Programming A Quirkey Controller

Overview

This document describes how a Quirkey chord keyboard's firmware is installed. The Quirkey hardware only uses a single controller board, six switches connected to the controller board IO pins, and a USB cable for connection to the user's PC. Current versions use either a small Arduino-compatible board with an AT32U4 processor such as a "Leonardo Micro", or a Raspberry Pi Pico (or Pico W) with an RP2040 processor. Other boards supporting CircuitPython and having a programmable USB controller that emulates an HID may be used.

Arduino

The Arduino version was the first target for the re-implementation of the original Microwriter device, which evolved into the Quirkey. Basic familiarity with the Arduino IDE is desirable. The IDE and beginner's tutorials can be found at https://arduino.cc

First retrieve the Microwriter.ino file from https://github.com/VikOlliver/Microwriter by clicking on the "Code <>" button and downloading the Zip file. Open the Microwriter-master.zip file and extract the Microwriter-master directory from within it. In your file manager, rename this directory to "Microwriter".

Plug the microcontroller into your PC and start the Arduino IDE. Open the File menu and select Open. Navigate to the "Microwriter" directory from above, and open the Microwriter.ino file. Open the Tools menu and select the Board sub-menu, then the "Arduino AVR boards". From there select the "Arduino Leonardo" type.

Open the Tools menu again, select the Port sub-menu, and from there select the serial port that the microcontroller is connected to (this will vary from PC to PC). Click on the Upload icon (circle containing an arrow pointing to the right).

The lower part of the screen should indicate that the PC is compiling, then uploading, and finally should show "Done uploading". Your controller is now ready for use in the Quirkey.

Raspberry Pi Pico

The Pico firmware is written in CircuitPython. This is an interpreted language, and so things are complicated by the need to install the CircuitPython interpreter first. You should download it as a UF2 file from https://circuitpython.org/board/raspberry_pi_pico – a comprehensive "Getting Started" guide is linked to on that page.

Open your file manager, connect the Pico to one end of the USB lead, and hold down the BOOTSEL switch on the Pico board. Keep holding it down and plug the USB lead in to your

PC. When your file manager shows a new USB storage device called “RPI-RP2” you can let go of the BOOTSEL button. Open this device. It should already contain “INDEX.HTM” and “INFO_UF2.TXT”. Copy the UF2 file that you downloaded above into the “RPI-RP2” directory.

Shortly after the copy is complete, the “RPI-RP2” will automatically disconnect and be replaced with a “CIRCUITPY” device. This indicates that you have correctly installed the CircuitPython interpreter. Open this device in the file manager.

The Quirkey code requires a library from Adafruit to function. The easiest way to do this is to go to the Adafruit CircuitPython HID example page at https://learn.adafruit.com/circuitpython-essentials/circuitpython-hid-keyboard-and-mouse and download the Project Bundle. This is a ZIP file.

Open the ZIP file and navigate into the CircuitPython_Essentials directory, then into the CircuitPython_HID_Keyboard directory. In there you will find a directory for the version of CircuitPython you are using (at time of writing 8.x). Navigate into that directory. Copy the “lib” directory from there into the CIRCUITPY device. This may overwrite an existing file, which is OK. You have now installed the necessary library.

Finally, retrieve the actual quirkey.py file from https://github.com/VikOlliver/Quirkey by clicking on the “Code <>” button and downloading the Zip file. Open the Quirkey-main.zip file and extract the Quirkey-main directory from within it. Navigate into this directory, and copy the quirkey.py file onto the CIRCUITPY device. Rename the copy to code.py. Unmount or eject the CIRCUITPY device. Your controller is now ready for use in the Quirkey.