Dwight Look College of ENGINEERING TEXAS A&M UNIVERSITY

Team 28: Guitar Entertainment System

Bi-Weekly Update 3

Rishabh Ruikar, Monte Martin III, Rawan Ibraheem

Sponsor: Souryendu Das (TA)

Project Summary

- Problem Statement: Current guitar amplifiers and effects systems often present a steep learning curve, deterring those with limited technical experience from fully exploring their sound potential.

- Solution proposal: Developing a user-centric guitar sound system, combining an amp, pedals, and a Bluetooth-connected app. This system simplifies sound customization through intuitive controls and presets, making advanced sound manipulation accessible to all skill levels.

Project/Subsystem Overview

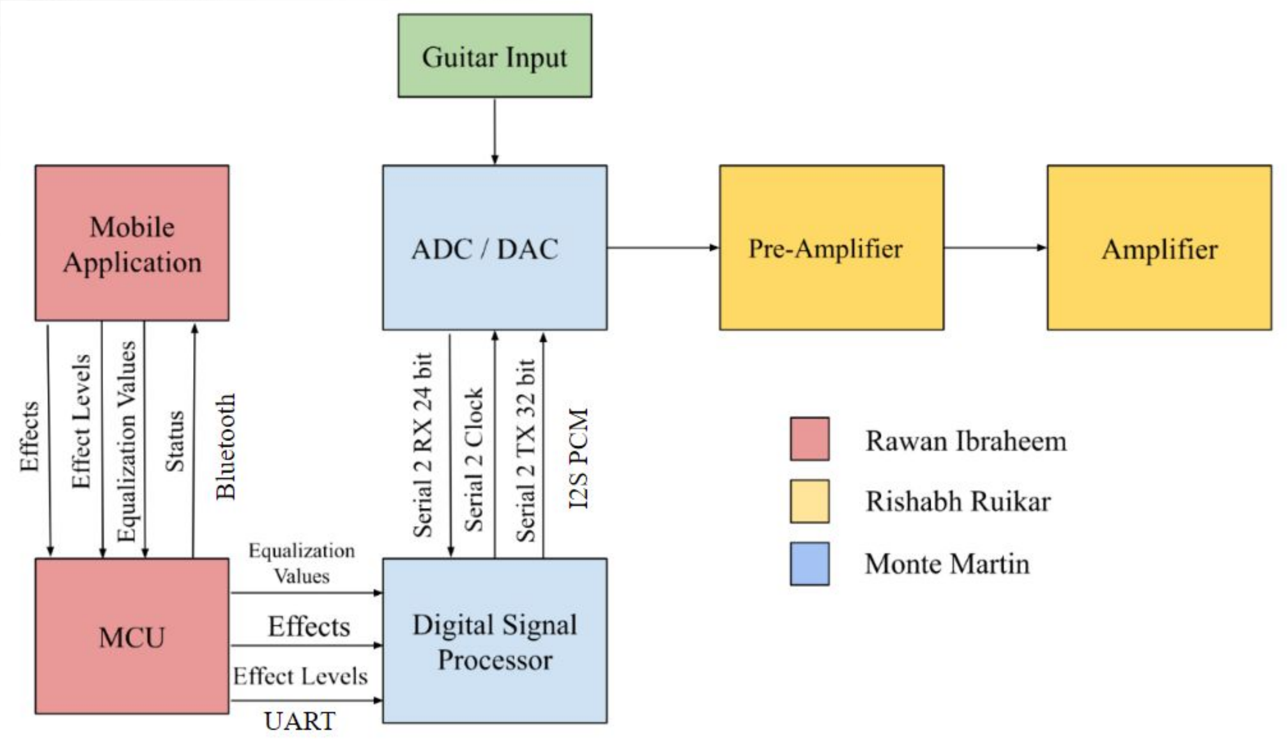


Figure 1: Subsystem overview.

Project Timeline

Subsystem Redesign (Amplifier; completed 1/26)	Pedal System tweaks (completed on 2/6)	Subsystem Redesign (App and Pedal System; to be complete by 2/20)	System integration (to be complete by 3/15)	Finish Housing (to be complete by 4/1)	Validation (to be complete by 4/10)

Subsystem 1: Amplifier

Rishabh

Accomplishments since last update 15 hrs of effort	Ongoing progress/problems and plans until the next presentation
● Preamplifier PCB has almost been soldered (had some parts issues) ● Housing design has reached a “rough draft” state	● Test the preamplifier board (done before lab 2/29) ● Solder and test the amplifier board (done before lab 2/29) ● Both board tests are preparation for pedal integration ● Heavy progress on pedal system integration ● Start drawing out the housing in SolidWorks

Subsystem 1: Amplifier

Rishabh

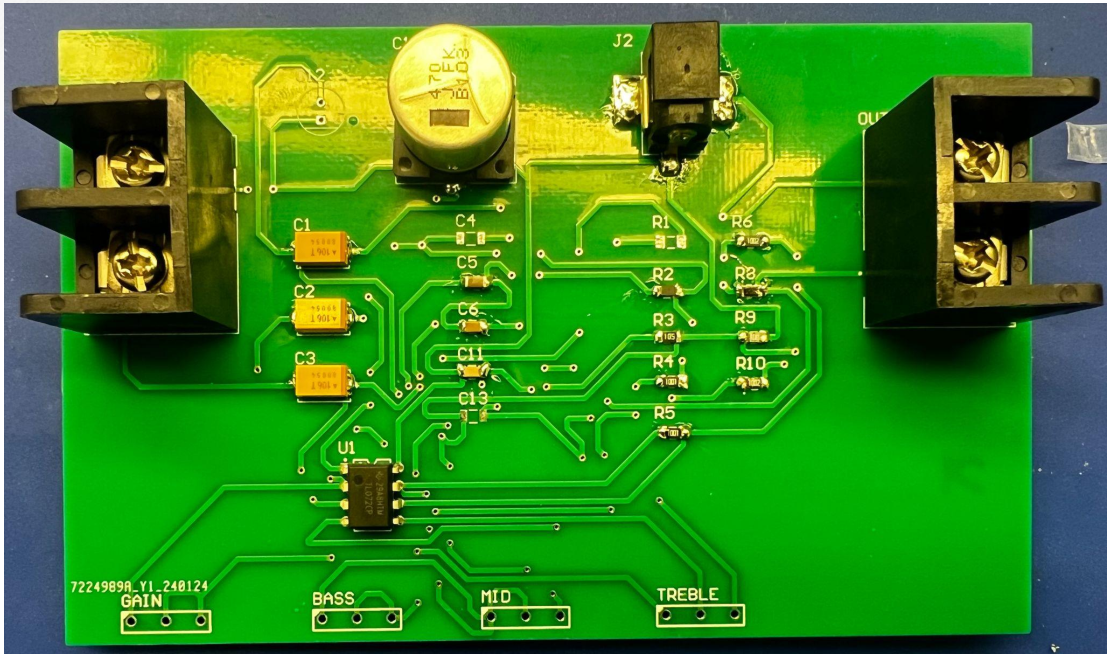


Figure 2: Amplifier PCB

- Preamplifier has been almost soldered (soldering *and* testing will be complete by 2/29)
- Amplifier PCB has encountered delays due to missing parts
- Amplifier PCB will be soldered and testing will be complete by 2/29

Subsystem 2: Pedal System

Monte Martin III

Accomplishments since last update 40 hrs of effort	Ongoing progress/problems and plans until the next presentation
● Finished PCB redesign and ordered PCB and parts, parts are in and PCB should be here this week ● Determined a method to integrate with the app for testing, and a way to validate it	● Assemble and test PCB ● integration of the app and pedal system ● validate the pedal system ● integration with the amplifier and testing stability of the connection under load

Subsystem 2: Pedal System

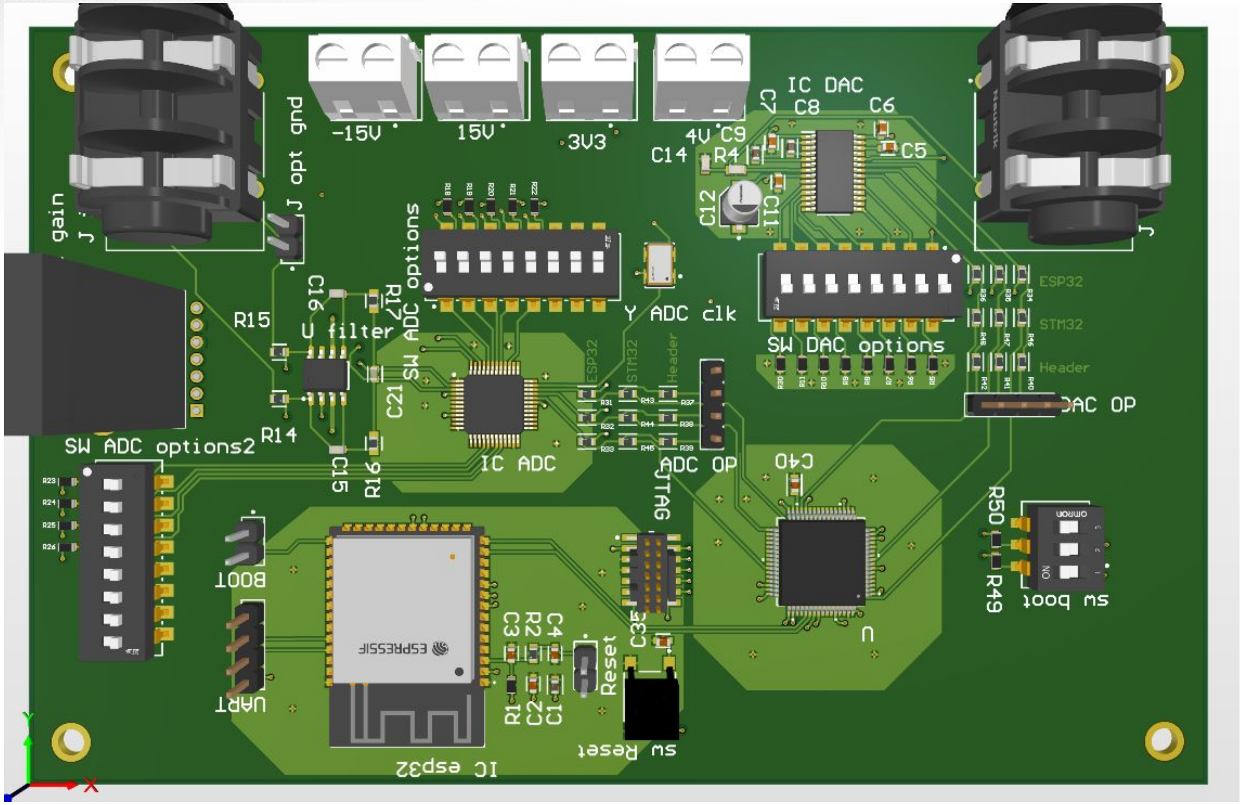


Figure 3: 3D view of Pedal PCB.

Subsystem 2: Pedal System

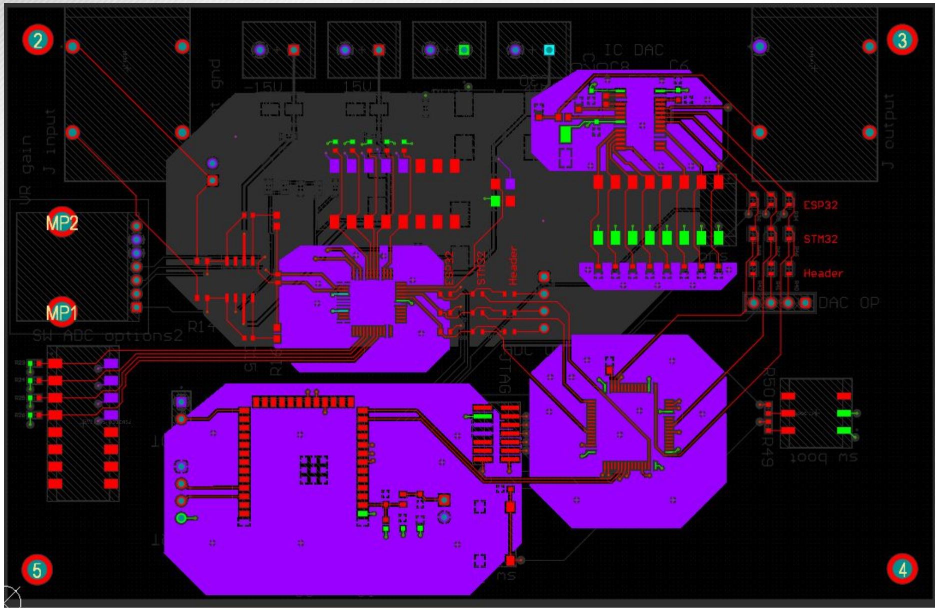


Figure 4: Layer 1 of PCB

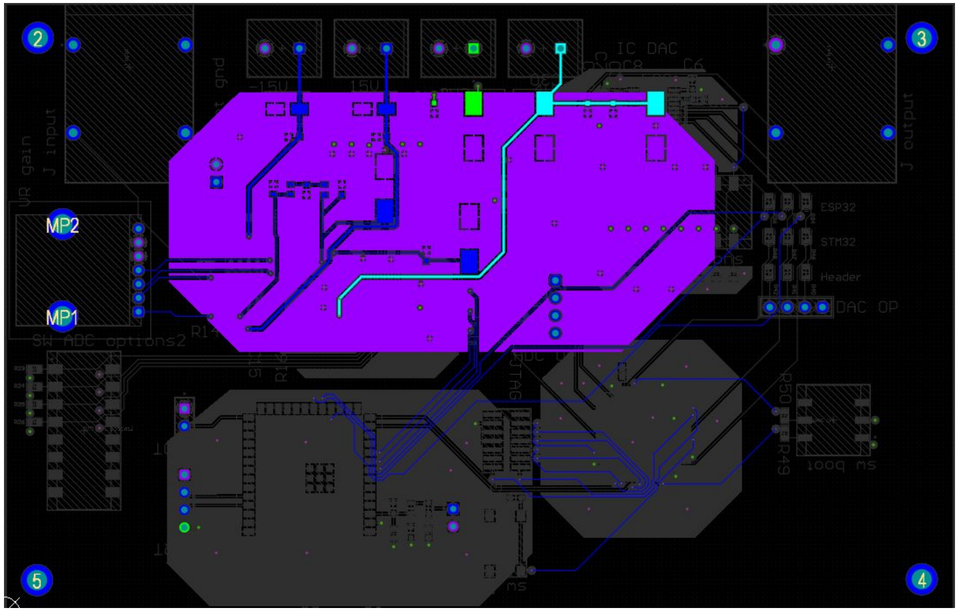


Figure 5: Layer 4 of PCB

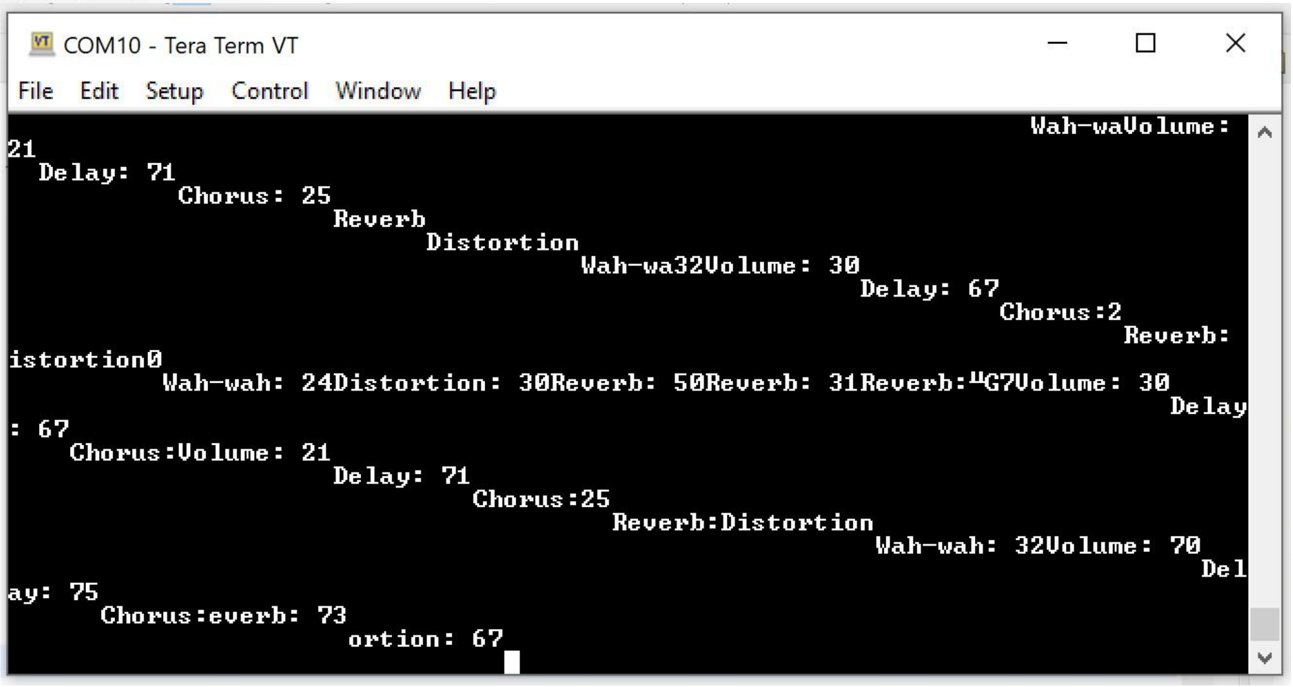
Subsystem 3: Bluetooth App

Rawan Ibraheem

Accomplishments since last update ~ 19 hrs of effort	Ongoing progress/problems and plans until the next presentation
● Completed integration of MCU with DSP ● ESP32 is transmitting an audio signal to STM32 through I2S (to simulate a guitar playing)	● Ongoing implementation of sound effects applied to guitar input signal ● Design housing for all three PCBs, along with the output speaker

Subsystem 3: Bluetooth App

Rawan Ibraheem



```
                                                                  Wah-waVolume:
21
  Delay: 71
           Chorus: 25
                     Reverb
                           Distortion
                                     Wah-wa32Volume: 30
                                                       Delay: 67
                                                                Chorus:2
                                                                        Reverb:
istortion0
          Wah-wah: 24Distortion: 30Reverb: 50Reverb: 31Reverb:4G7Volume: 30
                                                                           Delay
: 67
    Chorus:Volume: 21
                     Delay: 71
                              Chorus:25
                                       Reverb:Distortion
                                                        Wah-wah: 32Volume: 70
                                                                             Del
ay: 75
      Chorus:everb: 73
                      ortion: 67
```

Figure 6: Terminal output of STM32 Nucleo board. This data was sent from the application to the MCU through Bluetooth, then from the MCU to the DSP through UART communication.

System Integration

Accomplishments since last update	Ongoing progress/problems and plans until the next presentation
● Completed integration of MCU and DSP	● Integration and testing of amplifier and pedal PCBs ● Test that STM32 can apply each sound effect (volume, distortion, reverb, chorus, delay, wah-wah)

Integration

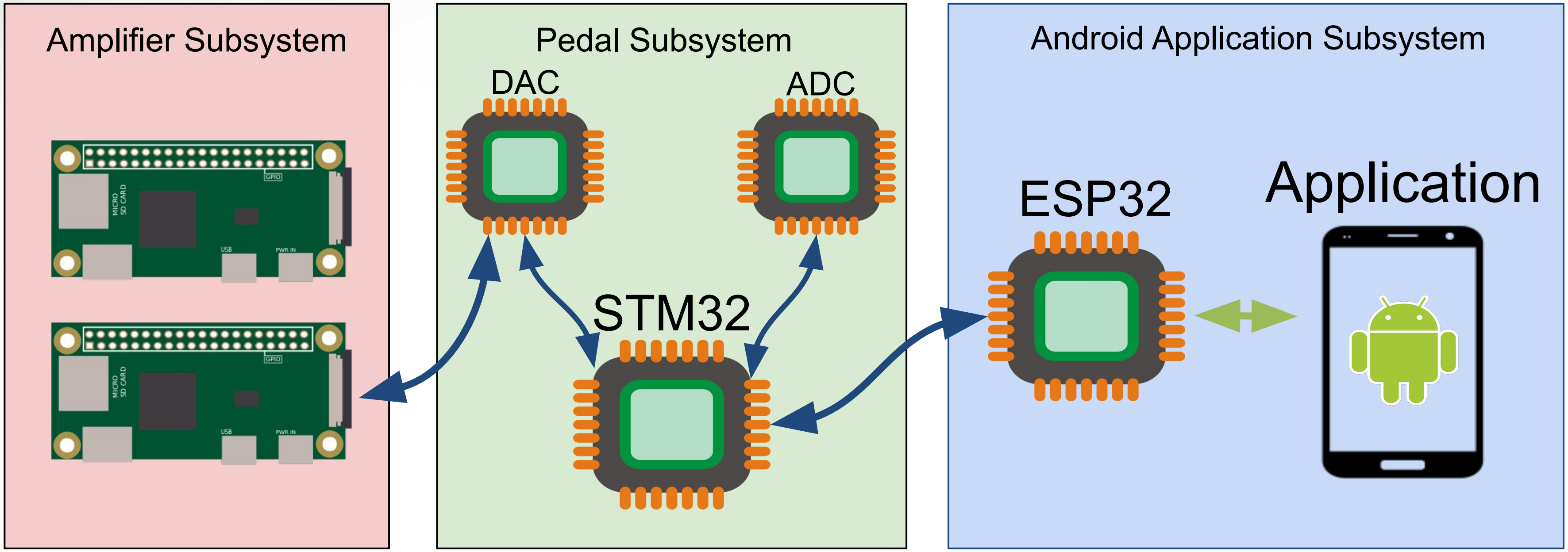
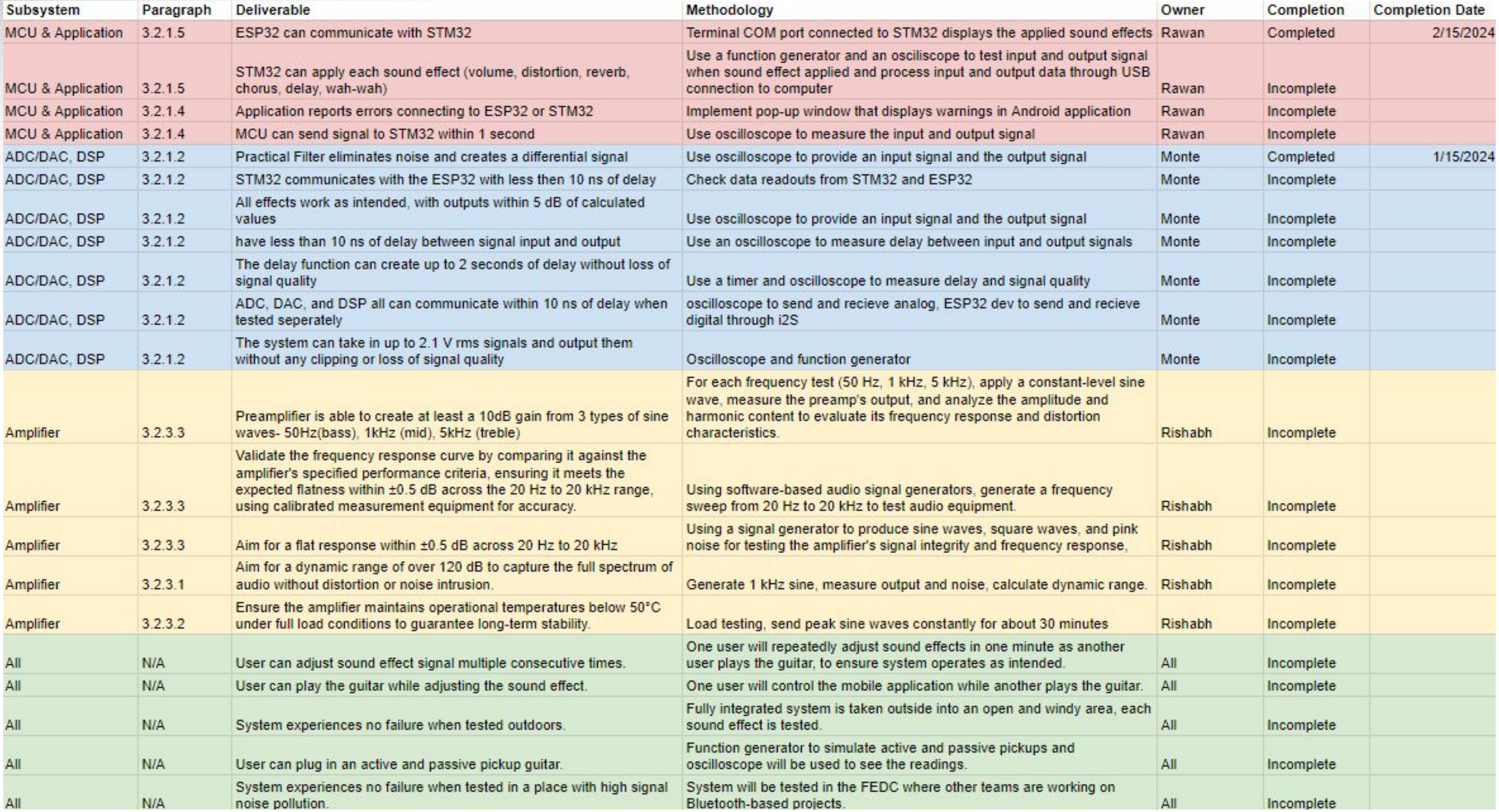


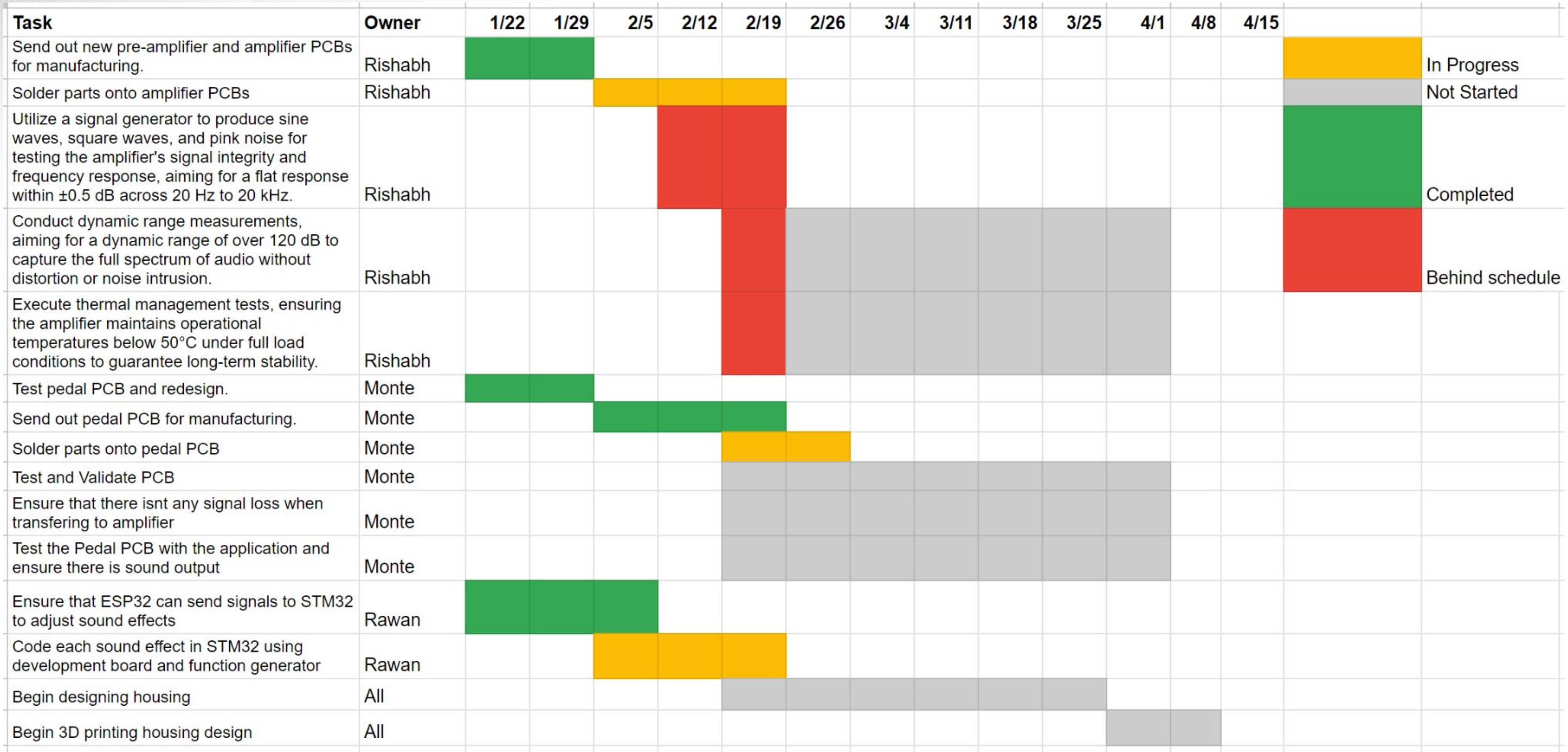
Figure 7: Overview of Integration.

Validation Plan



Subsystem	Paragraph	Deliverable	Methodology	Owner	Completion	Completion Date
MCU & Application	3.2.1.5	ESP32 can communicate with STM32	Terminal COM port connected to STM32 displays the applied sound effects	Rawan	Completed	2/15/2024
MCU & Application	3.2.1.5	STM32 can apply each sound effect (volume, distortion, reverb, chorus, delay, wah-wah)	Use a function generator and an osciliscope to test input and output signal when sound effect applied and process input and output data through USB connection to computer	Rawan	Incomplete	
MCU & Application	3.2.1.4	Application reports errors connecting to ESP32 or STM32	Implement pop-up window that displays warnings in Android application	Rawan	Incomplete	
MCU & Application	3.2.1.4	MCU can send signal to STM32 within 1 second	Use oscilloscope to measure the input and output signal	Rawan	Incomplete	
ADC/DAC, DSP	3.2.1.2	Practical Filter eliminates noise and creates a differential signal	Use oscilloscope to provide an input signal and the output signal	Monte	Completed	1/15/2024
ADC/DAC, DSP	3.2.1.2	STM32 communicates with the ESP32 with less then 10 ns of delay	Check data readouts from STM32 and ESP32	Monte	Incomplete	
ADC/DAC, DSP	3.2.1.2	All effects work as intended, with outputs within 5 dB of calculated values	Use oscilloscope to provide an input signal and the output signal	Monte	Incomplete	
ADC/DAC, DSP	3.2.1.2	have less than 10 ns of delay between signal input and output	Use an oscilloscope to measure delay between input and output signals	Monte	Incomplete	
ADC/DAC, DSP	3.2.1.2	The delay function can create up to 2 seconds of delay without loss of signal quality	Use a timer and oscilloscope to measure delay and signal quality	Monte	Incomplete	
ADC/DAC, DSP	3.2.1.2	ADC, DAC, and DSP all can communicate within 10 ns of delay when tested seperately	oscilloscope to send and recieve analog, ESP32 dev to send and recieve digital through i2S	Monte	Incomplete	
ADC/DAC, DSP	3.2.1.2	The system can take in up to 2.1 V rms signals and output them without any clipping or loss of signal quality	Oscilloscope and function generator	Monte	Incomplete	
Amplifier	3.2.3.3	Preamplifier is able to create at least a 10dB gain from 3 types of sine waves- 50Hz(bass), 1kHz (mid), 5kHz (treble)	For each frequency test (50 Hz, 1 kHz, 5 kHz), apply a constant-level sine wave, measure the preamp's output, and analyze the amplitude and harmonic content to evaluate its frequency response and distortion characteristics.	Rishabh	Incomplete	
Amplifier	3.2.3.3	Validate the frequency response curve by comparing it against the amplifier's specified performance criteria, ensuring it meets the expected flatness within ±0.5 dB across the 20 Hz to 20 kHz range, using calibrated measurement equipment for accuracy.	Using software-based audio signal generators, generate a frequency sweep from 20 Hz to 20 kHz to test audio equipment.	Rishabh	Incomplete	
Amplifier	3.2.3.3	Aim for a flat response within ±0.5 dB across 20 Hz to 20 kHz	Using a signal generator to produce sine waves, square waves, and pink noise for testing the amplifier's signal integrity and frequency response,	Rishabh	Incomplete	
Amplifier	3.2.3.1	Aim for a dynamic range of over 120 dB to capture the full spectrum of audio without distortion or noise intrusion.	Generate 1 kHz sine, measure output and noise, calculate dynamic range.	Rishabh	Incomplete	
Amplifier	3.2.3.2	Ensure the amplifier maintains operational temperatures below 50°C under full load conditions to guarantee long-term stability.	Load testing, send peak sine waves constantly for about 30 minutes	Rishabh	Incomplete	
All	N/A	User can adjust sound effect signal multiple consecutive times.	One user will repeatedly adjust sound effects in one minute as another user plays the guitar, to ensure system operates as intended.	All	Incomplete	
All	N/A	User can play the guitar while adjusting the sound effect.	One user will control the mobile application while another plays the guitar.	All	Incomplete	
All	N/A	System experiences no failure when tested outdoors.	Fully integrated system is taken outside into an open and windy area, each sound effect is tested.	All	Incomplete	
All	N/A	User can plug in an active and passive pickup guitar.	Function generator to simulate active and passive pickups and oscilloscope will be used to see the readings.	All	Incomplete	
All	N/A	System experiences no failure when tested in a place with high signal noise pollution.	System will be tested in the FEDC where other teams are working on Bluetooth-based projects.	All	Incomplete	

Execution Plan



Task	Owner	1/22	1/29	2/5	2/12	2/19	2/26	3/4	3/11	3/18	3/25	4/1	4/8	4/15		
Send out new pre-amplifier and amplifier PCBs for manufacturing.	Rishabh	🟩	🟩												🟨	In Progress
Solder parts onto amplifier PCBs	Rishabh			🟨	🟨	🟨									▨	Not Started
Utilize a signal generator to produce sine waves, square waves, and pink noise for testing the amplifier's signal integrity and frequency response, aiming for a flat response within ±0.5 dB across 20 Hz to 20 kHz.	Rishabh				🟥	🟥									🟩	Completed
Conduct dynamic range measurements, aiming for a dynamic range of over 120 dB to capture the full spectrum of audio without distortion or noise intrusion.	Rishabh					🟥	▨	▨	▨	▨	▨				🟥	Behind schedule
Execute thermal management tests, ensuring the amplifier maintains operational temperatures below 50°C under full load conditions to guarantee long-term stability.	Rishabh					🟥	▨	▨	▨	▨	▨					
Test pedal PCB and redesign.	Monte	🟩	🟩													
Send out pedal PCB for manufacturing.	Monte			🟩	🟩	🟩										
Solder parts onto pedal PCB	Monte					🟨	🟨									
Test and Validate PCB	Monte					▨	▨	▨	▨	▨	▨					
Ensure that there isnt any signal loss when transfering to amplifier	Monte					▨	▨	▨	▨	▨	▨					
Test the Pedal PCB with the application and ensure there is sound output	Monte					▨	▨	▨	▨	▨	▨					
Ensure that ESP32 can send signals to STM32 to adjust sound effects	Rawan	🟩	🟩	🟩												
Code each sound effect in STM32 using development board and function generator	Rawan			🟨	🟨	🟨										
Begin designing housing	All					▨	▨	▨	▨	▨						
Begin 3D printing housing design	All											▨	▨			

Thank you for listening.