

README

Disclaimer:

I am just a guy who builds stuff. Most of what I know is self taught. I strive to design for functionality and safety but I make mistakes as humans do. Having said that, do your due diligence and check my work before building this project. This is also probably not a beginner project due to the soldering difficulty and note that it's fairly expensive. Check the bill of materials and your sanity before committing. Keep in mind this is also just a version one prototype. There are some things that could have been done better.

Design Intent:

The purpose of this project was to develop a Raspberry Pi Development Platform that can be used to program and prototype Raspberry Pi projects. It started out with the idea of building a cyberdeck in my own style which just seemed like an interesting project to pursue. As I started designing it though, it evolved into more of a cyberstation due to its non-traditional size and modular features. The result is a design with the following features:

- 3D Printed Enclosures and Buttons
- Two 9 inch touchscreen displays
- Raspberry Pi 4 or Raspberry Pi 5 compatibility
- Internal USB hub
- External I2C port
- External 40 pin GPIO port
- External USB port
- 4 Programmable buttons tied to GPIO
- 1 Programmable Rotary Encoder Knob
- 1 Programmable Linear Potentiometer Slider converted to I2C
- Slide Power Switch activated N channel mosfet circuit for power up to 6A
- Dipswitch disconnect for buttons and encoder
- Quick eject handles for the Pi
- Micro SD card access

3D Printed Enclosures and Buttons

Part of the goal was to make this easy to replicate and assemble so I designed all enclosures and buttons to be FDM 3D printed. The enclosures all use a snap-fit mechanism. Screws are only used in places where it's absolutely necessary. So it is quite easy to put together and take apart.

Displays

I wanted two displays that could also be rotated into either portrait or landscape mode. I like to code in portrait mode and often use two displays to look at pinouts or references while writing code for a project. These monitors are also touchscreen because why not? I use a mouse but if you want to use it, it's there.

Raspberry Pi Compatibility

The alignment of the jacks being used and GPIO pinout are the same (mostly) so you can use either a 4 or 5. There is one caveat, however, there is a separate Eject Two printed part for each Pi model because they changed the USB connector position between versions.

Internal USB Hub

Raspberry Pi's are notorious for not playing nice with high power usb peripherals. They just can't supply enough power to handle things like backlit keyboards or usb powered displays. The hub supplies those peripherals with enough power (kind of) and sends the data along to the Pi. The hub I chose was the smallest one I could find with the highest current rating. But even that is borderline for the displays being used so I designed a custom usb power injector to be on the safe side. More on that later.

External I2C

Part of developing a project usually involves sensors. I like adafruit's breakout boards for sensors because they often utilize I2C which lets you connect a bunch of them to the same datalines. Adding this port enables you to connect whatever sensors you want to the station easily. If the sensor is analog just use an adafruit ADC to I2C. Only one port is needed since I2C sensors can be daisy chained together.

External 40 pin GPIO

This port allows you to add any HAT you might normally stack onto a Pi. Or using Adafruit's cyberdeck breakout you can add a third display for station status readout or terminal. All pins can be made available by disconnecting the programmable buttons and rotary encoder via the dip switch on PCB1.

External USB

For all things USB that you might want to connect. WiFi antenna, gps module, radio stuff, usb memory stick, whatever.

Programmable Inputs

The buttons, rotary knob, and slider are there for two reasons. You can program them to control PI functions like the slider knob for volume control, rotary for scrolling, and buttons for macros. Or use them to simulate events in a program you might be developing for a different project. Like using the slider to simulate water level or something.

Power Circuit

This station requires more power than a slide switch can handle so instead the little slide switch turns on the gate of a mosfet which is actually what switches on power to everything in the deck.

Quick Eject

The whole point of a development platform is with the idea that you're programming the Pi to go into a final project so I wanted to make it easy to swap Pi's. The ejection handles are tied to the USB and HDMI cables going into the Pi. All you have to do is pull and the Pi is free. There is also a micro sd card cover so you can easily swap cards as well.

Required tools and equipment:

- Soldering Iron (and hella skills)
- 3D Printer
- Phillips Screwdriver
- Metric Allen Wrench Set

Tasks and Skills:

- 3D printing
- Soldering SMT and Through-hole parts
- Threaded Heatset Insert assembly
- Programming

Components and Parts:

See the included Bill of Materials (BOM). I included part names, cost, description, and links to where I got them.