Dynaboard Prototype

Basic Assembly and Interaction Details

**Parts**

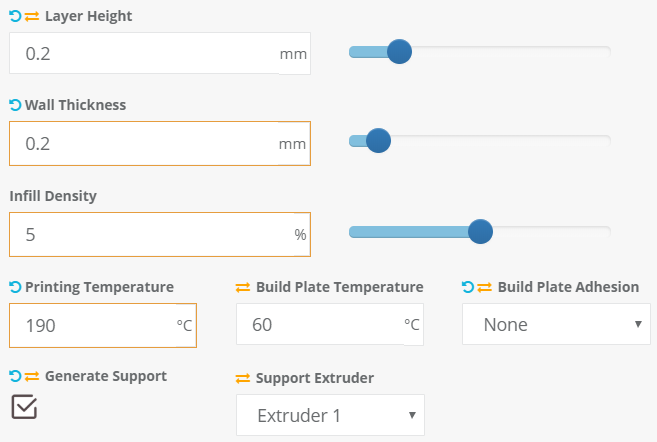
The Dynaboard key prototype involves a Raspberry Pi Zero (W works as well) acting as the control module that communicates with the interaction assembly that is comprised of three components:

|  |  |  |
| --- | --- | --- |
| **The Switch**  The switch is the part of the key that actuates when a downward force is applied, closing a PGIO circuit that indicates to the Pi that the configured character should be types.  For this prototype model, the switch being used is [this Cherry MX Black knock-off](https://www.amazon.com/ZugGear-67g-Koala-Mechanical-Keyboards/dp/B07ZFDP5RV/ref=redir_mobile_desktop/138-8490378-2912220?_encoding=UTF8&psc=1&ref_=sspa_mw_detail_1). Important details of this switch are:   * 67 g actuation force (super high) * PCB mounted (plastic pins on bottom for support) | **The Cap**  The cap is the part of the key that serves as an encasement for the OLED module and holds it to the stem of the switch. This part of the assembly will ideally take most of the “punch” of a key press.  For this prototype model, the cap has been custom designed and optimized for 3D printing with PLA, but is missing many features that would be required for a suitable production cap.        Full-size images in printing instructions section | **The OLED Screen**  The display shows what character has been configured to be typed when the switch is actuated.  This OLED module, though not explicitly sourced from the supplier at the linked domain, is described by [this datasheet](https://www.buydisplay.com/download/manual/ER-OLEDM0.49-1_Datasheet.pdf).  Important details:   * Designed to communicate on I2C bus * SSD1306 display protocol * “Technically” 128x64 display size, but actual display size is 64x32. The actual top-left corner begins at (64, 32).     Full-size image at end of document |

**Cap Printing Guide**

In our repository, under /hardware, lies the .stl file that is used to print the keyboard caps.

Using the 3dprinteros service available on campus, we can print these out in the student center. Based on trial and error, the optimal print (so far) has been achieved with the following slicing parameters:



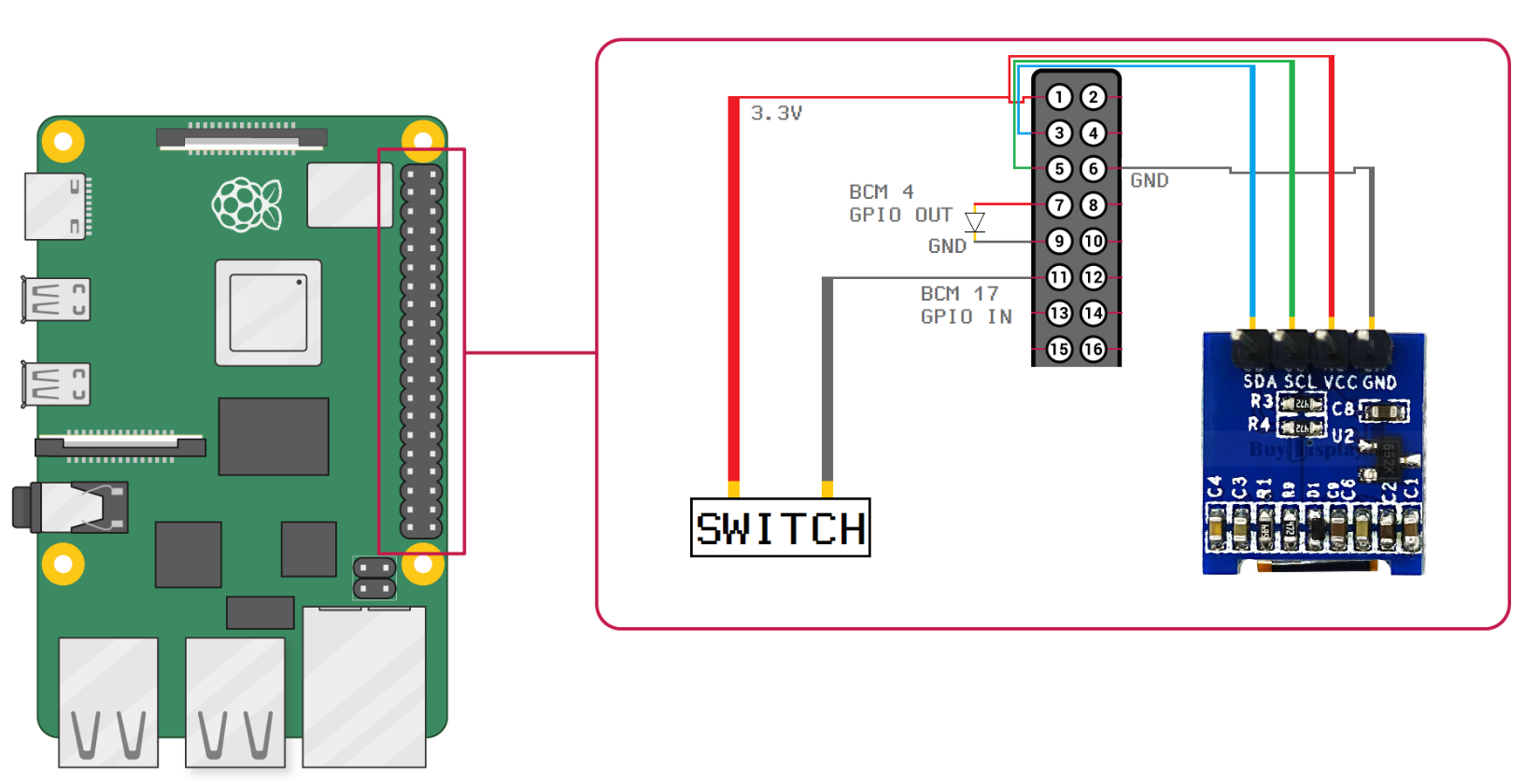
These are the results:

|  |  |  |
| --- | --- | --- |
|  |  |  |
| View of the printed cap from the bottom. The sleeve for the Cherry stem is shown here. | View of the printed cap from the top, before removal of printing artefacts.  I typically use a house key to break through the mesh and pry the supports out from the bottom. **Be careful**, as it is really easy to break off the stem and the walls on the bottom if you do not do this on a flat surface. | View of the printed cap from the top after build artefacts have been removed. You can see the depressions that allow the OLED module board to rest in the slot comfortably without being held up by its components. |

**Parts and Assembly**

*<TODO Solidworks Composer* *Manuals>*

**Wiring Guide**

****

This is consistent between all 40-pin Raspberry Pi’s including the Zero and Zero W.

**OLED Display Setup:**

The I2C module is disabled by default. The easiest way to ensure it can be communicated with is via the following procedure:

Logged into the terminal, run:

**$ sudo raspi-config**

Using arrow keys, navigate to option 5, “Interfacing Options” and press Enter. Navigate to option P5, “I2C” and press right so that the cursor goes to “<Select>” and press Enter. Select <Yes> and press Enter. Exit by selecting “<Ok>” then “<Finish>”.

Clone this repo: <https://github.com/adafruit/Adafruit_SSD1306>

Then:

**$ cd Adafruit\_SSD1306**

**$ sudo python setup.py install**

Then clone our repository: <https://github.com/zefaxet/dynaboard>

To test, navigate to /onboard and run:

**$ python screenwriter.py**

And type out some words to be displayed on the OLED module you have wired up.

**OLED Module Dimensioning:**

