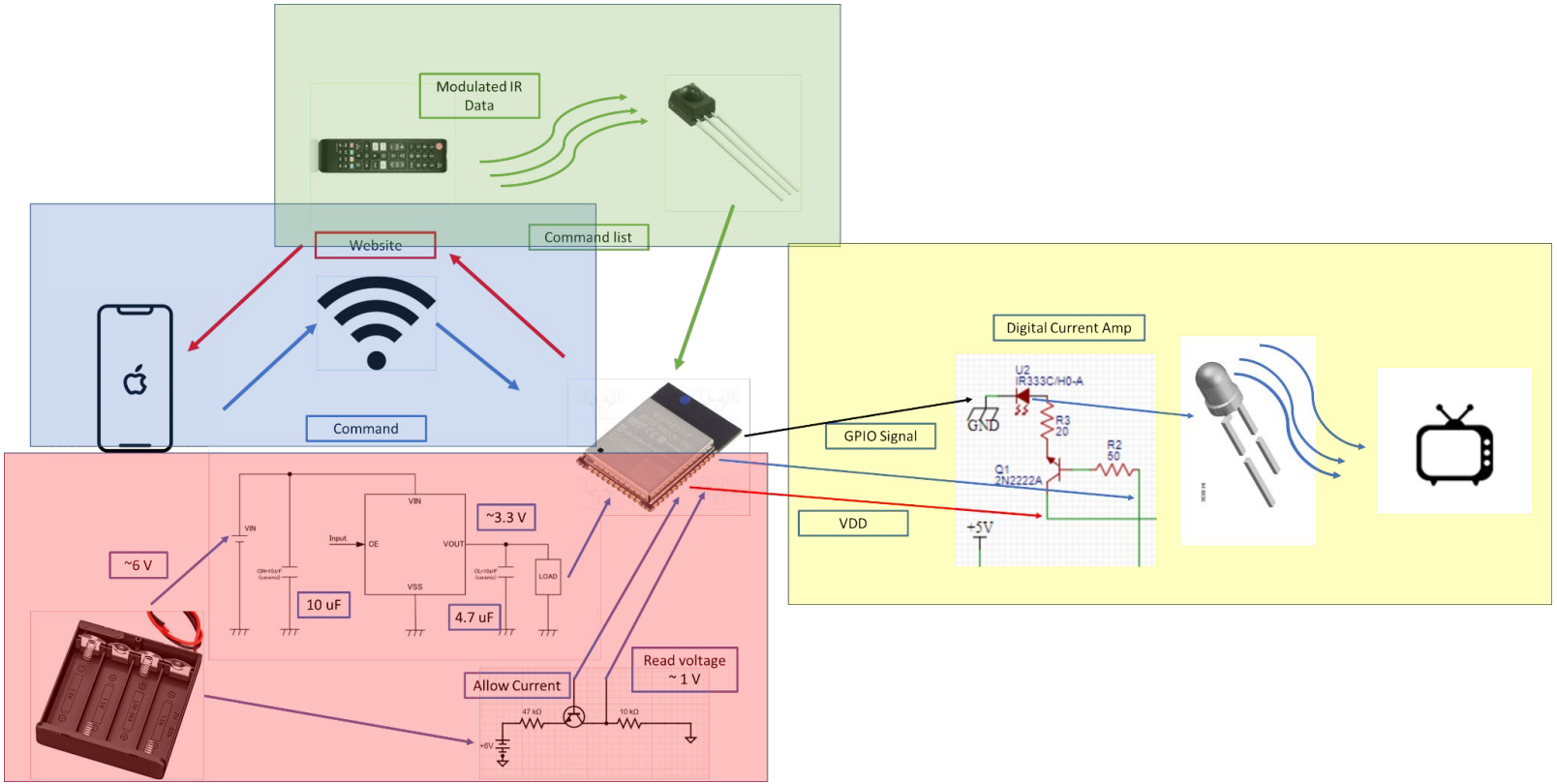
Design Review 1

IR Remote Project - Group 11

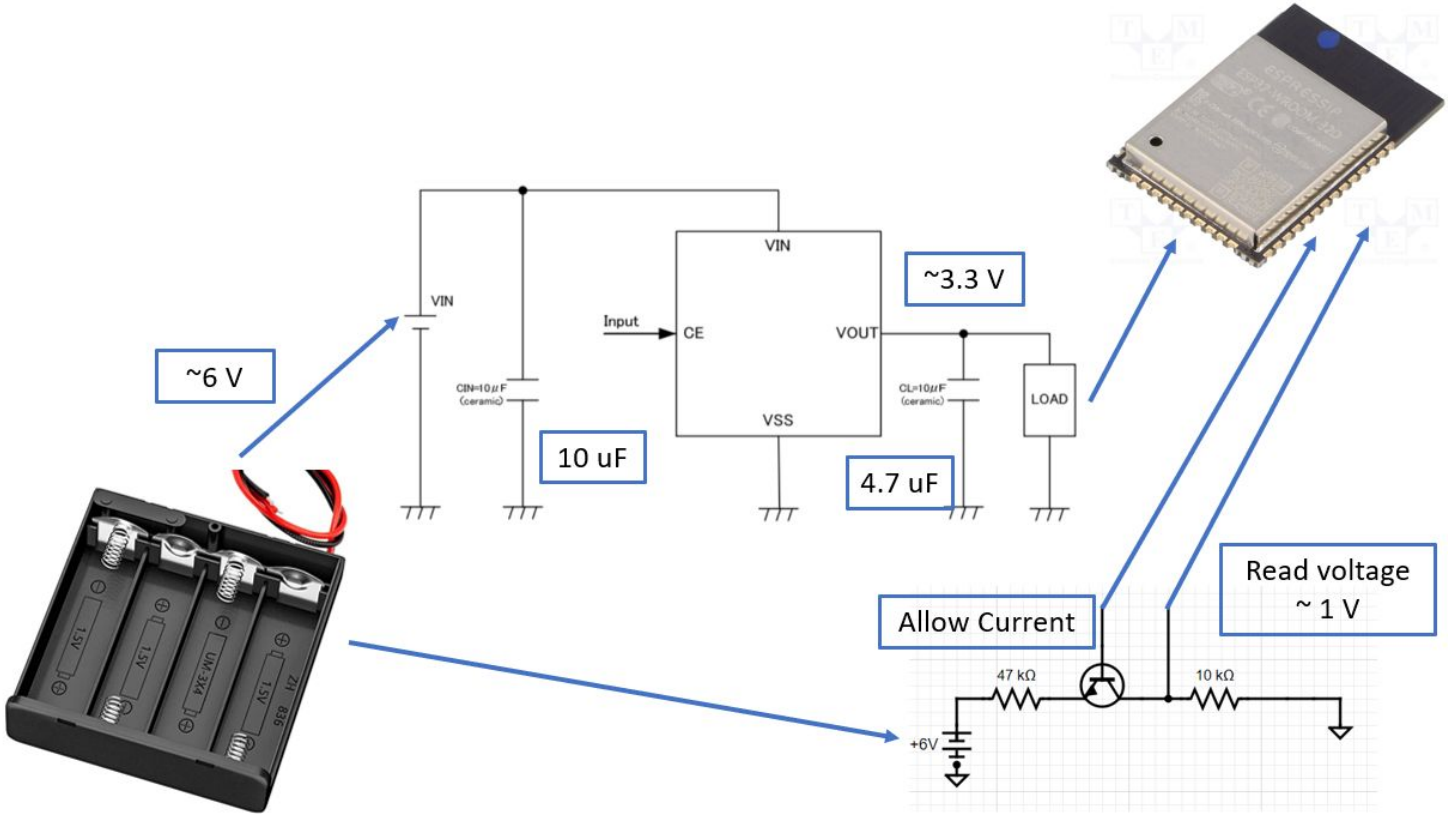
System Overview Broken Into Subsystems



Subsystem 1: Power - Design and Overview

- Overview:
  - We want to supply with external batteries and give 3.3 V out of a regulator
  - Needs to power C3 and IR actuator circuit (which has amplifier circuit)
- Power will be supplied via batteries
  - VIN on regulator maxes at ~6.5 V, using 9 V is too much
  - Therefore, we'll use series connection of 3-4 AA, standard battery carrier
  - Link
- GPIO pin monitors DC input voltage, notifies users when voltage is sagging
  - Resistor voltage divider that draws little to no current; ESP32 input voltage limit 3.3 V
  - Goes to ESP32, while operating, code will sample voltage on that pin and show it as power bar.
- XC6220
  - Output of 3.3 V, same passives as first board design from last semester

Subsystem 1: Power - Diagram



Subsystem 1: Power - Progress

- Selection of regulator
  - XC6220
- Selection of ESP32
  - We will use the C3; examples show single-core works for this application
  - 3.3 VIN will be used - passives selected (see diagram) - same as last semester
- Selection of battery powering and monitoring
  - User friendly, just need AAs.
  - Digitally allows users to know their battery status, **circuit designed.**

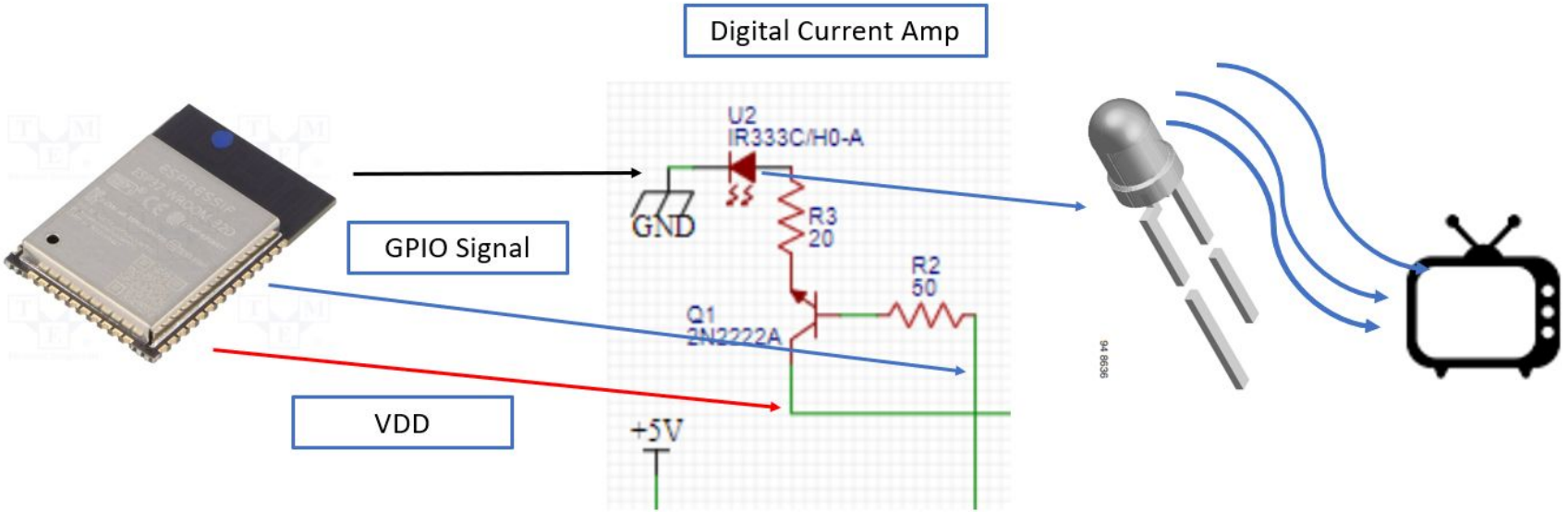
Subsystem 1: Power - Plan for Design Review 2

- Generate PCB
  - Use last semester design as a model
  - Passives for regulator will be the same; we chose C3, as we have seen 1-core examples for this type of project, therefore passive are same for ESP32 as last semester too
  - Purchase battery pack and some AAs; **include solder pads on PCB** so external battery pack can attach.
- Only change:
  - Need to add small voltage divider circuit on PCB
  - Need a **MOSFET** with gate controlled by ESP32; allows current to flow for readout.
  - See diagram

Subsystem 2: IR Actuator - Overview and Design

- Overview:
  - ESP32 outputs modulated data signal from GPIO pins. This circuit converts this to an IR signal.
  - ESP32 output amperage is ~40 mA, not quite enough for our IRED
  - We want simple amplifier circuit driven by GPIO pin of ESP32
- Design
  - A simple BJT transistor will work as amplifier; ESP32 signal goes to gate
  - Bias is 3.3 V, pick resistors accordingly;
  - Though our current configuration ~works~, we want ~100 mA through our IRED

Subsystem 2: IR Actuator - Diagram



SubSystem 2: IR Actuator - Progress

- Demonstrated our IRED works
  - We selected correct wavelength and bandwidth
  - Showed that ESP32 can output modulated IR signal
- That’s about it!

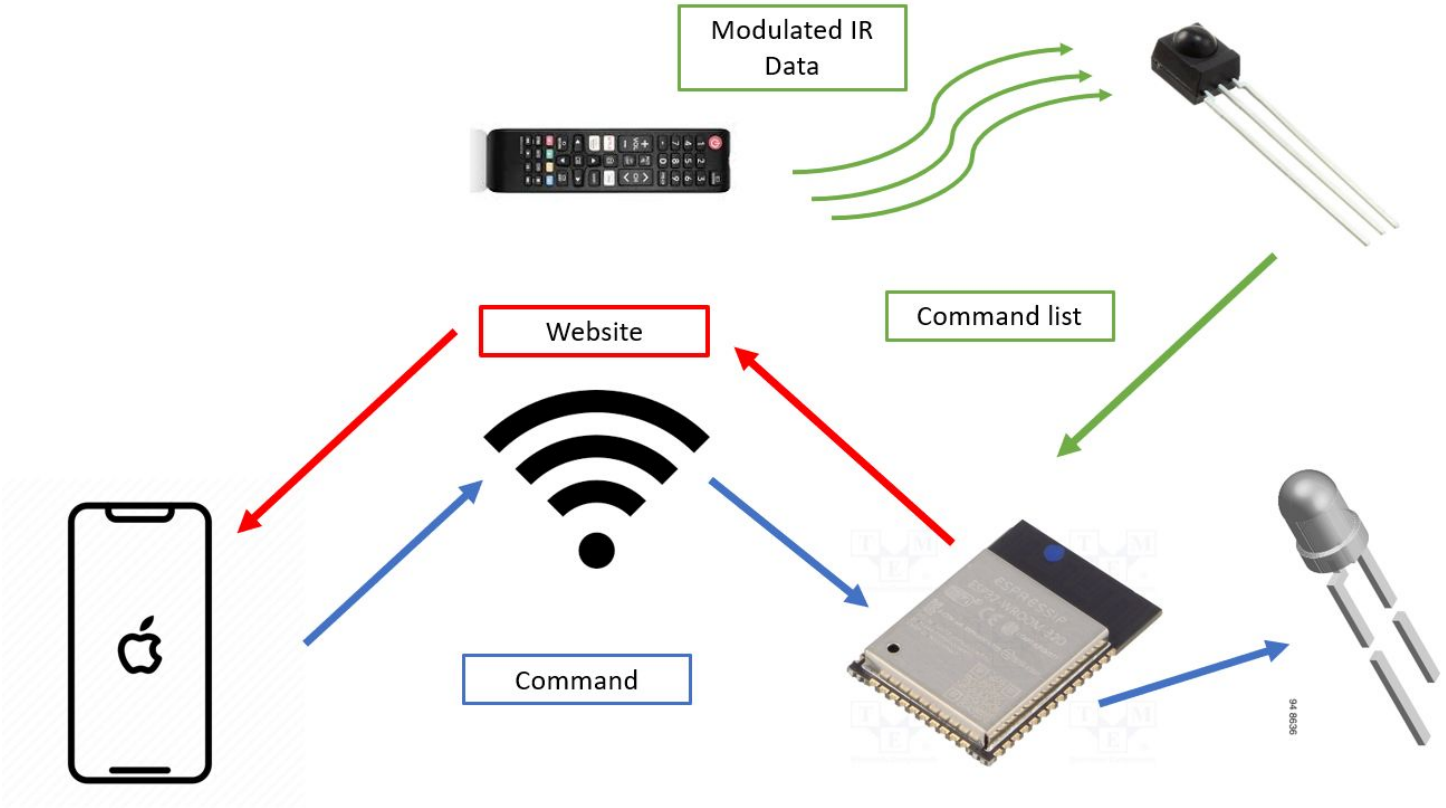
What needs to be done?

- Simulate - find resistor values in diagram above to get optimal output voltage and current
  - Voltage: 1.3 V drop across IRED
  - Bias voltage: 3.3 is the most we can hope to achieve
  - ESP32 output: pins output 3.3 as well, may need divided to eliminate high voltages
  - Output current: we want 100 mA! Select resistors accordingly.
- Design PCB:
  - Make all components surface mount, routes ESP32 to the subcircuit; keep everything close together, capacitance affects BW
  - Feedthrough holes for IRED, build near the edge of device.

Subsystem 3: IR Code - Overview and Design

- Overview: This system blends with the website, as they cannot be decoupled. However, this code consists of a list of commands for changing TV properties like power, volume, cursor, etc.
- Design: create a library for each type of TV we're interested in with relevant commands. Commands can be found online or via our receiver circuit.

Subsystem 3: IR Code - Diagram



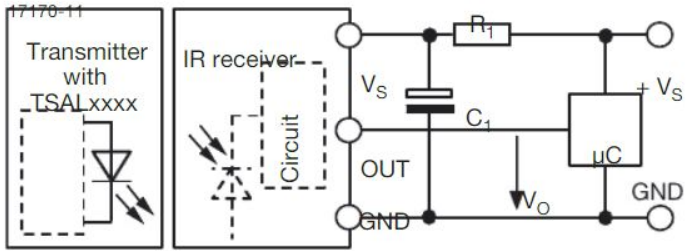
Subsystem 3: IR Code - Progress

- Receiver is up and running and we have a list of relevant codes for the TV in the SD lab
- Transmitter is up and running and we have used codes from remote to send signals with ESP32

Subsystem 3: IR Code - To do

- Keep abstracting / developing library
- Add code for new remotes (beyond samsung)
- Continue to integrate with website
- Create receiver circuit for PCB

APPLICATION CIRCUIT

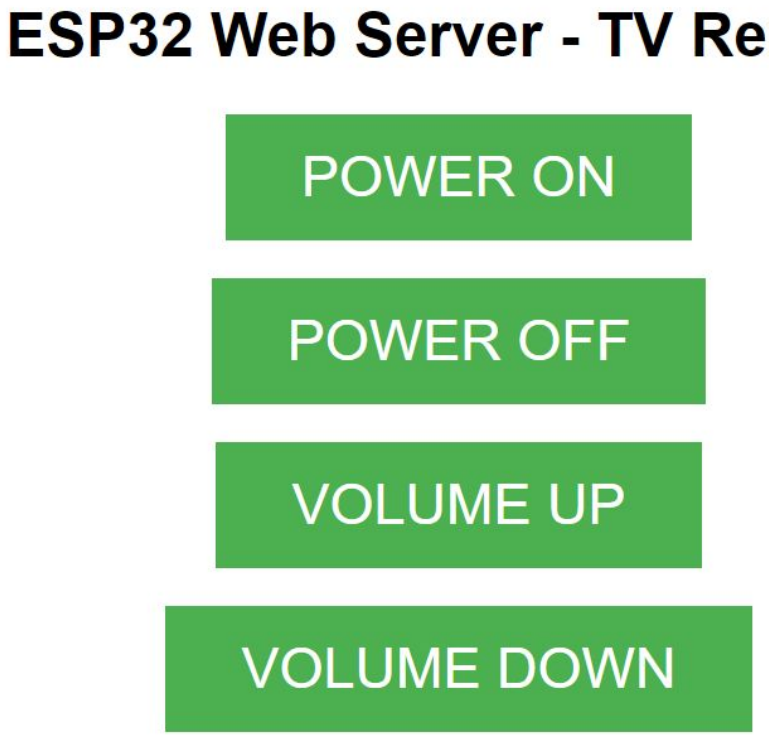


R_1 and C_1 recommended to reduce supply ripple for V_S < 2.8 V

Subsystem 4: Website - Overview and Design

- Website style:
  - HOME PAGE:
    - Select from current users, which will redirect to IP#\user_name\
    - Make new profile which will redirect to IP#\create_remote\
  - User Page
    - Just the remote made for that specific user
    - No ability to add buttons; would need to make new profile
  - Create Page
    - Page with bunch of prompts that allows people to make their own remote
- Website logistics:
  - Everything with customization is purely software; just need to code it out well
  - Buttons on the User Page need to link to hardware, i.e., call functions for turning on the TV.

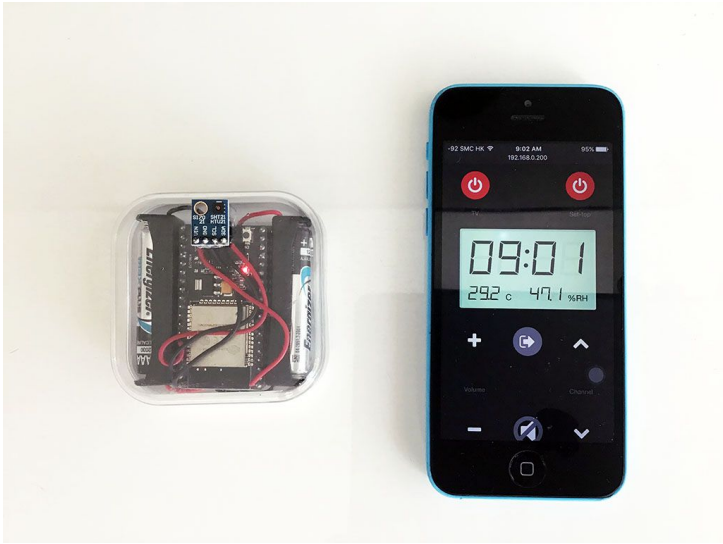
Subsystem 4: Website - Progress



Live demo time!

Subsystem 4: Website - To do

- Make it prettier and more complex
- WiFi auto connect interface with phone as previously discussed
- Add customization features (lower priority)
  - Home page with profile selection
  - Ability to add profiles with certain features
  - Example →



Ranked To Do List for Next Time

1. PCB Design - We have verified everything we need to and are ready to map out a board. We want the follow on the board (as a review):
    - Same VRM from first semester
    - Same ESP32 (C3) as first semester
    - Battery power monitor circuit: Voltage divider powered by battery connected to GPIO pin of C3; divider is an open circuit due to MOSFET whose gate is connected to a different GPIO of the C3. To read batteries, output high on MOSFET gate, read voltage divider.
    - BJT with bias resistors whose gate is connected to GPIO pin of C3 to power IRED. This is to maximize output current.
    - Purchase and implement battery holder. Make sure there are solder pads / through holes on board for this.
    - Circuit for receiver.
2. Website!

Question

- Do we have access to transistor for kitboard testing?
- Do we want to use surface mount transistors for final PCB or add through holes?
- Monitoring batteries and the usage of 4AA discussion