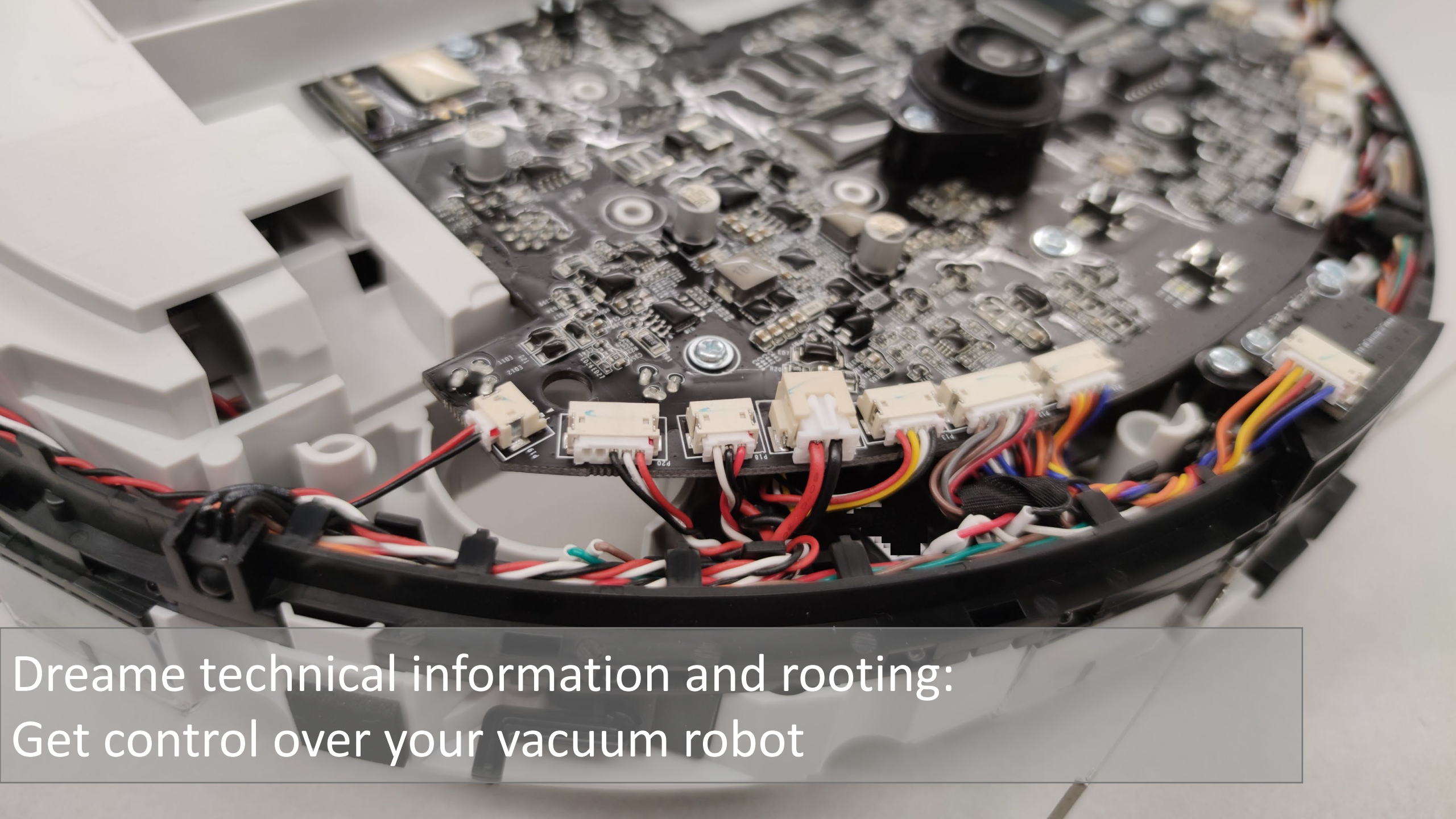


Dreame technical information and rooting:
Get control over your vacuum robot

Before you continue: Please look thru all slides and the FAQ before you start your adventure

All commands and FAQ can be found here:

https://builder.dontvacuum.me/dreame/

You might want to join the Telegram channel:

https://t.me/dust_announce

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

Currently tested devices (02.2022)

- Xiaomi Vacuum 1C: UART*/Reset**/Livesuit
- Dreame D9 (normal and Pro): UART*/Reset*/Livesuit
- Dreame F9: UART*/Reset*/Livesuit
- Xiaomi Vacuum 1T: Reset*/Livesuit***
- Dreame L10 Pro: Reset*/Livesuit***
- Dreame Z10 Pro: Reset*/Livesuit***
- Mova Z500: Reset*/Livesuit

* Method patched in new versions, check https://dontvacuum.me/robotinfo/

** Method might not be available in too old or too new versions.

*** Method likely to be patched with future firmware updates

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

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
- 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.

** if you backup the encryption keys, you can restore them at a future time. In special circumstances (dead hardware) we might provide you with stock firmware.

Important information

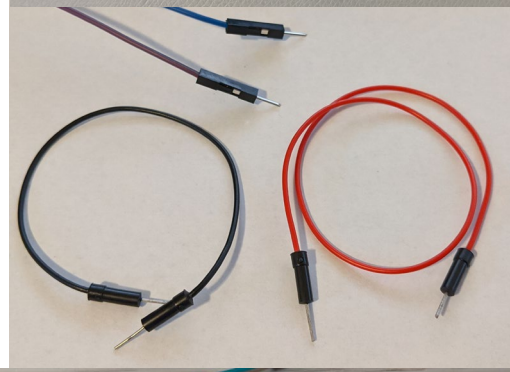
- DO NOT UPDATE THE FIRMWARE VIA THE APP if you ever plan to root your device
  - New firmwares have rooting methods patched
  - Root might be still possible, but is more complicated and risky
- Before rooting: test the device without using the app
  - Fully charge your device
  - Use the buttons to start the cleaning run
  - Let the robot clean multiple times
  - Background: Some devices come with defects from the factory. It is tricky to figure out if your robot behaves weirdly due to it being already broken or due to rooting

Current available rooting methods

- UART
  - Requires: UART connection
  - Idea: Interrupting the bootloader, boot into single-user-mode, flash firmware
- Reset (Currently recommended)
  - Requires: UART connection
  - Idea: trigger the secret login shell by pressing reset <=1 sec, compute root password, login, flash firmware
- Livesuit
  - Requires: USB connection, UART (recommended but not required)
  - Idea: creation of special factory image, triggering bootrom mode (FEL), flashing of all partitions via Livesuit software (requires Windows)

Tools required for root

- UART-USB adapter (3.3V, aka TTL adapter)
  - Typical chipsets:
    - FT232RL, FT232, PL2303TA or CP2102
  - Price ~10 USD/Euro
- Breadboard Jumper Wires
- 2mm pitch headers
- USB cable (for Livesuit/FEL)
  - (e.g. from a broken USB mouse)
- Alternative: custom PCBs



Opening the device

Opening the device

Opening the device

Opening the device

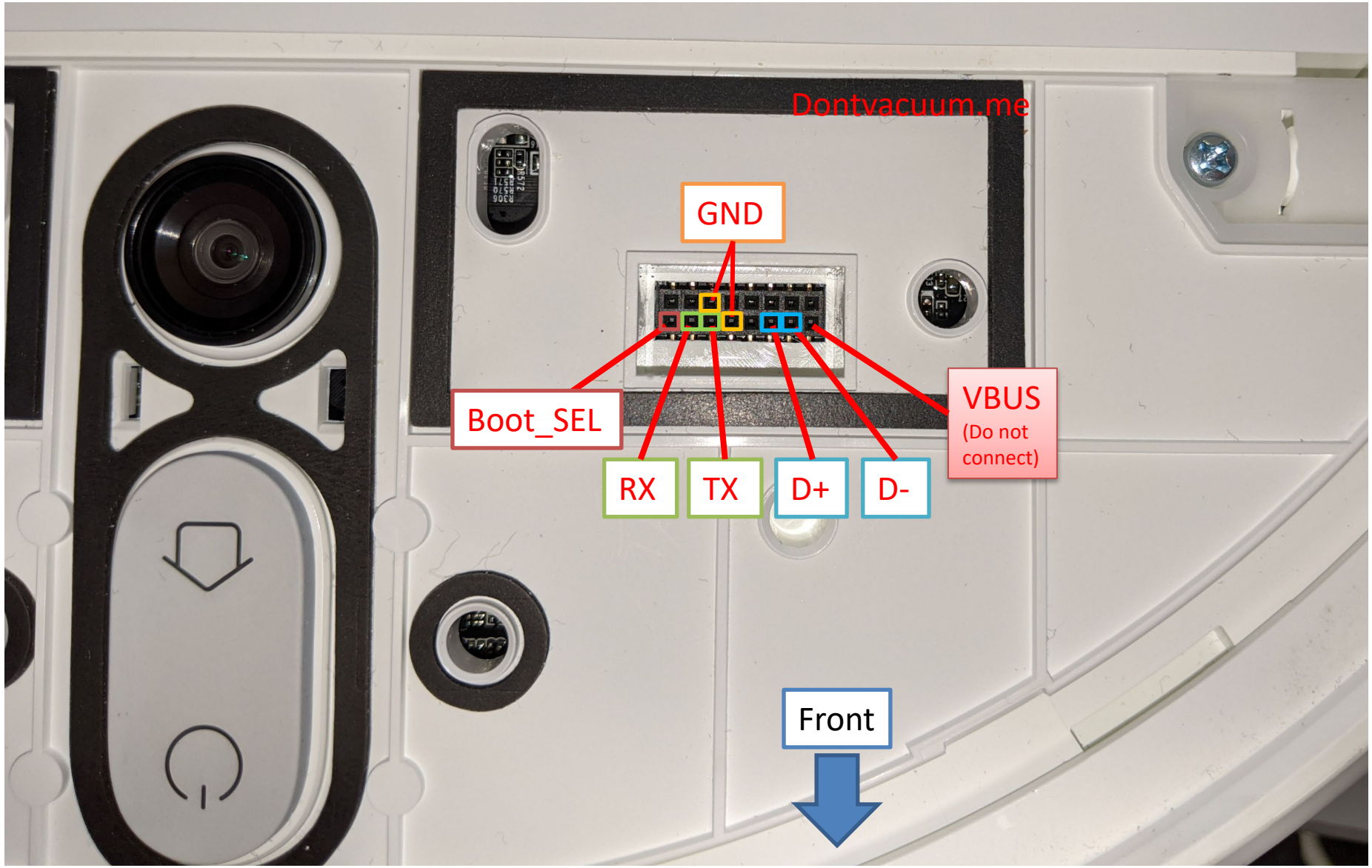
Opening the device

Debug pinout

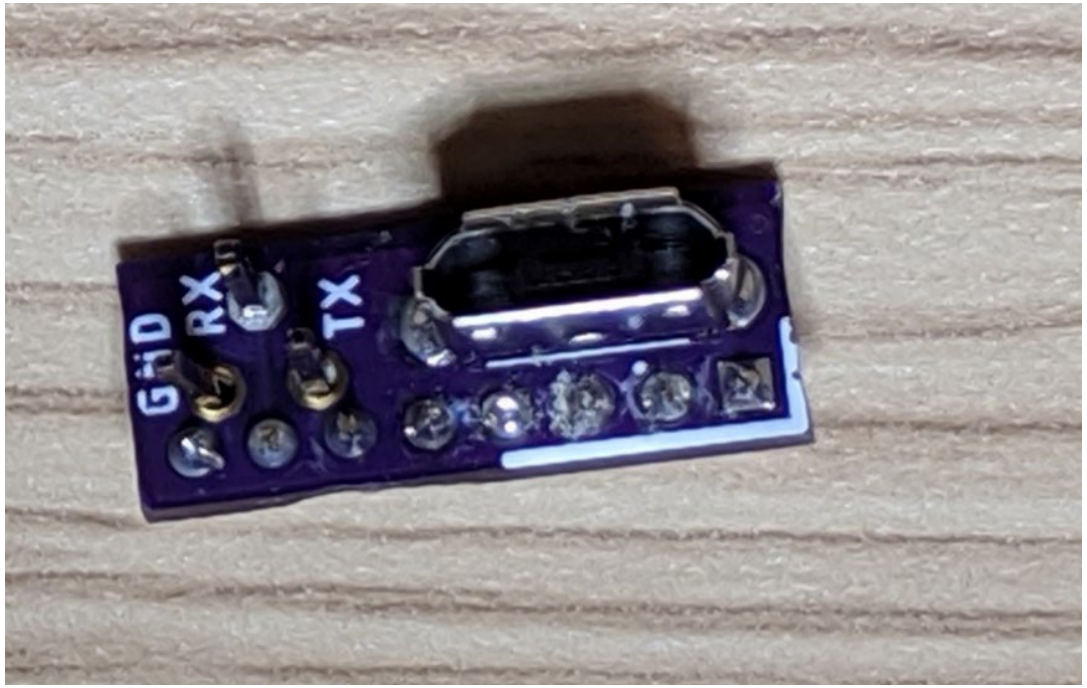
- Debuginterface
  - 2x8 pins
  - 2mm pitch size

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

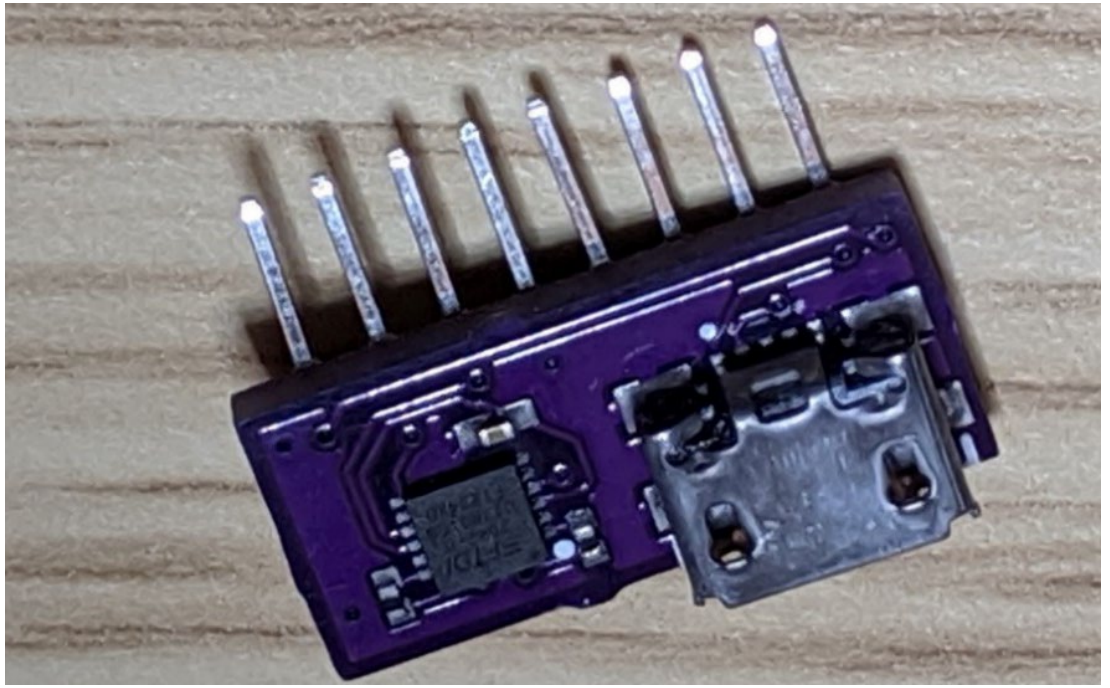
Warning:
Make sure you connect to the correct pins!



Rooting with custom PCBs



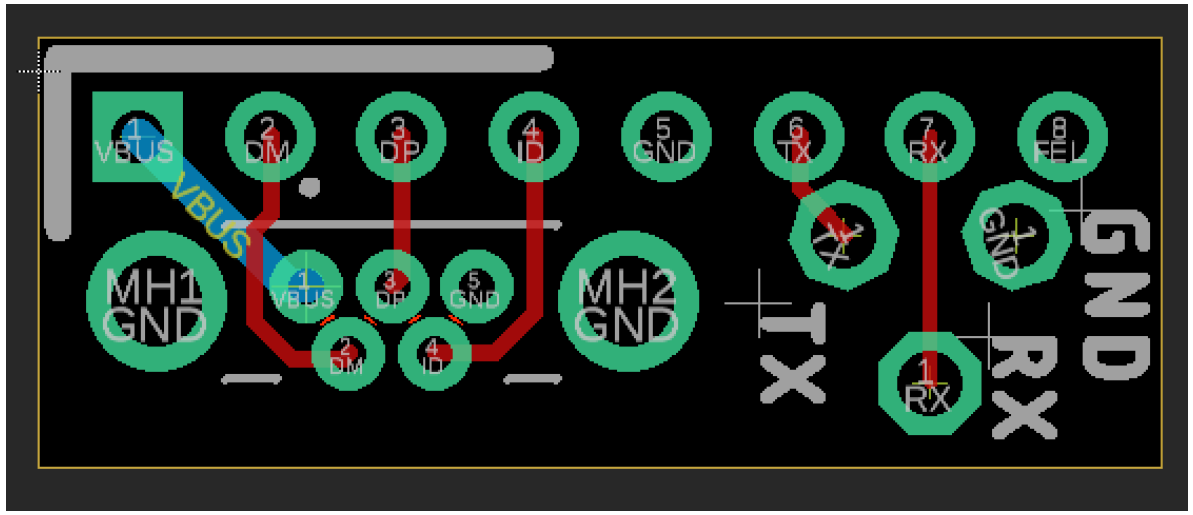
USB + UART headers
(aka basic PCB)



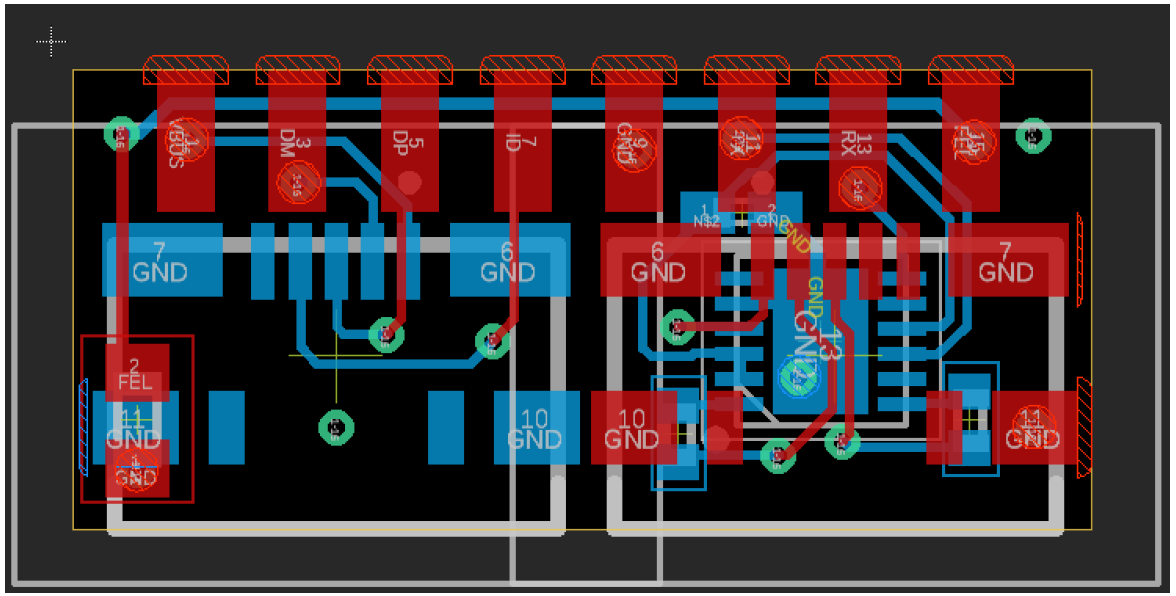
USB + integrated UART Adapter
(aka Advanced PCB)

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

Rooting with custom PCBs



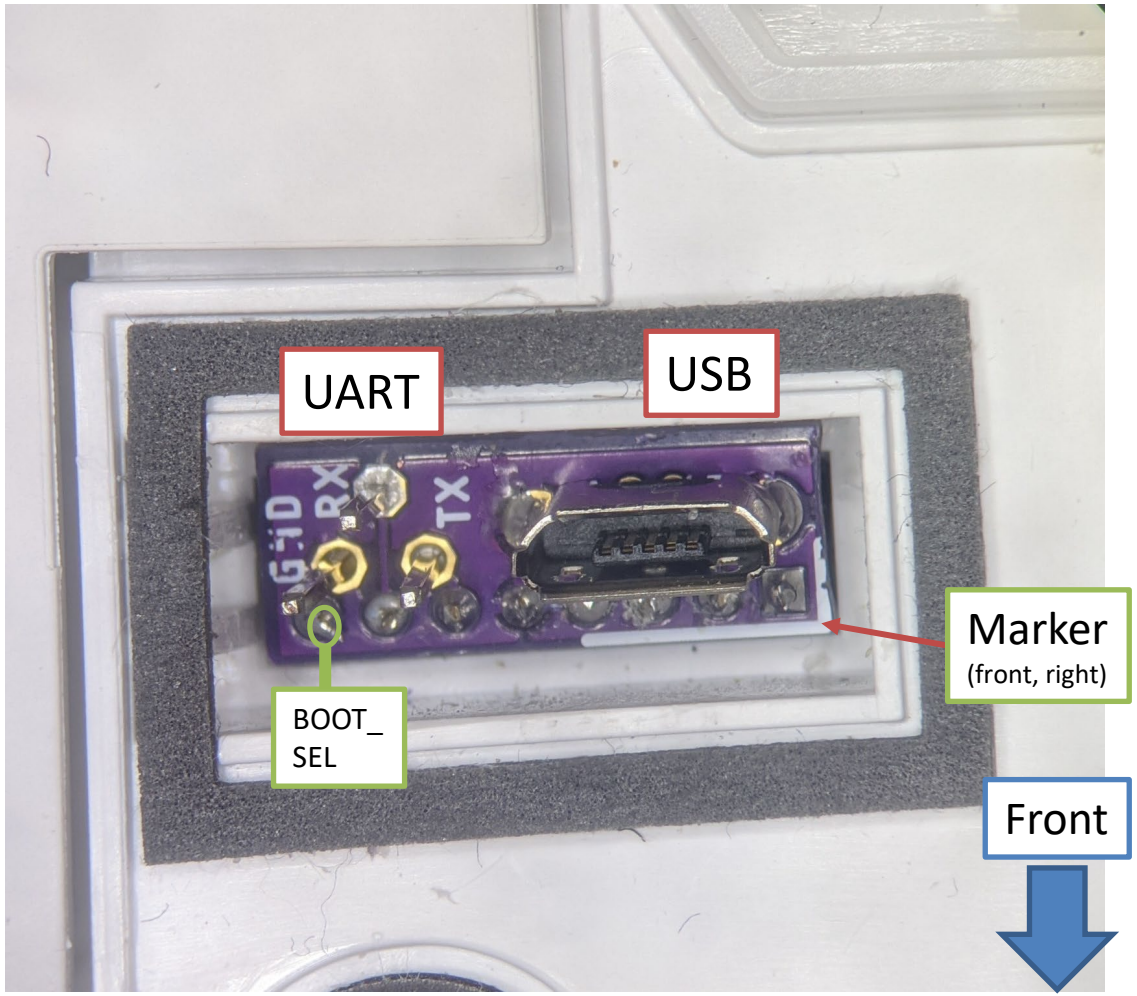
USB + UART headers
(aka basic PCB)



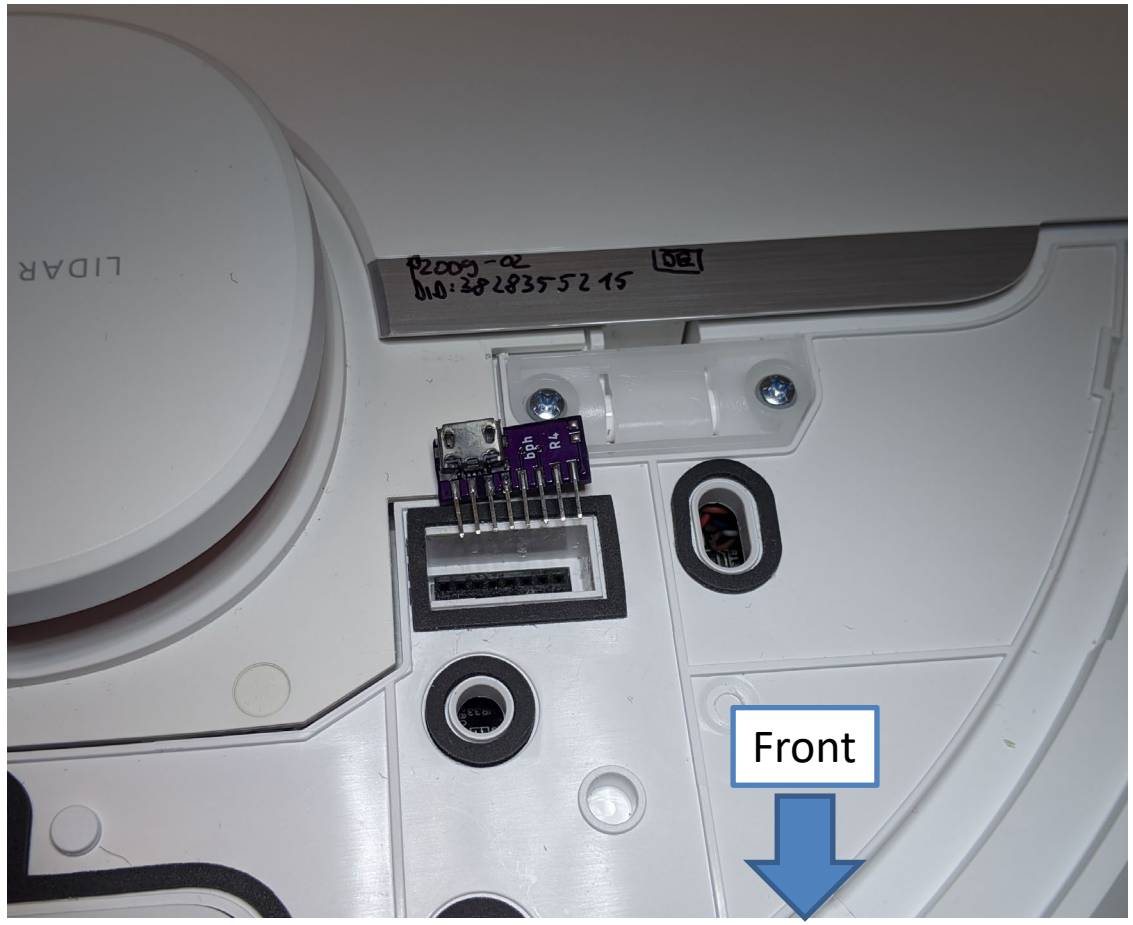
USB + integrated UART
Adapter
(aka Advanced PCB)

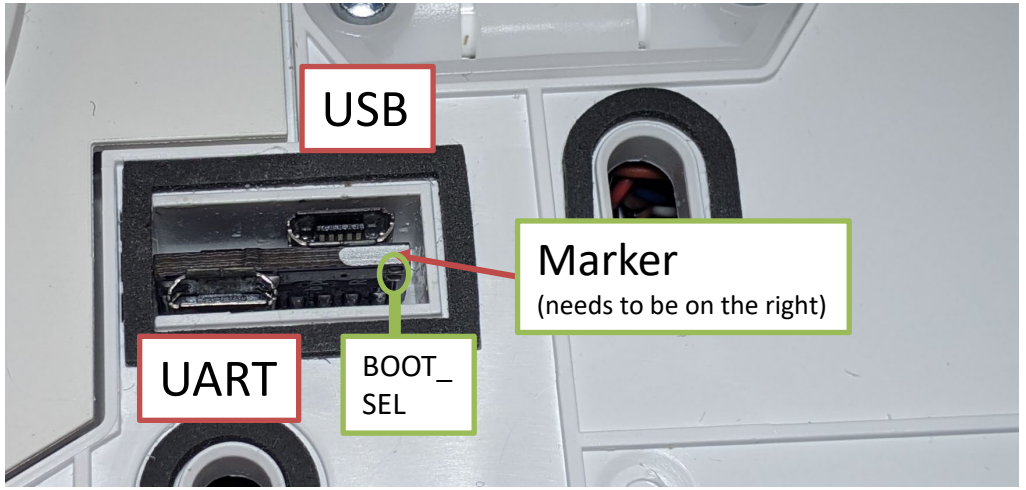
Check builder.dontvacuum.me/dreameadapter for the Gerber files

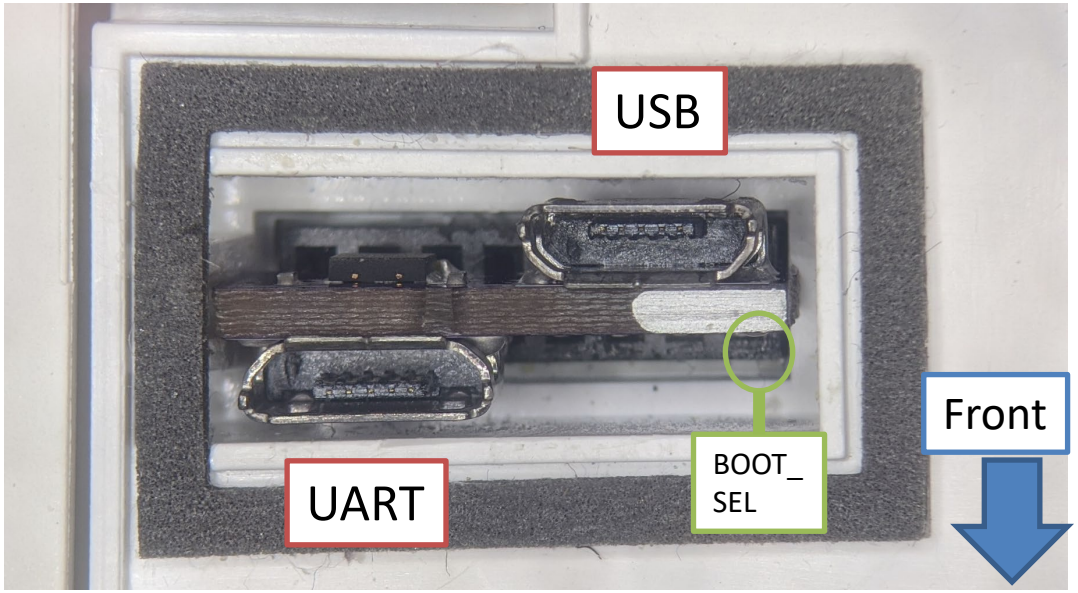
Usage of basic PCB



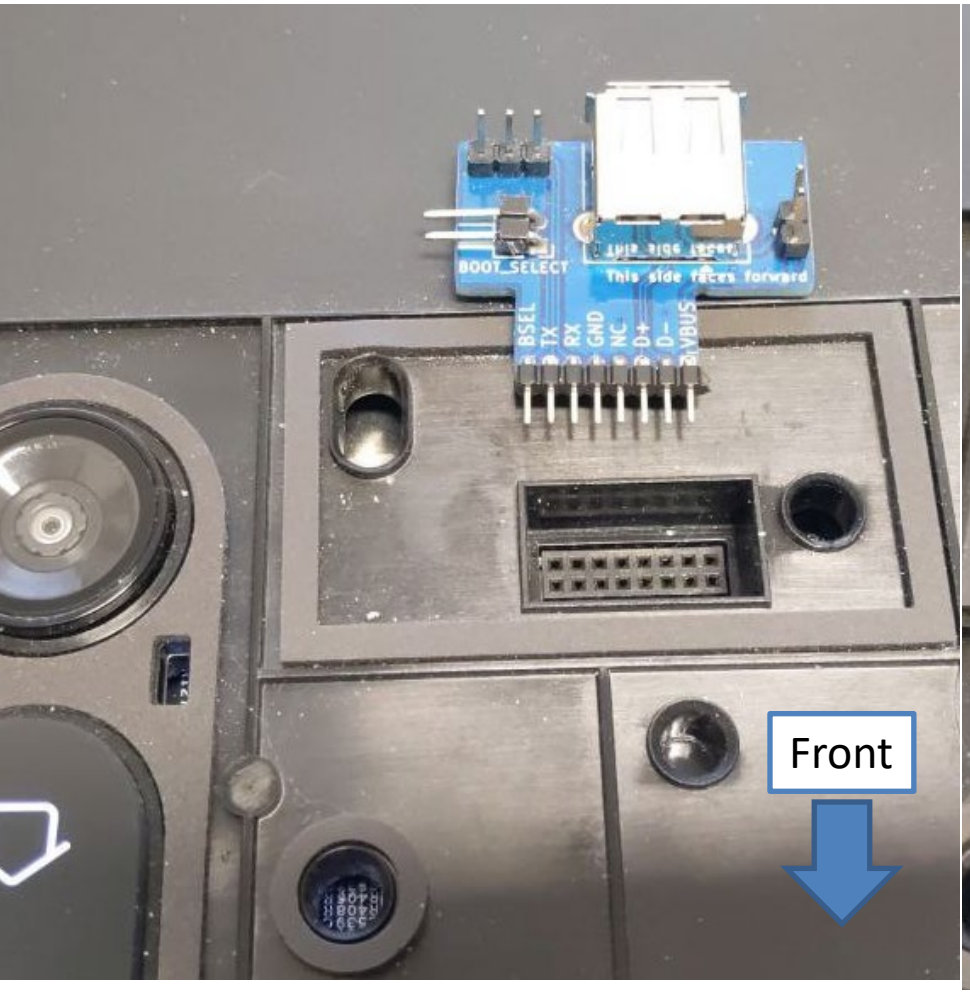
Usage of advanced PCB

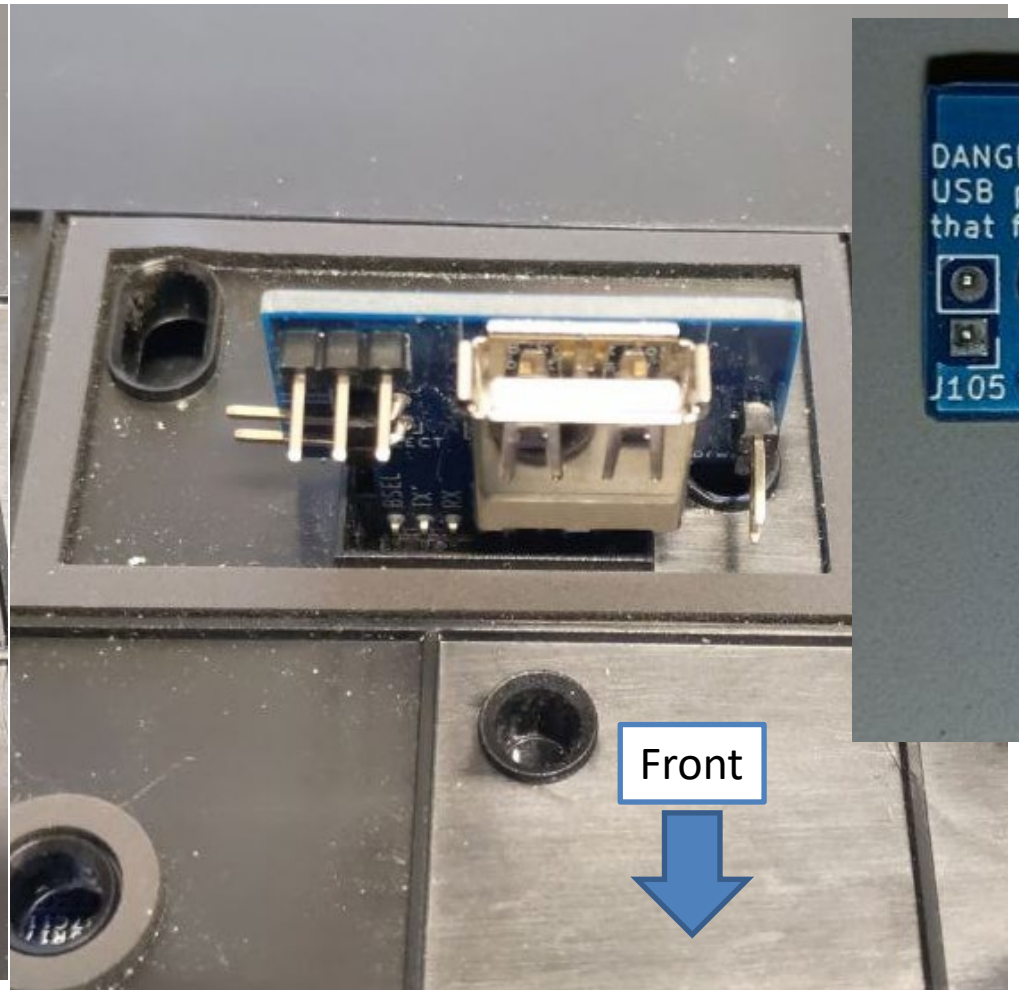






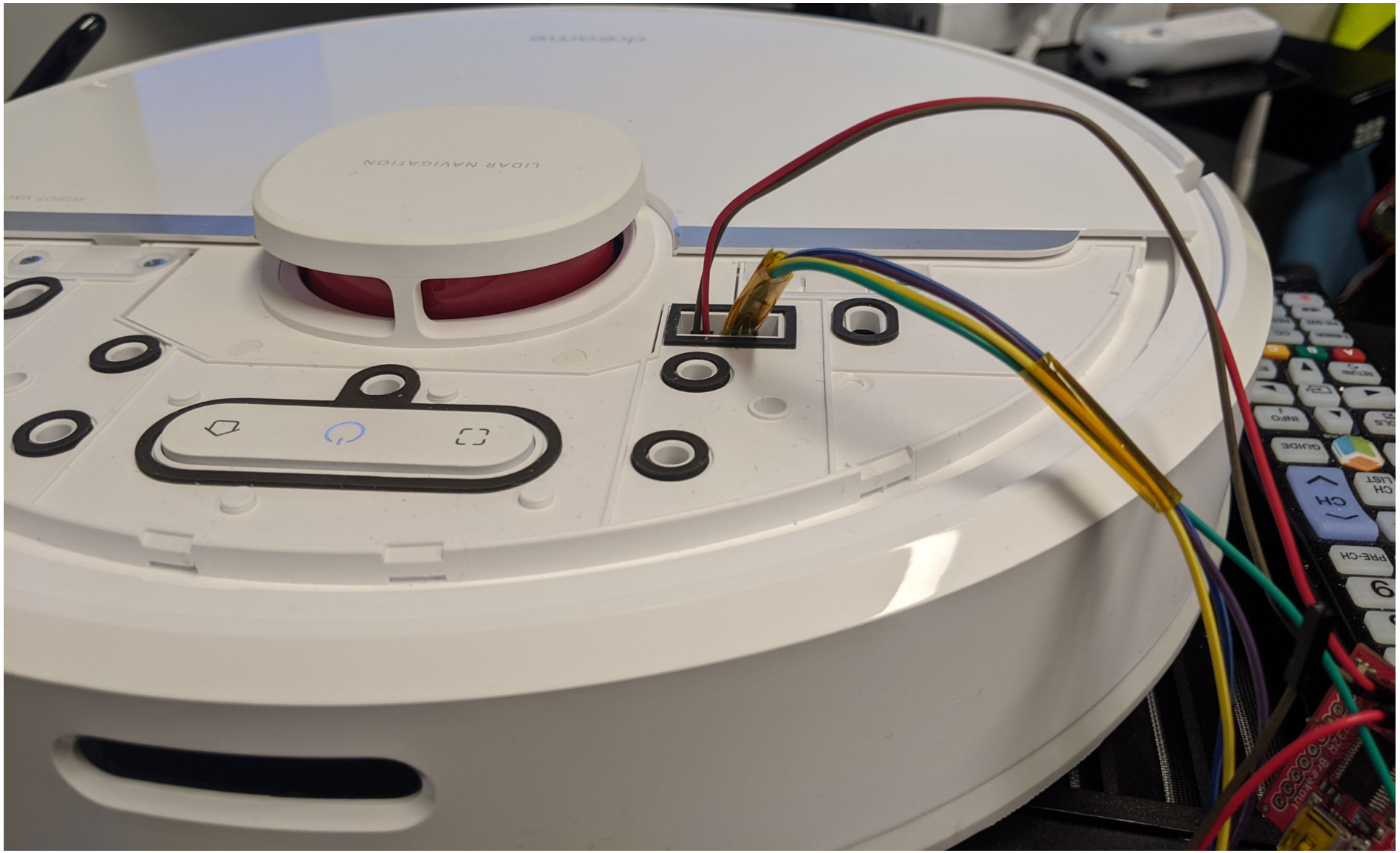
Usage of Archi's PCB*





* Created by Telegram User Sebastian (@ArchimedesMP)

Connecting jumper wires (2mm pitch)



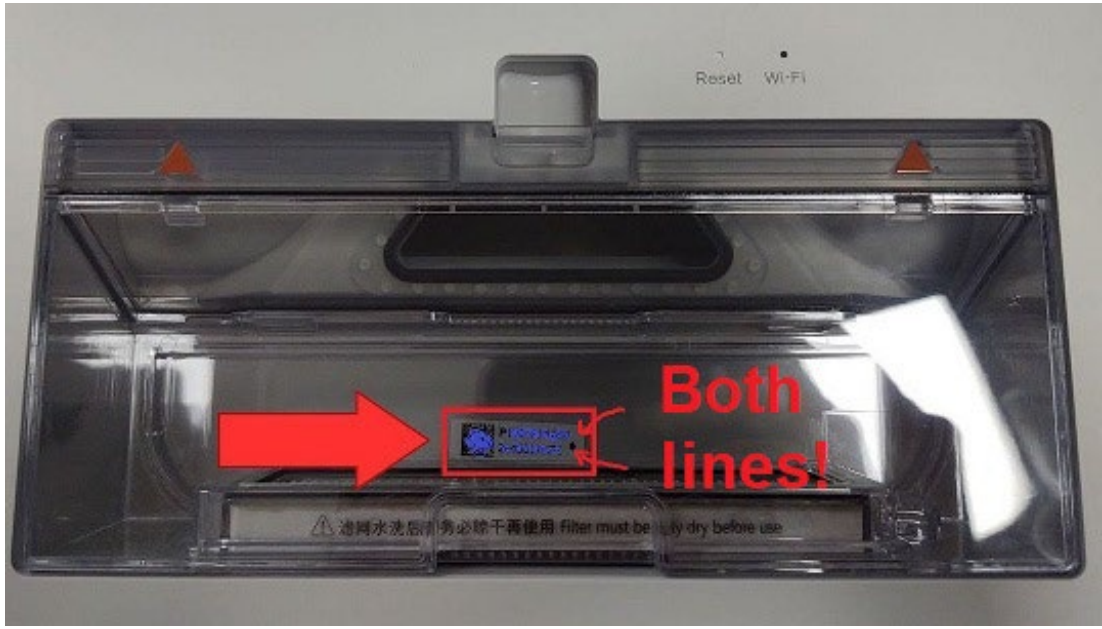
RESET METHOD

Reset Method

- You can find all commands and steps in a text file here:
  - https://builder.dontvacuum.me/dreame/cmds-reset.txt
  - It is recommended that you copy+paste the commands (instead of typing them)
- Background
  - The login shell is by default disabled on Dreame robots
  - The developers left a hidden function:
    - If reset button is pressed for <=1 second, a login shell is triggered
    - The root password is computed from the device serial number
  - This hidden function is patched in firmwares newer than 10.2021

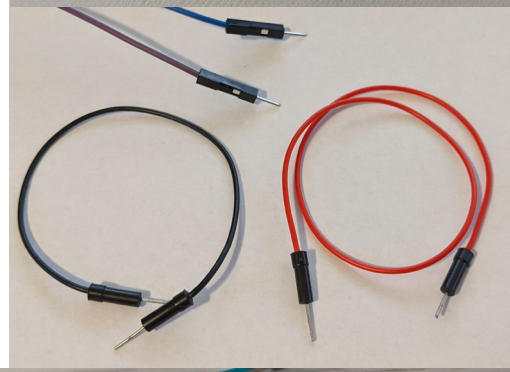
Reset Method Step 0

- Compute the root password from the serial number
  - Under Linux:
    - 'echo -n "P2009XXXXXXXXXXXXX" | md5sum | base64’ (where “P2….” is your serial)
    - The expected result is: 'Y2QxNjA4YjdhNWU1Y2RlNGQ3ODkzZjY4Yzk1MTVkOTAgIC0K’
  - Or use https://builder.dontvacuum.me/dreamepassword.php
- Important:
  - Use the serial under the dustbin
  - Use both lines
  - Serial number under the robot differs!!



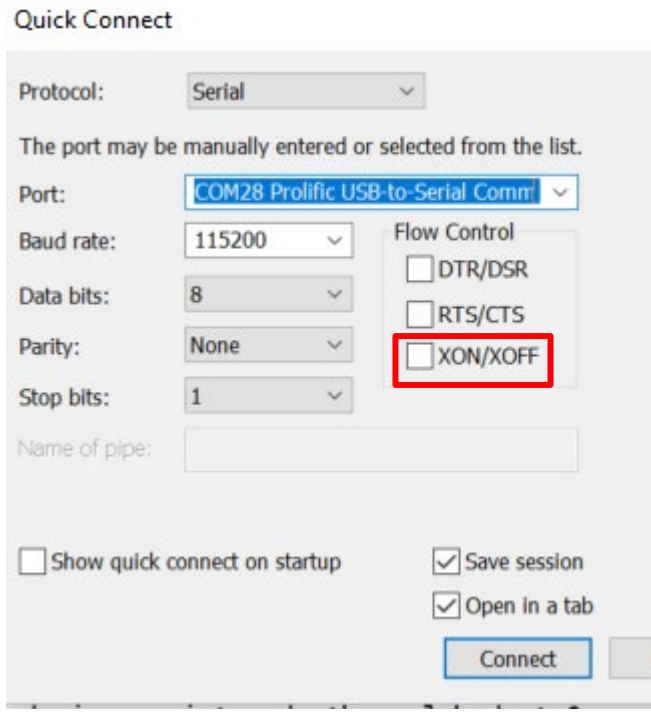
Reset Method Step 1

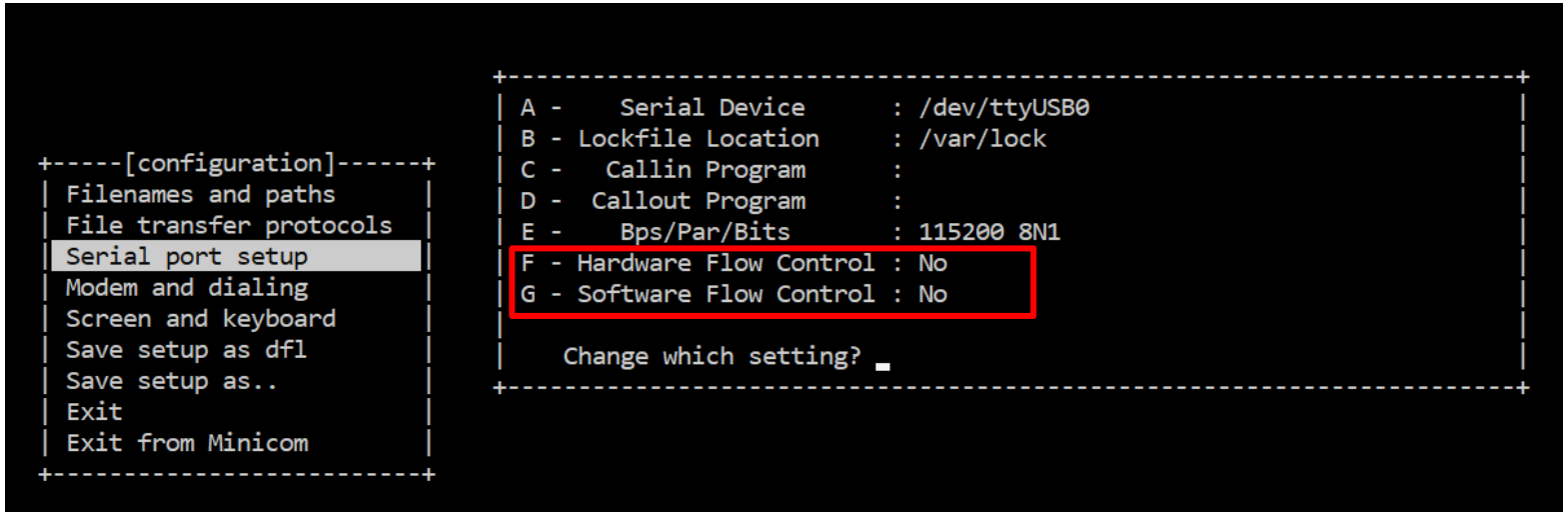
- Get an adapter
- Reminder: Works only on 1C/F9/D9
- UART-USB adapter (3.3V, aka TTL adapter)
  - Typical chipsets:
    - FT232RL, FT232, PL2303TA or CP2102
  - Price ~10 USD/Euro

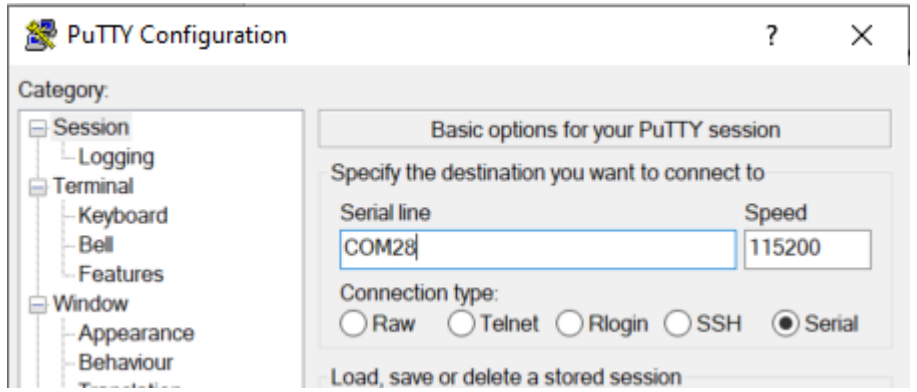


Reset Method Step 2a

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





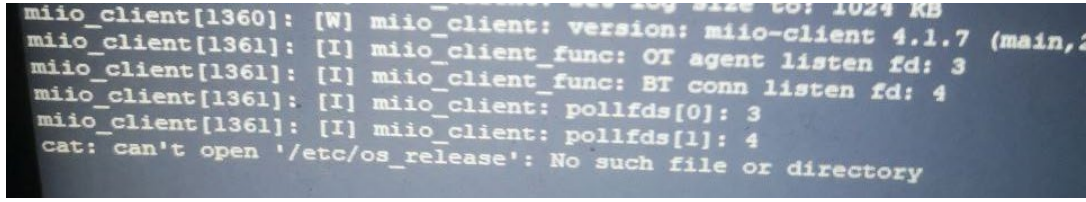


Reset Method Step 2b

- Power off the robot before proceeding
- Remove it from the base station
- Connect serial wires to PCB
  - Do not connect 5V (red cable)!
  - Test for correct connection
    - Press the power button for 3 seconds
    - You should see some output
    - If you don’t see anything, try to switch RX and TX

Reset Method Step 3

- Wait for the robot to boot
- Press reset button for <=1sec
  - The reset button is next to the WiFi LED
- If you see something like “/etc/os_release”, your firmware might too new (or too old for 1C/F9/D9) ☹



- If success full, you should see “pxxx_login:”
  - enter user “root”
  - enter password (that you computed in Step 0)
  - You should have a shell now

Reset Method Step 4

- Check this document: https://builder.dontvacuum.me/dreame/cmds-reset.txt

- Check the current installed version (check "fw_arm_ver"):
  - cat /etc/os-release
- Run these commands to print the configuration (save output):
  - grep "" /mnt/private/ULI/factory/*
- Run these commands to save the calibration (save output, outside of the robot):
  - grep "" /mnt/misc/*.json
  - grep "" /mnt/misc/*.yaml
  - cat /mnt/misc/*.txt
  - hexdump /mnt/misc/*.bin

Depending on your device model, some of the files might not exist for you. If you get file errors, it is totally fine and you can proceed.

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

Reset Method Step 5

- Check this document: https://builder.dontvacuum.me/dreame/cmds-reset.txt
- Make copies of the firmware encryption keys (e.g., to restore the robot to a factory state)
  - cat /etc/OTA_Key_pub.pem
  - cat /etc/publickey.pem
- Make sure that you save the output of the calibration and the encryption keys before proceeding
- Run these commands to create binary backups:
  - mkdir -p /tmp/backup
  - tar -cvzf /tmp/backup/misc_backup.tgz -C /mnt/misc .
  - tar -cvzf /tmp/backup/factory.tgz -C /mnt/private/ULI/factory/ .
  - cp /tmp/backup/*.tgz /data/

Reset Method Step 6

- Check this document: https://builder.dontvacuum.me/dreame/cmds-reset.txt
- If the robot is already in your WiFi, skip this step
- If the robot is still unprovisioned, you have 2 choices
  1. Connect the robot temporarily to the WiFi
     - Use the command: wifi_cli -c "SSID" "password“
     - You must reset the WiFi after you installed the custom firmware, otherwise Valetudo will not work!
  2. Connect your computer to the robot‘s WiFi, and run a local HTTP server on your computer
     - Example: use python http.server or SimpleHTTPServer*

     * https://developer.mozilla.org/en-US/docs/Learn/Common_questions/set_up_a_local_testing_server

Depending on your device model and firmware version, this might not work

Reset Method Step 7

- Check this document: https://builder.dontvacuum.me/dreame/cmds-reset.txt
- Make sure that your robot is fully charged and back in its dock
- Run this commands to download an install the rooted firmware:
  - cd /tmp
  - wget --no-check-certificate {url-of-firmware.tar.gz}
  - tar -xzvf {name-of-firmware.tar.gz}
  - ./install.sh

Only for 1C/F9/D9

UART METHOD

Do not use this method if the Reset method works for you!

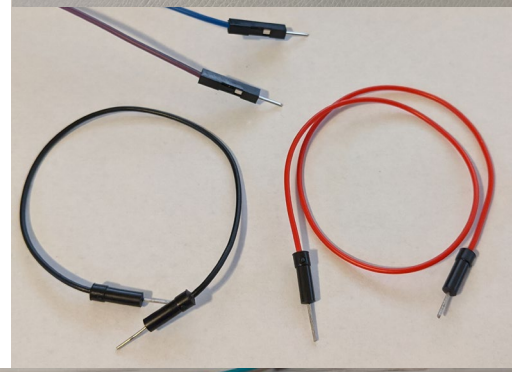
UART method

- You can find all commands and steps in a text file here:
  - https://builder.dontvacuum.me/dreame/cmds.txt
  - It is recommended that you copy+paste the commands (instead of typing them)
- 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
6. Install custom firmware

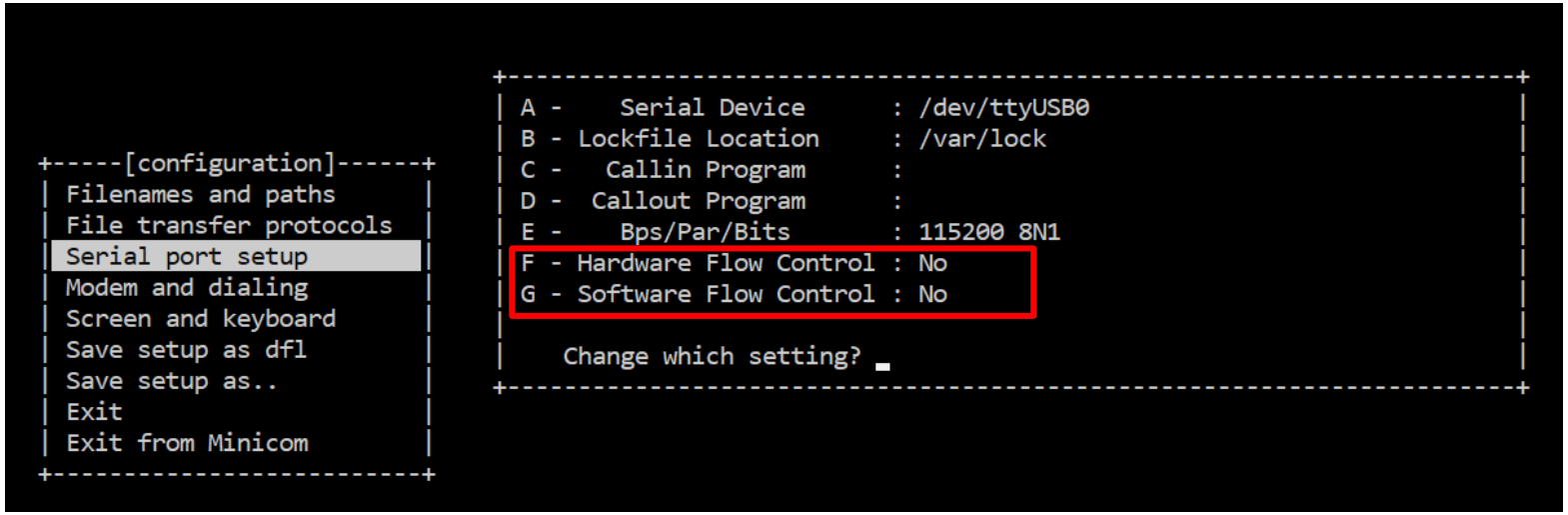
UART Method Step 1

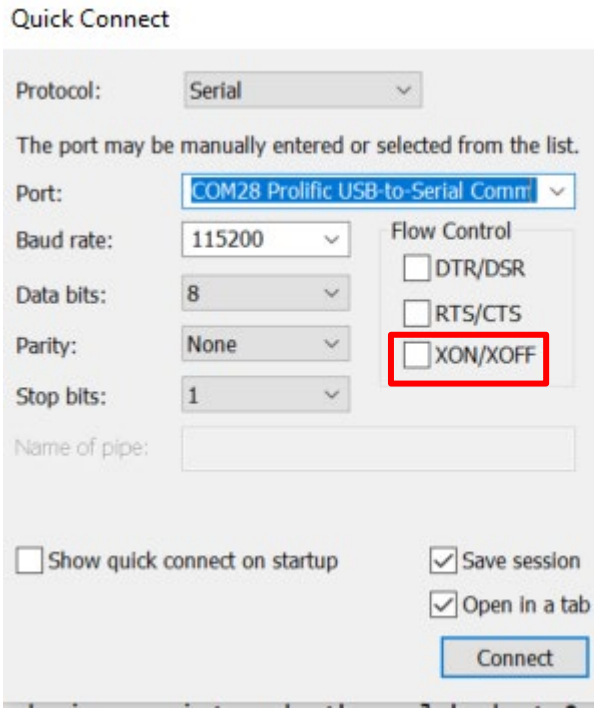
- Get an adapter
- Reminder: Works only on 1C/F9/D9
- UART-USB adapter (3.3V, aka TTL adapter)
  - Typical chipsets:
    - FT232RL, FT232, PL2303TA or CP2102
  - Price ~10 USD/Euro

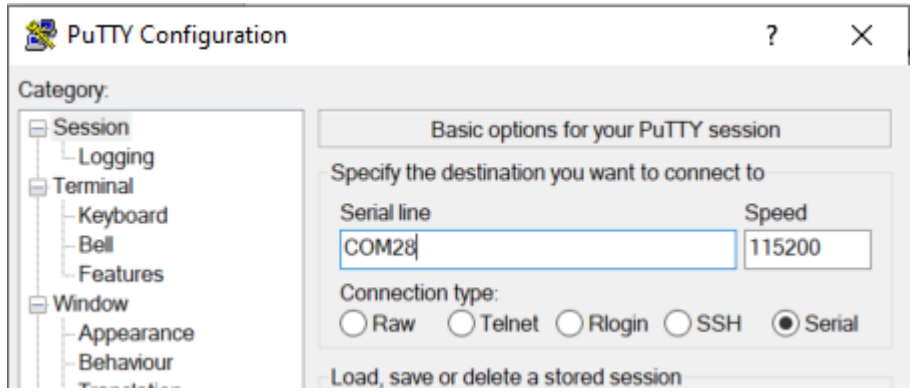


UART Method Step 2a

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





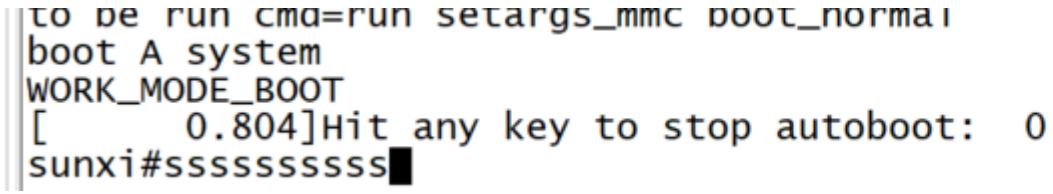


UART Method Step 2b

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

UART Method 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:



UART Method 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

UART Method Step 4b

- After the system booted, run these commands (copy+paste from the cmds.txt):

  `/etc/init.d/sysconfig.sh`

  `echo V > /dev/watchdog`

  `/etc/init.d/mount_private.sh`

  `/etc/init.d/mount_misc.sh`

  `mount /tmp`

  `mkdir /tmp/fakeetc`

  `cp -R /etc/* /tmp/fakeetc`

  `mount --bind /tmp/fakeetc /etc`

  `echo >> /tmp/fakeetc/inittab`

  `echo '::respawn:-/bin/sh' >> /tmp/fakeetc/inittab`

  `exec init`

UART Method 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

UART Method Step 5

- Check this document: https://builder.dontvacuum.me/dreame/cmds.txt
- Make copies of the firmware encryption keys (e.g., to restore the robot to a factory state)
  - cat /etc/OTA_Key_pub.pem
  - cat /etc/publickey.pem
- Make sure that you save the output of the calibration and the encryption keys before proceeding
- Run these commands to create binary backups:
  - mkdir -p /tmp/backup
  - tar -cvzf /tmp/backup/misc_backup.tgz -C /mnt/misc .
  - tar -cvzf /tmp/backup/factory.tgz -C /mnt/private/ULI/factory/ .
  - cp /tmp/backup/*.tgz /data/

UART Method Step 6

- Check this document: https://builder.dontvacuum.me/dreame/cmds.txt
- If the robot is already in your WiFi, skip this step
- If the robot is still unprovisioned, you have 2 choices
  1. Connect the robot temporarily to the WiFi
     - Use the command: wifi_cli -c "SSID" "password“
     - You must reset the WiFi after you installed the custom firmware, otherwise Valetudo will not work!
  2. Connect your computer to the robot‘s WiFi, and run a local HTTP server on your computer
     - Example: use python http.server or SimpleHTTPServer*

     * https://developer.mozilla.org/en-US/docs/Learn/Common_questions/set_up_a_local_testing_server

Depending on your device model and firmware version, this might not work

UART Method Step 7

- Check this document: https://builder.dontvacuum.me/dreame/cmds.txt
- Make sure that your robot is fully charged and back in its dock
- Run this commands to download an install the rooted firmware:
  - cd /tmp
  - wget --no-check-certificate {url-of-firmware.tar.gz}
  - tar -xzvf {name-of-firmware.tar.gz}
  - ./install.sh

LIVESUIT/FEL ROOTING

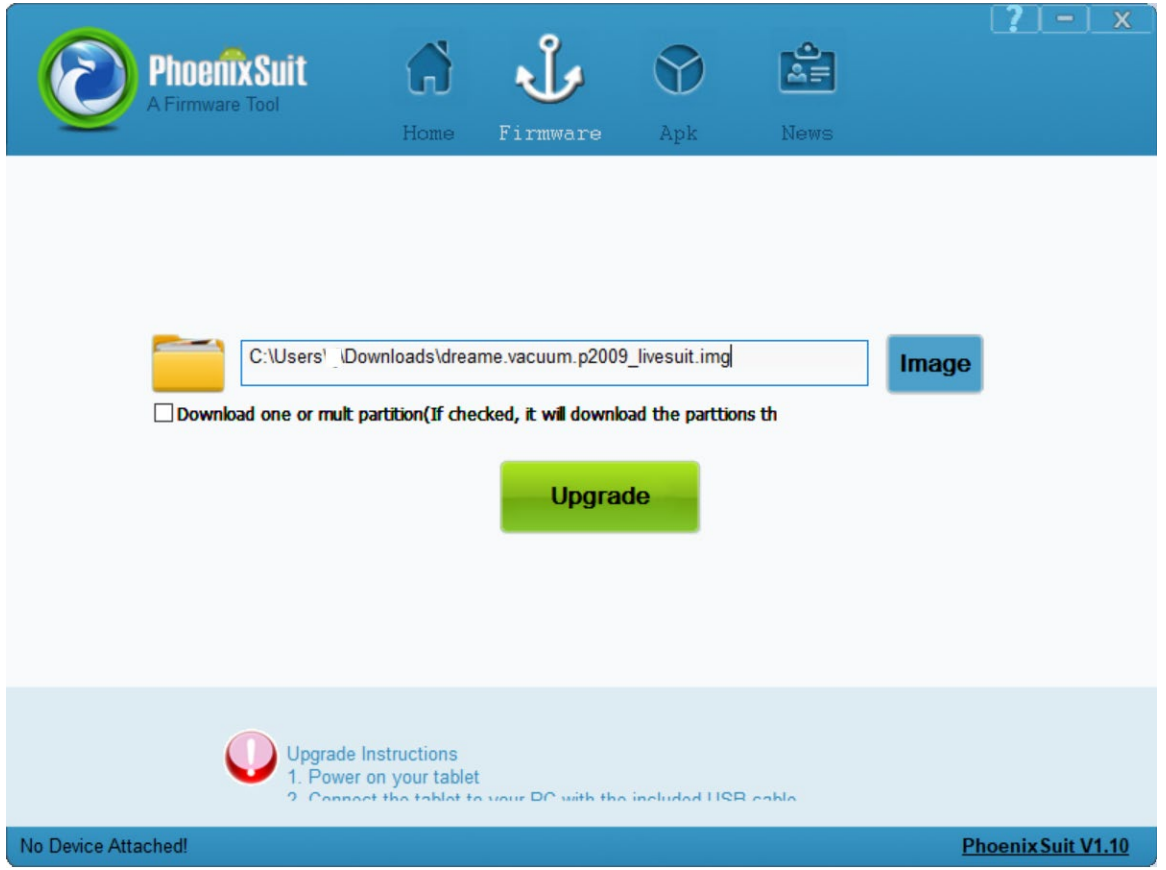
This method is our fallback method in case the other methods do not work. There is more risk involved. Before you use it, please ask us.

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

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

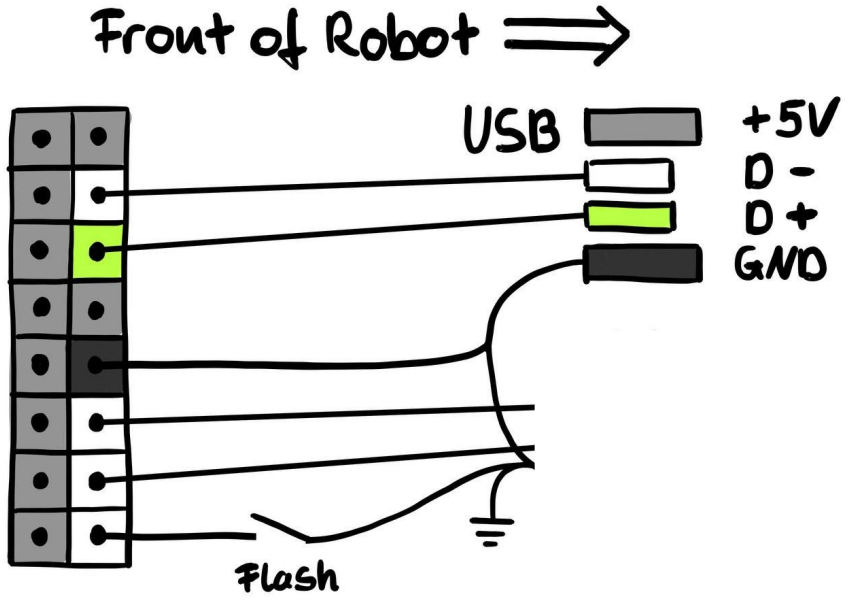


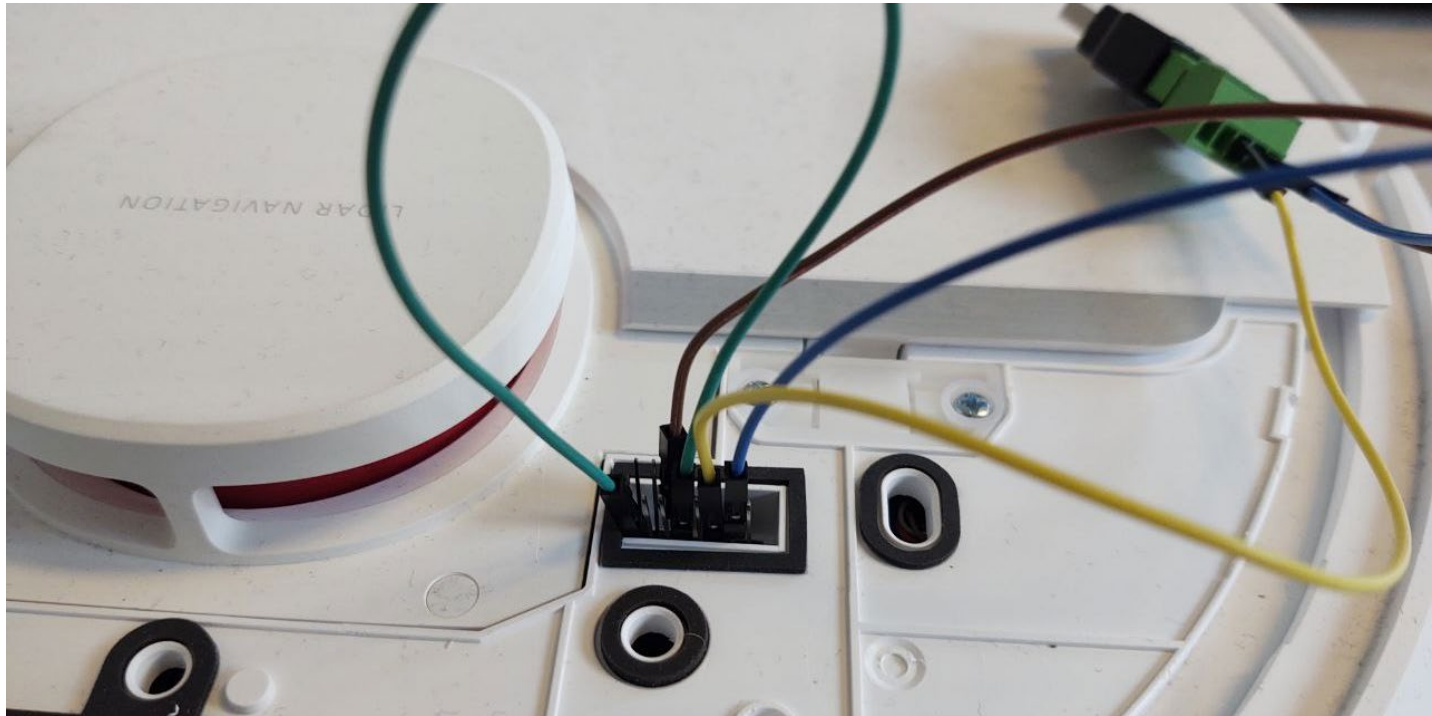
Root Step 2

- 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

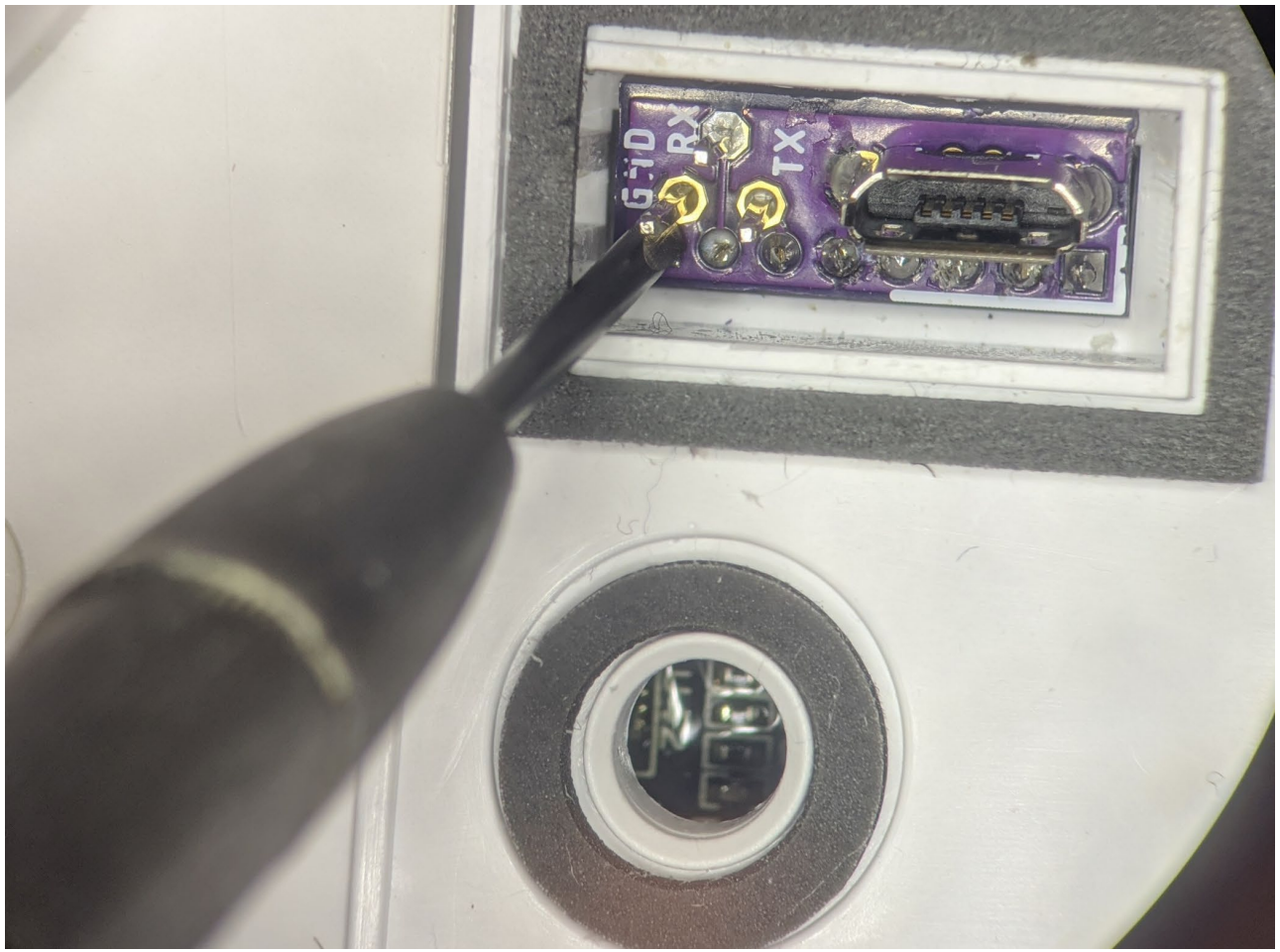
- Jumper wires: Ideally you have 2mm headers
  - Make sure that the USB connection stays stable before you trigger the update (check the Windows Device Manager)
  - 2.54 pitch jumper cables might fit but can cause issues





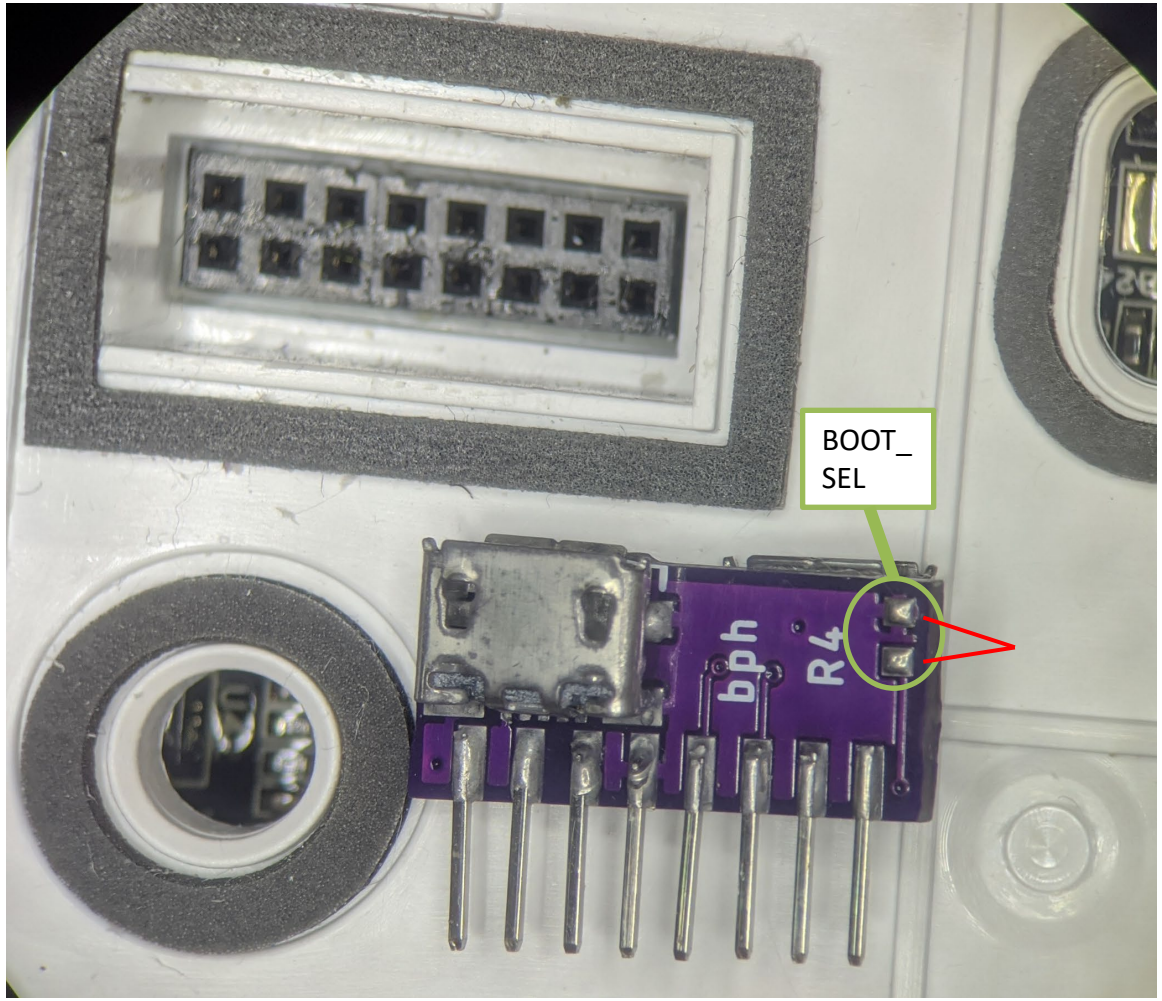
Root Step 2b

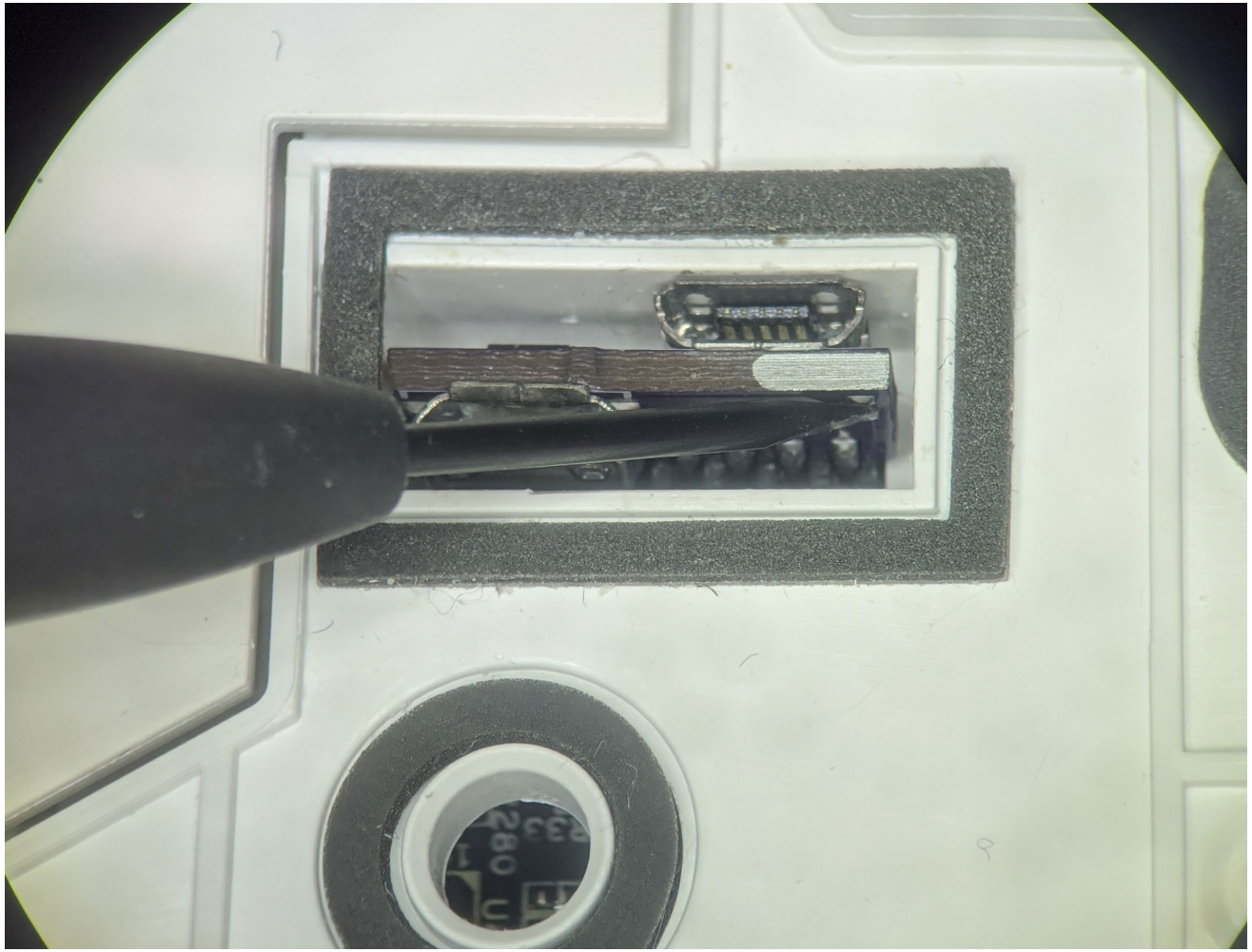
- Basic PCB: Short the left pin to GND (e.g. with screwdriver)



Root Step 2c

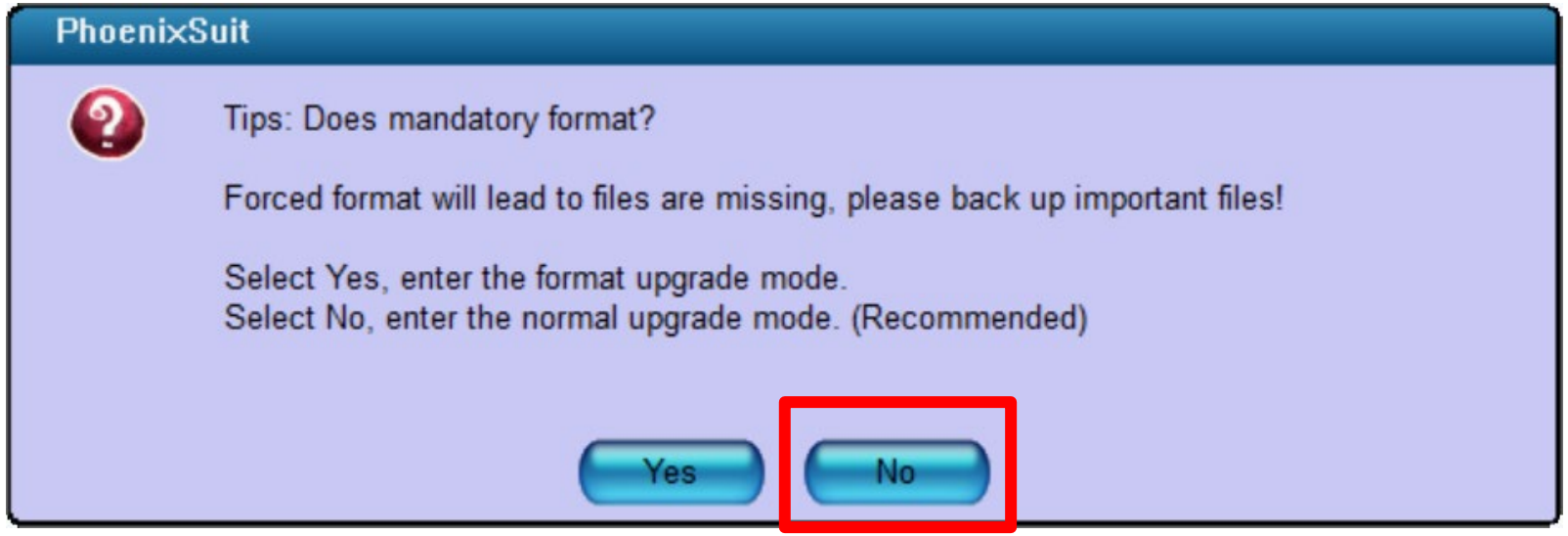
- Advanced PCB: Short BOOT_SEL pads





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
- Use only USB 2.0 connections
- 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

FIRMWARE CUSTOMIZATION

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

You might need to add “--no-check-certificate” if wget fails

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


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

Only relevant if you have a 1C/F9/D9 and Reset method does not work

BACKUPS VIA UART

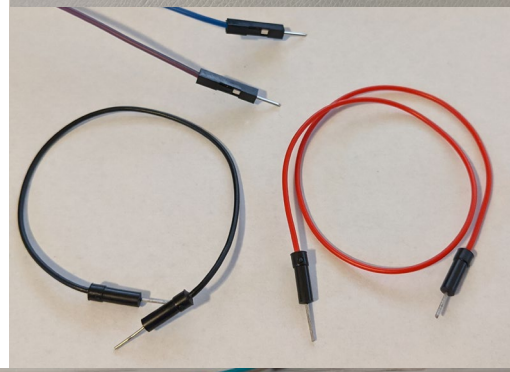
Backup of configuration and calibration

- 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

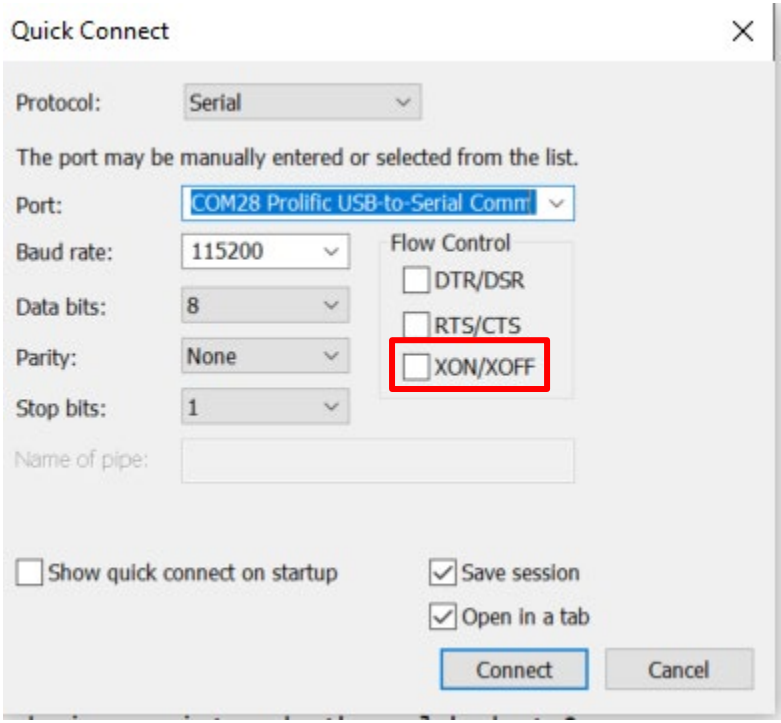
Backup Step 1

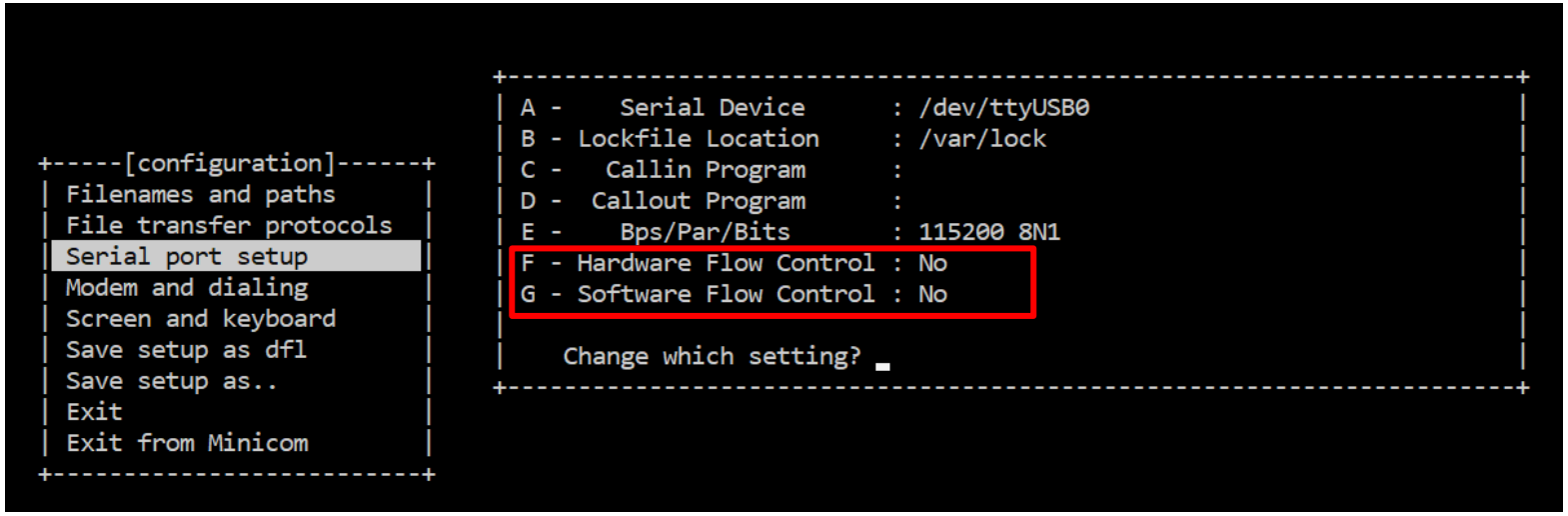
- Get an adapter
- Reminder: Works only on 1C/F9/D9
- UART-USB adapter (3.3V, aka TTL adapter)
  - Typical chipsets:
    - FT232RL, FT232, PL2303TA or CP2102
  - Price ~10 USD/Euro

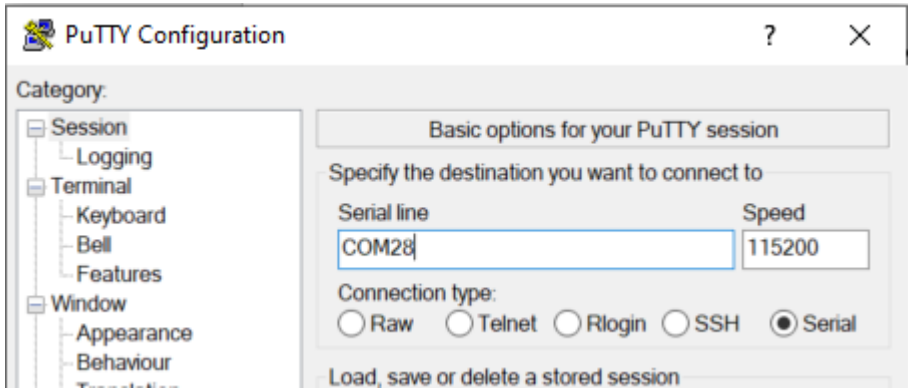


Backup Step 2a

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





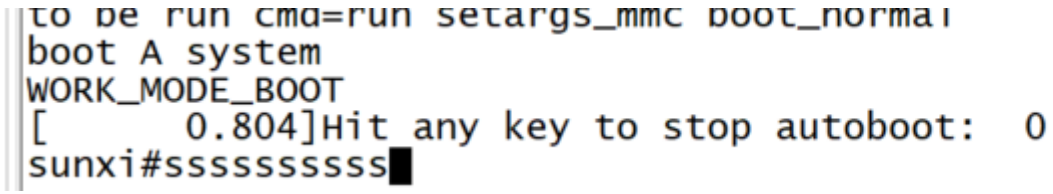


Backup Step 2b

- 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!