

Vector Panel

The purpose of this document is to describe the setup of the underlying hardware of the panel. It also outlines some basic steps on how to connect and briefly test the panel.

Hardware Setup

The panel is modified from it's OEM specs to make use of an [Adafruit RGB Matrix Hat](#) for the Raspberry Pi, and a [Raspberry Pi Zero 2 W](#).

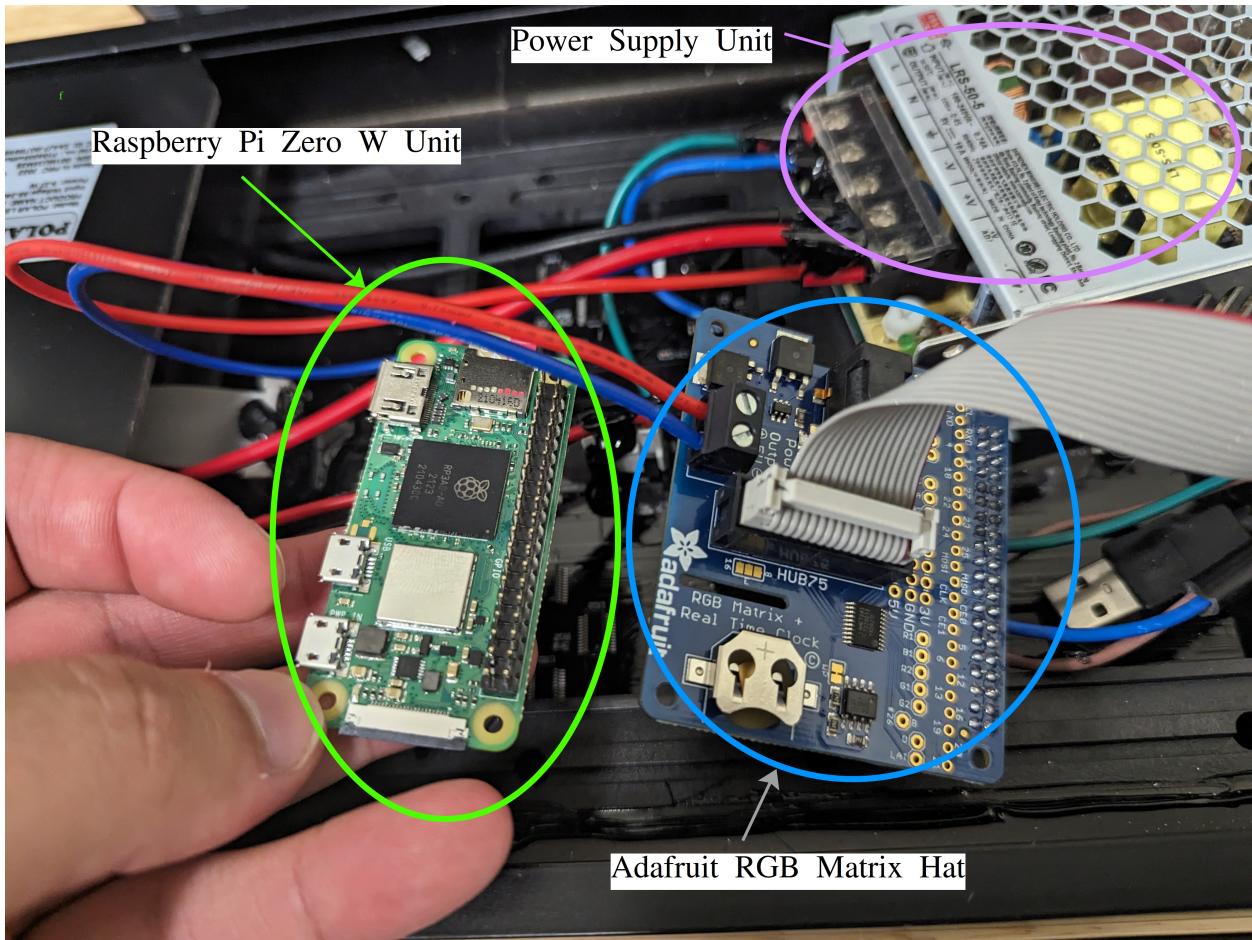


Figure 1: A Raspberry Pi Zero 2 W alongside the Adafruit RGB Matrix Hat.

The Adafruit hat is connected by the panel PSU (OEM), and connects to the LED panels via a HUB75 male header. The Raspberry Pi Zero 2 W comes soldered with a 2x20 male header that connects directly to the Adafruit hat.

Connecting & Testing The Panel

1. Connect the panel power cable to an outlet. Turn it on.
2. Connect to the VECTOR_Panel network
 - The password is VECTOR_1
3. Opening up a terminal, ssh to the panel: `ssh rgb-panel@192.168.0.0`
 - You will be prompted to enter a password. It is VECTOR
4. Run a quick demo to ensure working functionality:
 - From the root of the `rpi-rgb-led-matrix` directory, cd into `examples-api-use` and run the following as `root`
`./demo -D 0 --led-gpio-mapping=adafruit-hat --led-rows=16 --led-cols=32 --led-chain=3`
 - You can also run additional demos as well to further test functionality of the panel and also see what the LED library can do
5. Run a test image:
 - Back from the root of the directory, cd into the `utils` directory and run the following:
`./led-image-viewer --led-gpio-mapping=adafruit-hat --led-rows=16 --led-cols=32 --led-chain=3 [path-to-image]`

Ensure to include the path to image you'd like the panel to display. From the user directory, there is an `images` directory that comes with a sample `encoded-message-VECTOR.png` file, so the path to this image is as such: `~/images/encoded-message-VECTOR.png`

Controlling the RGB LED Display

The Adafruit hat is driven by a [LED matrix library](#) written in C++. Please note that this is licensed with the GNU GPL V2.0, meaning that any subsequent products/projects that make use of this software must be open-sourced.

Natively, the API is in C++, but it does support [C](#) and [Python3](#) bindings.

For the most part, making use of the existing compiled samples works well in displaying different images, but it is possible to develop different programs/scripts if it turns out it's not enough. Please look through the source code of the samples to see how they are done.

Connect to Internet

The Raspberry Pi Zero 2 W comes with WiFi, which is leveraged in creating an access point to be able to connect to it. However, doing so prevents the unit from accessing the internet, preventing the installation and updating of additional libraries/dependencies and tools.

Fortunately, we can turn off the access point and be able to connect to an internet WiFi network. Unfortunately, the Raspberry Pi unit will have to be removed from the panel and connected with peripherals: **1**) a mini-HDMI to whichever display port of your choice (i.e. HDMI/DP), **2**) a microUSB to USB to connect a wireless keyboard to, and **3**) a microUSB power source.

Once done, ensure to reconnect the Raspberry Pi Zero 2 W back to the Adafruit hat and close the panel.

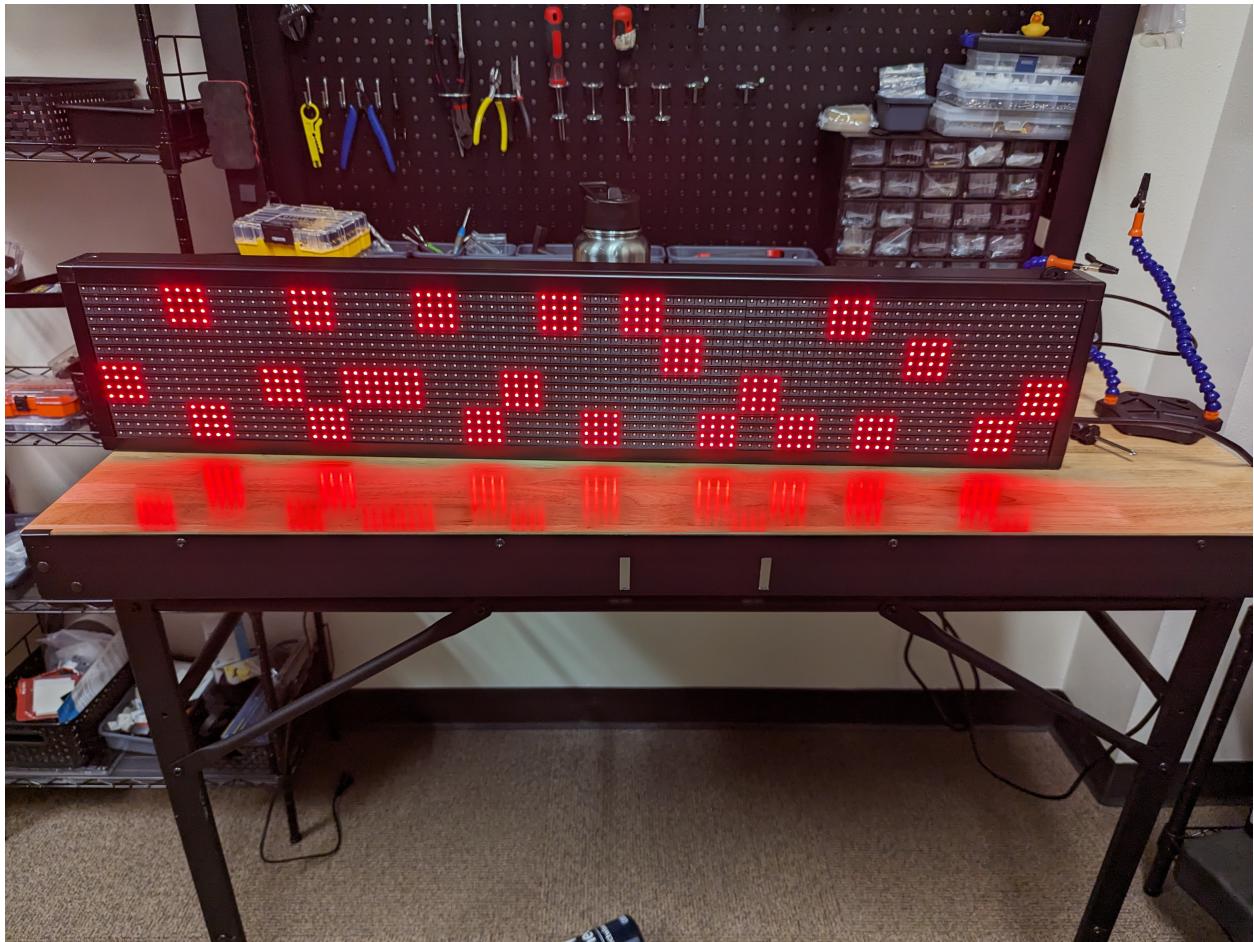


Figure 2: The LED panel displaying the ‘encoded-message-VECTOR.png‘ image.

Disabling the WiFi Access Point

1. Return DHCP to default settings:
 - `sudo cp /etc/dhcpcd.conf.orig /etc/dhcpcd.conf`
 - Restart the service: `sudo systemctl restart dhcpcd`
2. Stop the following services:
 - `sudo systemctl stop dnsmasq`
 - `sudo systemctl stop hostapd`
3. Disable the following services:
 - `sudo systemctl disable dnsmasq`
 - `sudo systemctl disable hostapd`
4. Reboot the system.
5. If not already connected to a network, run the `raspi-config` tool as root:
 - System Options -> Wireless LAN
6. Run `ping google.com` to verify connection.

Re-enabling the WiFi Access Point

1. Return DHCP to static IP settings:
 - `sudo cp /etc/dhcpcd.conf.static /etc/dhcpcd.conf`
 - Restart the service: `sudo systemctl restart dhcpcd`
2. Re-enable the `dnsmasq` and `hostapd` services:
 - `sudo systemctl enable dnsmasq`
 - `sudo systemctl enable hostapd`
 - `sudo systemctl start dnsmasq`
 - `sudo systemctl start hostapd`
3. Reboot the system.
4. Once the system is back, check for the `VECTOR_Panel` WiFi network.