**Required Hardware:**

1. [ESP-EYE](https://www.espressif.com/en/products/hardware/esp-eye/overview)
   1. Optional [ESP-EYE case](https://www.thingiverse.com/thing:3586384)
2. [PyPortal](https://www.adafruit.com/product/4116)
   1. Optional [PyPortal case](https://www.thingiverse.com/thing:3469747)
3. Two 3.3v usb to outler adapters and two usb to usb mini cables

**OR**

1. Two 3.3v micro usb wall outlet chargers

Build Steps:

1. Clone the [on-air](https://git.sr.ht/~n0mn0m/on-air) repo.

Cloudflare Worker:

1. Setup [Cloudflare](https://www.cloudflare.com/dns/) DNS records for your domain and endpoint, or setup a new [domain](https://www.cloudflare.com/products/registrar/) with Cloudflare if you don’t have one to resolve the endpoint.
2. Setup a [Cloudflare workers](https://workers.cloudflare.com/) account with worker K/V.
3. Setup the [Wrangler](https://developers.cloudflare.com/workers/tooling/wrangler) CLI tool.
4. `cd` into the `on-air/sighandler` directory.
5. Update `[wrangler.toml](https://git.sr.ht/~n0mn0m/on-air/tree/master/sighandler/wrangler.toml)`
6. Run `wrangler preview`
7. `wrangler publish`
8. Update `[Makefile](https://git.sr.ht/~n0mn0m/on-air/tree/master/sighandler/Makefile)` with your domain and test calling.

PyPortal:

1. Setup CircuitPython 5.x on the [PyPortal](https://circuitpython.org/board/pyportal/).
   1. If you’re new to CircuitPython you should [read](https://learn.adafruit.com/welcome-to-circuitpython/circuitpython-essentials) this first.
2. Go to the directory where you cloned on-air.
3. `cd` into display.
4. Update `[secrets.py`](https://git.sr.ht/~n0mn0m/on-air/tree/master/display/secrets.py) with your wifi information and status URL endpoint.
5. Copy `code.py`, `secrets.py` and the bitmap files in `screens/` to the root of the PyPortal.
6. The display is now good to go.

ESP-EYE:

1. Setup `[esp-idf](https://docs.espressif.com/projects/esp-idf/en/latest/esp32/get-started/)` using the 4.1 release branch.
2. Install [espeak](http://espeak.sourceforge.net/) and [sox](http://sox.sourceforge.net/).
3. Setup a Python 3.7 virtual environment and install Tensorflow 1.15.
4. `cd` into `on-air/voice-assistant/train`
5. `chmod +x orchestrate.sh` and `./orchestrate.sh`
6. Once training completes `cd ../smalltalk`
7. Activate the `esp-idf` tooling so that `$IDF\_PATH` is set correctly and all requirements are met.
8. `idf.py menuconfig` and set your wifi settings.
9. Update the URL in `[toggle\_status.cc](https://git.sr.ht/~n0mn0m/on-air/tree/master/voice-assistant/smalltalk/main/http/toggle_status.cc)`
   1. This should match the host and endpoint you deployed the Cloudflare worker to above
10. `idf.py build`
11. `idf.py --port <device port> flash monitor`
12. You should see the device start, attach to WiFi and begin listening for the wake word “visual” followed by “on” or “off”.