Design Review 1 Agenda

2/25

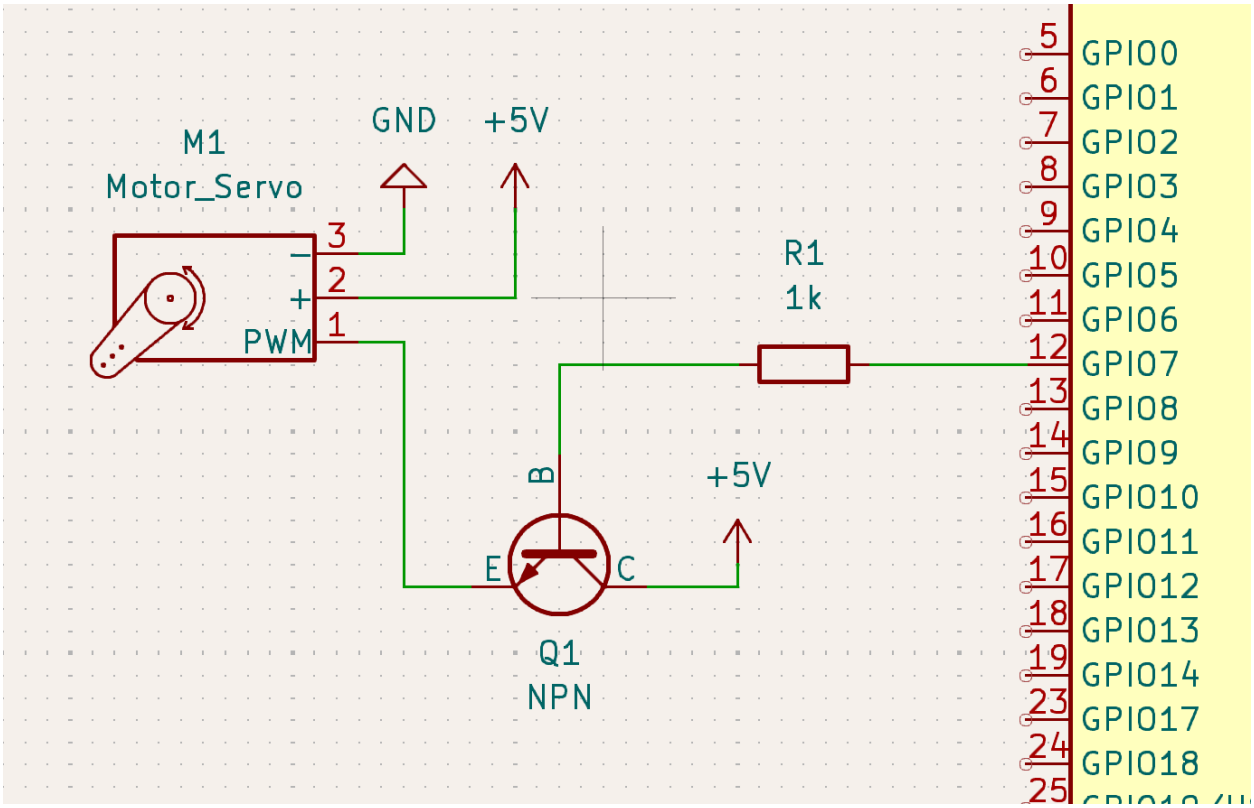
1. Why we chose the ESP32 S3 for our board
    a. Benefits of a Dual Core Processor
    b. PSRAM Compatible
    c. AI integration
    d.
2. Go over each major subsystem
    a. Power
        i. How power is distributed to each other subsystem
    b. Motion
        i. Body Design
        ii. Motors
        iii. Movement
    c. UI
        i. Speaker
        ii. Mic
        iii. Display
        iv. Controller
    d. AI
        i. Creating a live feed
3. How we are going to demonstrate each subsystem

ESP32 S3

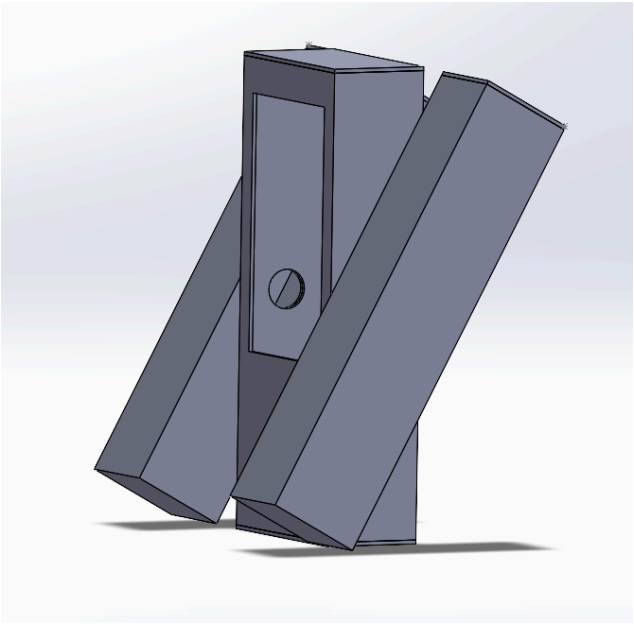
- Wifi and bluetooth connectivity
    - The Remote will require bluetooth
    - To use the AI API, will require the connection to WIFI
- AI Acceleration Support
    - Considering the use of AI in our project, accelerated support is welcome
- I2C and SPI
    - The mic, speaker, and display all require the use of either I2C or SPI
- Dual Core Processor
    - Essential for the many tasks we want to run

Major Subsystem/Major Components

Motion: Xander and McGarrity



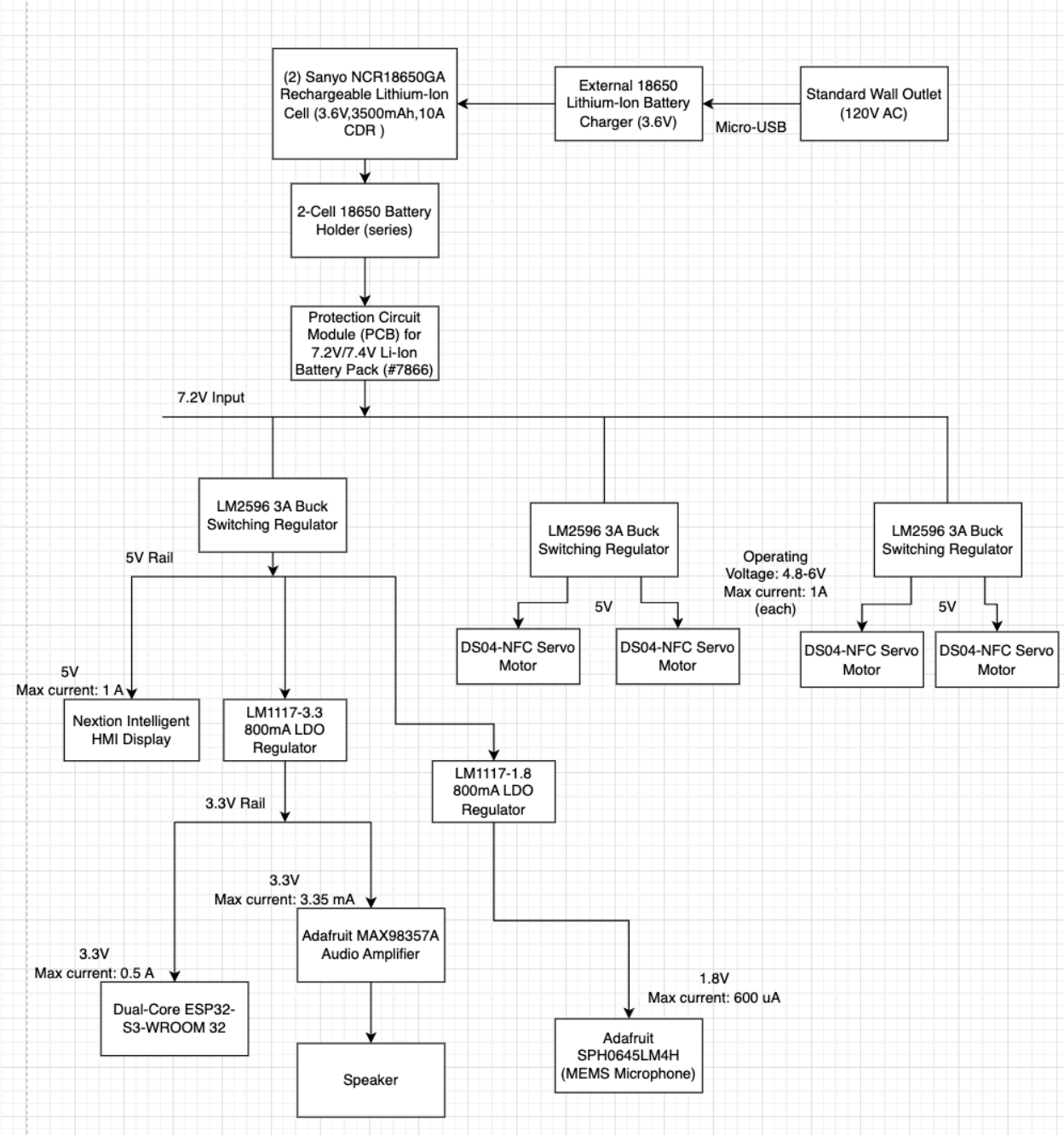
- Control of Motor through GPIO pin and BJT as pictured above.
- Motors: DS04-NFC
- BJT: TPN2222A
- Power Requirements for Motors: 4.8V-6V, Max 1A



- First Frame Design: Sent to EIH for Printing
- Crafting Frame with Cardboard to test

Power: Jack Corrao

- Subsystem Block Diagram:



- Battery: Sanyo NCR18650GA 3500mAh 10A Battery (Quantity 2)
    - Need at least 5V, CDR (constant discharge rate) of at least 5.5A and relatively large capacity >3000mAh
        - Voltage: 7.2V (2 cell in series)
        - Capacity: 3500mAh
        - Constant Discharge: 10A
    - Relatively cheap ($5.99)

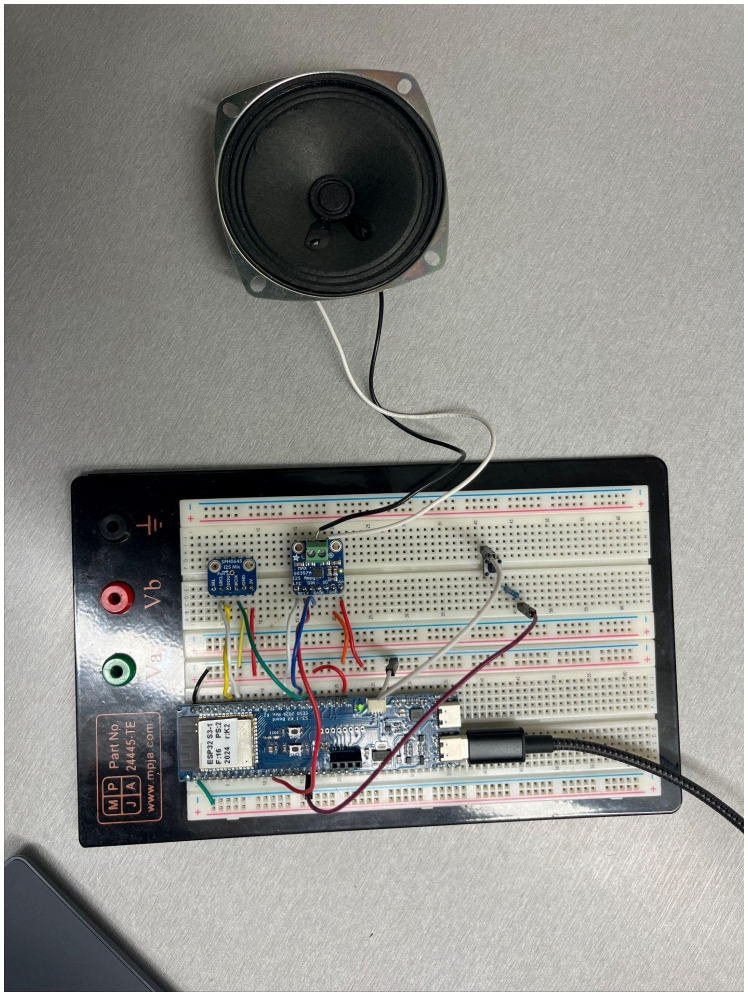
    - Easy to charge with relatively common external Li-Ion 18650 chargers and can plug into 18650 battery holders like the Victagen one in 205
    - Small size, shouldn't take up too much space or add a lot of weight
    - Con: No over discharging voltage/current protection, need an external PCM PCB to regulate over discharging of the battery or else design one ourselves
- Charger: External 18650 Lithium-Ion Battery Charger (3.6V)
    - Easy to use/safe
    - Has overcharge protection
    - May be able to use one like the Victagen one already in 205
- 2-Cell 18650 Battery Holder
    - Allow us to use two 3.6V Li-Ion cells in series to get 7.4V so we can power our 5V peripherals.
- Battery Protection Circuitry: Protection Circuit Module (PCB) for 7.2V/7.4V Li-Ion Battery Pack
    - Prevents overdischarge of the two series cells in the battery holder because the two cells do not have one built in as many battery packs do
- Buck Converters and LDOs
    - LM2596 3A Buck Switching Regulator (5V version) (Quantity 3)
        - Servo Motors: Need Vout = 5V, Iout = >1A
        - Nextion Display: Need Vout = 5V, Iout = >1A
        - More efficient than an LDO which is especially important for high-current applications such as motors and our display to keep dissipated power low.
    - LM1117-3.3 (Quantity: 1)
        - Microcontroller: Need Vout = 3.3V, Iout = >0.5A;
        - Audio Amplifier: Need 2.5V-5.5V, Iout = >3.35mA
        - Can handle 800mA output current, drops voltage from 5V to 3.3V for the microcontroller and audio amp
        - Acceptable dropout voltage (between 1.1V and 1.3V)
    - LM1117-1.8 (Quantity 1)
        - Microphone: Need Vout = 1.8V, Iout = > 600 uA
        - Can handle 800mA output current (far more than needed here), drops voltage from 5V to 1.8V for the MEMS microphone.
        - Acceptable dropout voltage (between 1.1V and 1.3V)

AI: Garrett and Matthew

- Connecting to the ESP32 over WIFI
- Streaming real time data to the computer
- Using ChatGPT API
- Displaying the response

UI: Garrett and Matthew

- Speaker Amp: Max 98375A
    - Easy to integrate and use
    - Cheap
    - Loud enough for our project
    - Working code
    - I2S
    - 3.3V
- Microphone: MEMS 321
    - Ease of implementation
    - Working code
    - I2S
    - 3.3V
- Display
    - 7 inch display provided by schafer
    - SPI
- Playstation Controller
    - Connect with bluetooth
    - Controls movement
    - Easily accessible



Demonstration for Design Review 2

1. **Demonstration/Prototyping of Motion Subsystem:**
    - Use existing kit and development board of ESP32-S3 for demonstration/prototyping of motion subsystem.
    - Verify motors are functional and capable of moving our frame as desired.
    - Verify that motion subsystem and frame design meets previously outlined requirements
2. **Demonstrating AI and UI**
    - Integrate entire workflow of listening → recording → streaming to server → receiving and streaming response from ChatGPT
    - Demonstrate the readiness of the Microphone to “listen” and understand speech
        - Display the words that are processed on the LED
    - Relay the response on the Display and over the speaker
    - Show the connectivity of the controller to our ESP

3. **Demonstration/Prototyping of Power Subsystem:**
    - Convince Professor Schafer that this will work and not start a fire