OSP Hardware - PCD Setup (macOS, Linux)

Created: May 18th, 2021

Last Updated: May 22nd, 2021

For Windows installation, see the PDF document “documentation_pcdSetup_macOS_Linux.pdf” or the public Google doc.

Table of Contents

Disclaimer - Device Overheating Issue

While the device is plugged in with a USB cable(s) and operating at the same time, the device will eventually overheat, which can reduce its audio quality and performance. The OSP team is working to resolve this.

For the time being, it is recommended to set up a SSH connection so that you don’t have to use a cable to connect the Processing and Communication Device (PCD) to your computer.

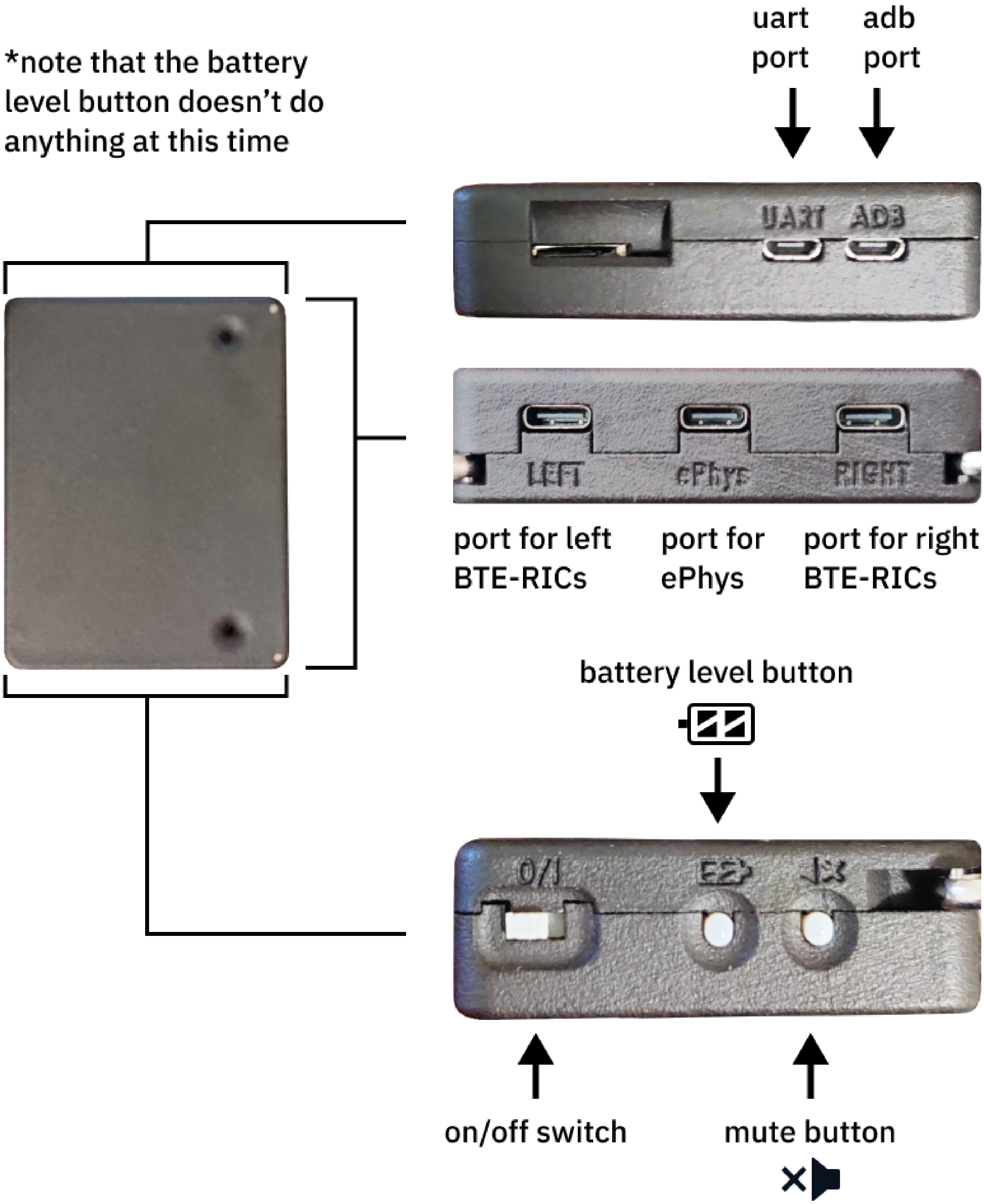
1 - Purpose of Document

- To install new software on OSP’s Processing and Communication Device (PCD), the main OSP hardware device. The instructions in this guide may also refer to the PCD as "the device".
- To connect the PCD using a serial port and ssh.
- To customize the device operation.

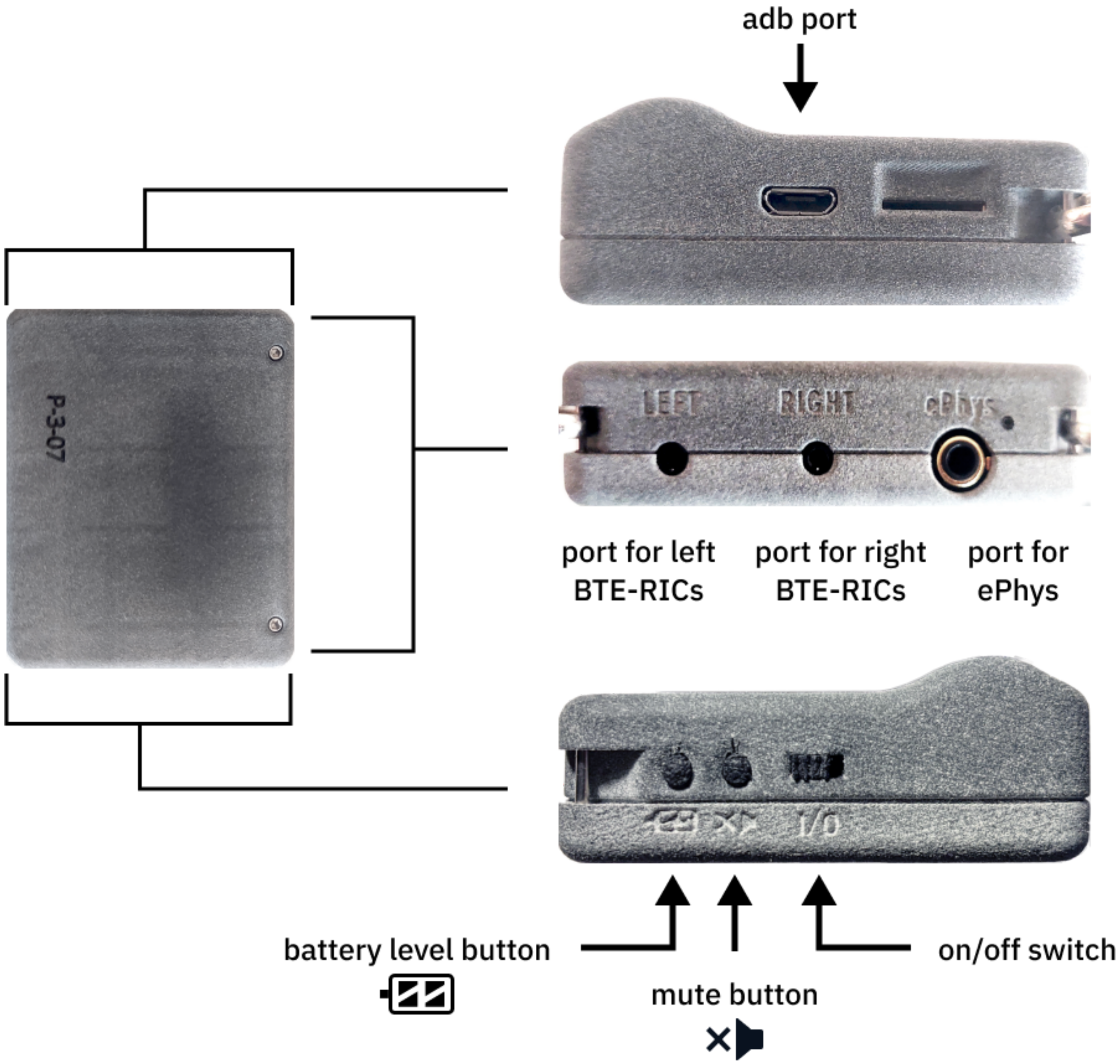
For more information not covered in this guide, please visit this directory "OSP Hardware Documentation" via OSP's GitHub page.

V8 Version of the Processing and Communication Device (PCD)

*note that the battery level button doesn't do anything at this time



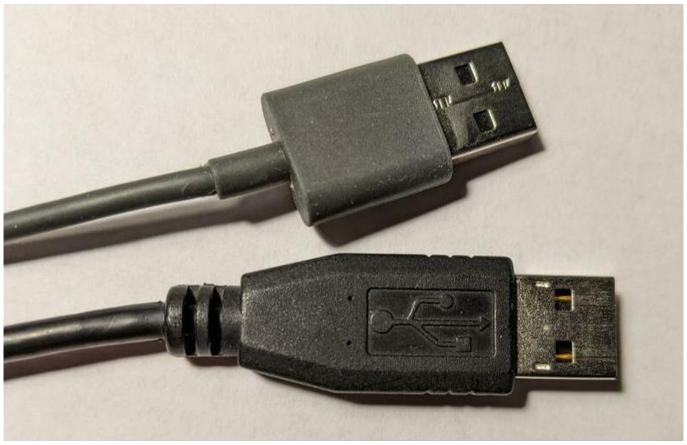
V9 Version of the Processing and Communication Device (PCD)



2 - Requirements

- The device (PCD), either the V8 or V9 version.
    - *V8 version* - Has two micro-usb ports labelled "UART" and "ADB".
    - *V9 version* - Has a single micro-usb port.

- Micro-usb cable(s) that **can exchange data**.
    - 2 cables for the V8 version, 1 cable for the V9 version.
    - Use cables that can exchange data and verify that they can, even if they have the trident logo. Do not use any cables that only charge the device (aka "charge-only").



The top cable is a "charge-only" usb cable, do not use these. Bottom cable is a usb cable with the trident logo, which typically signifies that you can exchange data.

- Knowledge of using a Mac or Linux computer to update (flash) the device.
- Knowledge of using a terminal. Almost all of the terminal commands are covered in this guide.
- Software packages for your Mac or Linux computer. Steps for installing these packages will be covered in this guide.
    - *Linux Only* - "fastboot" package from android tools
    - *Linux and macOS* - Python3, and python packages "pyserial" and "colorama". Use Homebrew to install Python3.

Windows can be used to connect to the serial USB port or ssh into the device once the device is flashed.

3 - Installation Steps for Hardware

3.1 - Download, Save, and Open the PCD .zip File

Go to this link to the public "Releases" folder in Google Drive and download a .zip file named "pcd-[date].zip" of the most recent version.

It is recommended to save the .zip file somewhere convenient on your computer, such as your “Downloads” or “Desktop” directory.

3.2 - Extract the .zip File

Go to your file manager application (such as Finder for Mac), and extract the contents of the zip file, and open the extracted folder.

In step 3.7, you will need to use the terminal to navigate inside the "pcd-[date]” folder that gets produced from the extracted .zip file.

For example, if you have extracted a .zip file named “pcd-18May21”, you will see a folder named “pcd-18May21”.

3.3 - Connect the Device to Your Computer Using Micro-USB Cables

Depending on which version of the device you have, do one of the following:

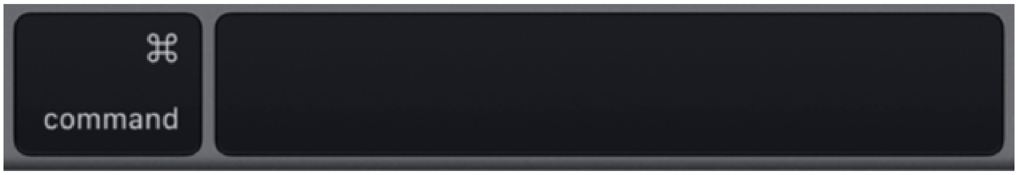
- **If you have the V8 version with two micro-usb ports**, attach one micro-usb cable to the “ADB” port. Attach the other cable to your computer or the USB hub.
- **If you have the V9 version with one micro-usb port**, attach one micro-usb cable and connect the cable to your computer via the USB hub.

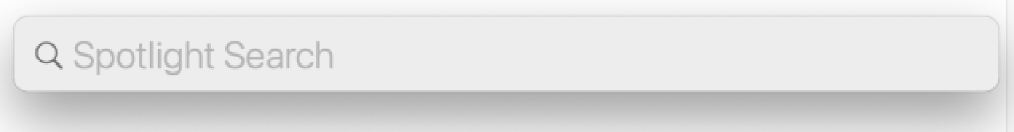
Reminder to only use cables that can transfer data. The micro-usb ports are used for charging, flashing, and connecting as a serial port.

3.4 - Open a Terminal Window

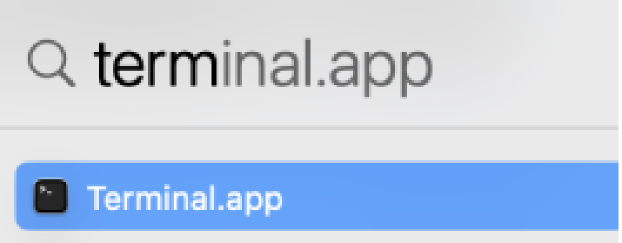
For Mac

1. Open Spotlight Search (or simultaneously press the command and spacebar keys), which looks like a long horizontal area in your screen with a Search icon.



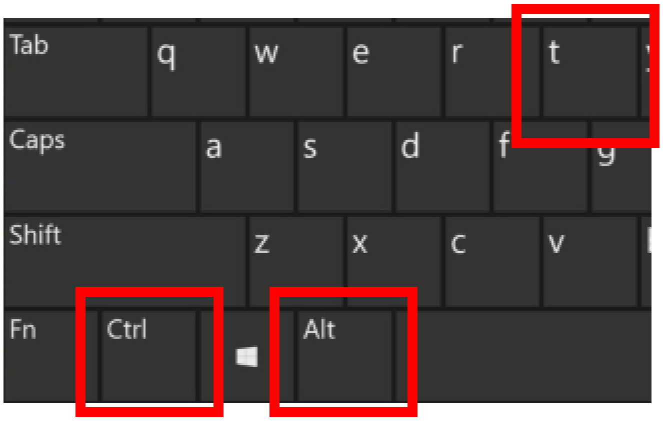


2. Type in “terminal”, and the application should show up automatically. Press enter.

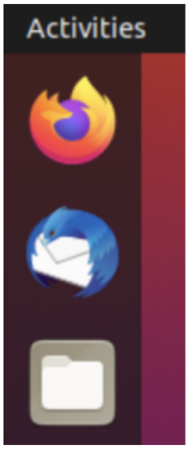


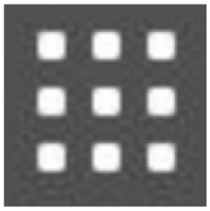
For Linux

- Simultaneously press the "Ctrl", "Alt", and "t" keys on your keyboard to open a terminal.



- If this doesn’t work, select “Activities” on the upper left corner of your screen or click on the 3x3 icon on the side of your screen. Use the search feature to type in "terminal" and open the terminal app.





3.5 - Install Required Software Packages

For Linux, install the "fastboot" package by typing this command into the terminal (and pressing enter): `sudo apt install fastboot`

For macOS and Linux, Python3 must be installed using Homebrew.

- On macOS, enter the command: `ruby -e "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/master/install)"`
- On Linux, use your package manager to install it. Depending on the type of Linux you're using (e.g. Debian or Fedora), the command(s) used for Python3 may vary.

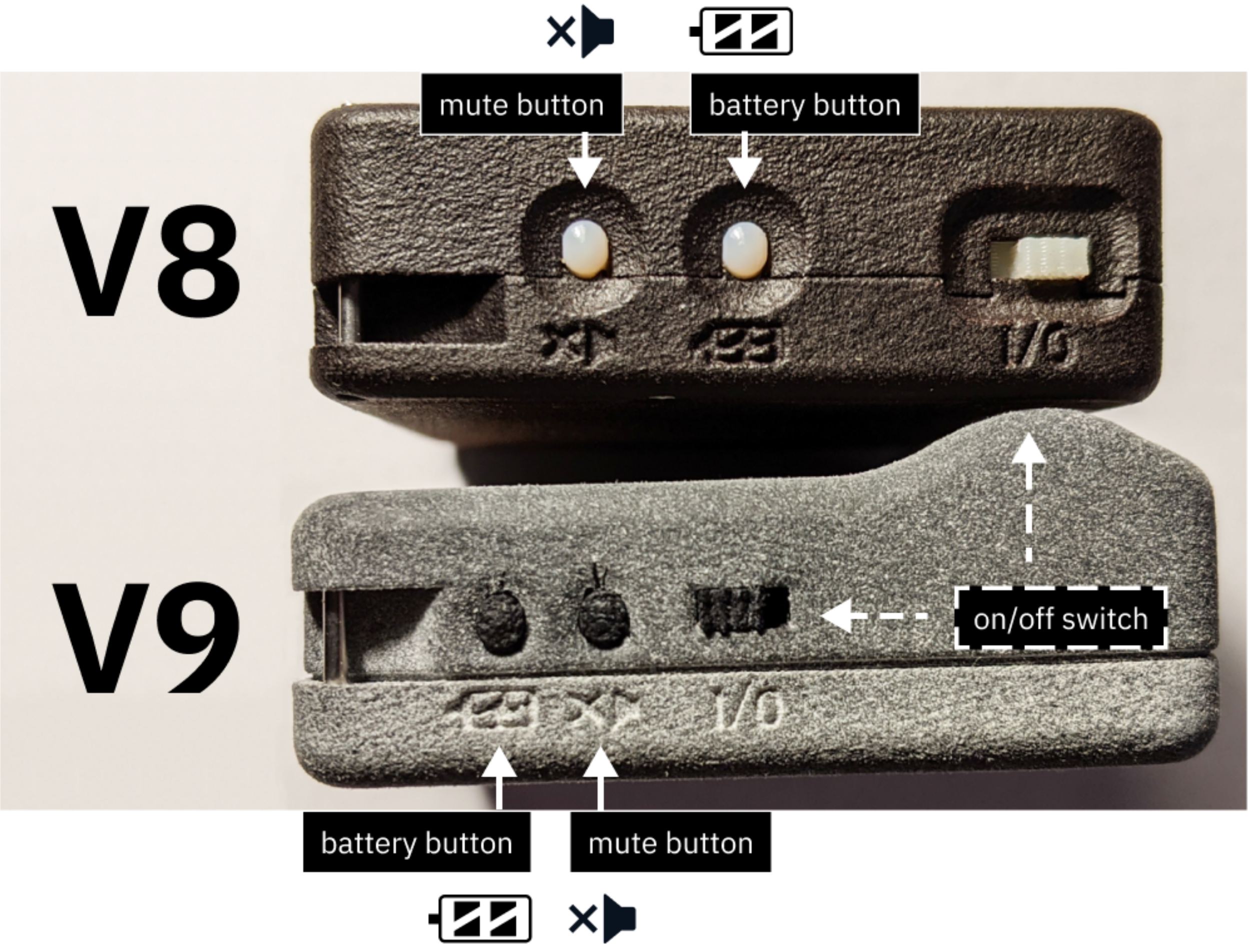
Then, enter the terminal command: `pip3 install pyserial colorama` to install the two non-standard Python3 packages "pyserial" and "colorama". You may need to prefix this command with `sudo` depending on how Python3 is installed on your system.

3.6 - Turn on the Device in Boot Mode

To flash the device, the device must first be turned on and set to Boot mode. Do one of the following for your device version (also see image below).

- *V8 Version* - While pressing and holding down the *battery button,* flick the on/off switch **towards the battery button**.
- *V9 Version* - While pressing and holding down the *mute button,* flick the on/off switch **towards the mute button.**

If you have received one of the pre-production V9 units with the mislabelled on/off switch, you may need to flick the on/off switch away from the mute button. **For these units, the device will turn on when the switch is towards the "0"**.



(top area) V8 device. The battery button is closer to the on/off switch.
(bottom area) V9 device. The mute button is closer to the on/off switch.

3.7 - Navigate to the “pcd-[date]” Folder

In your terminal, navigate to the “pcd-[date]” folder that was produced from step “3.2 - Extract the .zip File”. You can do this by entering the command: `cd [PATH_TO_“pcd-[date]”_FOLDER]`

For example, if you have extracted a .zip file named “pcd-18May21”, you will see a folder named “pcd-18May21”. If that folder was saved within “Downloads”, you would enter: `cd Downloads/pcd-18May21`

```
MacBook-Pro:~ USER$ cd Downloads/pcd-18May21
MacBook-Pro:pcd-18May21 USER$
```

3.8 - Flash the Image to the Device

The "pcdtool" utility can flash the filesystem, boot the device, connect to it, and transfer files to and from it.

Enter the terminal command `./pcdtool flash` to flash the device.

You may be prompted to enter your computer password before the terminal proceeds executing the command.

```
> ./pcdtool flash

 FLASHING DEVICE.  Hold mute button and power on
device

Flashing rootfs and rebooting

Password:
sending 'boot' (16406 KB)...
OKAY [  0.518s]
writing 'boot'...
OKAY [  0.342s]
finished. total time: 0.860s

target reported max download size of 268435456
bytes
sending sparse 'rootfs' 1/19 (230025 KB)...
```

If something goes wrong, the bootloader in the device may have been corrupted. If so, try entering this terminal command: **`./pcdtool flash -b`**

To view all the available commands for the device, enter the command **`./pcdtool`**

You’ll see the following commands below.

```
> ./pcdtool
usage: pcdtool [-h] {connect,conn,flash,recv,send}
...

Serial port utility for PCD Devices

pcdtool conn[ect]
    Connects to attached pcd device and displays
output. Hit ^Z to exit.

pcdtool flash [-b]
    Connects to attached pcd device, flashes the
the rootfs
    and restarts the device.  If -b flag is there,
it first flashes the bootloader.

pcdtool send {filename}
    Sends the file to the current directory on the
device.

pcdtool recv {filename}
    Receives a file from the device.

A device must first be connected via USB on
/dev/ttyUSB0 (or /dev/cu.usbserial* for Mac)

positional arguments:
  {connect,conn,flash,recv,send}
```

```
optional arguments:
  -h, --help            show this help message and
exit
```

3.9 - Connect to the Device Using the Serial Port

You can use a terminal program, such as **pcdtool**, to connect the PCD to a usb port (/dev/ttyUSB0 on Linux or /dev/cu.usbserial* for Mac). You may need to specify the device name and speed (baud rate) to use so that the terminal program can start the connection up. If you have multiple devices, you may see ttyUSB1, cu.usbserial1, etc. and they only appear when the PCD is attached and powered on.

If you're continuing to use **pcdtool** after flashing the device, you need to connect to the device by entering the terminal command: **./pcdtool conn**

```
> ./pcdtool conn


Hit Ctrl-Z to exit

```

```
OSP VERSION: 08Feb21

Hotspot Mode
Connect to the device at SSID "ospboard" password
"hearingaid" then
open a browser and connect to
http://192.168.8.1:5000 and/or http://192.168.8.1
If you wish to put the board back into
NetworkManager mode then use following command:

set_mode nm
root@ospboard:~>
root@ospboard:~>
```

SSID is the wifi name when you typically view available wifi networks. If you don't see any terminal output, you may need to press the "Enter" key once or twice for terminal feedback (to confirm feedback, you'll see **`root@ospboard:~>`**).

If you are considering an alternative to **`pcdtool`**, **PuTTY** works well (link to PuTTY download) for any installation step that requires the serial port. In PuTTY, set the baud rate (speed) of 115200, which is required to communicate with the PCD.

Troubleshooting Error - "can't open device "/dev/ttyUSB0": Permission denied"

If you receive this error from the terminal, you need to add yourself to the ‘dialout’ group.

To do so, enter the command: `sudo usermod -a -G dialout $USER`

You will then need to reboot the device. If this doesn’t work, you may need to log out and log back into your computer.

Check that you are in the groups by entering the command: `groups`

```
> groups
mmh adm dialout cdrom sudo dip plugdev lpadmin
sambashare docker
```

After flashing the device, you should expect the following:

- You should see “ospboard” show up in the wifi SSID list on a laptop or cell phone.
- You should be able to connect to the device with a serial connection over the USB cable.
- You should be able to see lights flashing inside the case.

3.10 - Hotspot and NetworkManager Modes

The device has two different modes, "Hotspot mode" and "NetworkManager mode".

You can check the device's current mode by entering the command:

```
print_mode
```

You can also have the option to set up SSH to connect to your PCD without needing a USB cable.

Hotspot Mode - Introduction

When the device is first flashed, the device is in hotspot mode by default.

- *Benefits* - This mode allows you to connect to the PCD using a browser-enabled phone or computer via wifi SSID. The PCD has a consistent IP address and is not dependent on any external wifi router.
- *Drawbacks* - The device has no internet connectivity and when you connect your phone or laptop to the PCD, it will lose internet too.

To set the PCD to hotspot mode, enter the command `set_mode hs` and let the device reboot itself.

Entering the `print_mode` command shows the following terminal output for hotspot mode.

```
root@ospboard:~> print_mode
Hotspot Mode
Connect to the device at SSID "ospboard" password
"hearingaid" then
open a browser and connect to
http://192.168.8.1:5000 and/or http://192.168.8.1
If you wish to put the board back into
NetworkManager mode then use following command:

set_mode nm
```

You can connect to the PCD using the PCD's native wifi hotspot by connecting to the wifi “ospboard”, with the password “hearingaid”.

Hotspot Mode - Using SSH to Connect to the Device

To use ssh to connect to the device, enter the command: **`ssh -i ospboard_id root@192.168.8.1`**

“ospboard_id” is a file containing an ssh key that is included in the zip file. To simplify ssh connections to the PCD, you should copy ospboard_id to "~/.ssh" and add an entry to your ".ssh/config" file such as:

```
Host osp_hs
```

```
HostName 192.168.8.1
User root
IdentityFile ~/.ssh/ospboard_id
```

Then, you can enter the command **ssh osp_hs** instead to connect to the device in Hotspot mode.

Hotspot Mode - Setting the HotSpot SSID

For 01Mar2021 and later PCD file releases, you can set the Hotspot SSID and passphrase.

Simply enter the command: **set_hs_name -n ospboard2 -p my_password**

Then, let the PCD reboot itself.

```
root@ospboard:~> set_hs_name -n ospboard2 -p
my_password
Done. Reboot for changes to take effect
root@ospboard:~> print_mode
Hotspot Mode
Connect to the device at ssid=ospboard2 with
wpa_passphrase=my_password
```

```
open a browser and connect to
http://192.168.8.1:5000 and/or http://192.168.8.1
If you wish to put the board back into
NetworkManager mode then use following
command:
set_mode nm
```

NetworkManager Mode - Introduction

NetworkManager mode allows you to connect to a local wifi SSID. Its address will be allocated by the wifi router.

- *Benefit* - Easy connectivity from your phone, laptop, or desktop computer.
- *Drawback* - Device must connect to a specific wifi network. If you want to talk to the device via SSH, you need a specific IP address reserved.

To set the device in NetworkManager mode, enter the command `set_mode nm` and let the device reboot itself.

Once the device has finished rebooting, entering the `print_mode` command shows the terminal output below with a list of SSIDs the device can see. (The list may not be complete, so you can always try other SSIDs.)

```
root@ospboard:~> print_mode
NetworkManager Mode
Connect to your wifi with one of the following:
> nmcli dev wifi con MYSSID password "MY PASSWORD"
or
> nmtui
IN-USE  SSID  MODE   CHAN  RATE        SIGNAL  BARS  SECURITY
        DNRC  Infra  1     130 Mbit/s  49      __    WPA2
        DNRC  Infra  6     130 Mbit/s  42      __    WPA2
        DNRC  Infra  6     130 Mbit/s  40      __    WPA2
If you wish to put the board back into hotspot mode
then use following command:

set_mode hs
```

NetworkManager Mode - Connect to Your Wifi

Connect to your wifi by entering the command: `dev wifi con MYSSID password "MY PASSWORD"`

```
root@ospboard:~> nmcli dev wifi con MYSSID password
"MY PASSWORD"
[  203.944162] wlan0: authenticate with
70:3a:cb:2a:06:d8
```

```
[  203.989349] wlan0: send auth to
70:3a:cb:2a:06:d8 (try 1/3)
[  204.000409] wlan0: send auth to
70:3a:cb:2a:06:d8 (try 2/3)
[  204.002978] wlan0: authenticated
[  204.006728] wlan0: associate with
70:3a:cb:2a:06:d8 (try 1/3)
[  204.023939] wlan0: RX AssocResp from
70:3a:cb:2a:06:d8 (capab=0x1431 status=0 aid=4)
[  204.057159] wlan0: associated
[  204.164330] IPv6: ADDRCONF(NETDEV_CHANGE):
wlan0: link becomes ready
Device 'wlan0' successfully activated with
'2ce4abfd-9c36-446c-9a25-6511d85c9dea'.
```

When entering the **`print_mode`** command, note the assigned IP address number to the right of "inet". This number will be different for your network.

```
root@ospboard:~> print_mode
NetworkManager Mode
        inet 192.168.86.33  netmask 255.255.255.0
broadcast 192.168.86.255
```

```
Connected
If you wish to put the board back into hotspot mode
then use following command:
set_mode hs
```

At this point, the device should be running on your local wifi and will continue to use this mode after it reboots.

You can now simultaneously hit the "ctrl" and "z" keys to exit from the terminal.

You can also disconnect the USB cable if you want.

You will periodically need to reconnect it to a charger to recharge the PCD's battery.

NetworkManager Mode - Using SSH to Connect to the Device

From another computer, you can connect the device by entering the command: `ssh -i ospboard_id root@192.168.86.33`
Replace "192.168.86.33" with your IP address number.

At this point, it is highly recommended that your wifi router reserves that IP address for your device so it doesn't change in the future.

To simplify ssh connections to the PCD, you should copy “ospboard_id” to ~/.ssh and add an entry to your “.ssh/config” file, like the following:

```
Host osp
     HostName  192.168.86.33
     User root
     IdentityFile ~/.ssh/ospboard_id
```

This way, you can simply enter the command `ssh osp` for future use instead of `ssh -i ospboard_id root@192.168.86.33`.

4 - Using Visual Studio Code

Visual Studio Code is an integrated development environment that, when used with the Remote SSH Extension, is a great way to do development on the device.

You will need to setup your “~/.ssh/config” file as described above.

5 - Monitoring and Controlling the OSP Processes

By default, the PCD boots up in Hotspot mode. You can connect to the PCD using the PCD's native wifi hotspot by attaching to SSID “ospboard”, password “hearingaid”.

Then, open a browser window/tab and enter http://192.168.8.1:5000 and http://192.168.8.1:8080 to launch OSP’s software interface.

If you are using NetworkManager mode, you will need to use the IP address assigned by your wifi router.

OSP has three processes that are started automatically in “/etc/rc.local”

```
screen -L -Logfile /var/log/osp.log -dmS
osp_startup ionice -c 1 /opt/release/bin/osp -m
screen -L -Logfile /var/log/ews.log -dmS
ews_startup /opt/release/bin/start-ews
screen -L -Logfile /var/log/ews-php.log -dmS
php_startup /opt/release/bin/start-ews-php
```

You can cat or tail the log files. You can also kill any of the processes you don't need.

For osp, you can monitor CPU usage by entering the command: **ps_osp**.

osp runs realtime threads in CPUs 1-3.

```
root@ospboard:~> ps_osp
CPUID CLS PRI %CPU   LWP COMMAND
    0  TS  19  0.0   558 OSP
    1  FF 130 22.3   566 OSP: Chan 0
    2  FF 130 21.8   567 OSP: Chan 1
    3  FF  41 13.0   607 OSP: AudioCB
```

In the above example, the command **kill 558** will kill all the OSP threads. You can restart it later with the command **nohup ionice -c 1 /opt/release/bin/osp -m > /var/log/osp.log &**

If you wish to disable all the OSP processes, simply create a file named **/root/.no_osp_startup** and reboot. As long as that file exists, OSP will not be run on startup.

Use the command **touch /root/.no_osp_startup** to create the file.

Device Temperature

To check the CPU temperature, enter the command **temp**.

To monitor the temperature every 5 seconds, enter the command `temp -t 5`

6 - Charging the Device

Currently, the devices do not have a way to measure the battery level. They also have some charging quirks, which would be fixed in later hardware revisions.

If a device in Hotspot mode is not creating an "ospboard" SSID, the battery may be too low. If it is in NetworkManager mode and you cannot ping or ssh to it, the battery is likely discharged.

V9 Device Charging

V9 PCDs will charge at up to 600mA when powered on and when powered off, BUT only if the PCD is turned off while the USB port is connected.

If a device is not responding, attach a USB cable and plug the PCD into a computer or charger. Switch the power on. Normally the PCD will charge briefly then start working. If it does not start after 15 minutes, turn the PCD off then on again. It may take 6-8 hours to fully charge.

If you disconnect the USB cable or unplug the charger/laptop, keep the device power on. Once it is moved, wait 5 seconds then you can turn the power off and it will keep charging. If the device was off, you should turn it on, wait 30 seconds then turn it off.

Disclaimer - Device Overheating Issue

While the device is plugged in with a USB cable(s) and operating at the same time, the device will eventually overheat, which can reduce its audio quality and performance. The OSP team is working to resolve this.

For the time being, it is recommended to set up a SSH connection so that you don’t have to use a cable to connect the Processing and Communication Device (PCD) to your computer.

V8 Device Charging

V8 devices charge whenever the USB port labelled "ADB" is connected.

System Check

To see a summary of the PCD’s system status, enter the command:

```
system_check
```

7 - Next Steps

At this point, your PCD should be ready to go.

To take advantage of the PCD's software features over the browser, refer to the "OSP Software - User Guide".

8 - Resources

Below are hyperlinks to OSP-related resources:

- OSP Website - General information of OSP.
- OSP Forum - Community discussion of issues and suggestions for improving OSP.
- "Releases" folder in Google Drive - You will find downloadable .zip and installer files for OSP hardware and software setup.

There are separate hardware guides for doing the sanity checks on the software within the PCD. Below are more hyperlinks:

- OSP Hardware Sanity Check Guide - Node.js Version of EWS - Guide to testing that the Node.js version of the embedded web server (EWS) within the PCD works as intended.

- OSP Hardware Sanity Check Guide - PHP/Laravel Version of EWS - Guide to testing that the PHP/Laravel version of the embedded web server (EWS) within the PCD works as intended.

If you want to simply download OSP software without the device, you may refer to the Software Getting Started Guide, which is a combined version of the following list of separate guides.

- Installation Requirements and Steps (included in the Software Quick Start Guide)
- Software Troubleshooting Guide - Covers steps for possible issues during OSP software installation.
- Software Sanity Check - Audio Input and Output Sources - Guide to checking that your audio input and output sources are connected for OSP software usage.
- Software Sanity Check - Node.js Version of EWS - Guide to testing that the Node.js version of the embedded web server (EWS) works as intended.
- Software Sanity Check - PHP/Laravel Version of EWS - Guide to testing that the PHP/Laravel version of the embedded web server (EWS) works as intended.

9 - Acknowledgments and Bibliography

This work is supported by:

- NIH R33-DC015046: Self-fitting of Amplification: Methodology andCandidacy
- NIH R01-DC015436: A Real-time, Open, Portable, Extensible SpeechLab to University of California, San Diego.
- NSF IIS-1838830: Division of Information & Intelligent Systems, ”AFramework for Optimizing Hearing Aids In Situ Based on PatientFeedback, Auditory Context, and Audiologist Input”
- The Qualcomm Institute

Bibliography

1. Harinath Garudadri, Arthur Boothroyd, Ching-Hua Lee, Swaroop Gadiyaram, Justyn Bell, Dhiman Sengupta, Sean Hamilton, Krishna Chaithanya Vastare, Rajesh Gupta, and Bhaskar D. Rao, "A Realtime, Open-Source Speech-Processing Platform for Research in Hearing Loss Compensation", In Signals, Systems and Computers, 2017 51st Asilomar Conference, IEEE, 2017.
2. James M. Kates, "Principles of digital dynamic-range compression," Trends in Amplification, vol. 9, no.2, pp. 45-76, 2005.

3. Shilpi Banerjee, The compression handbook: An overview of the characteristics and applications of compression amplification, 4th Edition, Starkey Laboratories, 2017.
4. James M. Kates, Digital hearing aids, Plural publishing, 2008.
5. Toon van Waterschoot and Marc Moonen, "Fifty years of acoustic feedback control: State of the art and future challenges," Proceedings of IEEE, vol. 99, no. 2, pp. 288-327, 2011.
6. Johan Hellgren, "Analysis of feedback cancellation in hearing aids with Filtered-x LMS and the direct method of closed loop identification," IEEE Transactions on Speech and Audio Processing, vol. 10, no. 2, pp. 119-131, 2002.
7. Hsiang-Feng Chi, Shawn X. Gao, Sigfrid D. Soli, and Abeer Alwan, "Band-limited feedback cancellation with a modified filtered-X LMS algorithm for hearing aids," Speech Communication, vol. 39, no. 1-2, pp. 147-161, 2003.
8. Donald L. Duttweiler, "Proportionate normalized least-meansquares adaptation in echo cancelers," IEEE Transactions on Speech and Audio Processing, vol. 8, no. 5, pp. 508-518, 2000.
9. Jacob Benesty and Steven L. Gay, âAIJAn improved PNLMS al- ˘ gorithm,âA˘ I in Proceedings of IEEE International Conference on ˙ Acoustic, Speech, and Signal Processing (ICASSP), pp. 1881-1884, 2002.
10. Constantin Paleologu, Jacob Benesty, and Silviu CiochinĂC, "An ˘ improved proportionate NLMS algorithm based on the l0 norm," in

Proceedings of IEEE International Conference on Acoustic, Speech, and Signal Processing (ICASSP), pp. 309-312, 2010.

11. Arslan, Levent, Alan McCree, and Vishu Viswanathan. "New methods for adaptive noise suppression." Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on. Vol. 1. IEEE, 1995.

12. Hirsch, Hans-GÃijnter, and Christoph Ehrlicher. "Noise estimation techniques for robust speech recognition." Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on. Vol. 1. IEEE, 1995.

13. Cohen, Israel, and Baruch Berdugo. "Noise estimation by minima controlled recursive averaging for robust speech enhancement." IEEE signal processing letters 9.1 (2002): 12-15.

14. Lee, Ching-Hua, Bhaskar D. Rao, and Harinath Garudadri. "Sparsity promoting LMS for adaptive feedback cancellation". In Signal Processing Conference (EUSIPCO), 2017 25th European, pp. 226-230. IEEE, 2017.