ESP Home and Sonoff Basic

2022.05.16

- Chris Yarger has a handy ESP Home templates page.
- Alas: Chromium and Brave do not enable **WebSerial** for security reasons. Both Chris's site and ESPHome's on web installer complain about that.
- However, one can do it in pieces.
- First grab the ESPHome Flasher
- Specifically as of this moment, ESPHome-Flasher 1.4.0
- Make sure you're a member of the `dialout` group. (This goes without saying, if you're doing ANYTHING at all beyond using a web browser and a word processor.)
- As it needs to be an executable…

```
$ chmod 755 ESPHome-Flasher-1.4.0-Ubuntu-x64.exec
```

- Pop the bottom off of the Sonoff Basic. It's just held in place by plastic snaps.
- TTL serial not RS232 serial. Specifically Chris is using a FT232RL USB to TTL Adapter for Development Projects, USB to Serial Converter Module with Genuine FTDI USB UART available from Amazon for $10.00.
    - Clear-to-send (CTS) and Ready-to-Send (RTS) are broken out for working with older devices that have hardware flow control. (Modern devices use software flow control.)
    - XMIT and RECV LEDs
    - Switchable 5V or 3.3V
- Adafruit has a FTDI Serial TTL-232 USB Cable which looks like the programmer thing I built at HacDC to flash the Atmel Tiny MEGA, and possibly the cable I built for Julia. But I'll be damned if I know where it is.
- I THINK this is the same from Sparkfun: USB to TTL Serial Cable
- Connect the above serial cable (whichever one you get or make) to the edge of **Sonoff**:
    - **3.3V** to **3.3V**
    - **GND** to **GND**
    - **XMIT** to **RECV**
    - **RECV** to **XMIT**
- A 4-pin "header" (may need to break off four pins from a longer header) will help connect to the Sonoff. (Gravity is your friend.)
- Flint's Sonoff Setup documentation on Docbox

2022.05.23

- Flint's cable is verified: A loopback / echo test via `screen` verified that yea verily, his keystrokes are being echoed back when the XMIT and RCV
- Hold the button down while plugging in the USB to put the system in a **reflash** mode.
- Start the ESP Home flasher downloaded from last week. Refresh the ports list with the left-right arrow icon. Then use the pull-down and select the device (typically `/dev/ttyUSB#`). Browse for Chris's Sonoff Basic template – again from last week. Click **Flash**.
- If all goes well, the flash happens, and a new device shows up on the network.
- If all does NOT go well… Troubleshooting
- If the board is a **Sonoff RF R2 Power V1.4**, there is a **KEY** pin on the board and it should be grounded. While KEY is grounded, repeat the button-hold / connect cycle above.
- Assuming we're back to the "all goes well" and a new device has shown up, browse to that address, enter your gateway router address into the form, and **Save**.
- Now go to the address the gateway thinks the device is. Chris's(?) basic Sonoff interface should show up. Write down the serial number sonoff-basic-########. Etch the ######## on the board.
- Play with the web interface.
- `curl` can be used to control the beastie instead of the web browser See the web API documentation
- Flint's documentation: Sonoff Setup
- **Next week:** Using the Home Assistant to modify YAML configuration?, etc., etc.

Controlling the light with Sonoff Basic module

2022.07.04 (KJC)

- We're using ESPHome
- Wireshark captured a request sent to `192.168.1.219`:

```
POST /light/sonoff_basic/turn_on HTTP/1.1
```

- Experimentation with the above gets a status response, but does not change the state of the light:

```
$ curl http://192.168.1.219/light/sonoff_basic/turn_on
{"id":"light-sonoff_basic","state":"OFF","color_mode":"onoff","color":{}}
```

- Removing stuff from the end of the URL reveals that the minimum requirement for getting the same status report is:

```
$ curl http://192.168.1.219/light/sonoff_basic/
```

- In fact, it seems to ignore anything added after that.
- Since the response is suspiciously JSON… my stack overfloweth… How do I POST JSON data with cURL?

- It seems, based on the ESPHome API that we're not(?) sending a **POST** but rather, a **GET**, which is why we're getting a status back.
- See HTTP POST and GET using cURL in Linux? and How to send a header using a HTTP request through a cURL call?
- Ladies and gentlemen, we have a winner!

```
$ curl -X POST http://192.168.1.219/light/sonoff_basic/turn_on
$ curl -X POST http://192.168.1.219/light/sonoff_basic/turn_off
```