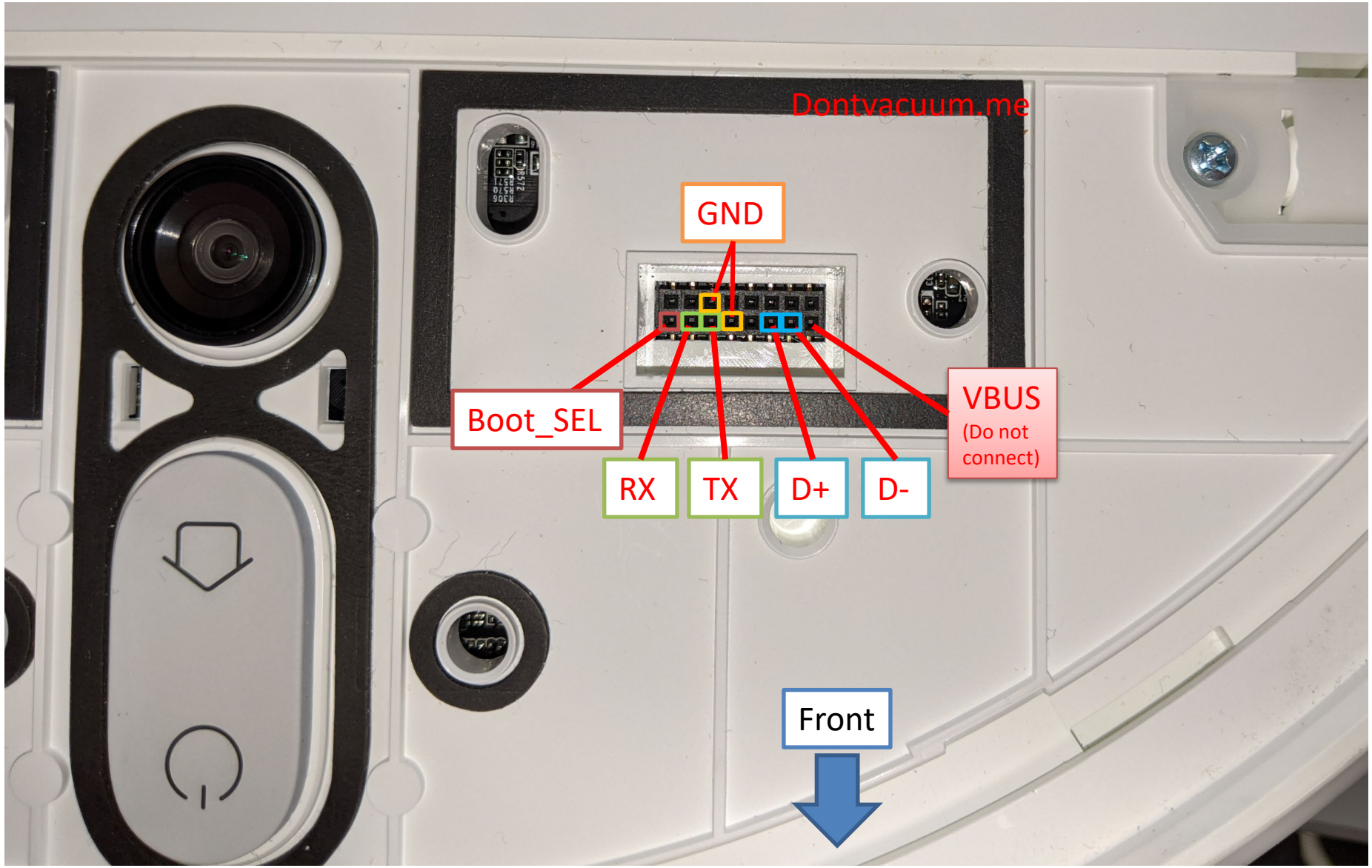
Opening the device

Debug pinout

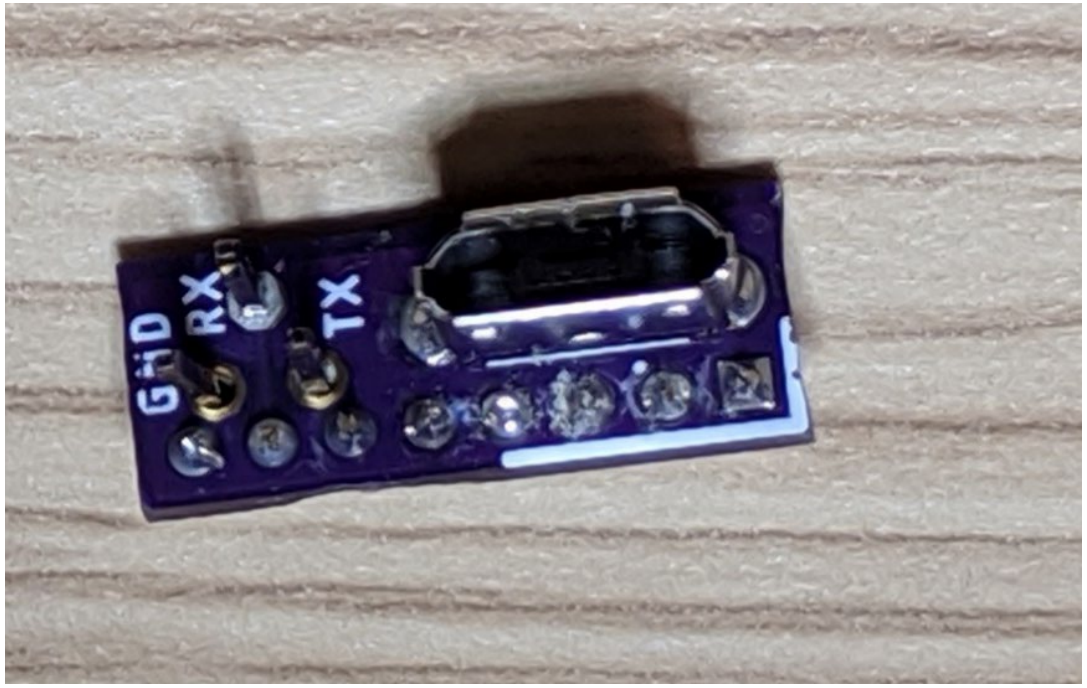
- Debuginterface
  - 2x8 pins
  - 2mm pitch size

Warning:
2mm pitch size is way smaller than the usual 2.54 mm

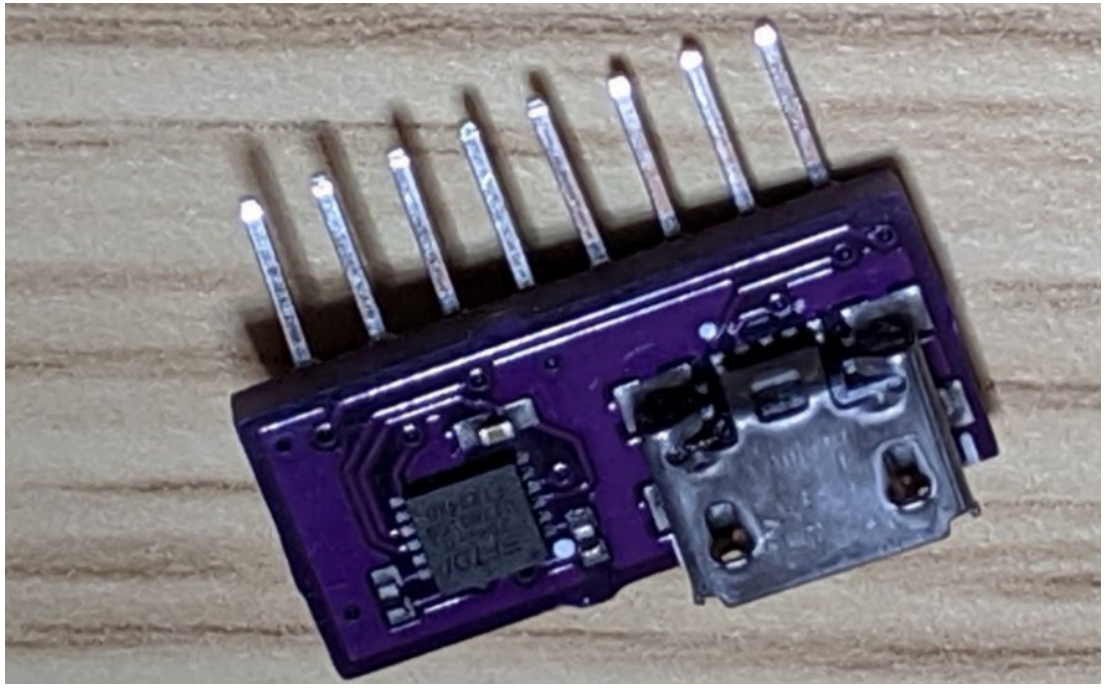
Warning:
Make sure you connect to the correct pins!



Rooting with custom PCBs



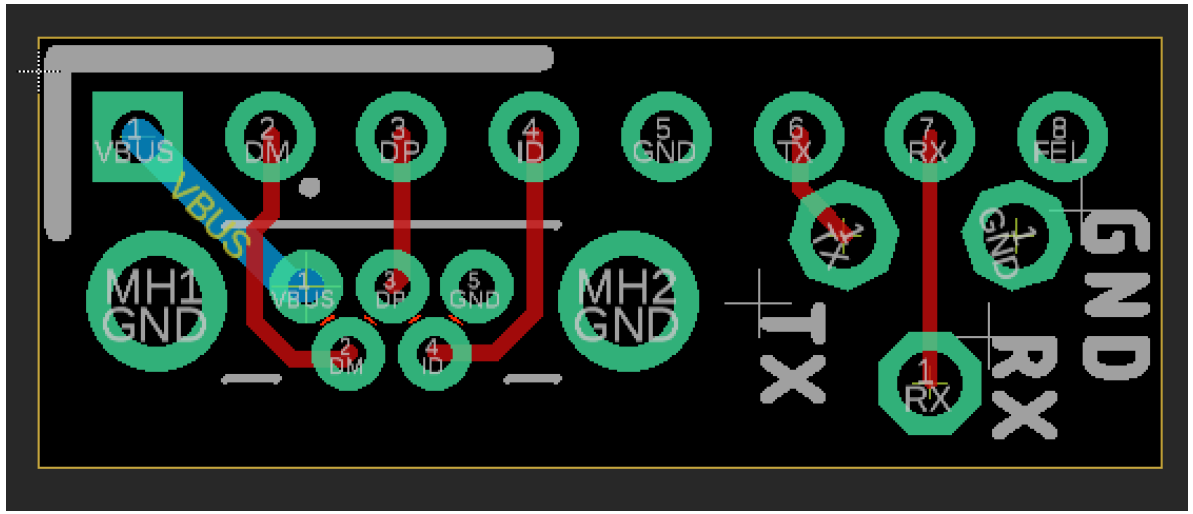
USB + UART headers
(aka basic PCB)



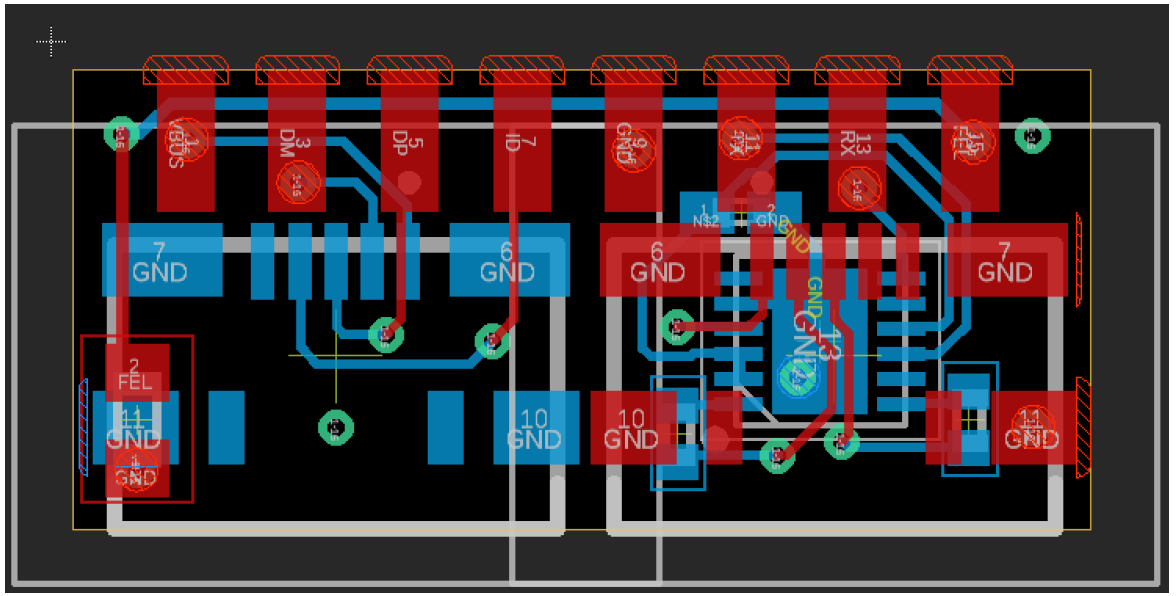
USB + integrated UART Adapter
(aka Advanced PCB)

Check builder.dontvacuum.me/dreameadapter for the Gerber files

Rooting with custom PCBs



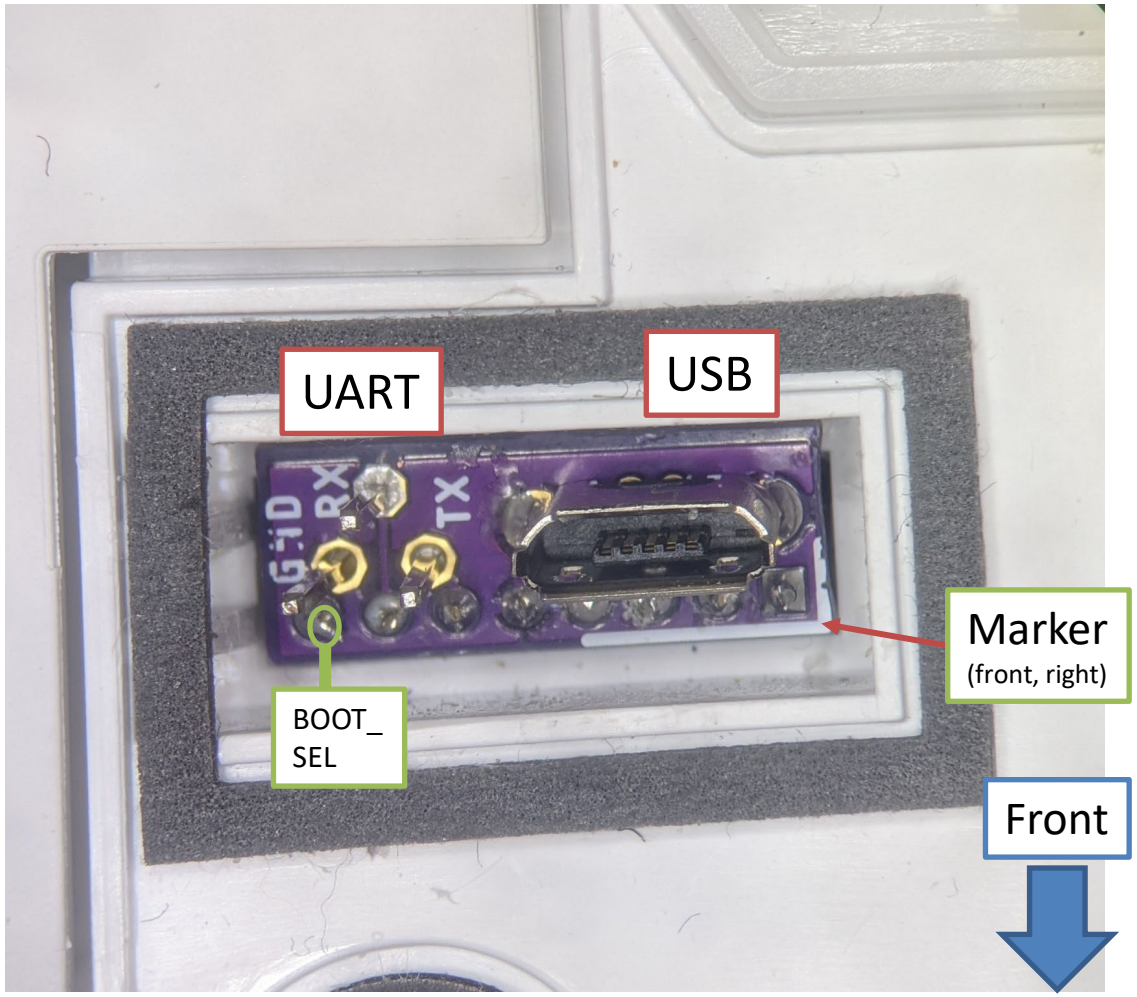
USB + UART headers
(aka basic PCB)



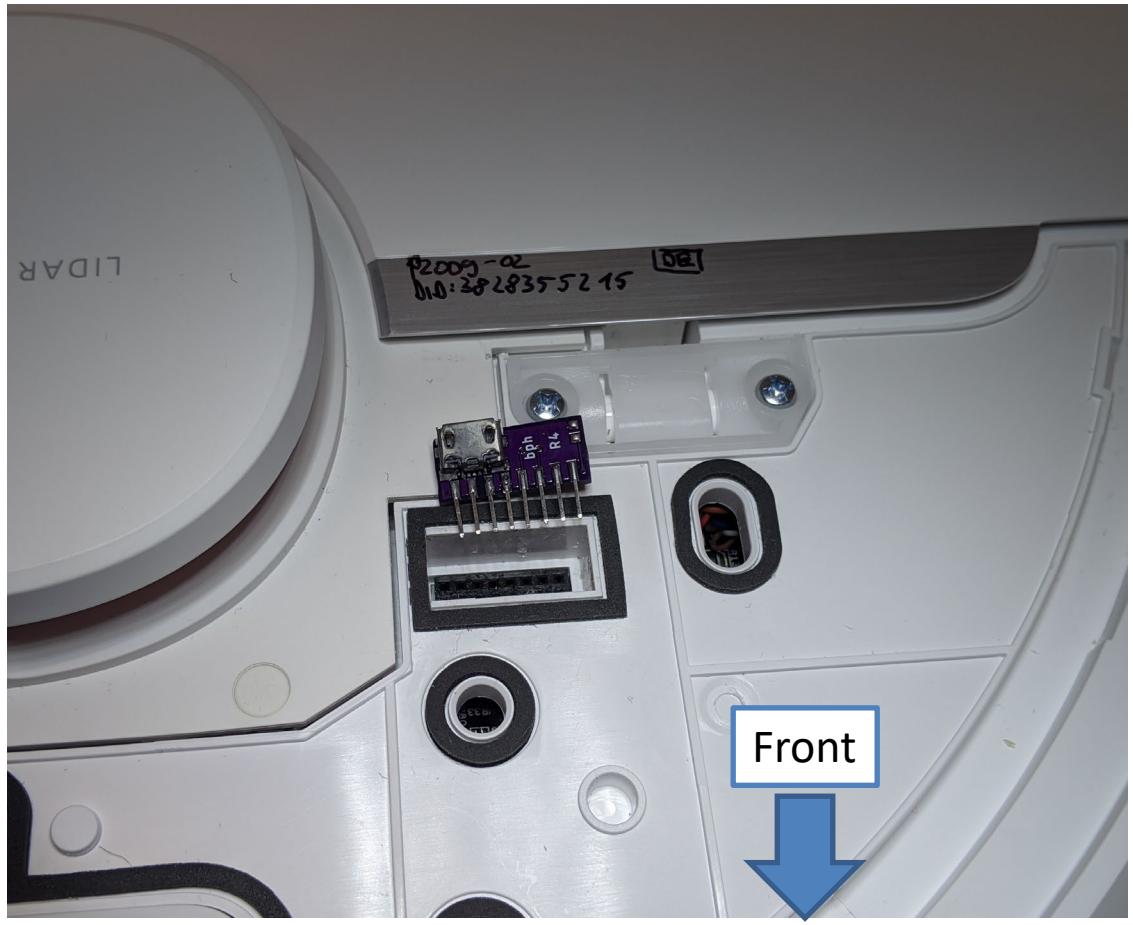
USB + integrated UART
Adapter
(aka Advanced PCB)

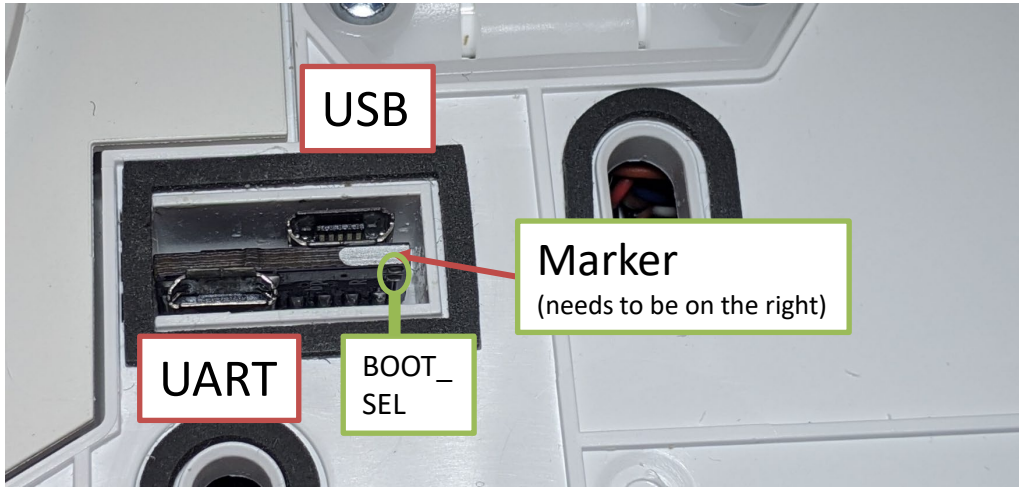
Check builder.dontvacuum.me/dreameadapter for the Gerber files

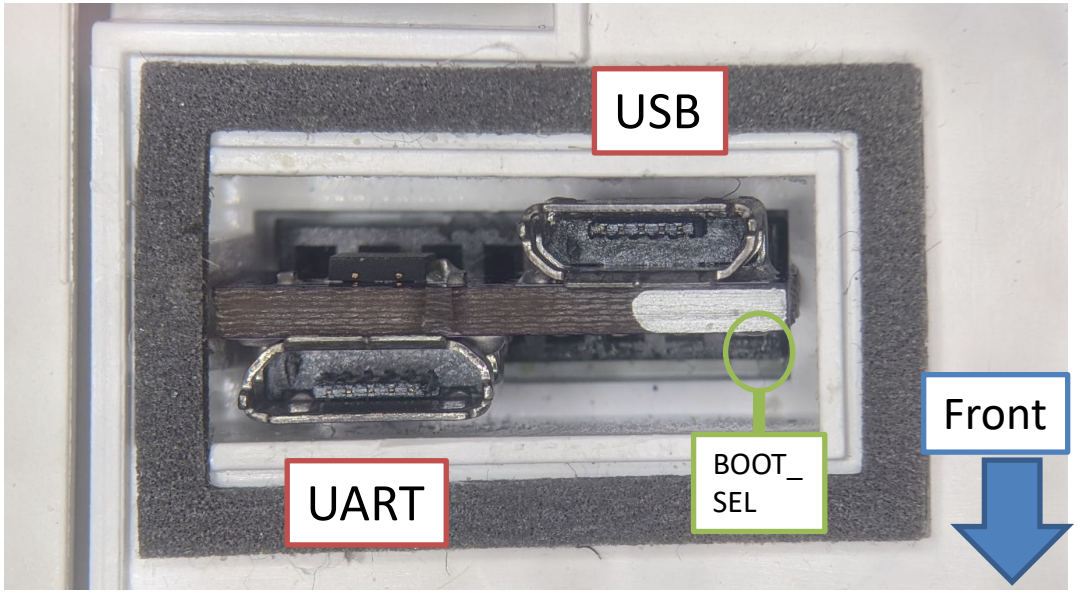
Usage of basic PCB



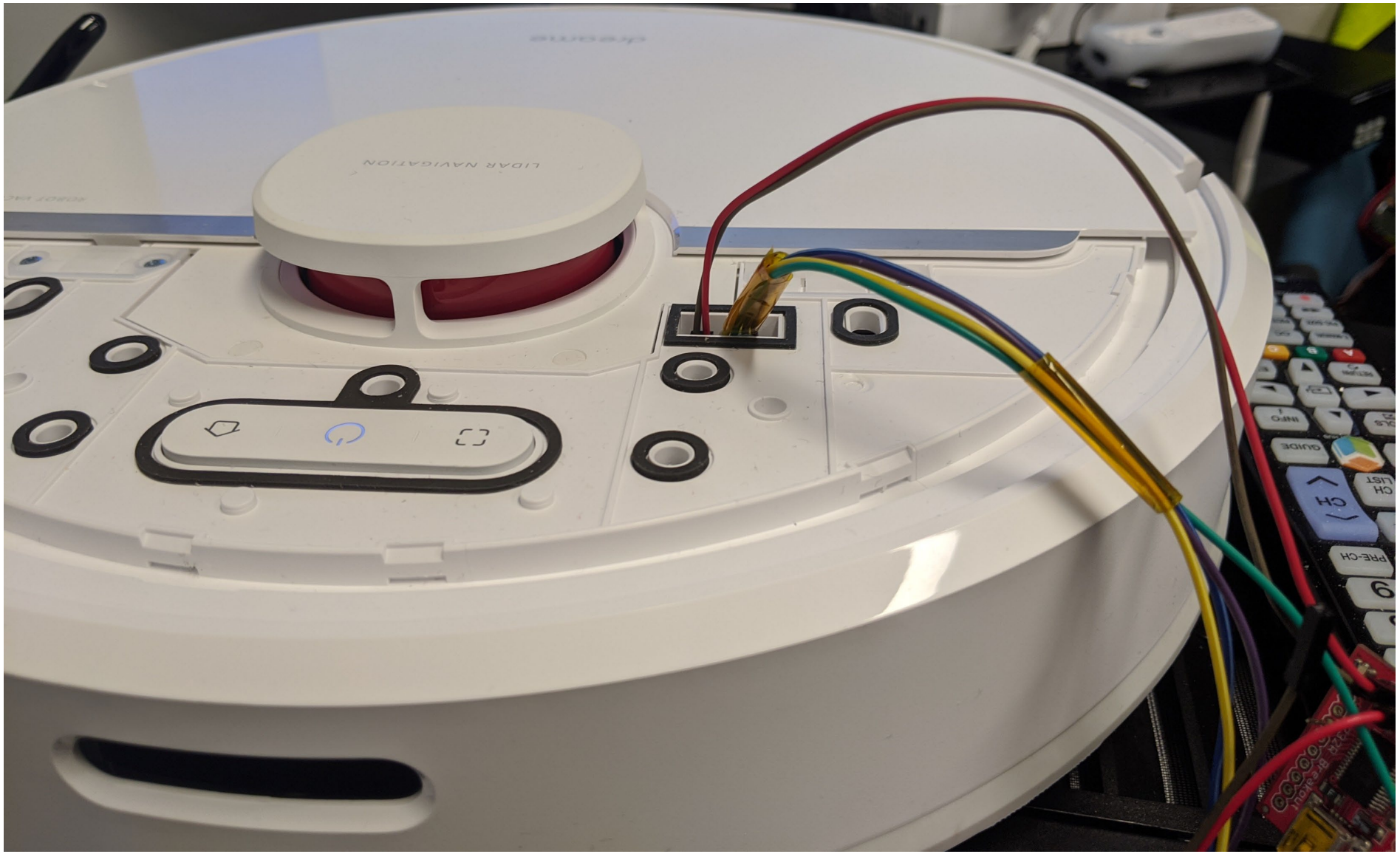
Usage of advanced PCB







Connecting jumper wires (2mm pitch)



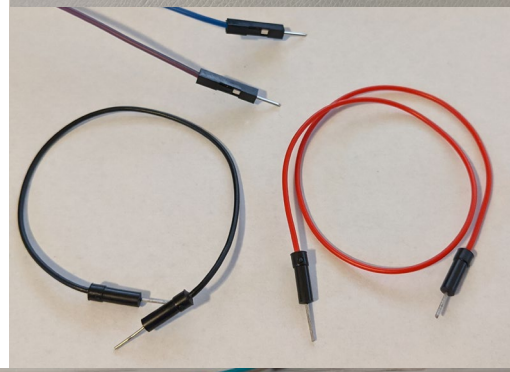
Backup of configuration and calibration

- Background: If flashing the custom firmware fails, the robot might delete the configuration and calibration files
- Idea: Interrupt U-Boot, boot in single user mode, backup files
- Limitation: works only on 1C/F9/D9

1. Power off the robot
2. Connect to UART (115200 baud, no flow control)
3. Power on the robot and keep key “s” pressed
4. Modify the command line and boot
5. Print files over UART

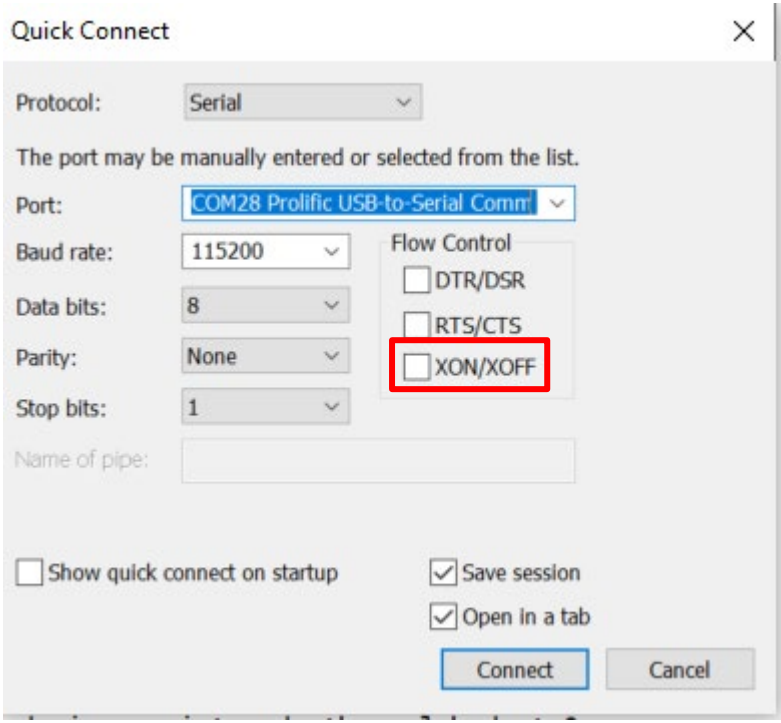
Backup Step 1

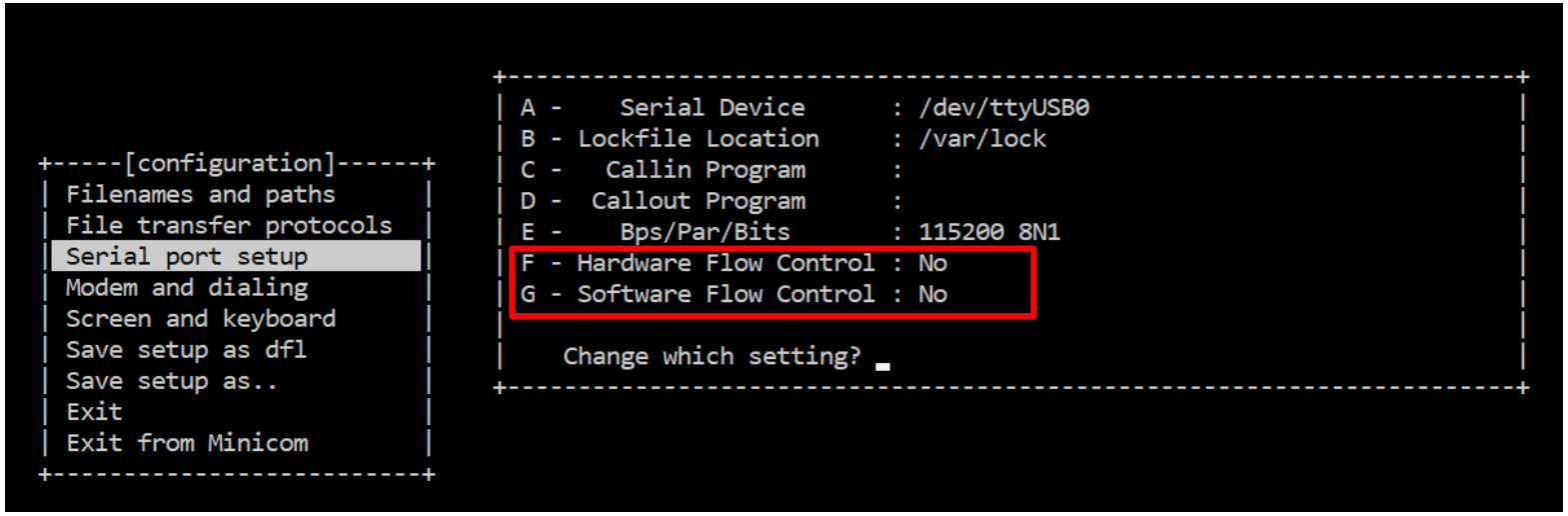
- Get an adapter
- Reminder: Works only on 1C/F9/D9
- UART-USB adapter (3.3V, aka TTL adapter)
  - Typical chipsets:
    - FT232RL, FT232, PL2303TA or CP2102
  - Price ~10 USD/Euro

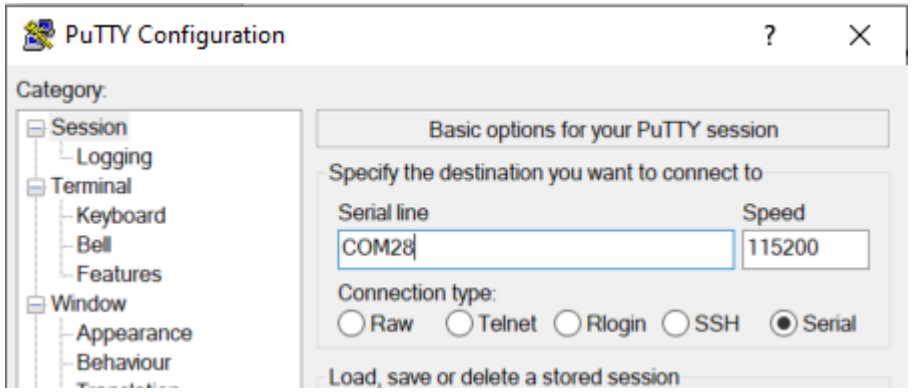


Backup Step 2a

- Know where RX and TX on your adapter is
- Configure your UART program
  - Baud: 115200
  - Flow control: off (!)
- Test the settings without robot





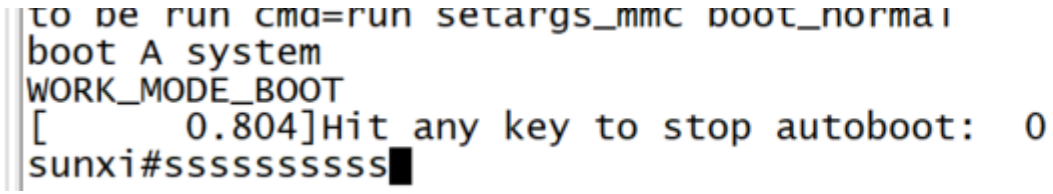


Backup Step 2b

- Connect serial wires to PCB
  - Do not connect 5V (red cable)!
  - Test for correct connection
    - Press middle button (<1s)
    - You should see some output

Backup Step 3

- Inside the terminal program
  - Hold “s” key on your keyboard
  - At the same time: Press middle button for 3 seconds
  - We want to see this:



Backup Step 4a

- In the U-Boot shell run this commands:

  setenv init /bin/sh

  setenv boot_partition boot1

  run setargs_nand

  run boot_normal
- Your robot should boot and present you a shell

Check https://builder.dontvacuum.me/dreame/cmds.txt for a copy-pastable command list

Backup Step 4b

- After the system booted, run these commands:

  /etc/init.d/sysconfig.sh

  echo V > /dev/watchdog

  /etc/init.d/mount_private.sh

  /etc/init.d/mount_misc.sh

Check https://builder.dontvacuum.me/dreame/cmds.txt for a copy-pastable command list

Backup Step 4c

- Run these commands to print the configuration (save output):

  *grep "" /mnt/private/ULI/factory/**

- Run these commands to save the calibration (save output):

  grep "" /mnt/misc/.json*

  grep "" /mnt/misc/.yaml*

  cat /mnt/misc/.txt*

  hexdump /mnt/misc/.bin*

Some files might not exist on your device. That is normal.

Make sure that you copy the full output to a text file and save it

Check https://builder.dontvacuum.me/dreame/cmds.txt for a copy-pastable command list

Rooting preparations

- Fully charge your vacuum robot
- Required software: Phoenixsuit
  - Download it here:
    - https://androidmtk.com/download-phoenixsuit
- Generate a custom firmware
  - Go here:
    - https://builder.dontvacuum.me/
    - Select your model
    - Fill out the form
    - Select “Patch DNS” if you plan to use Valetudo (this disables the cloud)
    - Use voucher “dreameroot”
    - Select Livesuit image

Alternative Method for 1C/F9/D9

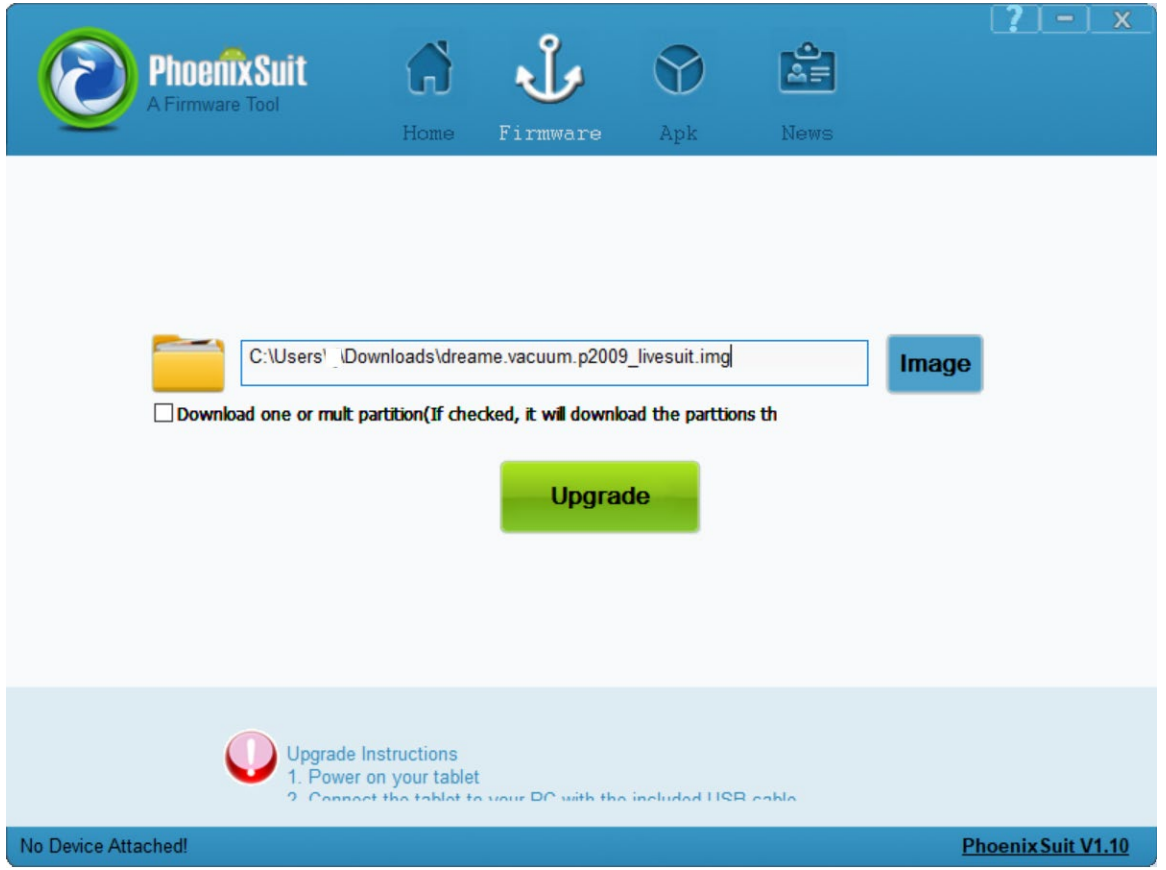
- You can root your device via UART
  - Method might be safer
  - Risk of soft-bricking reduced
  - Check this link: https://gist.github.com/stek29/5c44244ae190f3757a785f432536c22a
  - You need to build a normal firmware update (non-livesuit)
  - Hint: Your robot must be docked and charged before you run the installer!

For 1C/F9/D9 only:
DO NOT PROCEED if you do not have a backup of the configuration and calibration!!
(check also the alternative method)

Please report any issues directly (e.g. misflash, non-booting, etc)

Root Step 1

- Make sure that the robot is not connected over USB
- Open Phoenixsuit and select image

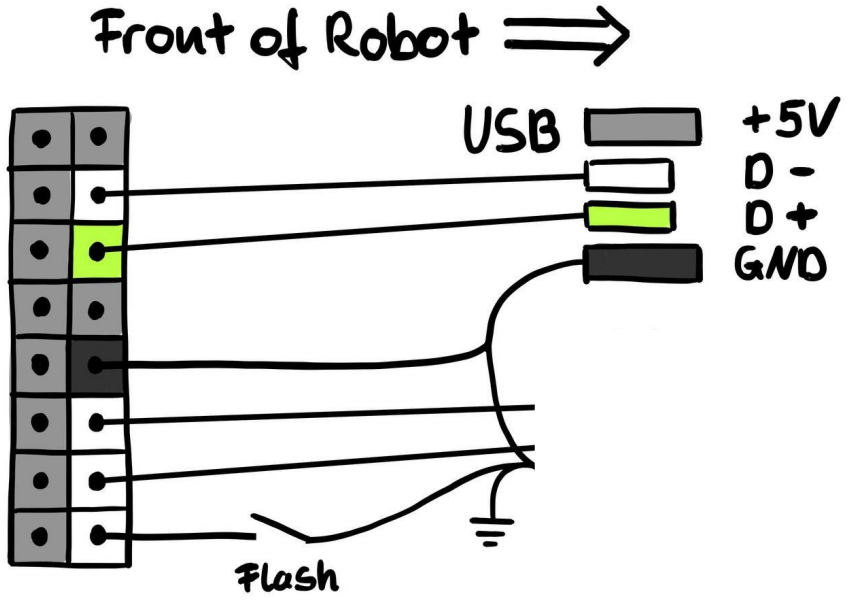


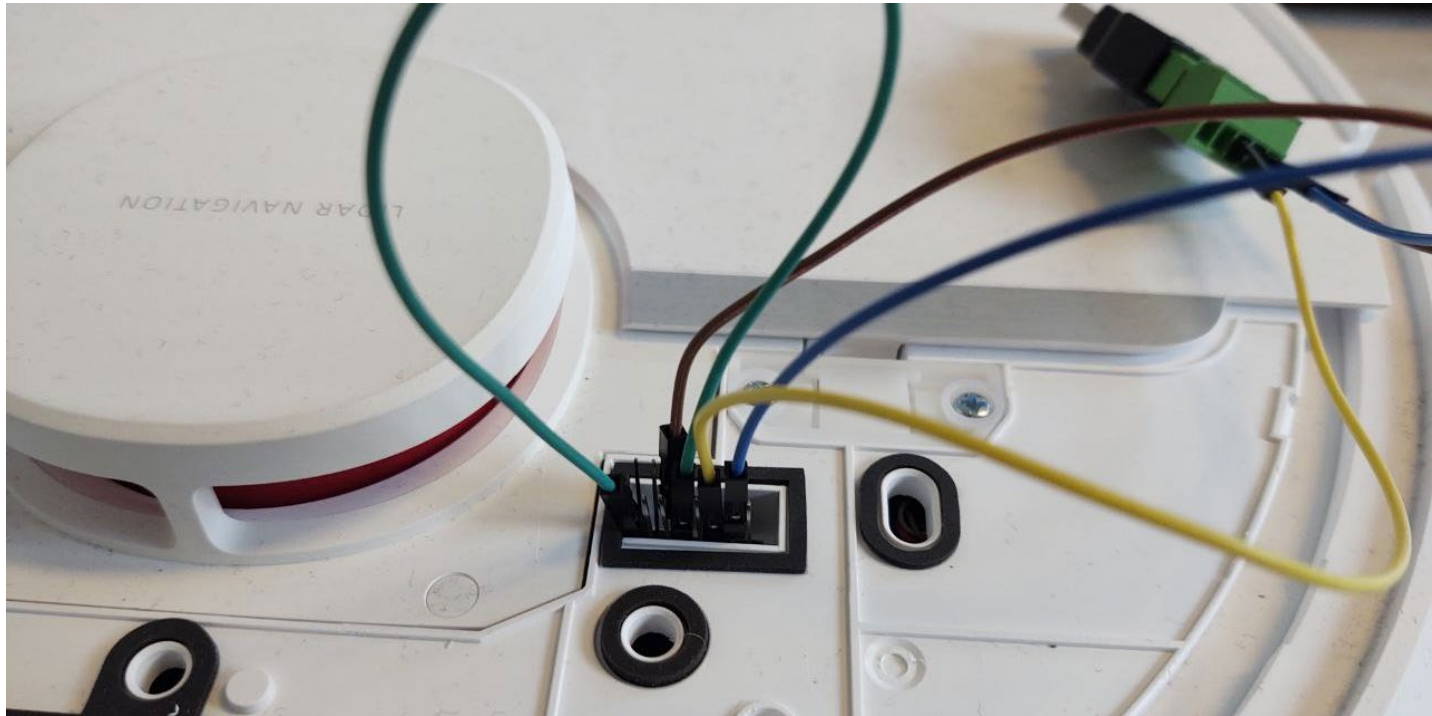
Root Step 2

- Boot the Robot in FEL mode
  - Connect MicroUSB to your computer (no USB hub, preferred USB 2.0 port)
  - Connect BOOT_SEL to GND
    - Use jumper wire (see 2a)
    - Basic adapter (see 2b)
    - Advanced adapter (see 2c)
  - Press the power button for 3 seconds
    - USB device should show up on your computer
    - You might need to install the Phoenixsuit drivers located in “Drivers/AW_Driver”
      - (via Device Manager -> Unknown Device -> Update drivers)
      - If that does not work, download it here: https://builder.dontvacuum.me/usbdriver.zip

Root Step 2a

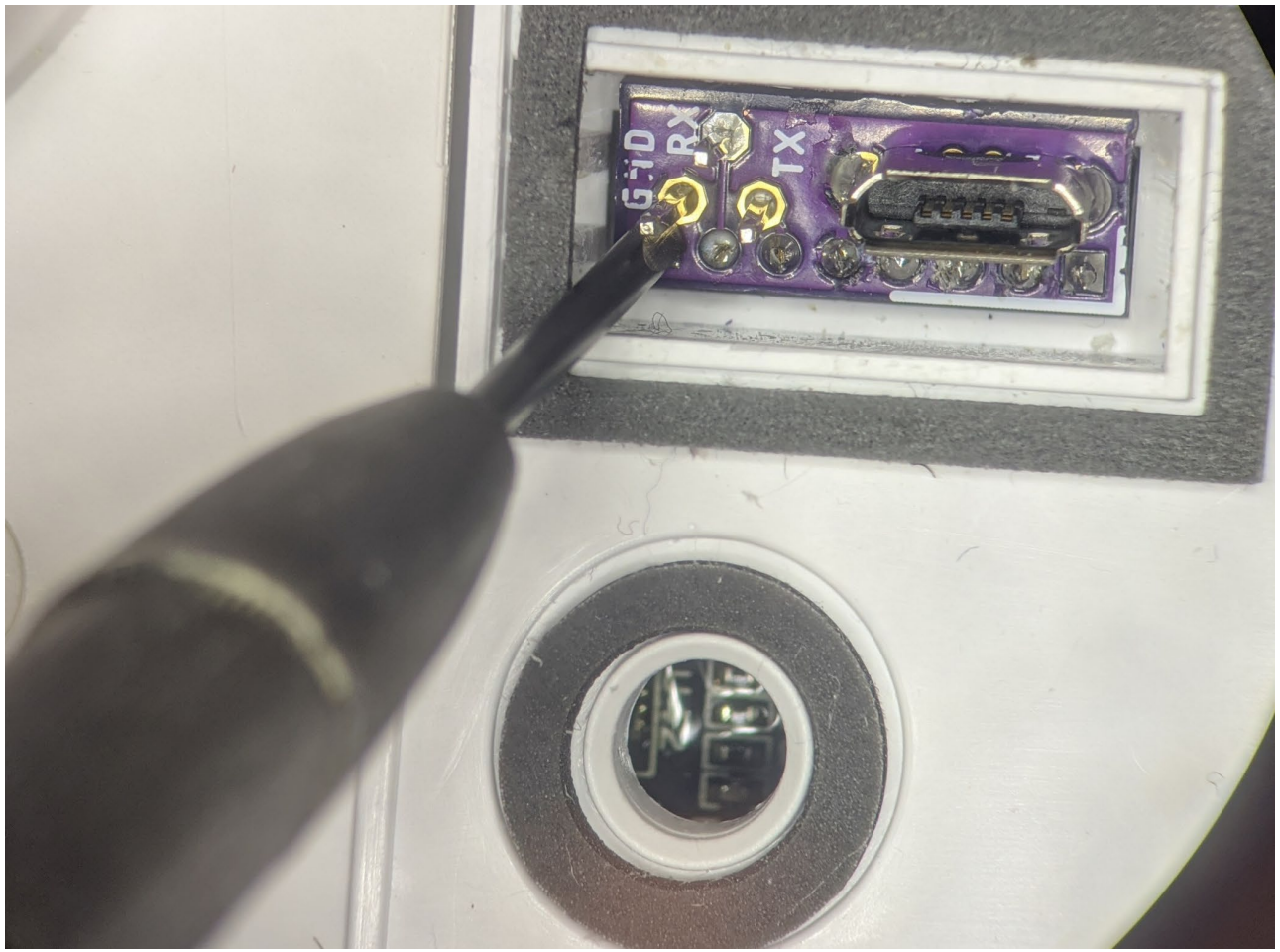
- Jumper wires: Ideally you have 2mm headers
  - Make sure that the USB connection stays stable before you trigger the update (check the Windows Device Manager)
  - 2.54 pitch jumper cables might fit but can cause issues





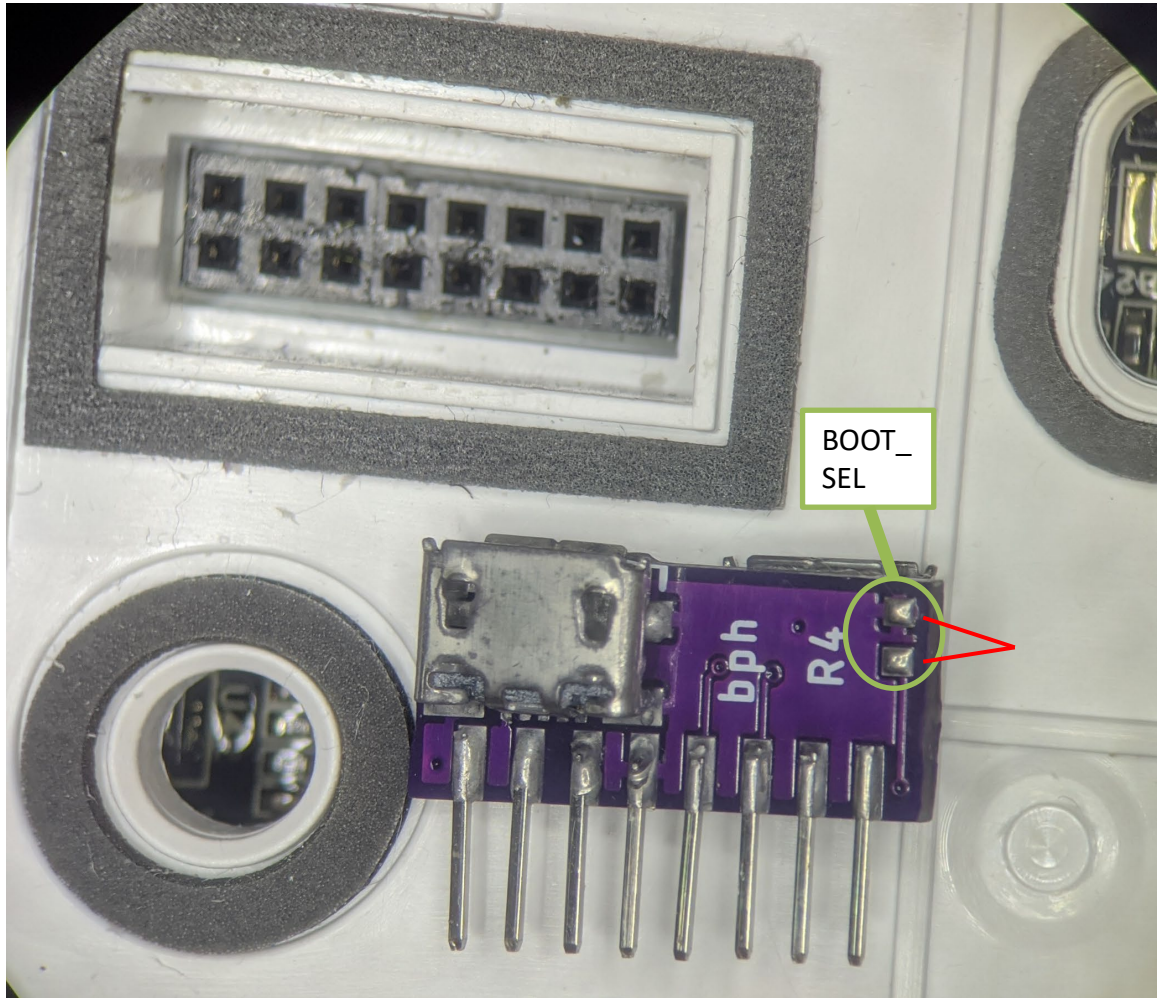
Root Step 2b

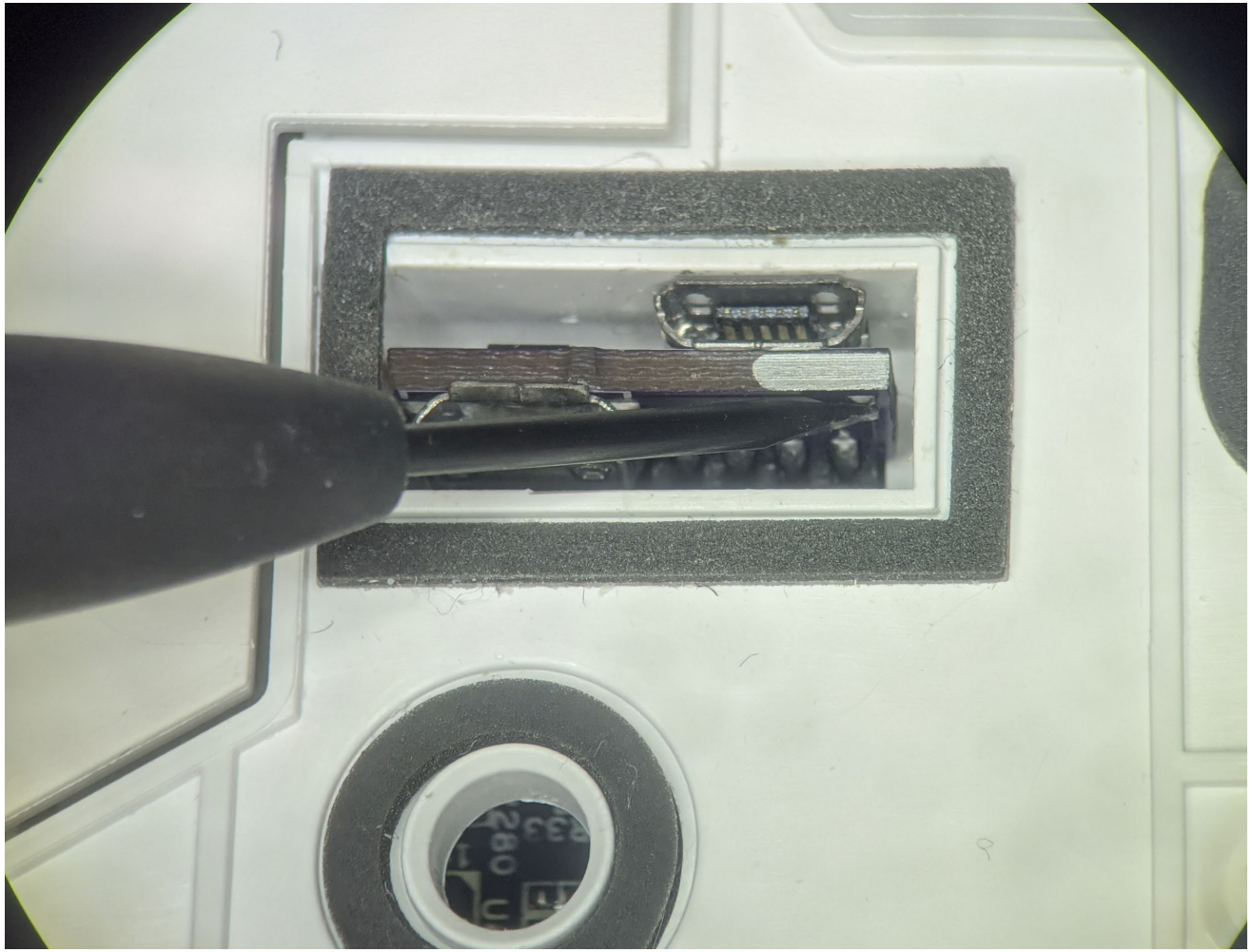
- Basic PCB: Short the left pin to GND (e.g. with screwdriver)



Root Step 2c

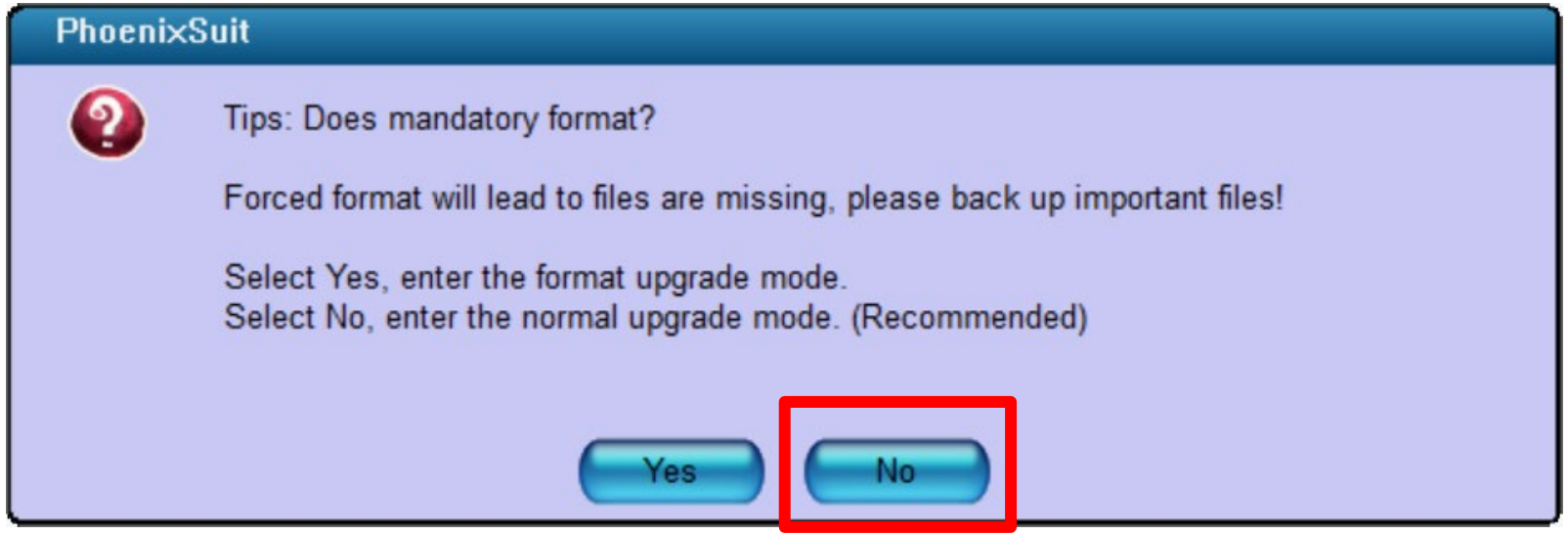
- Advanced PCB: Short BOOT_SEL pads





Root Step 3

- Disconnect BOOT_SEL before proceeding
- In Phoenixsuit
  - Click on Update
  - If asked about mandatory format, select “No” !
  - Wait for Update to finish



Root Step 3 Trouble shooting

- Do not use USB hubs
- If the device is detected initially, but disappears:
  - Try to use a different USB port
  - If you use VMs: Make sure that the USB filter is correct

You have now installed a custom firmware ;)

Customization of firmware

- The firmware builder adds to hooks to the start process
  - /data/_root_sysconfig.sh (gets executed early)
  - /data/_root_postboot.sh (gets executed after boot)
- At each boot we check for the presence of the custom files
  - Safety measurement: In case something goes wrong, a factory reset will delete this files
- An example file can be found in /misc

Valetudo installation

- Download valetudo binary to /data:
  - https://github.com/Hypfer/Valetudo/releases
  - Device dependent (check our website):
    - 1C, F9, D9: valetudo-armv7-lowmem
    - 1T, L10 Pro: valetudo-aarch64
  - “wget https://github.com/Hypfer/Valetudo/releases/latest/download/valetudo-armv7-lowmem -O /data/valetudo”
- Make valetudo executable
  - “chmod +x /data/valetudo”
- Enable boot script
  - “cp /misc/_root_postboot.sh.tpl /data/_root_postboot.sh”
  - “chmod +x /data/_root_postboot.sh”
- Reboot

Check https://builder.dontvacuum.me/dreame/cmds.txt for a copy-pastable command list


Thank you for watching!
@dgi_DE
Website: dontvacuum.me