Build an ESP32 IR Remote (and Web Server)

Henry Cheung

Hackware V5.8
3 December, 2019

About Me

- Henry Cheung - Tinker, Blogger, Whatever
  - Tinker - Make and break all things electronics
  - Blogger - Run a web blog https://www.e-tinkers.com that is hosted on a Raspberry Pi 3
  - Electronic-trained (Radio Communication and Embedded Microprocessor)
  - A software developer (Web, IoT) in Python, Golang, JavaScript/Nodejs, C/C++

About ESP32 IR Remote

- My TV remote corroded by leaking battery
- To watch TV, I need two remotes, TV and Set-top box
- Wouldn’t it be nice if I could control my TV and Set-top box using my iPhone?
- Why not use an ESP32 to build an IR remote?
- I also have a SHT21 Temperature/Humidity breakout board that I’d like to incorporate into the design

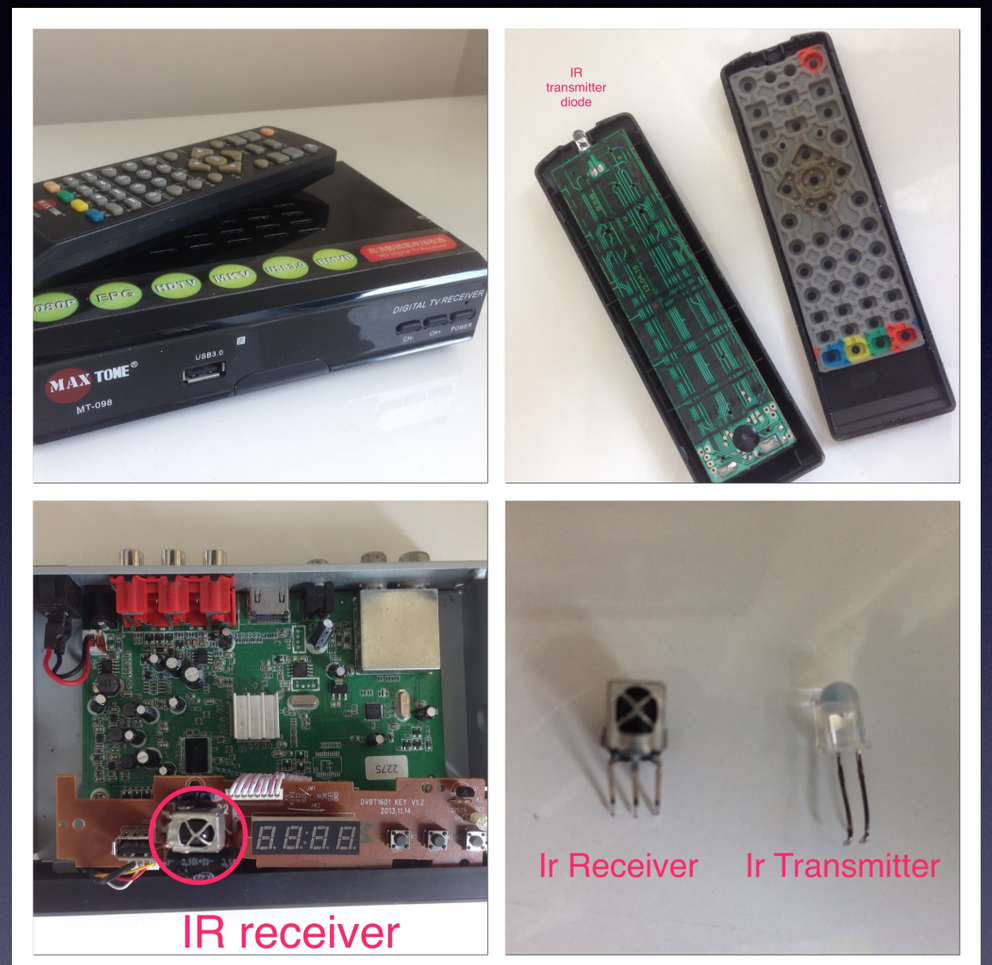
ESP32

Arduino	ESP8266	**ESP32**
AVR ATMega328P	Tensilica Xtensa LX106	**Tensilica Xtensa LX6**
8 bit	32 bit	**32 bit**
1 core	1 core	**2 cores***
20 MHz	80/160 MHz	**160/240 MHz**
2 KB RAM	160 KB RAM	**520 KB RAM**
32 KB Flash	1 or 4MB Flash	**4 or 16 MB Flash**

IR Transmitter and Receiver

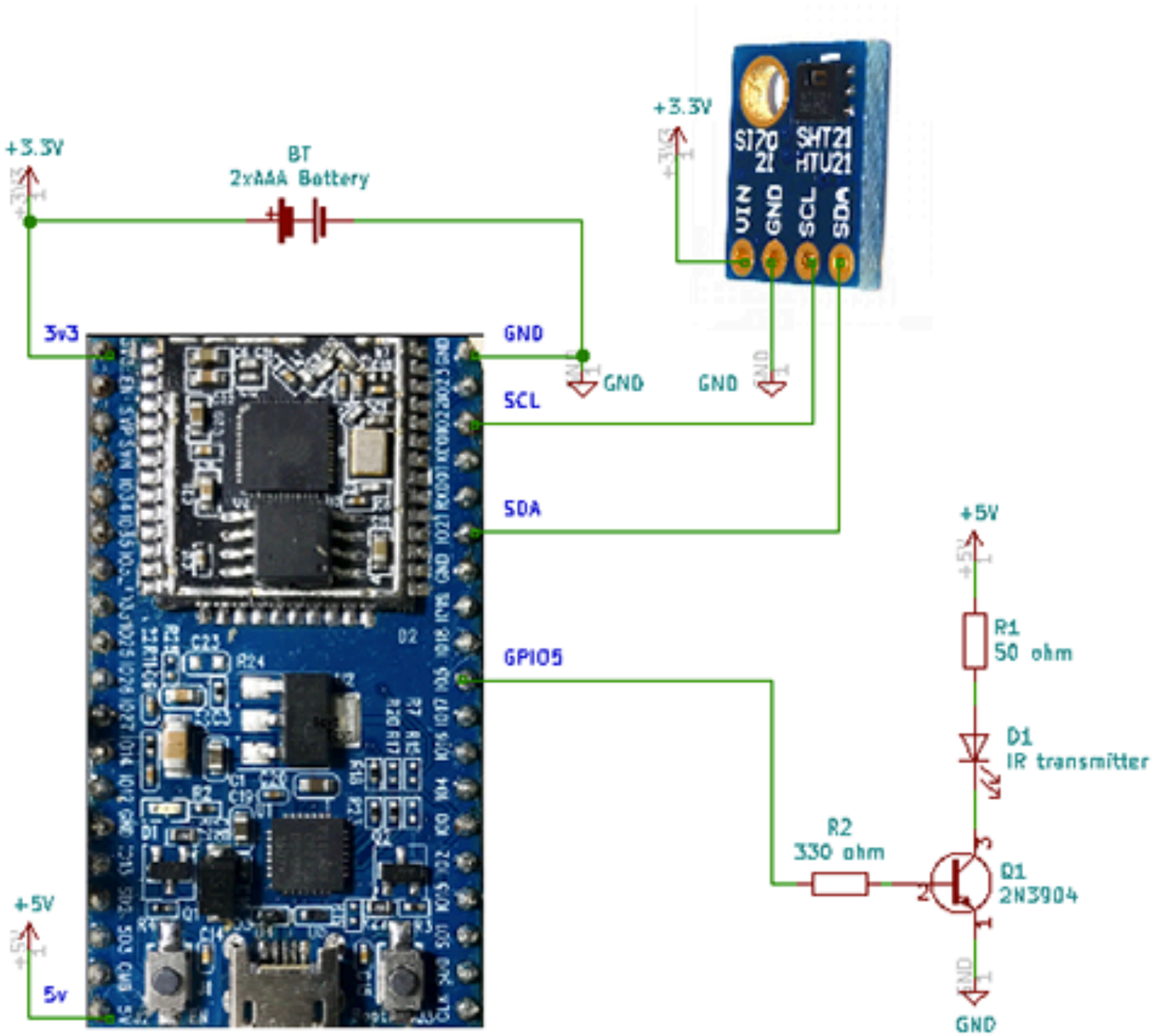
Wrote a simple Arduino sketch to decode the code and protocol used by my TV and Set-box remote controls

Remote	Protocol	Key	IR Code
TV	Samsung	TV on/off	0xE0E040BF
		TV Source	0xE0E0807F
Set-top Box	NEC	Set-top On/Off	0x807F807F
		Volume Up	0x807F827D
		Volume Down	0x807F42BD
		Channel Up	0x807FE817
		Channel Down	0x807F58A7
		Mute	0x807F48B7

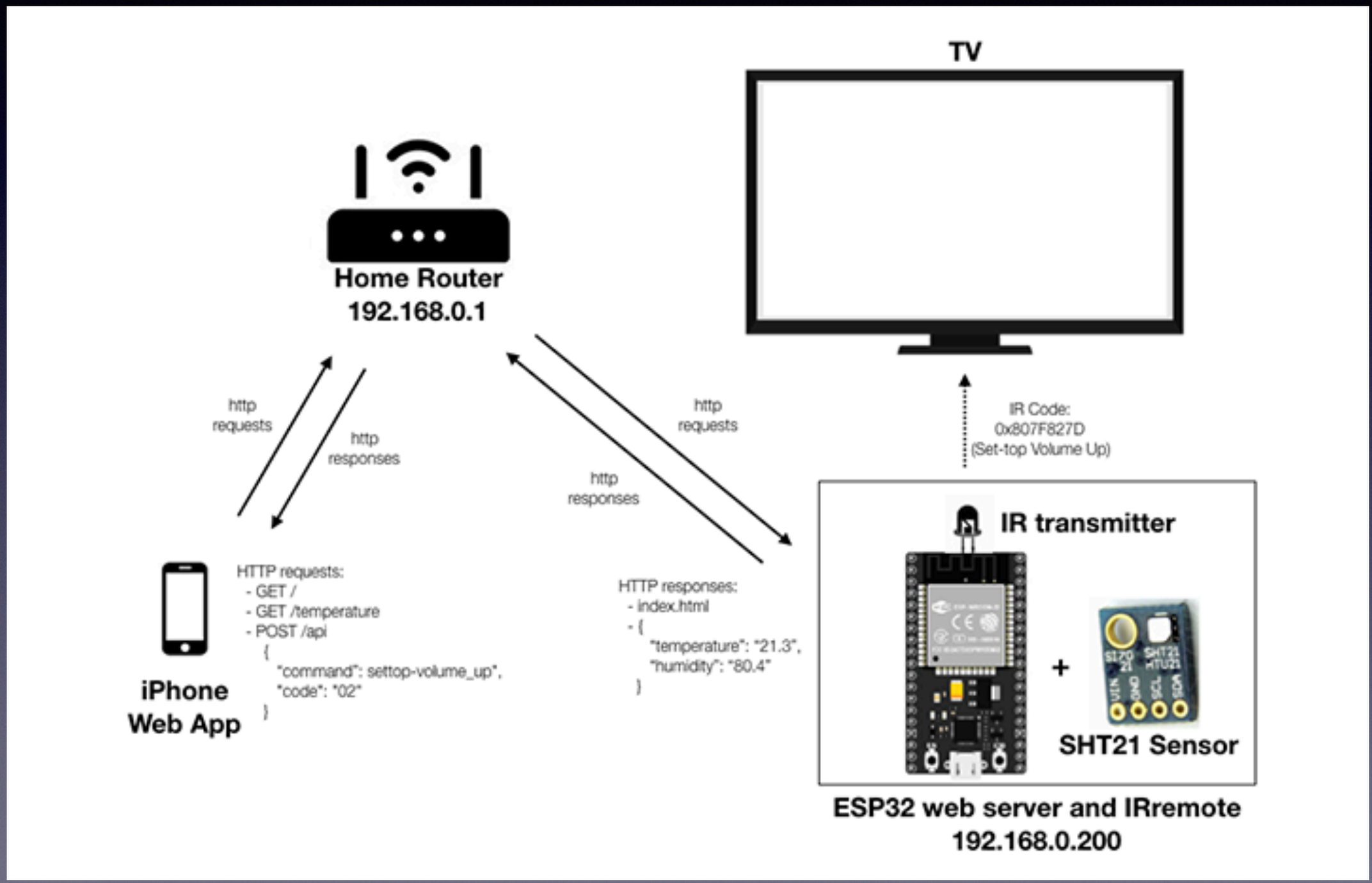


Get a pair of IR transmitter and receiver from an old set-top box

ESP32 IR Remote Circuit



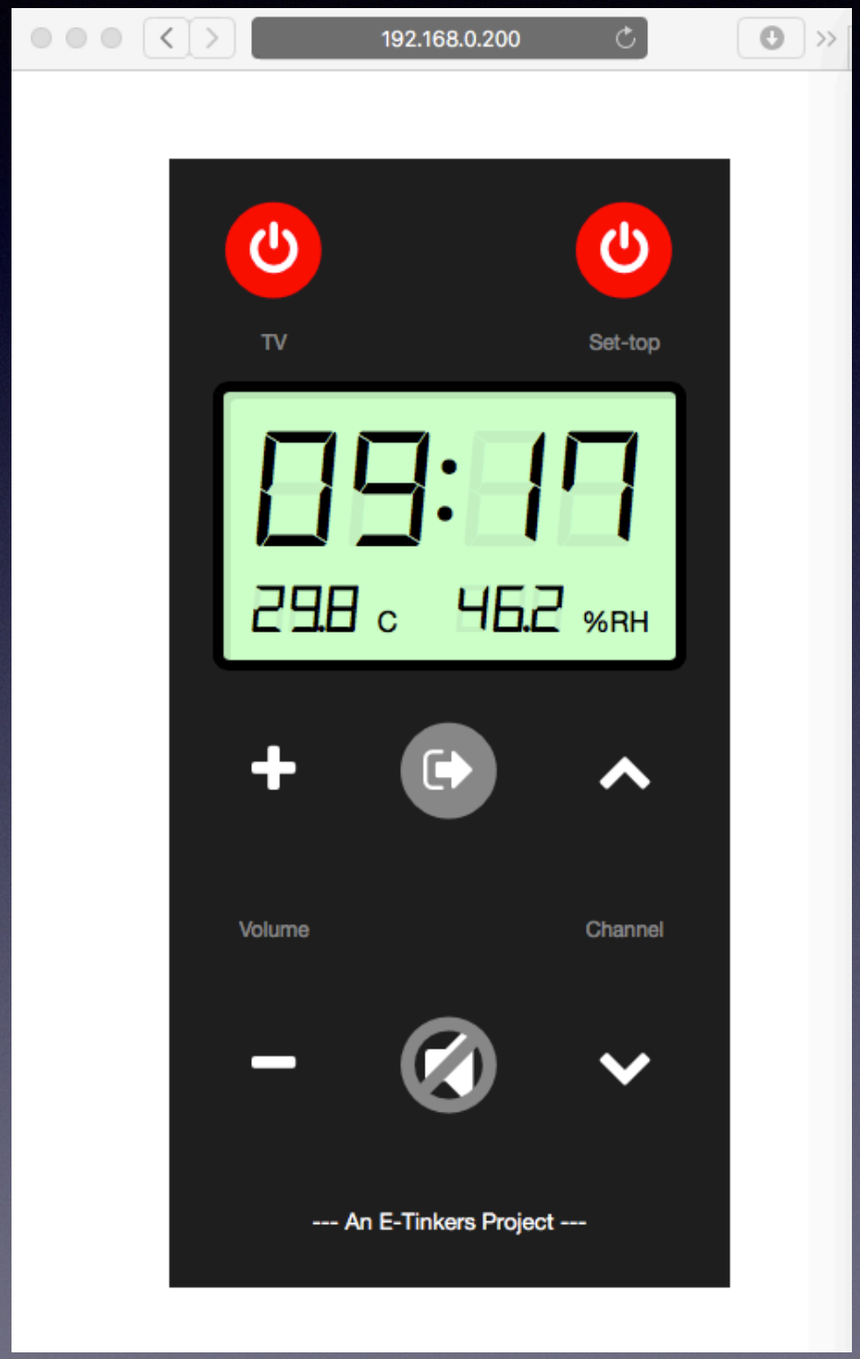
ESP32 Web Server

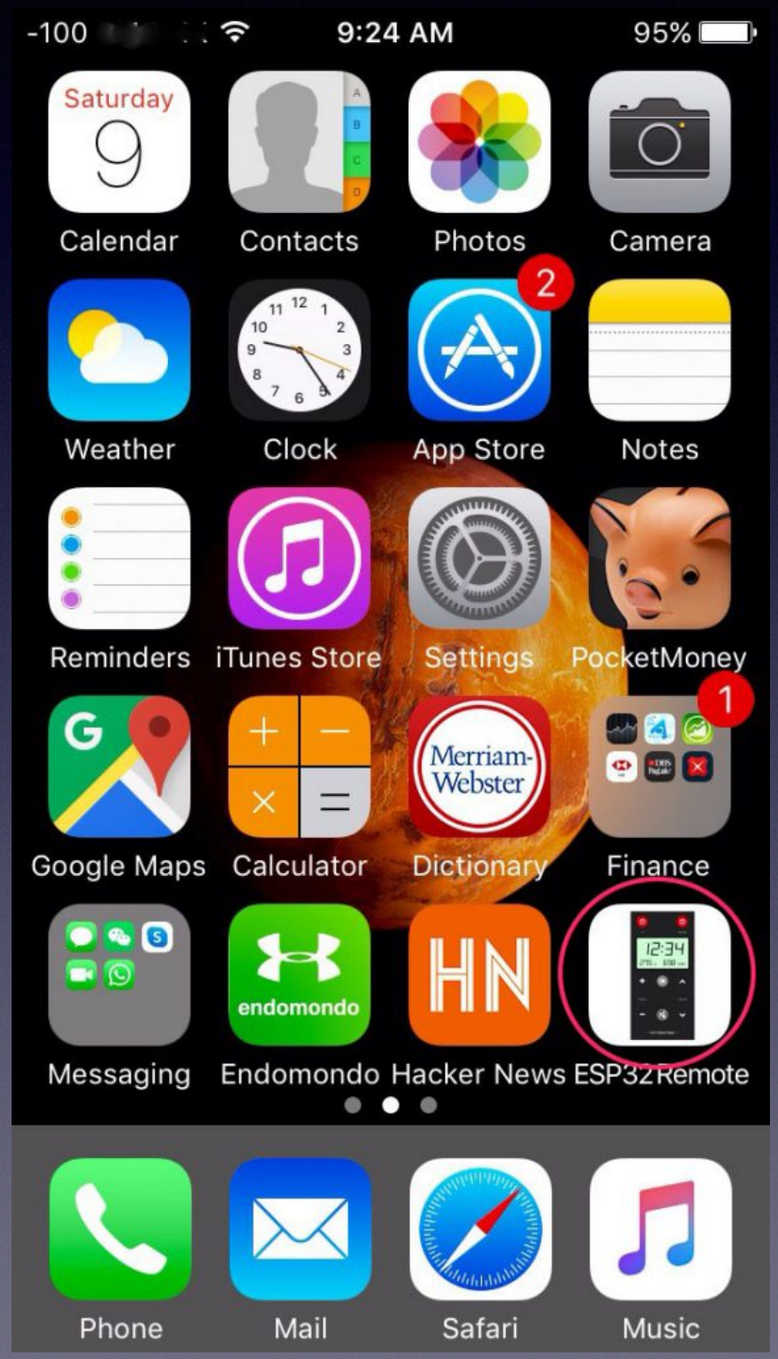


ESP32 Web Front-End

- Accessable via browser at http://192.168.0.200
- HTML responsible for creating a UI that resembling a TV remote. It also provides a display area for displaying temperature and humidity information, along with the current time.

ESP32 Web Front-End





ESP32 Web Front-End

- JavaScript is used to setup event listener for each remote control's key, and two Ajax requests
  - GET /temperature - send GET request to server. Web Server will return a json response with temperature/humidity information

```
{
  “temperature”: “28.3”,
  “humidity”: “80.4”
}
```

  - POST /api - send POST request with a json object when a key is pressed

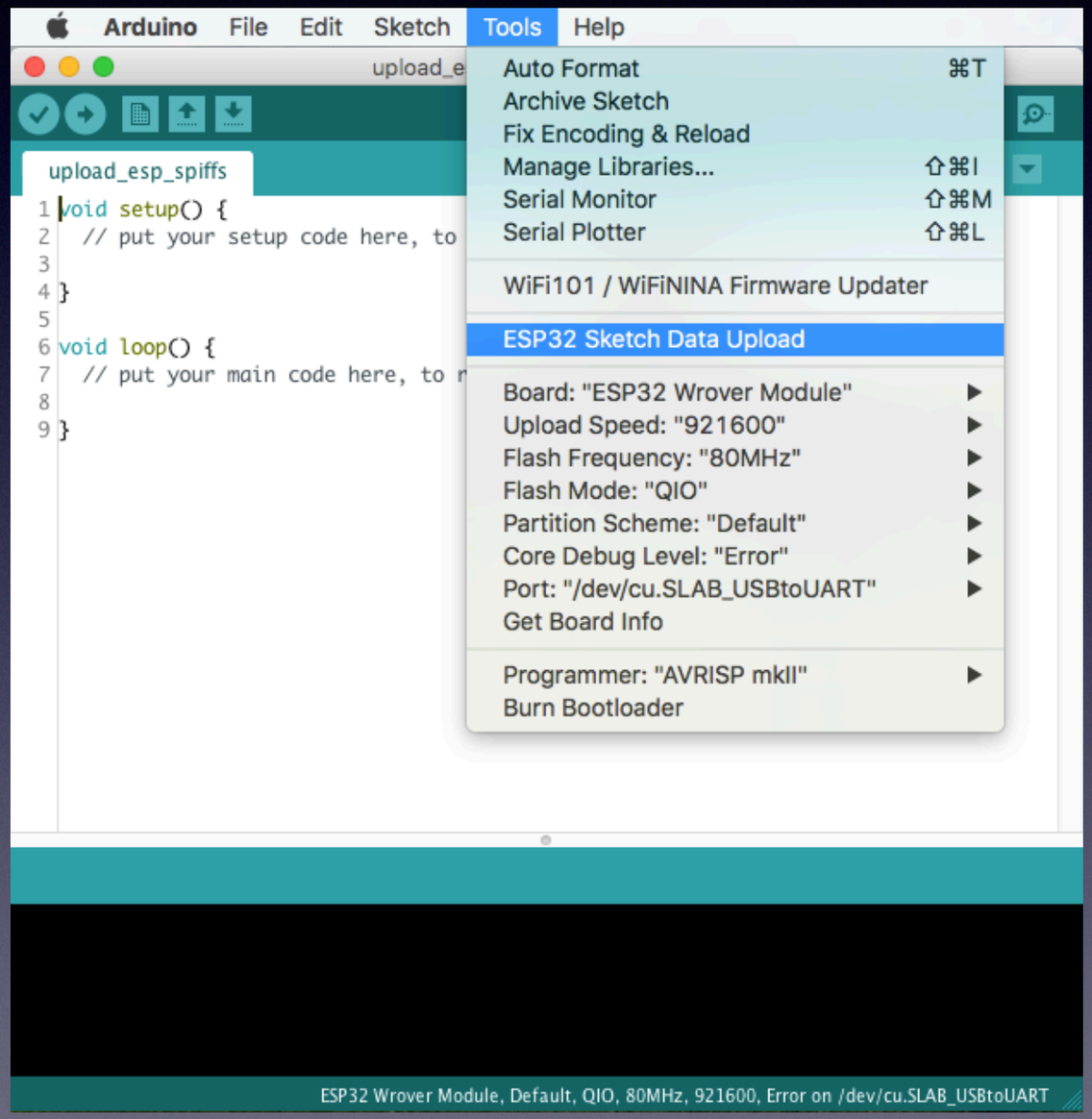
```
{
  “command”: “tv-on_off”,
  “code”: “00”
}
```

ESP32 SPIFFS

- ESP32 supports Serial Peripheral Interface Flash File System (SPIFFS) - lightweight filesystem for micro controllers with flash memory, using SPI protocol/bus

- Allows files to be stored separately within flash memory with a structure that “looks” like separated in different directories

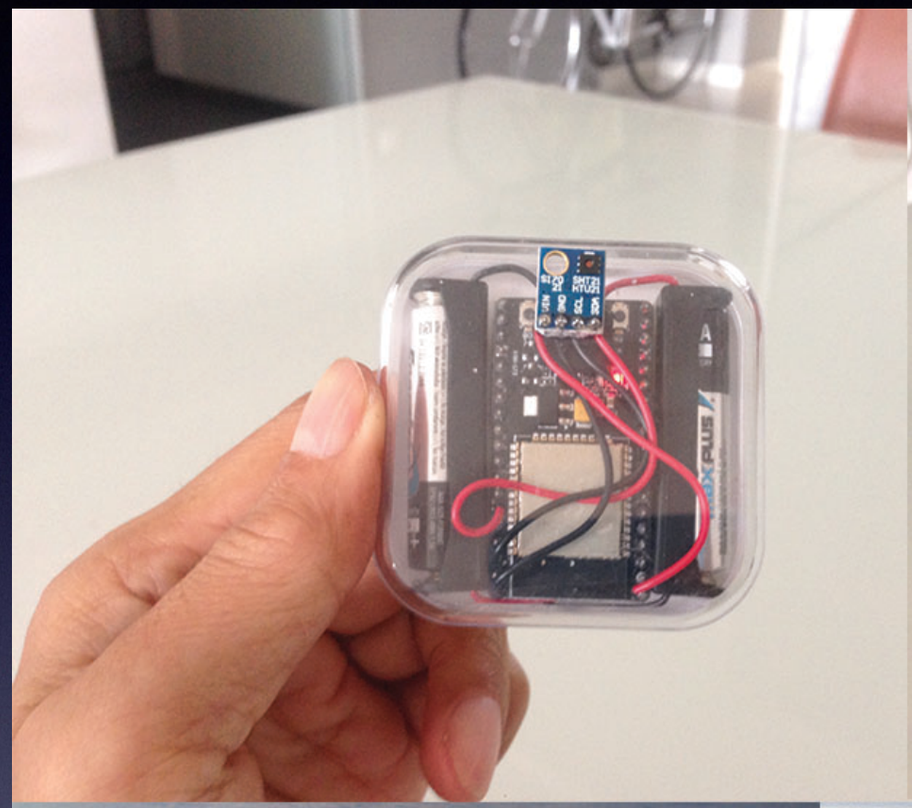
- All of html, css and JavaScript files can be stored in a directory called “data” and uploaded to ESP32 SPIFFS via a special Arduino IDE plugin

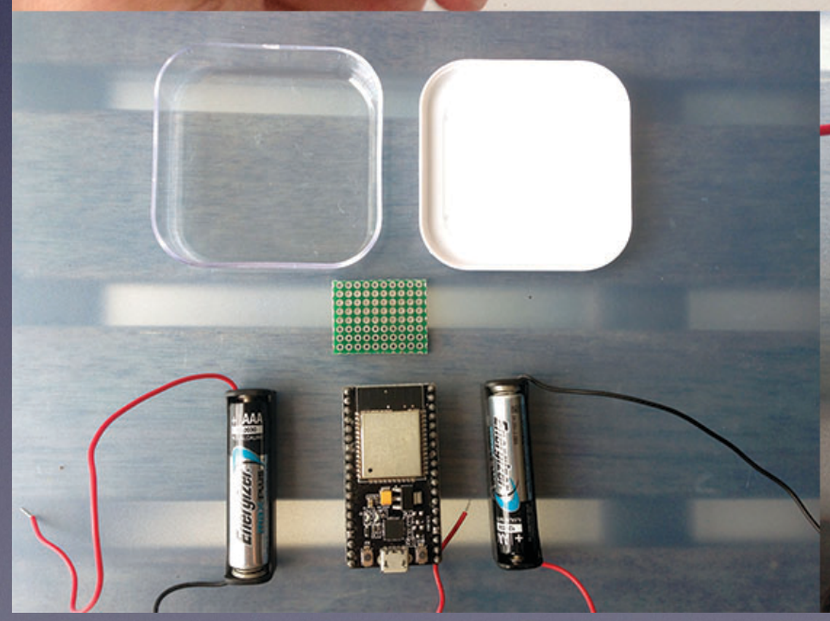


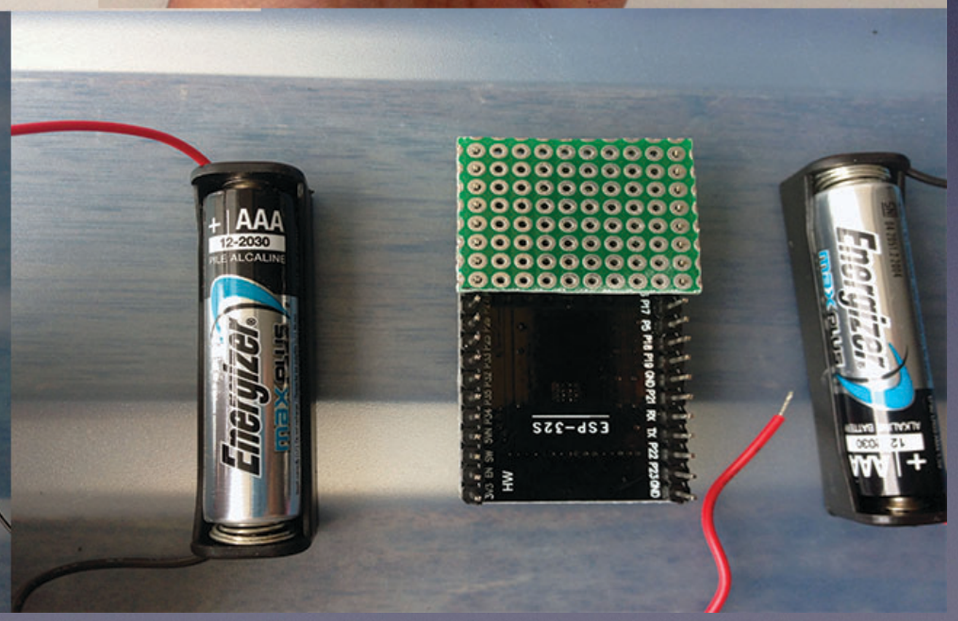
ESP32 Web Server

- The Web Server back-end code (i.e. main.cpp) has a few dependencies (libraries) that need to be installed
- The Web Server setup WiFi station mode with an static IP at 192.168.0.200
- Setup a few routes and callback functions for handling the routes:
  - GET / - serve index.html from SPIFFS
  - GET /temperature - onTemperature() send data to client
  - POST /api - onBody() parse received command and generate IR code to control TV/Set-top box

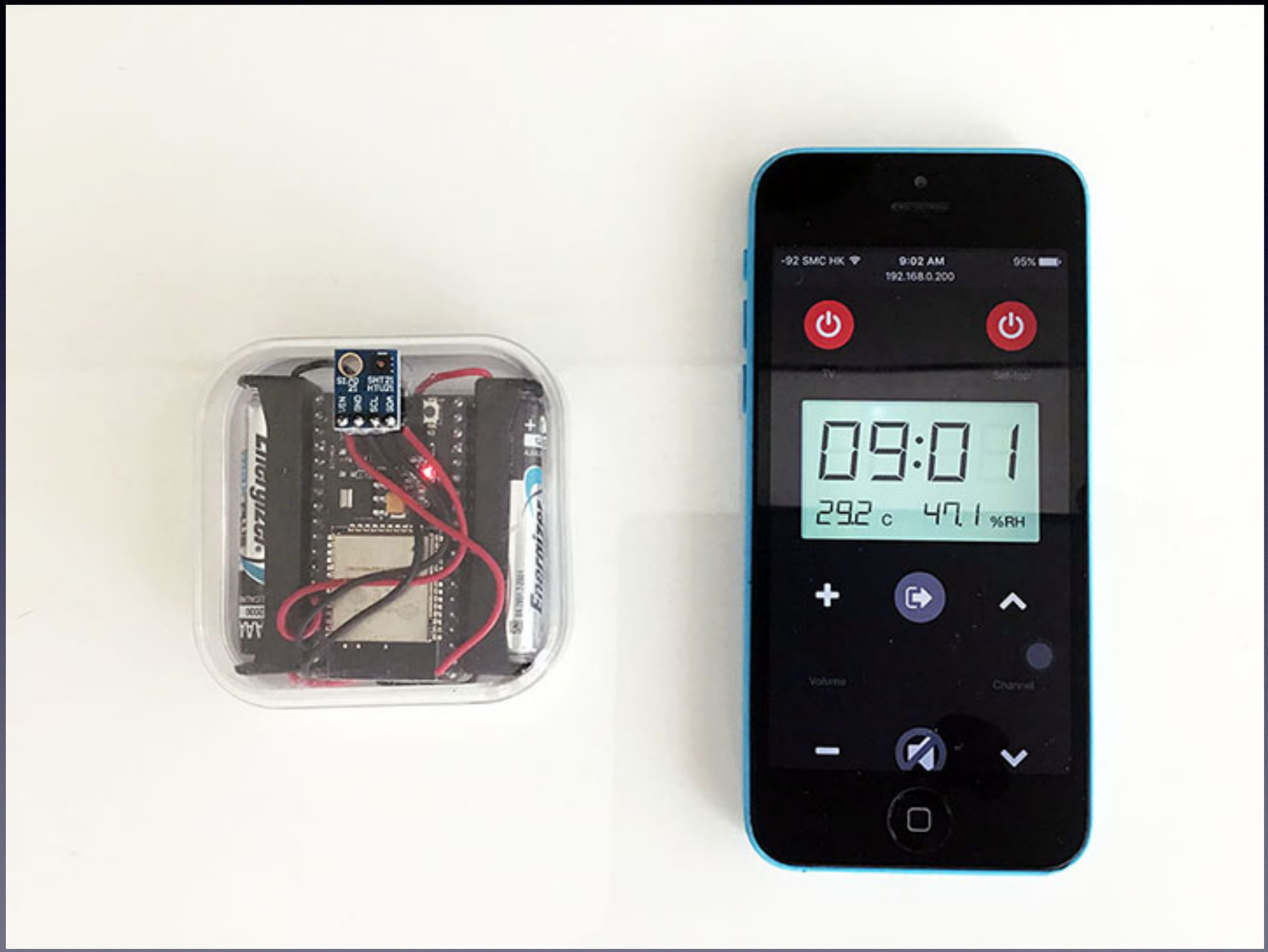
ESP32 IR Remote Hardware



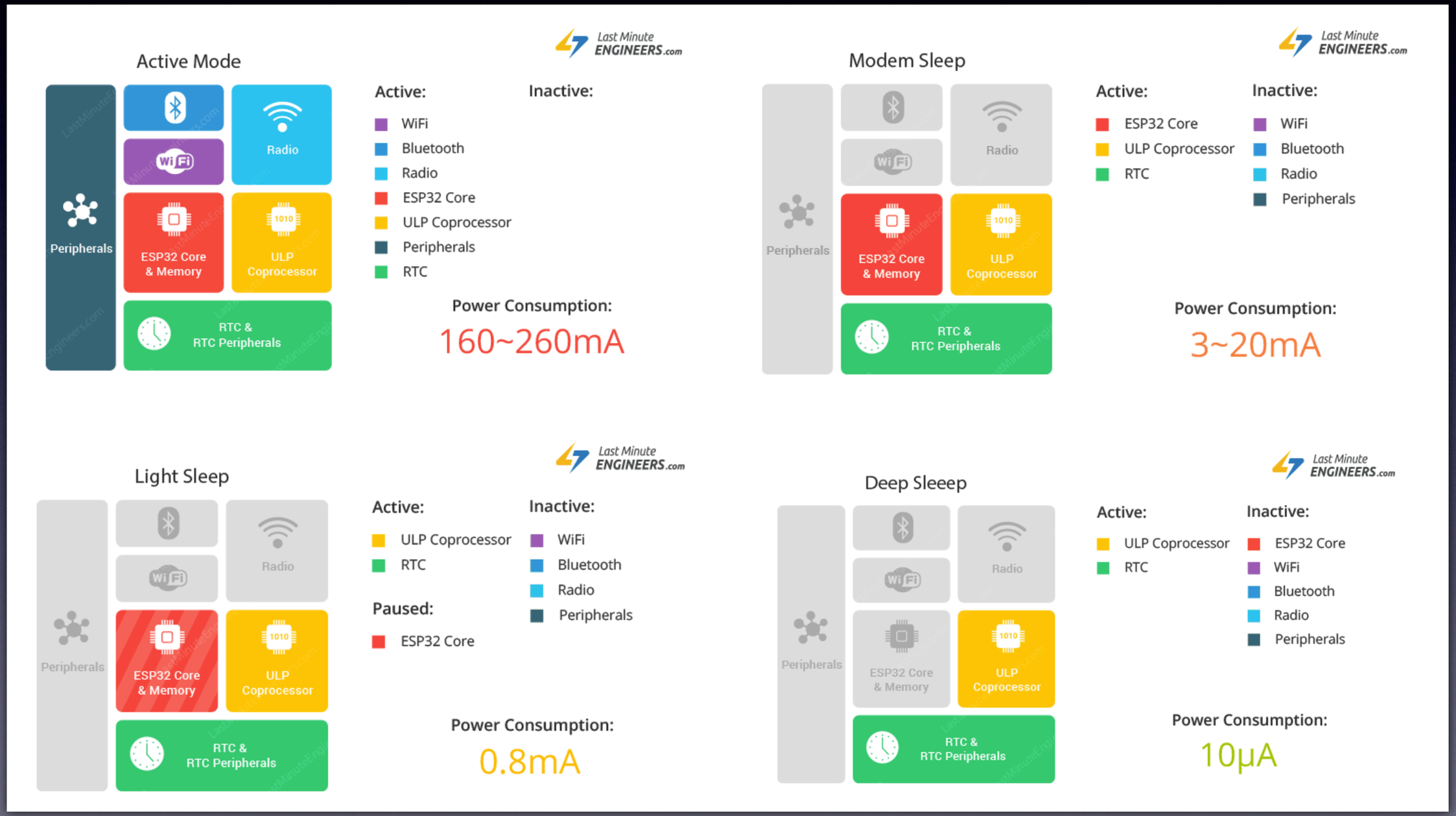




Final Product



Challenge - Battery Life



Source: https://lastminuteengineers.com/esp32-sleep-modes-power-consumption/

Project Resources

- Blog:
  - https://www.e-tinkers.com/2019/11/build-an-esp32-web-server-and-ir-remote/
- Github:
  - https://github.com/e-tinkers/esp32_ir_remote