CHILLER: Assembly & Setup Guide

Ripollés Lab

A detailed guide on how to build and set up a CHILLER.

Contents

This guide contains a number of website hyperlinks that may not function properly if you are previewing the document on GitHub. For the best experience, download this document as a PDF and view it locally.

1 Required Hardware and Equipment

This section covers the hardware needed to set up, assemble, and use a single CHILLER.

1.1 Important Notes

***Read this section thoroughly before proceeding**; it contains critical information about what to do first, which hardware options to consider, and how to understand the rest of the guide.*

IMPORTANT TIMELINE INFORMATION: your first hardware considerations should be:

- **Ensuring access to compatible WiFi**
  Detailed information about the compatibility of different types of wireless internet networks is given near the end of section 1.3, but you should know now that **a compatible wireless internet connection is required** during CHILLER setup and use and that **not all networks are compatible** (in brief, most personal/home networks are compatible, while most institutional/enterprise networks, which usually require a username & user-specific password, are not). When deciding where to set up and use your CHILLER, remember that access to compatible WiFi will be essential.
- **Making arrangements for the 3D-printed CHILLER case**
  Detailed information about 3D-printing options is given at the end of section 1.3 (along with a link to the freely available case-printing file), but you should know now that **getting a case made can take some time** and should not be left until the last minute.

Note on alternative hardware: in the interest of practical and financial flexibility, we have included multiple options for certain hardware items described below. Importantly, however:

- **We cannot guarantee** that all hardware options will work exactly the same way.
- **We do not specifically cover** all hardware options when giving step-by-step instructions for setup, assembly, and use of the CHILLER.

For clarity and simplicity, later sections will refer *only* to the main/first option for each piece of required hardware; specific instructions for every hardware option will not be included.

This does not mean that the various hardware options presented here (or others meeting the same requirements) will not work equivalently - but know that *you may have to do additional research and troubleshooting* along the way, particularly if you use alternative hardware.

Note on abbreviations: In the interest of brevity and clarity, we use several abbreviation conventions throughout this guide.

- When specific guide sections are referenced, we will sometimes use "S" as a shorthand for "section" (e.g. "see S1.3" means "see Section 1.3").
- When referring to certain hardware items, we will often use a condensed naming convention for the sake of brevity (e.g. "HDMI-MiniHDMI cable" means "HDMI-to-Mini-HDMI cable"; "Raspberry Pi Zero WH" is often shortened to "Pi Zero").

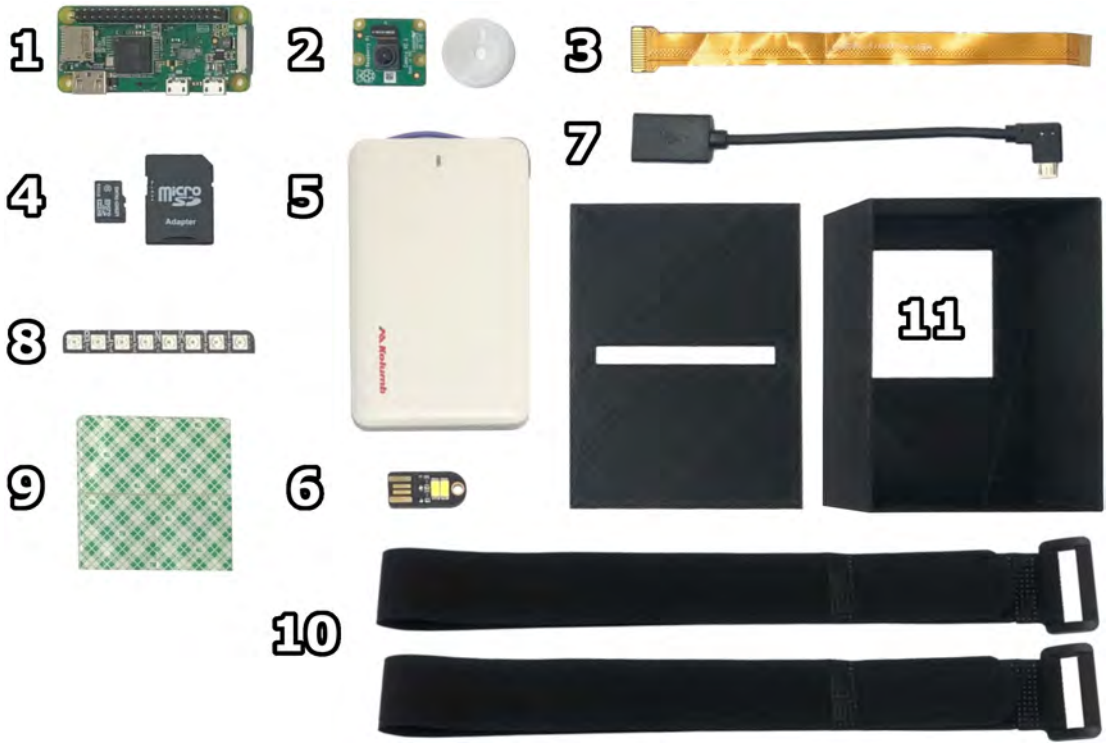
- The names of many hardware items are underlined when first introduced and may be *italicized* when mentioned later on. When italics are used in this way, it is to indicate that we are referring to a specific piece of hardware that was described in detail earlier in the guide, rather than using some more general term (e.g. "*camera cable*" will be used to refer specifically to the Pi Zero Camera Cable adapter (150mm) described in detail in S1.2).

  As many different cables, adapters, and so on are mentioned in this guide, we want you to know when we are referring back to a specific item. When ambiguity is minimal, we usually drop the italics (e.g. "Pi Zero" is never italicized, as it always refers to the Raspberry Pi Zero WH and is unlikely to be mistaken for some more general term).

1.2 Components of the CHILLER

These are required components of the device itself and will be "consumed" during assembly. Each item is covered in detail on the next page.

Figure 1: Components Overview



(1) Raspberry Pi Zero WH **(2)** Camera & Focus Tool **(3)** Camera Cable **(4)** MicroSD Card **(5)** Power Bank **(6)** LED Stick **(7)** USB-MicroUSB Adapter **(8)** Biofeedback LED Strip **(9)** Mounting Pads **(10)** Velcro Straps (shown folded) **(11)** 3D-Printed CHILLER case+lid

1. 1 x Raspberry Pi Zero WH, which includes a built-in 2x20 pin header strip
   Alternative: 1 x Raspberry Pi Zero W & 1 x 2x20 GPIO pin strip dual male header
   We recommend using the Pi Zero WH instead, as it comes with a built-in, professionally soldered *pin strip header* (the Pi Zero W does not). As the *pin strip header* is required for connecting the biofeedback *LED strip* (more on this later), you will need to buy and install one yourself if using a Pi Zero W model. We have not tested this alternative setup extensively and do not include specific instructions for it here.

2. 1 x Adjustable-Focus Raspberry Pi Camera Board (V2-8 Megapixel, Focus Tool included)
   Focus must be adjustable; do not use an auto- or fixed-focus model.

3. 1 x Pi Zero Camera Cable, 150mm
   The *camera cable* adapter will be required to use your *camera board*, as the cables included with most cameras will not be compatible with the Pi Zero; they are usually intended for use the larger camera ports of newer (and more expensive) Pi models.

4. 1 x MicroSD card, minimum memory 8GB, to act as a hard drive
   Kingston, SanDisk, or similar to act as a hard drive. Most MicroSD cards come with an SD adapter, which may be needed for setup (see S1.3 for more information).

5. 1 x Ultra-Slim Power Bank, ~2500mAH capacity, to act as a battery
   Kolumb or equivalent with similar capacity and **equal or smaller dimensions (especially thickness)**. Larger models may not fit inside the case.

6. 1 x Mini-LED Stick for illuminating the skin during recording
   Drock or similar stick-mounted style that plugs into a USB port.

7. 1 x 'Female-to-Male' USB-MicroUSB adapter
   Monoprice or equivalent with a similar length, a 'female' USB end, and a 'male' MicroUSB end that is right-angled in the same way (ensure that the microUSB is not 'upside down'). This will be used first to connect a keyboard and mouse during configuration and then to illuminate the skin with the *mini-LED stick* while the CHILLER is in use.

8. 1 x LED strip for biofeedback (Pimoroni Blinkt!)
   This *biofeedback LED strip* will light up when a chill is detected.

9. 1 x Pack of foam mounting pads (at least 8) to secure the components during assembly.
   Scotch 24-pack or similar pack of ~1 inch square double-sided sticky pads.

10. 2 x Velcro straps, ~18" long (many options available)
    Two straps will be needed to secure the CHILLER on the arm during recording. To fit most arms, the straps need to be at least 18 inches long.

11. 1 x 3D-Printed CHILLER case. The printing file is available for free online, and printing instructions are covered in detail in detail in the next section.

1.3 Equipment for CHILLER setup

*This equipment is required for CHILLER setup but will not be 'consumed' in the process. You need **access to** this equipment, but you do not necessarily need to own it yourself.*

- Access to a WiFi-enabled computer with MATLAB installed (e.g. personal laptop)

  A computer will be needed during set up (covered in S2) to configure CHILLER hardware and install / update software on the *MicroSD card*, and then again during use (covered in S4) to run the CHILLER using MATLAB scripts.

  While the configuration steps covered in S2 may vary depending on your particular operating system, and compatibility for all systems cannot be guaranteed, most personal computers running a recent release of a standard OS (Windows, Mac, Unix) should be able to connect to the *MicroSD card* with one of the following options:

  - ***Option 1**:* Computer has an SD card reader slot, and you have a *MicroSD-SD adapter* for your *MicroSD card*. Many cards come with an adapter included.

    Note that some newer computers even have a dedicated *MicroSD* reader slot, in which case the adapter is not required.
  - ***Option 2**:* Computer has no SD slot, or you have no *MicroSD-SD adapter*, but the computer does have a USB port. In this case, you will need to connect the card to the computer using a USB-MicroSD Adapter (e.g. Vanja).

    While this option should work, we have not tested it and will not cover it further.

- Access to an HDMI-compatible screen & an HDMI-MiniHDMI cable setup

  During configuration, you will need to connect the Pi Zero (which has a MiniHDMI port) to a screen (most modern computer monitors and TVs have an HDMI port). As the Pi Zero uses a MiniHDMI port, you will need an *HDMI-MiniHDMI cable setup*:

  - ***Option 1**:* 1 x HDMI cable (e.g. Twozho) & 1 x MiniHDMI adapter (e.g. CableMatters)
  - ***Option 2**:* 1 x HDMI-MiniHDMI cable (e.g. Rankie)

- Access to an all-in-one keyboard-mouse setup

  While a keyboard and mouse will each be needed to configure the Pi Zero in S2, the Pi Zero only has a single mini-USB input port. The output of both keyboard and mouse can be delivered to the Pi Zero (via the *USB-MicroUSB adapter*) in several ways:

  - ***Option 1**:* 1 x mouse-keyboard all-in-one combination (e.g. Hongaga, Logitech)
  - ***Option 2**:* 1 x USB keyboard & 1 x USB mouse & 1 x USB Hub (e.g. Anker)
  - ***Option 3**:* If you have access to a separate USB keyboard and USB mouse but no USB Hub, it is possible to complete the configuration steps covered in S2 by switching the connection back and forth between the mouse and keyboard (i.e. unplugging one and plugging in the other as needed, repeatedly).

    This option is workable but time consuming; it is not recommended.

- Access to a compatible wireless internet network, i.e. WiFi (non-Enterprise)

  In order to complete the configuration steps covered in S2, you will need to connect the Pi Zero to the Internet over WiFi (a WiFi connection is again required to run the CHILLER as covered in S4, but the network does not have be connected to the broader Internet in that instance). While your Pi Zero should be able to connect to many standard WiFi networks, including those that require a password, Pi Zero models may *not* compatible with the main networks of large institutions that use more stringent security protocols. See RaspberryPi.org for detailed Zero W(H) specifications.

  In general, Pi Zero W(H) models are:

  – **Compatible with personal wireless networks** typically used in homes, small businesses, and certain schools. Such networks generally *do not require a username* to connect, but may require a password (e.g. WPA-Personal, WPA2-Personal, unsecured)

  – **Not compatible with enterprise wireless networks** typically used in large universities and corporations. Such networks generally *require both a username and password* to connect. (e.g. WPA-Enterprise, WPA2-Enterprise)

- Access to 3D-printing services for printing the *CHILLER case*

  The case (including the lid) can be 3D printed using the ChillerCase.amf file found in our CHILLER GitHub repository (the file is in the "ChillerCase" folder).

  If you have direct access to a 3D printer, you can print the case yourself. You may also have access through a local institution (e.g. your school, company, or public library). If not, you may need to place an order with a 3D-printing company. Note that if you plan to build more than a few CHILLERs (~6-8 or more), it may be more cost effective to buy an inexpensive 3D printing kit ($100-$200) and print the cases yourself, as 3D-printing companies can charge a large premium for their services and shipping costs: if you own a 3D printer, the cost ratio between ordering a printed case versus printing it yourself can be as large as 20:1 (e.g. ~$30 charge vs ~$1.50 worth of printing materials).

  However you decide to get your case(s) printed, the process can take time and should not be left to the last minute.

  Here is a more detailed overview of each option:

  – ***Option 1***: If you own a 3D printer, use it to print the case from the .amf file

    Warning: always follow the safety guidelines and printing instructions for your specific 3D printer. These will vary from printer to printer.

    There are safety concerns when using any 3D printer; printer components can become dangerously hot during use, and the printer can be damaged if used improperly.

    Methods of loading/using the .amf file to print the case will likewise depend on your specific printer. Some printers may require you to convert the .amf file to another format before it can be used; some printers require printing files to be loaded using a MicroSD card, while others may be able to receive files over WiFi. Refer to your printer manual for specific instructions.

- ***Option 2**:* Find out if you have access to a 3D printer through a local institution

  Your school, employer, public library, or community center may have a 3D printer that you can use, directly (by printing the case yourself, as described in *Option 1*) or indirectly (e.g. by submitting a request). If not, your school or employer may still be able to help you access 3D-printing services some other way.

- ***Option 3**:* If you not have access to a 3D printer, you can use a 3D-printing service

  There are many online services that can print the case for you from the .amf file and then mail you the finished product (e.g. shapeways, sculpteo). Research the available options to figure out which service is right for you and reach out to them directly for specific instructions. **Be aware that fabrication and shipping can take a while.**

- ***Option 4**:* Buy a 3D printing kit (& printing filament) to use as described in *Option 1*

  We use a Kingroon DIY Aluminum Printer to print our own CHILLER cases, which cost ~$150 at the time of purchase. While setting up and using such a kit takes time and effort, it is a relatively inexpensive option that gets the job done. Similar models should presumably work just as well, but make sure to research the options thoroughly and choose a model that will work for printing the cases.

  Depending on your kit, you will likely need to buy **printing filament** as well, as sufficient filament material may not be included with the kit. With our Kingroon kit, we use basic Geeetech spools, one of which is sufficient for up to ~16 prints (it can be fewer than 16 in practice, as prints are occasionally unsuccessful and a certain amount of filament is used during routing maintenance / testing). When purchasing filament, make sure that it is compatible with your particular 3D printing kit.

  The relative benefit of buying a kit versus ordering prints will depend on your individual circumstances, and the decision is ultimately up to you. There are many factors to consider: the prices of printing kits, printing material, and printing services will vary over time, and much depends on the number of cases you intend to print, your level of confidence with technology / troubleshooting, and the amount of time you are willing to put into setting up and using a printing kit. You should also consider how much use you foresee in owning a printing going forward, as you will be able to print things other than CHILLER cases. yourself are all important factors. the prices charged by 3D-printing services, the cost of the printer, the price charged by the (we use and print the cases yourself (and a spool of printing filament, if not included with your material, if not included

2 Software Installation, Configuration, and Hardware Testing

Detailed information for each piece of hardware referenced below can be found in Section 1.

As noted at the beginning of S1: while there are multiple options for certain hardware, the instructions going forward will refer *only* to the main/first option for each hardware item; specific/alternative instructions (if any) for alternative hardware options are not included.

2.1 Installing Raspbian

Raspbian is a free Linux-based operating system for Raspberry Pi. We will install the Raspbian OS on the *MicroSD card* so that the card can act as a hard drive for the Pi Zero.

1. Connect the *MicroSD card* to your computer.

   First insert the *MicroSD card* into the *MicroSD-SD adapter*, then insert the adapter into the SD-reader slot on your computer.

   As mentioned in S1.3, the adapter is not required if your computer has dedicated slot for MicroSD cards. It is also possible to connect the *MicroSD card* to the computer using a *USB-MicroSD Adapter*, but we have not tested this option ourselves.

Figure 2:
Mounting the MicroSD.

2. Install Raspberry Pi Imager on your computer and use it to install Raspbian (the Raspberry Pi OS) on the *MicroSD card* (for more instructions, see: www.raspberrypi.org/software/)

3. Once Raspbian has been installed, eject the *MicroSD card* and remove it from the adapter.

2.2 Preparing the Hardware for Configuration

Note that the final, full assembly of the CHILLER will take place later *(in Section 3).*
At this point, we will simply connect the components for configuration and testing; there is no need to touch the 3D printed case or stick anything together with adhesive. While the camera and SD card configuration will remain the same in S3, you can fully disconnect the components at the end of S2 before proceeding to S3 if desired.

1. Insert the *MicroSD card* into the Pi Zero slot labeled "MICRO SD CARD".

2. Connect the Pi Zero to the screen using your *HDMI-MiniHDMI cable setup*.

   First insert the HDMI end of the *cable setup* into your screen's HDMI port, then insert the

other end of the *cable setup* into the MiniHDMI slot on the Pi Zero. This slot may not be labeled, but it is the Pi Zero's only slot with the appropriate shape.

3. Connect the *keyboard-mouse setup* to the Pi Zero via the *USB-MicroUSB adapter*.

   First insert the *keyboard-mouse setup*'s USB output into to the *USB-MicroUSB adapter*'s USB slot, then insert the adapter's MicroUSB end into the Pi Zero slot labeled "USB".

   Important: The Pi Zero only has one MicroUSB port (labeled "USB").
   The second slot with a similar shape (labeled "PWR IN") is *only* for powering the device. This is why you need a *keyboard-mouse setup* with a single USB output (see S1.3).

4. Connect the camera (**see figure below** for illustrations).

   - **Loosen the "port locks"**:

     Looking at the back of the *camera board* (the side without the lens), you will see that the there is a white plastic port with a black plastic strip on its outside edge. Loosen the black strip by gently pushing on its two ends until the strip slips slightly outwards; this "opens" the port, which will allow us to insert the *camera cable*.

     Repeat this procedure with the Pi's camera port (labeled "TV") to unlock it, too.

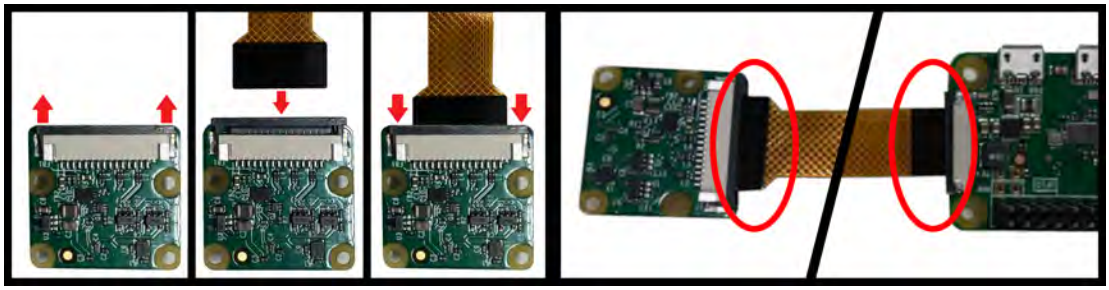
   - **Connect the *camera cable***:

     First insert the wide end of *camera cable* into the *camera board*'s unlocked port, **making sure that it is not "upside down"** (the cable's dark side should face *away* from the camera board; see figure below). Gently push the cable inwards.

     With the cable snugly inserted, lock the port by pushing its black plastic strip back inward. The camera and cable should now be securely attached.

     Repeat this procedure to connect the cable to the Pi Zero's TV slot.
     Insert the thin end of the cable into the Pi Zero's unlocked camera port, again making sure that the cable is oriented correctly (see figure below). With the cable fully inserted, re-lock the Pi's TV port. The cable and Pi should now be securely attached.

Figure 3: Connecting the Camera Module to the Pi Zero with the Camera Cable.



Left: Procedure for connecting the camera & cable; repeat this process with the Pi & cable.
Right: When the camera board, cable, and Pi Zero are correctly connected, as shown, the camera faces *away* from you and the cable's black strips (circled) face *towards* you.

5. Connect the *biofeedback LED strip* to the Pi Zero's *pin strip header*.

   The two strips of 20 pins should fit into the back of the *LED strip*; carefully position the strip over the pins, making sure that the rounded corners of the strip are facing towards the edge of the Pi Zero as shown below; push gently downwards to connect.

Figure 4:
Positioning the LED strip.

The strip's rounded corners should align with the Pi Zero's.

2.3 Configuring Raspbian and Updating via WiFi

1. Connect the Pi Zero to the *power bank* to start up the Raspbian operating system.

Figure 5:
Raspian OS booting up.

This is the first thing you will see on the screen if the OS starts up correctly.

   You should see the OS begin booting up on the screen. If you do not, make sure that the *MicroSD card*, *HDMI-MiniHDMI cable setup*, and *power bank* are all connected correctly (as shown below), and that each component is in good working order.

Figure 6: Full hardware testing setup.

With all the hardware connected for testing, your setup should look something like this.

Clockwise from upper left, the components connected to the Pi Zero are: *camera board* connected via *camera cable*; *power bank*; *keyboard-mouse* wireless USB connected via *USB-MicroUSB adapter*; *HDMI-MiniHDMI cable setup*, which connects to the screen; *MicroSD card*; *biofeedback LED strip*.

2. Follow the on-screen instructions to configure the OS using your *keyboard-mouse setup*.
   - The default user will be "pi"; you can change it to something else if desired.
   - Select a password of your choosing (one that you will remember!)
3. Connect to a WiFi network as you would on any personal computer.

   Click the WiFi icon (on the upper right of the screen), select a compatible network, and enter the network's password (if required).

   If you expect to see a given network here but it is not listed, the security protocols of the network may be incompatible with the Pi Zero; see S1.3 for more information.
4. Once connected to WiFi, check for updates to the operating system, install them, and reboot as needed.

2.4 Configuring Camera and Remote Connectivity

We will now enable the camera and set up remote access to the Pi Zero via VNC.
See the end of this subsection for more information about remote access.

1. Open a terminal window (look for the terminal icon on the top left).
2. In the terminal window, execute the following command*:

   $ sudo raspi-config

   *To run a command in terminal, you simply need to type it in and press Enter on your keyboard. Here, you would type in "sudo raspi-config" and press Enter. You should not type in the "$ ", which is just notation to indicate that what *follows* is a terminal command.

   If you have not used terminal before or need a refresher, this guide may help.

   After running the command, the raspi-config settings window will appear.

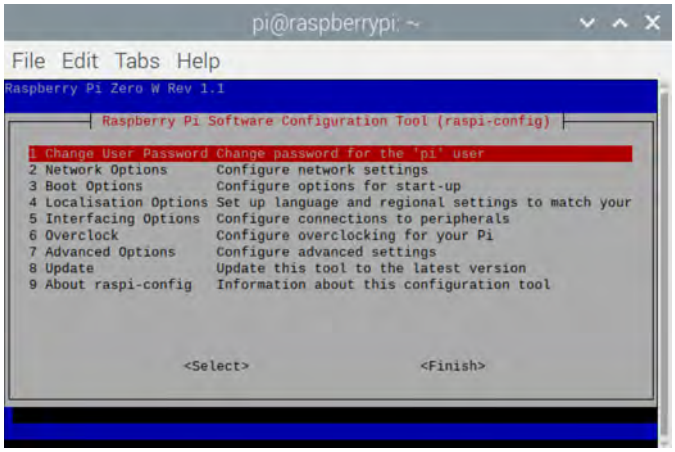


Figure 7:
Raspi-config settings window.

The "Interfacing Options" sub-menu (5th option) will be our main focus here.

You will navigate this window with your keyboard using the Arrow keys (to move the selection, highlighted in red) and the Enter key (the confirm the highlighted selection).

3. Highlight "Interfacing Options" using the Arrow keys and press Enter to select it.
   This will bring up the Interfacing Options sub-menu.

4. Enable the camera connection:
   In the Interfacing Options sub-menu, select the "Camera" option; a prompt will appear asking you if you want to enable the camera; select "<Yes>".

5. Enable VNC:
   Navigate back to the Interfacing Options sub-menu, select "VNC", and enable it the same way you enabled the camera. More information on VNC is given below.*

6. Enable SSH:
   Return once more to the Interfacing Options sub-menu and enable SSH the same way you enabled the camera and VNC.

7. With the Camera, VNC, and SSH now enabled, highlight "<Finish>" on the main menu (using the Right Arrow key) and press Enter.

8. Reboot your Raspberry Pi.
   Click on the Raspberry icon (upper left of the screen) to bring up a drop-down menu, then click on the "Logout" option at the bottom of the menu. In the "Shutdown options" box that appears, select "Reboot".

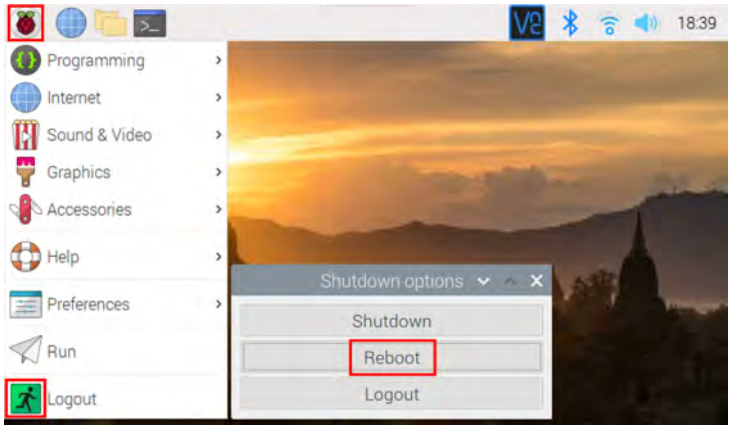


Figure 8:
Rebooting Raspbian.

**Note on remote access, a.k.a. "Headless Mode":*

The SSH and VNC options were activated so that you can access your Pi Zero remotely (i.e. over WiFI, from another computer) in the future if you want to; this may be easier than physically connecting the Pi Zero to a screen, keyboard, and mouse every time you want to use Raspbian.

VNC ("virtual network computing") is a system that allows for remote control of/access to a computer. We use VNC Connect by RealVNC to implement remote access, but there are many VNC options out there, and setting this up is ultimately optional.

2.5 Testing and Focusing the Camera

We will now confirm that the camera works.

1. After restarting your Pi Zero, open up a new terminal window.

2. In the terminal window, write and execute the following command:

   $ raspivid -o videoname.h264 -t 10000

   In the above command:

   - "videoname" is the name of the video file that will be saved.
     The file will be saved, by default, to your home folder.
     You can substitute another filename if you wish.
   - If you do not include "-o videoname.h264", no file will be saved.
     You will just see see the video feed on the screen.
   - "10000" is the number of milliseconds of camera feed that will be recorded.
     This value will record 10 seconds of video; feel free to change it.

3. Focus the camera.
   It is very important that you use the focusing tool (a white plastic ring-shaped device, or similar) that came with the camera to adjust the lens focus.

   - Attach the *focusing tool* to the lens by carefully placing it on the lens and gently pushing downwards.
   - Using the tool, rotate the lens counterclockwise as much as you can, making sure that the lens does not fully unscrew and fall out. If it does fall out, carefully place it back and use the tool to screw the lens a tiny bit in the clockwise direction.

2.6 Configuring the LED Strip

The instructions here assume that you are using Pimoroni's Blinkt as your biofeedback LED strip.

1. Install Blinkt by following the instructions here: https://github.com/pimoroni/blinkt

2. In a terminal window, open a python console by executing the simple command:

   $ python

   This will turn your terminal window into a python console, meaning that you can now type in an execute commands written in the python programming language.

   For the code given below, note that ">>>" is the python equivalent of terminal's "$"; do not actually type it in.

3. Use python to check whether the LED strip works.

   Execute the following lines of code, line by line, to light up the LEDs:

>>> import blinkt
>>> blinkt.set_all(255, 255, 255, brightness=None)
>>> blinkt.show()

In the "blinkt.set_all(...)" line of code above:

- The first three arguments ("255") correspond to RGB values: the amount of red, green, and blue light that the LEDs will emit. These can each be any integer from 0 to 255.
- The final argument ("brightness=None") sets the LEDs' brightness. This can be any number from 0 (off) to 1 (max). When brightness=None is passed, the most recent brightness value (or the default of 0.2) will be used.

Try running the "blinkt.set_all(...)" command again with different RBG and brightness values. To see the effect, you will need to run "blinkt.show()" again.

For example, you can run the following commands to turn off all the lights at once:

>>> blinkt.set_all(0, 0, 0, 0)
>>> blinkt.show()

Feel free to experiment with different parameters for the color and brightness (and even try out other commands to, for example, control each LED individually), but make sure to turn off all the lights before moving on from this section.

To close the python console and return to the terminal window, simply run the command:

>>> exit()

2.7 Getting the IP address of your Pi Zero:

There are multiple ways of getting the IP address of the Pi Zero in your network.
Here, we will do so by looking for the information in your router's system.

An easy way to do this is by executing the following command in terminal:

$ hostname -I

This will return a variety of networking information, including the IP address of your Pi Zero. The IP address will look something like "192.168.1.100" (with different numbers).

MAKE SURE TO WRITE THE IP ADDRESS DOWN SOMEWHERE, as you will need it later.

Note that this IP address corresponds to your current wireless network; it will be different if you connect the Pi to a different network. If you end up using the device on multiple networks, you will need to get the IP address for each network individually.

2.8 Installing MATLAB's Raspberry Pi Hardware Add-on

Here we will be installing the "MATLAB Support Package for Raspberry Pi Hardware".

IMPORTANT: READ EVERY STEP OF THIS SECTION WITH PARTICULAR CARE.

Choosing the wrong option or failing to read the notes here can cost you quite a lot of time.

Note: For the installation to work properly, both your computer running MATLAB and your Pi Zero should be connected to **the same Wi-Fi network (with working internet access)**.

Note: You need to input your Pi's **IP address** that you collected previously (section 2.7), along with your Pi's **username and password** (section 2.3; the default username is "pi", and the default password is "raspberry").

Note: **You may run into trouble** if you install the MATLAB add-on using a computer with a particular operating system (e.g. Windows) and then later try to use the CHILLER using computer with a different OS (e.g. Linux). We have tested **same-OS** add-on installation + CHILLER usage across various operating systems, but we cannot guarantee that the CHILLER will work properly with **cross-OS** installation + usage.

1. Open MATLAB on your computer.
2. Click on the "APPS" tab (on the upper left, next to the "HOME" and "PLOTS" tabs).
3. Click the "Get More Apps" button (on the upper left, just under the tabs bar).
4. In the "Add-On Explorer" window that pops up, locate the add-on "MATLAB Support Package for Raspberry Pi Hardware". To locate the correct add-on, scroll down until "Filter by Type" appears in the navigation column on the window's left edge, and check the box next to "hardware support packages". Look for the add-on in the results that appear; it should be one of the first options. **Make sure that the name of the add-on is exactly "MATLAB Support Package for Raspberry Pi Hardware".**

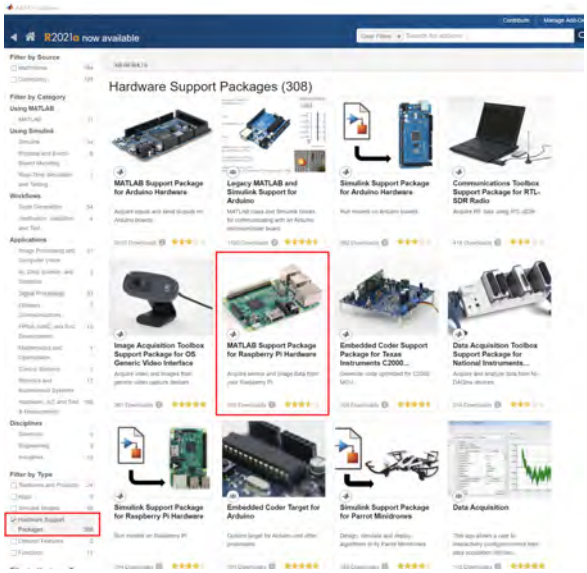


Figure 9:
Locating the "MATLAB Support Package for Raspberry Pi Hardware" Add-On in the "Add-On Explorer" window.

5. Choose the version for the Pi Zero and then select "modify software" to install.

6. **IMPORTANT: When prompted to "Select Linux Operating System" *make sure* to select the "Customize the existing operating system" option**; the other option ("Setup hardware with MathWorks Raspbian image") will ***ERASE THE CONTENTS OF THE MICRO SD CARD***, and you will have to start this entire section again from scratch.

7. When prompted to "Review Required Packages and Libraries", it is safest to add any missing software to ensure compatibility.

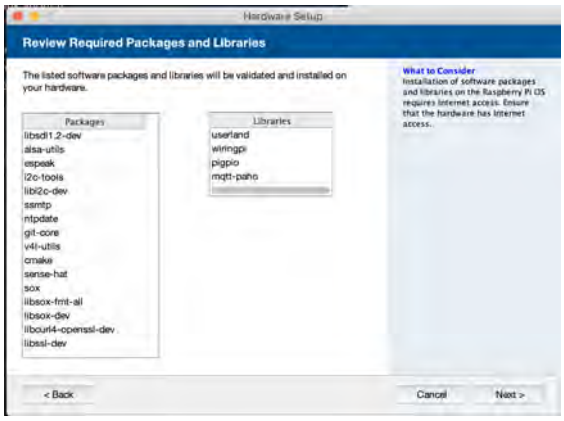


Figure 10:
Add any missing libraries to ensure compatibility with your MATLAB installation.

8. When prompted to "Configure Peripheral Modules", set all of the modules to "Enabled". This ensures that all the hardware components will be able to communicate with MATLAB.

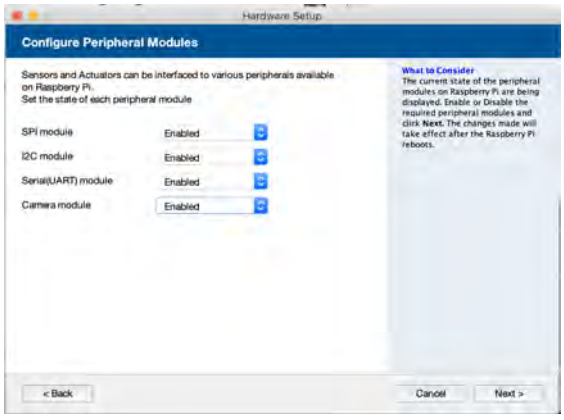


Figure 11:
Enable all peripherals when prompted.

9. Continue through the steps of the installation until the download and installation begins. **This can take quite a while;** now would be a good time to take a break. If everything installs correctly, you can move onto the next section and begin assembling your CHILLER.

3 Assembling the CHILLER

With all hardware is tested and software installed, it is now time to build the CHILLER.

You may still have certain components connected after performing the setup in section 2; this is fine, but this section will assume that everything is disconnected at the start.

1. Insert the *MicroSD card* into the Pi Zero slot labeled "MICRO SD CARD".

2. Connect the camera the same way described and illustrated in section 2.2.4.

3. Attach the Pi Zero, *power bank*, and CHILLER lid using sticky foam pads:

   - Stick a foam pad on the back of the Pi Zero, centered behind the *pin strip* nubs.

Figure 12:
Pi Zero with foam pad attached to its back.

   - Make sure the *biofeedback LED strip* is disconnected from *pin strip header*; we will connect it again once the lid is in place (with the pins sticking through the lid).
   - Position the Pi Zero and lid so that the *pin strip header* fits through the lid's slot. On the other side of this slot, roughly centered, stick a foam pad to the lid (position shown below). Remove the backing from this pad, stick another pad on top of it, and then do this once more, such that a stack of 3 foam pads - all stuck together - is attached to the lid. **Leave the backing on the outermost pads for now.**

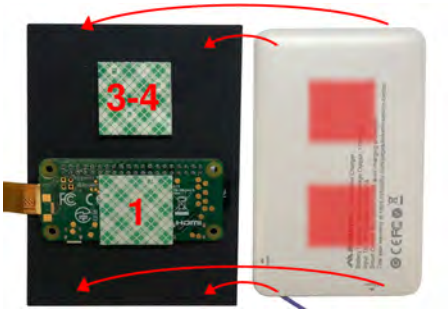


Figure 13:
Pi Zero, lid, pads, & power bank.

Pi Zero pad and pads stacked on lid (shown with backing of outermost pads still in place) will stick to the power bank in the approximate positions marked by red squares.

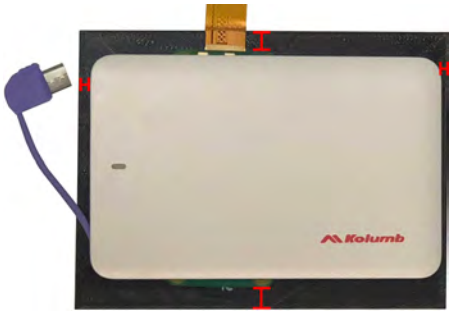
   - The outer pads will ultimately stick to the back of the power bank, but it is important to get the position and height right first. An additional pad may be needed to make the stack fit nicely with the battery.

     To determine if another pad is needed, lift the pi and lid up in one hand (so that the *pin strip* goes fully through the hole) and lower the power bank onto the stacked pads with your other hand. If the height is correct, the power bank should rest neatly on the both the stacked pads and the single pad on the back of the Pi Zero.

Figure 14: Short pad stack.

This power bank was attached with a stack of 3 pads;
it would have been better to use 4 pads here
(note the height difference between the two sides).

- Once you have added a 4th pad (if needed), peel the outer backing off the pads and prepare to stick the power bank to them; the sticky pads should approximately line up with the red squares shown in the figure above when you flip over, position, and attach the power bank. When attaching the power bank, it is important to ensure that it is positioned properly (see figure below) by checking that the power bank:
  - Is **oriented properly**, with the correct side facing up, so that the power cable can reach the Pi Zero's "PWR IN" port.
  - Fits **within the borders** of the lid in all four directions – otherwise, it will not fit into the case when the lid is lowered.
  - Is **centered horizontally** over the Pi Zero, so that neither the *camera cable* nor the *MicroSD* card will be crushed when the lid is lowered onto the case.

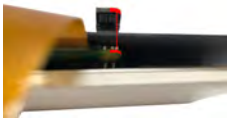


  Gently position the power bank on the pads to achieve the correct fit; if adjustments are needed, pull the power bank off the pads and try again (you can even tear the pads off entirely and start again with fresh pads, if needed).

  Once the correct position has been attained, firmly push the power bank together with the pi and with the lid to secure them in place. Be sure not to crush the pins protruding through the lid; it may help to connect the *LED strip* first (see next step).

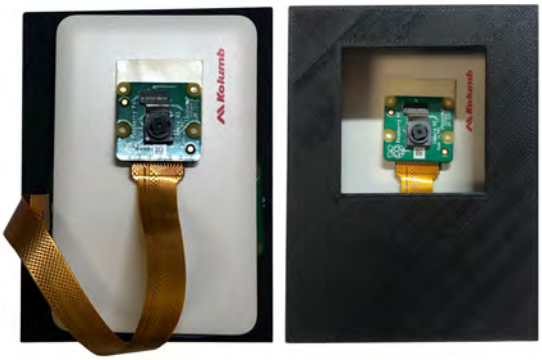
4. Connect the *biofeedback LED strip* to the *pin strip header*.

   The two strips of 20 pins should fit into the back of the *LED strip*; carefully position the strip over the pins, making sure that the rounded corners of the strip are facing towards the edge of the Pi (see figure below), then gently push downwards to connect.

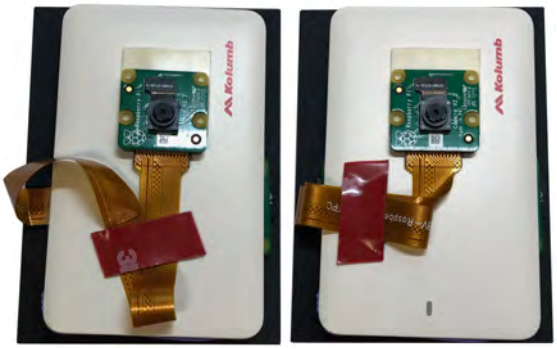


5. Attach the *camera board* and *camera cable* to the front of the *power bank*.

   - First apply a foam pad to the back of the *camera board*.
   - Remove the backing from the pad, and attach the camera to the power bank such that, When the case and lid are joined, the camera is positioned approximately at the center of the hole of the case; this ensures that the camera will have a clear view of the skin. Use the case to check the position, as shown below, then set it aside again for now).



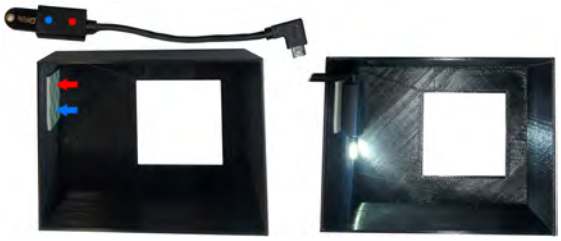
   - To keep the cable out of the way of the camera, secure it to the front of the power bank in one of the positions shown below (the first position may be easier).
     Note that, while we used red tape, you can easily cut a sticky foam pad in half and use it in place of tape (but leave the backing on the outer surface of the pad).



6. Secure the *USB-MicroUSB adapter* to the inside of the CHILLER case.

   - First insert the *Mini-LED Stick* into *USB-MicroUSB adapter*.

- Apply a foam pad to the interior of the case, remove the backing, and attach the USB end of the *USB-MicroUSB adapter* to it such that the LED will shine inwards (shown illuminated below), and is centered and as far down as possible. You may have to push the wire into the case's corner to achieve this position.



Note how the LED shines INTO the case.

7. All that remains is to connect the *USB-MicroUSB adapter* and the *power bank* to the Pi Zero. Make sure to do so with the camera in the correct orientation, so that the lid lowers nicely onto the case without any wires sticking out. When you connect the power bank, the LED should light up. You can now close the lid.

When not using the CHILLER, be sure to disconnect the power bank to conserve battery; if the battery runs low, charge up the power bank before the next use. If desired, tighten the Velcro straps around the CHILLER to keep the lid secured to the case.

Congratulations! You have fully configured and assembled a CHILLER!

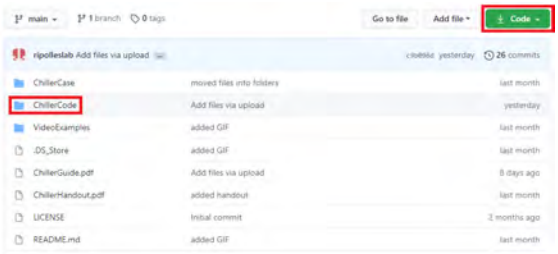
4 Running the CHILLER

***You will need a computer with MATLAB installed and the CHILLER connected to the same WiFi network**; it does not matter if the network is local only (not connected to the broader Internet). Once again, you will need to know the IP address of the CHILLER in your network.*

1. Download the CHILLER GitHub repository.

   All you *really* need is the ChillerCode folder, which contains the MATLAB scripts you will need to collect data, but it is simplest to just download the entire repository by clicking the green "Code" button.

2. Once you have the code downloaded, make sure that the folder containing them has been added to the MATLAB path.



- To add the ChillerCode folder to the path, first click on SetPath in MATLAB's HOME Tab (it is in the toolbar section labeled ENVIRONMENT).
- In the SetPath window, click the Add Folder button, then navigate to the ChillerCode folder. Select it, then hit Enter.
- Click the Save button before closing the SetPath window.

3. Open the CHILLER.m script in MATLAB.

4. Replace the IP address, username, and password of your Pi Zero by editing the script.

   There is a line of code that looks like this: *rpi = raspi('192.168.0.9','pi','raspberry');*

   Replace the three arguments (inside the parentheses) with the the IP address, username, and password of your own Pi Zero.

   The updated code might look something like this: *rpi = raspi('10.23.10.186','pi','chiller');*

   You will find more documentation about how to run and use the code in the comments of the script itself.

5. Save the edited script, but don't run it yet.

   It is okay to run the script if you want to make sure that it works, but you should attach the CHILLER to your arm first before running the script if you intend to collect data.

6. Position the CHILLER on your non-dominant arm and secure it in place on the forearm using the Velcro straps as shown.

7. Run the modified ChillerMain.m script in MATLAB.

8. The *LED strip* will at first shine red; **remain as still as possible until the LED strip lights are no longer red.**
   During this time, a baseline is being established by the camera (against which changes will be compared to detect the presence and intensity of goosebumps).

   The camera feed and biofeedback will now begin to run:

- You will be able to see what the camera is recording in real time, along with a plot showing the intensity of goosebumps whenever they are detected.
- Whenever you experience goosebumps, the *LED strip* will shine green.

If the image is blurry, it means that you need to readjust the camera lens using the *focusing tool*. Move it counter-clockwise until you can see the patch of skin with minimal blurriness.

Congratulations! You have successfully set up and can now use your very own CHILLER.