

RPV-80 Model 100

A modern revision of the Tandy TRS-80

It is a popular trend among makers to re-purpose old electronic housings for modern single-board computers. Many call them “cyberdecks,” based off William Gibson’s concept of a portable computer from his 1984 book, *Neuromancer*. I have wanted to undertake a similar project for some time using a Raspberry Pi to develop a portable and functional development platform for programming microcontrollers with Python. I recently acquired an original Tandy TRS-80 Model 100 with the intention of converting it for that purpose. The unit, listed as “for-parts and not functioning,” arrived at my house last week. I opened it and discovered a very clean, well-cared for, unit which seemed odd. So, I opened it and got to work with my multimeter. With minimal troubleshooting, I found and replaced a few damaged components on the main board. Next thing you know, I had the computer booting into its

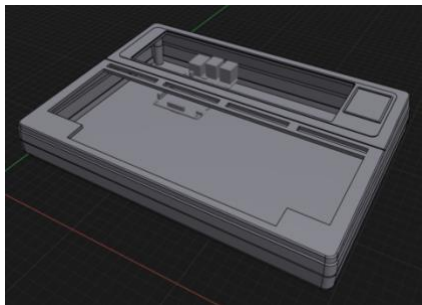


How can you say no to that face? Or that dim scanline?

pre-loaded operating system, **TRSDOS (Tandy Radio Shack Disk Operating System)**.

Now, these computers are considered by many to be one of the first versions of the modern-day laptop and have become somewhat of a collector’s item. The Model 100 was even considered one of Bill Gates’ favorite machines to write code for while he was on the development team at Microsoft. These computers have a rich history, and I just can’t bring myself to destroy a perfectly functional (if not ungodly slow and quirky) unit.

SO... I decided to revise the original model and make a “modern” version of the TRS-80. A project I named the [RPV-80 Model 100](#) (Raspberry Pi Version). I have a bit of experience 3D modeling so I started by taking a lot of measurements and drawing it up in CAD.



My 3D sketch of the Model 100 shell

Once I had the model complete and some test pieces printed, I started working on the most important part of the build, the interface; Specifically, the keyboard and mouse. The trouble with the original model is that it lacks a number of modern keyboard conventions such as independent bracket, flat/back slash, command, alt, and direction keys. So, I started to design a more modern keyboard layout that respected the original design, but added keys needed to write code in modern languages like Python.

I ended up designing a unique keyboard layout; it is a modified 60 keyboard that should provide functionality while also protecting aesthetic. I will also be incorporating 12 function keys above the keyboard similar to the original. Using an online keyboard building tool, I was able to



Check out that sweet double spacebar



It's actually happening!

generate a file that could be imported into CAD to create a 3D model of a switch plate. I printed this design, and started building the first physical prototype (below). I also plan to incorporate a mini trackball (think blackberry) on the top right of the keyboard. I was not able to source these parts quickly so I may be testing that with a breadboard and switches depending on shipping times from China.

This is where my final project for CS200 comes in. Once I finalize the design of my keyboard layout I will need to hand-wire a switch matrix and program a microcontroller as an interface device. There are numerous options for mapping software available online. However, what better way to honor the intention of this device than to write my own script in MicroPython! Adafruit has a few HID libraries and documentation available to assist with this. I plan to sort through their source code (which is thankfully written in Python), learn how it works, and write my own script from scratch. The end goal is to have a functional keyboard with integrated mouse that I can use inside my new computer case. I want to make sure it functions without noticeable lag, supports different key layers, can accept simultaneous inputs without issue (ctrl+alt+del) and interfaces with my Raspberry Pi 4 model B.

If I have time, I would like to also implement input acceleration for mouse, BLE connection, RGB controls for trackball, and MAYBE some custom keycaps.