SAN DIEGO STATE UNIVERSITY

Electrical and Computer Engineering
Senior Design (EE491W)

Risk Management and Prototyping Plan
Team 9 - HANDS-EMG

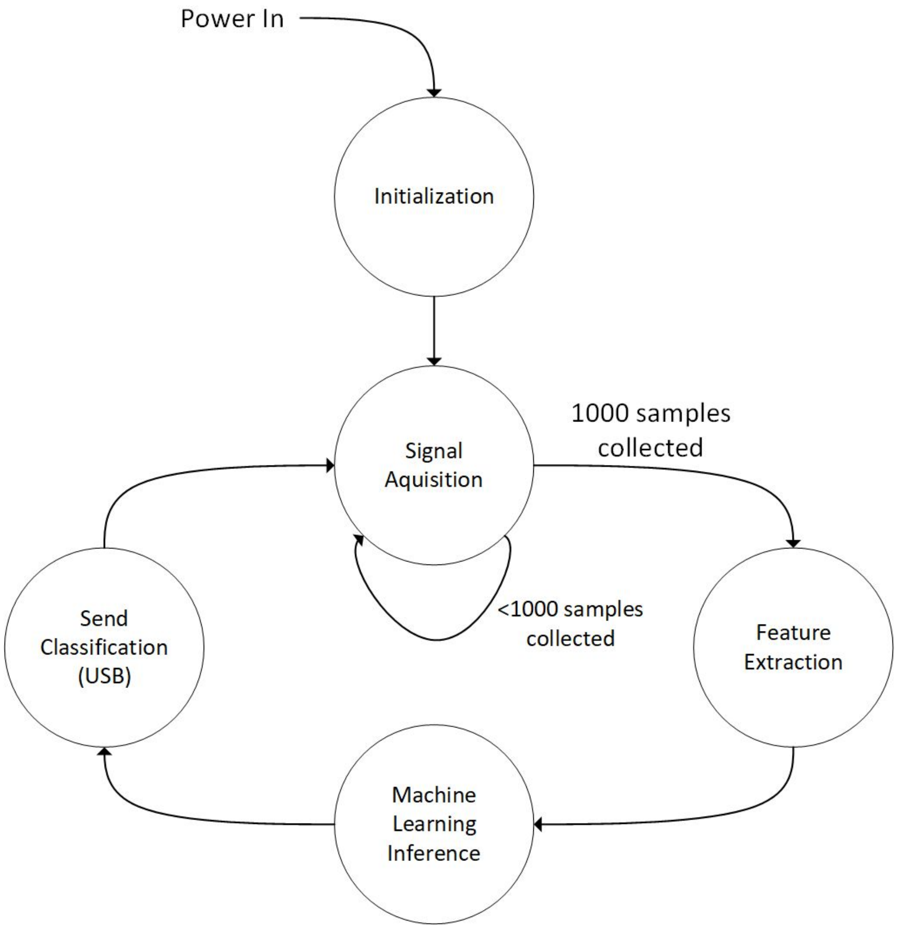
Hand **A**ctivity **N**eural **D**etection using **sEMG**

System Description

Our ECE team is developing a battery-powered surface electromyography (sEMG) sensor system designed to classify hand movements based on the wearer's muscle activity. The device utilizes four channels of wet electrodes placed on the user's forearm to capture sEMG signals. These signals are processed through our procured analog front-end module operating at a minimum of 2000 samples per second. The digitized signals are transmitted to our microcontroller via SPI, where a machine learning algorithm classifies the hand movements based on a pre-trained model. Our responsibilities include configuring the analog front-end module, programming a low-power microcontroller, integrating the system on a custom PCB, and training the machine learning model for accurate classification. The system will output the movement classifications via USB, which can be visualized on a PC simulator. Designed for efficiency, the device operates at a voltage of 3.7V provided by a rechargeable lithium polymer battery for portability.

Finite State Diagram

The finite state diagram figure shows the main processes that our device performs when powered on.



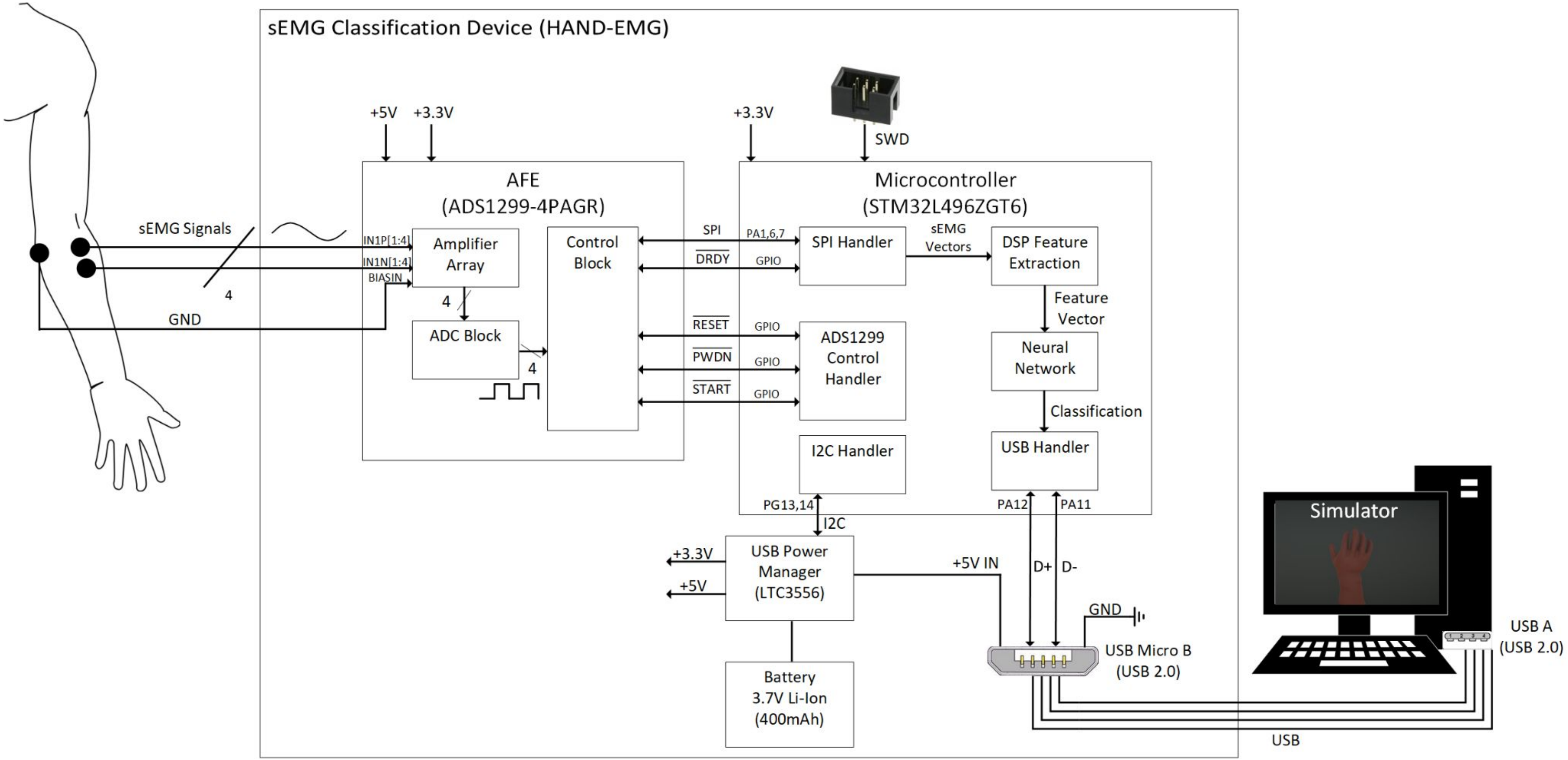

sEMG Classification Device (HAND-EMG)
sEMG Signals
4
GND
+5V
+3.3V
AFE
(ADS1299-4PAGR)
IN1P[1:4]
IN1N[1:4]
BIASIN
Amplifier Array
4
ADC Block
4
Control Block
SPI
DRDY
RESET
PWDN
START
+3.3V
SWD
Microcontroller
(STM32L496ZGT6)
PA1,6,7
GPIO
GPIO
GPIO
GPIO
SPI Handler
sEMG Vectors
DSP Feature Extraction
Feature Vector
ADS1299 Control Handler
Neural Network
Classification
I2C Handler
USB Handler
PG13,14
I2C
PA12
PA11
+3.3V
+5V
USB Power Manager (LTC3556)
+5V IN
D+
D-
GND
Battery 3.7V Li-Ion (400mAh)
USB Micro B (USB 2.0)
Simulator
USB A (USB 2.0)
USB

Figure 2

Software Layout

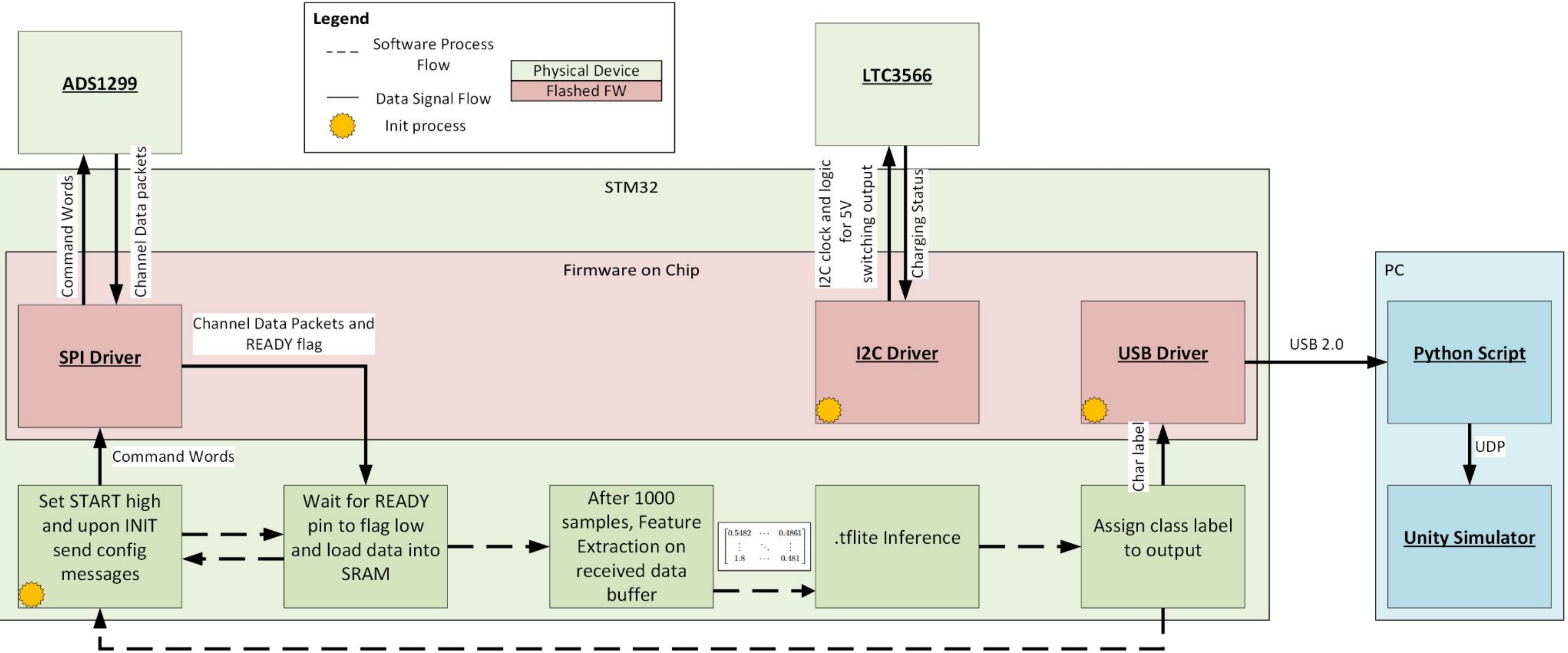


Figure 3

Project Risks

Risks that affected project definition

	Risk	Description	Project Change
1	Use of dry electrodes	Dry electrodes presented impractical complexity due to the added impedance and noise. In our analysis there was also a lack of data for dry electrode based classification.	Project was changed to use wet electrodes. These remove the complexity faced with dry electrodes and have a ubiquitous amount of public datasets for classification models.
2	Time to design custom power management system	Designing a custom battery charger and regulation circuit was impractical due to the extensive time required.	Project was changed to procure a USB power management IC.
3	Research and design for a custom analog front-end module	Designing an analog front-end that would amplify, filter, and sample our emg signals was impractical due to the engineering complexity.	Project was changed to procure an analog front-end module designed for biomedical purposes that handle the listed activities.

Summary of low/non-risk items

	Risk	Reason
1	Power Management	Providing power to our ICs via a 3.7V rechargeable li-ion battery is well documented for both the STM32 and ADS1299 including implementation examples in their respective datasheets. USB power manager (LTC3556) has been selected as our solution.
2	Communication Protocols	The SPI, I2C, and USB communication protocols are well defined and documented. Communication is handled by the STM32's hardware modules for simplicity. Team member has experience with SPI and I2C.
3	Enclosure	A plastic enclosure will be procured from DigiKey and holes will be drilled for external interfaces.
4	Public Data Availability	Team has identified public database "Ninapro" with 10 datasets including over 180 data acquisitions. Team has identified other public datasets as backup.
5	Deploying Machine Learning on STM32	Deploying a TFLite Micro model to STM32 is well documented with open source projects and internal STMicroelectronic documentation available. The documentation includes over 5 examples, a walkthrough, and troubleshooting information.

Risk Cube

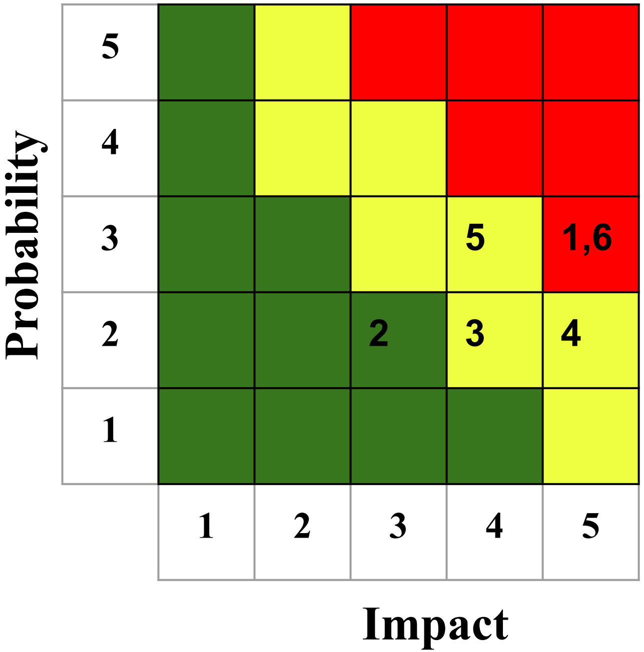


Figure 4

	Risk	Probability	Impact	Mitigation
1	ML Classification Accuracy	3	5	Testing will take place with public data and AFE acquired data.
2	ML Model Size	2	3	ML model will be adjusted incrementally beginning in phase 1.
3	DSP Algorithms	2	4	Close communication with project mentor. Algorithms are already being prototyped in C.
4	AFE Configuration	2	5	Signal acquisition testing begins early in phase 1 with all materials ordered. Allowing team 4 weeks of prototyping.
5	PCB Design Complexity	3	4	Three separate PCBs will be designed for each module. Then integrated into a single board after debugging.
6	System Integration	3	5	Modules will be tested individually with debugging tools to ensure proper technical characteristics before integration.

Risk Table

	Risk	Description	Probability	Impact
1	ML Classification Accuracy	An accurate machine learning model is necessary to make consistent classifications of movements. Standard accuracy of between 70% and 90% is necessary.	Datasets with over 180 acquisitions specifically for this application were identified. Analysis of similar models coupled with a considerable amount of data show this accuracy is possible. The probability is estimated as 3.	A system with a low accuracy model will often provide incorrect classifications. A completely inaccurate model will not provide any classifications. The impact is estimated to be a 5.

Risk Table

	Risk	Description	Probability	Impact
2	ML Model Size	Machine learning models can be computationally expensive and large in size. Deploying this on a microcontroller with limited memory is a risk factor.	The STM32L4 we selected has 1MB of flash. Analysis of literature and existing projects show EMG classification modules operating on 1MB devices. The STM32U5 with 4MB has been identified as an alternate option with the same pinout. The probability is a 2.	Model complexity is an important factor to its size. If model size presents itself to be an issue, then a new model will be trained with less gestures to classify. The impact is a 3.

Risk Table

	Risk	Description	Probability	Impact
3	DSP Algorithms	The digital output of the ADS1299 needs to be processed prior to feature extraction to ensure that captured data matches training data, if the features extracted are not within an acceptable range, the inferences will be inaccurate	The features being used are non-intensive, and upon analysis, the datasets being used are in an easy to modify format. STM also runs subroutines to dedicate majority processing power to feature extraction. Additionally, the literature backs up our specified features as an industry standard. Probability is a 2.	If the signals are not properly conditioned, or the MCU is unable to perform feature extraction as on our PC, the machine learning model performance will suffer, thus the impact is a 4.

Risk Table

	Risk	Description	Probability	Impact
4	AFE Configuration	The ADS1299 module has many configuration options including amplifier gain and sampling frequency. The proper combination of these options is necessary to receive a clean sEMG signal for classification.	Existing research for the use of the ADS1299 for EMG has been identified. The ADS1299 was selected at the advice of our project advisor who has used it previously for similar applications. For this reason the probability is a 2.	Our project relies heavily on proper signal acquisition. Without proper signals, the microcontroller will fail to classify the movement. The impact is a 5.

Risk Table

	Risk	Description	Probability	Impact
5	PCB Design Complexity	Currently, the proposed system utilizes three ICs with a combined chip area of 564mm^2, and a total number of 236 pins. This increases the complexity needed for the PCB layout substantially	Each devices data sheets provide examples of schematic setups for implementing on a custom PCB. Breaking up the PCB design by component, we can more easily finalize. Our eval boards will give us hands on knowledge of the boards and necessary pinouts. Probability is 3	If design of PCBs is not done properly, there could be problems with noise or signal degradation, which would affect the data received/outputted giving inaccurate results. Impact is 4.

Risk Table

	Risk	Description	Probability	Impact
6	System Integration	Our proposed system has 3 major components. Power management, analog front-end, and the microcontroller which all have different control and communication protocols.	While creating each PCB, there will be thorough testing and debugging for each system to ensure they work individually before full integration. Also, the communication modules will have a working prototype Probability is 3	Since we are on putting all three systems on one PCB, if the subsystems are unable to communicate properly, or the physical layout was done incorrectly, then the total system would fail. Impact is 5

Prototyping Plan

Prototyping Roadmap

Note the bulk of prototyping activities are present in phase 1 & 2, as most tasks from phase 3 onward involve considerable debug effort

Phase 1: Isolated Subsystem Prototyping	Phase 2: Subsystem Integration Pairing	Phase 3: Total Subsystem Integration	Phase 4: Subsystem Integration on PCB	Phase 5: Total System Integration on PCB
Tasks: **1.** ADS Eval kit is configured to find proper biasing of amplifier to match training data **2.** STM32 FSM subroutines are designed and tested **3.** Power delivery tested on breadboard with DC bench supply with voltage swing **4.** Machine learning architecture is created and tested with existing dataset **5.** SPI comms established & verified by logic analyzer	**Tasks:** **1.** ADS and STM32 SPI comm. Established **2.** ADS and STM32 powered with simulated computational load from power delivery circuit **3.** Machine learning model (.tflite) uploaded to STM32 aiming to classify training data gestures via USB 2.0 **4.** Simulator demonstrates proper articulation of poses **5.** DSP algorithms developed to match preexisting data	**Tasks:** **1.** Breadboard power system supplies to both eval kits, EMG measurements passed to STM, inference passed to simulator **2.** Every subsystem is now connected as demonstrated within the main block diagram	**Tasks:** **1.** Development of PCB for each individual subsystem, with their respective I/O, or ESD considerations accounted for. Same system setup as phase 3 **2.** Multi-layer boards for high pin IC packages **3.** Heat dissipation assemblies for packages past ambient temperature dissipation threshold	**Tasks:** **1.** Total integration of all subsystems on one PCB **2.** PCB subsystems routed with considerations for additional noise and heat **3.** Added layers
Risks Addressed: 1 and 4	**Risks Addressed:** 2,3 w/ stretch goal 5	**Risks Addressed:** 6 w/ stretch goal of 5	**Risks Addressed:** 5 and 6	**Risks Addressed:** 5 and 6

10/24 - 10/24	10/24 - 11/24	11/24 - 12/24	12/24 - 2/25	2/25 - 4/25
Phase 1	Phase 2	Phase 3	Phase 4	Phase 5

Phase 1 Description

Our prototype activities in phase 1 will target Risk #1 (ML Classification Accuracy) as well as Risk #4 (AFE Configuration). These risks were targets for initial prototyping in phase 1 because of their high impact on the project. The prototyping activity will consist of coding our subroutines on the STM to ensure proper functioning, building and training our ML model to ensure viable accuracy, finding the correct biasing of the PGA, configuring our PMIC to properly output the desired voltages, and verifying the SPI communication follows standard protocols. The devices and tools needed to complete this phase have already been procured. The activity will involve programming the MCU and the ADS using their prebuilt GUI, identifying proper protocols in both sets of SPI interfaces, and finally measuring test points along the QFN-28 to DIP-28 connector.

Phase 1 Activity Steps

1. ADS Configuration
    a. Place electrodes on arm as described in “Electrode Placement Protocol” and connect them to the eval board
    b. Connect the eval kit to the PC, and open the TI ADS1299EEGFE Software
    c. Adjust the gain of the channel preamplifiers to match the frequency content and amplitude of the acquired signals to that of the training data
2. STM32 Subroutine
    a. Connect the STM32 board to the PC
    b. Open the CubeIDE software to program the STM32
    c. Design and debug subroutines to follow the functionality of our block diagram
3. LTC3556 Verification
    a. Solder the QFN-28 LTC3556 onto the DIP-28 converter (designed item) with header pins
    b. Plug into breadboard
    c. Locate the pin labels on the converter, and align them to match the input voltage
4. Machine Learning Model Construction
    a. Use preexisting TensorFlow Lite libraries to construct a ML model that the STM is capable of utilizing
    b. In Matlab, format the data as it will be done on the MCU, including the windowing size, method, and actual features extracted from each window
    c. Separate data into a 70/20/10 split for training, validation, and testing data
    d. Train the model, adjusting hyperparameters as needed
5. SPI Communication Establishment
    a. Utilizing the firmware packages on the STM32, send some example messages via SPI to a logic analyzer
        i. Tune the device if unexpected messages are occurring (likely timing errors within the code)
    b. While capturing data, pulse the START pin high and visualize the output of the SPI lines along the logic analyzer

Phase 1 | Phase 2 | Phase 3 | Phase 4 | Phase 5

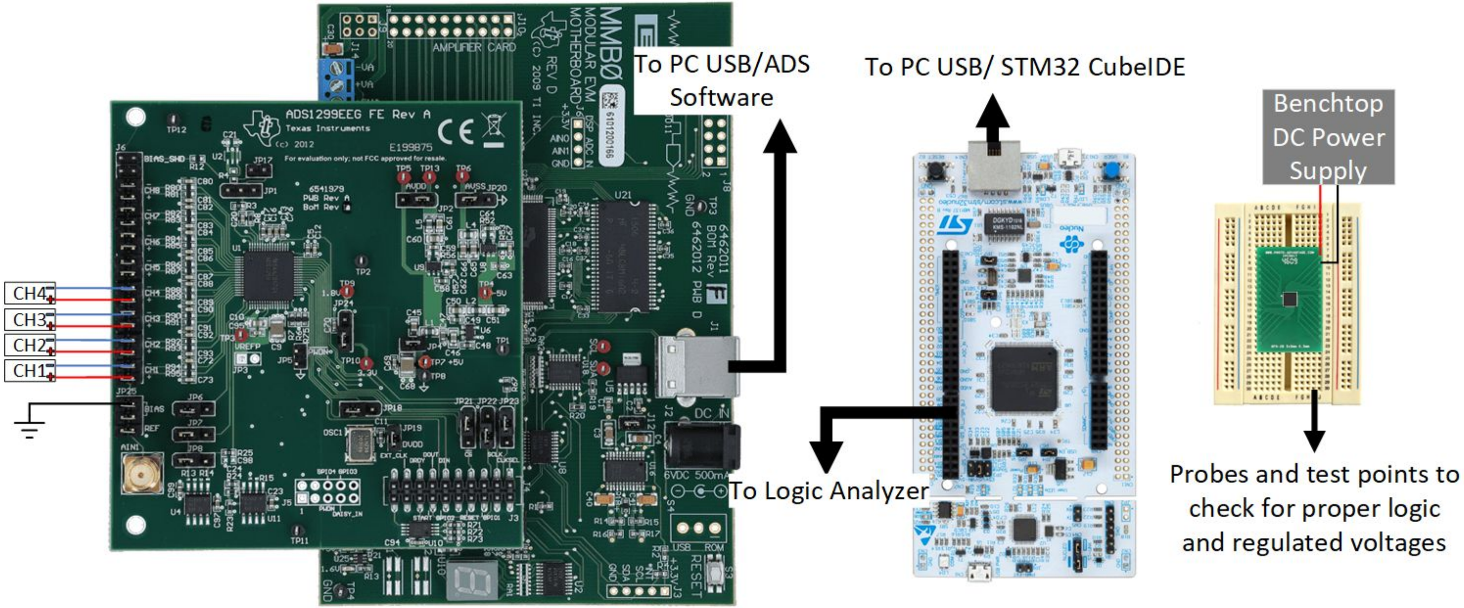


Figure 6

Figure 5 shows our isolated subsystem prototyping. Connections are simplified for readability

Phase 2 Description

Our prototype activity in phase 2 will target Risk #2 (ML Model Size), and Risk #3 (DSP Algorithms) with a stretch goal of Risk #5 (PCB Design Complexity). These risks were targets for early prototyping in phase 2 because of their high impact on the project. The prototyping activity will consist of writing the necessary code to establish communication between the ADS and STM32, powering both devices with our power delivery circuit with a simulated computational load, uploading the trained machine learning model to the STM32, and developing the necessary DSP algorithms to produce features. The devices necessary to conduct this prototyping phase have already been procured. The activity itself will involve capturing the communication between the STM32 and ADS using the logic analyzer to visualize the waveforms on matlab, and feeding the training data to the uC via USB and viewing the classification that is outputted.

Phase 2 Activity Steps

1. Power Management Verification
   a. Connect power management circuit to USB Micro B power delivery.
   b. Connect +3.3V output to STM32 Vin.
   c. Connect +5V output to ADS1299 Vin.
   d. Verify both boards constant power and are able to connect to PC for programming.
2. ADS1299 and STM32 Communication
   a. Write SPI communication code for STM32
   b. Connect ADS1299 SPI lines to STM32 SPI lines.
   c. Use ADALM2000/AD3 to verify SPI communication occurring.
   d. Use ADALM2000/AD3 script to capture SPI packets and visualize sEMG signal sent.
3. PC Simulator Verification
   a. Open the open source Unity simulator on PC.
   b. Create python script that will send fixed test poses to simulator via UDP socket.
   c. Verify simulator displays the poses in the sent test sequence.

Phase 1 | Phase 2 | Phase 3 | Phase 4 | Phase 5

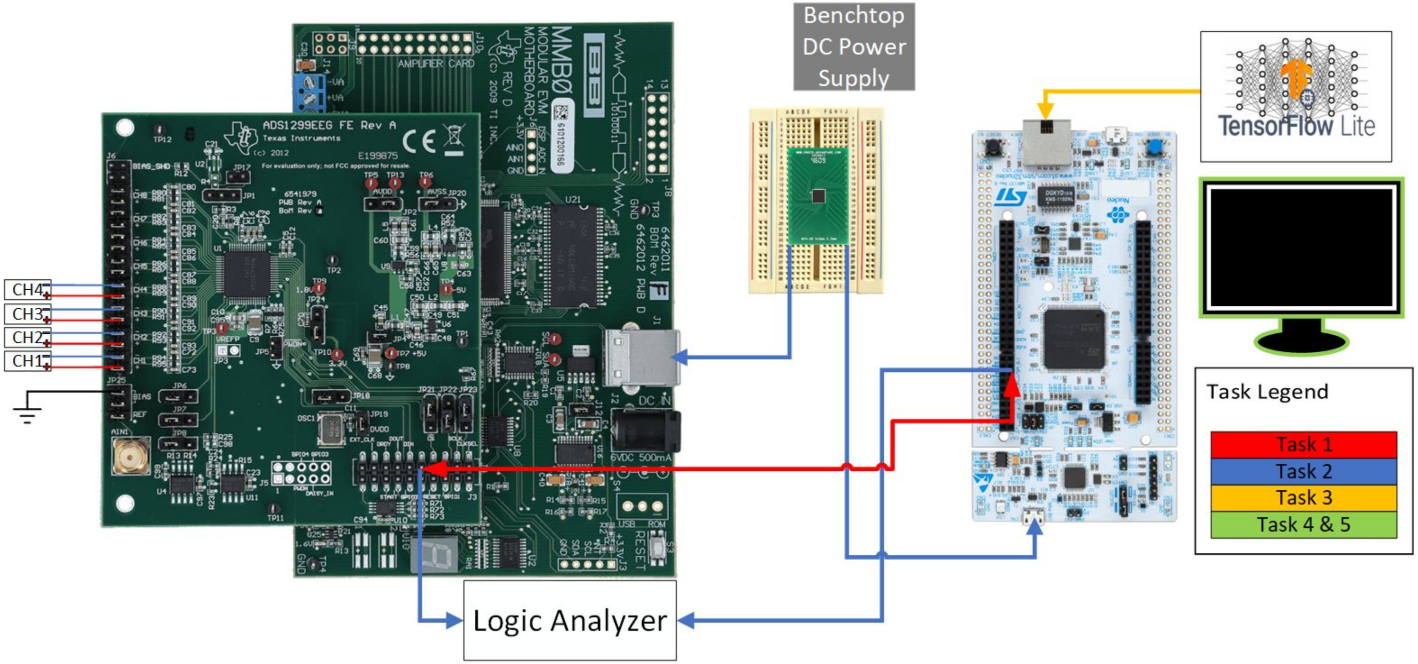


Figure 7

Note: The task legend in figure 5 refers to all tasks listed in phase 3 within figure 4 (Prototyping Roadmap). Every task that is a different color are tasks that occur independently, with the exception of task 4&5, which share the same color, but are also independently performed

Phase 3 Description

Phase 3 addresses Risk #6 (System Integration) with a stretch goal of Risk #5 (PCB Design Complexity). This phase showcases the integration of an EMG measurement subsystem using evaluations kits, highlighting the capture and transmission of muscle signals for analysis. The setup with an EMG sensor attached to the target muscle, measuring electrical signals generated by contractions and sending them to the (AFE), which amplifies and filters the signals.

The ADS1299 ADC then converts these amplified signals into digital data. This data is processed by the STM32L496ZG6ZGT6 microcontroller, programmed to initialize the ADC and set a sampling rate of 2000 samples per second (SPS). The microcontroller continuously reads the data and transmits it via I2C or SPI to a computer running simulation software.

The simulation software visualizes real time EMG waveforms, offering immediate feedback on muscle activity and analytical tools for filtering and peak directions.

Phase 3 Activity Steps

1. Total Subsystem Integration
    a. Place all subsystems and eval kits on the same ESD safe workspace
    b. Connect and remove necessary jumpers on the ADS1299EEGFE as per the documentation instructions
    c. Wire the SPI Lines according to the pinout specified in the block diagram (figure 2)
    d. Plug the micro USB into the USR_DTA port of the STM32 eval kit
    e. Wire the I2C from the LTC3556 to the STM32 according to the pinout specified in the block diagram (figure 2)
    f. Locate the AVdd pin on the ADS eval board, and connect the 5V output of the PMIC to that pin
    g. Locate the DVdd pin on the ADS eval board, and the PWR_IN pin on the STM32 eval kit, and connect that to the "Always on" 3.3V line of the PMIC
    h. Finally connect the battery, and wait for the system to initialize
        i. After init period, the data should be displaying on a PuTTy window, which verifies functionality of the device

Phase 1 | Phase 2 | **Phase 3** | Phase 4 | Phase 5

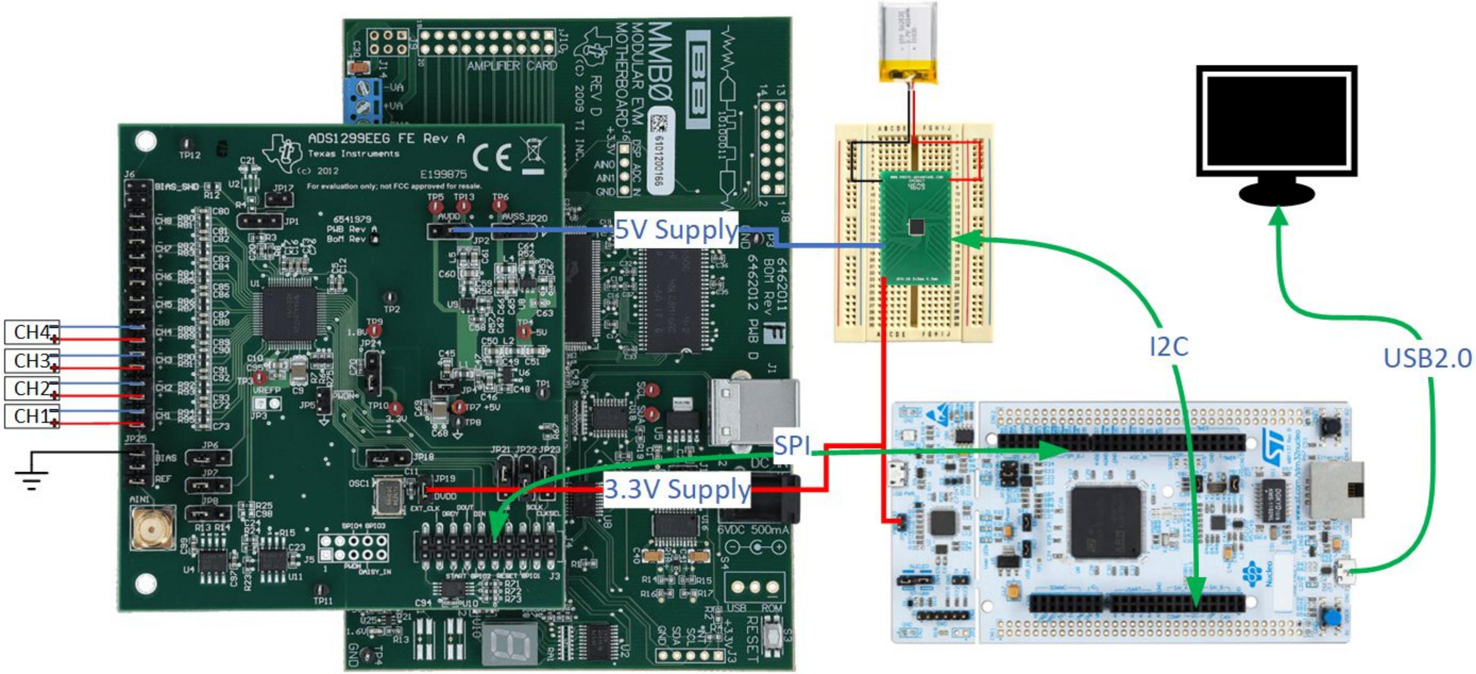


Figure 8

All devices now show a physical connection. This phase addresses the integration issue, by ensuring connectivity between devices before being placed on a PCB (which will begin its preliminary design process during this phase).

Phase 4 Description

Our Prototype activity in phase 4 is targeting Risk #5 (PCB Design and complexity) as well as Risk #6 (System Integration). These had posed a significant impact on the project so addressing them was made to be an important part of the prototyping process. This prototyping process will be the application of the eval testing into KiCad with the necessary pinouts found in phase 3 onto individual PCBs for each subsystem. The power delivery circuit will power both the ADS and the STM32, input signals will be processed and sent to the microcontroller, and the microcontroller will be able to receive those ADS signals to process and classify as movements for the simulator, as well as monitor the charging circuit for the power circuit. ESD considerations also accounted for (via Decoupling coupling and proper ground planes), as well as heat dissipation methods where necessary (via capacitors near power pins and trace designs that can handle higher currents AKA wider traces where needed).

Phase 4 Activity Steps

1. Modeling PCBs
   a. Based on phase 3 testing, use KiCad to create each PCB subsystem with the correct pinouts
   b. Test each circuit in the PCB simulation and verify all pinouts are correct and can communicate with the other subsystems where necessary before printing
2. Power Management Circuit
   a. Connect battery to USB Micro B
   b. Connect +5V output to ADS1299 Vin
   c. Connect +3.3V output to ADS1299 Vin and STM32 Vin
3. ADS1299
   a. Verify connecting points of electrodes and ensure the board has proper grounding for the analog and digital inputs voltage inputs
   b. Connect via SPI to the STM32
4. STM32
   a. Ensure connection from ADS, power circuit, and connect Via USB micro to simulator
   b. Run through commands to receive signals from ADS and process them with correct classification onto the simulator

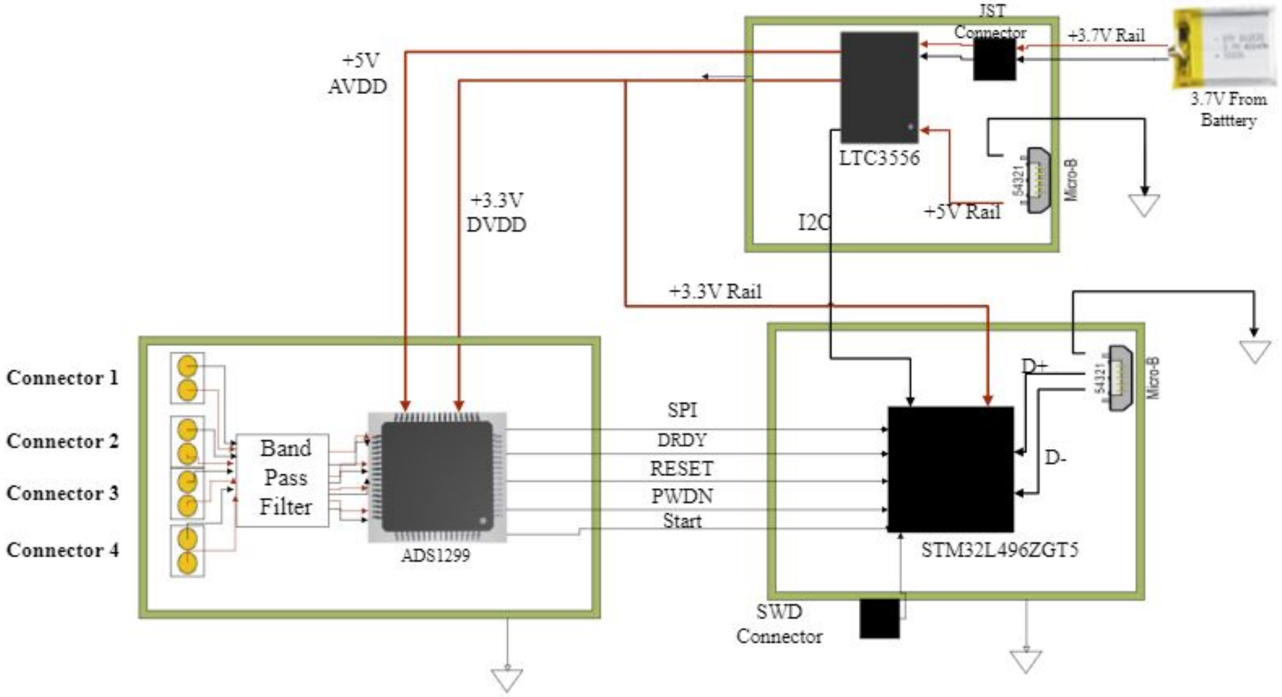


Figure 9

Each PCB design incorporates the I/O connections and ESD protection now, as well as necessary heat sinks and grounding planes for each one. This phase is ensuring each PCB functions and communicates properly with each other before full integration.

Phase 5 Description

Our prototype activity in phase 5 will target Risk #5 (PCB Design Complexity), and Risk #6 (System Integration). The phase 5 activity involves taking the three separate subsystems and integrating them into a single PCB. As previously mentioned in phase 4, the subsystems consist of power management, ADC, and the microcontroller. Each subsystem must be precisely routed in order to minimize the amount of noise interference and heat dissipation. Throughout the design and layout process we must ensure that the analog and digital components stay separate, use short signal paths, and opt for a multilayer PCB to isolate power and signal planes. This activity is crucial in order to preserve the signal integrity of our analog inputs.

Phase 5 Activity Steps

- Integrate power management, ADC, and microcontroller subsystems into a single PCB
- Route signal paths as short as possible to minimize noise interference
- Place decoupling capacitors near each subsystem’s power pins to filter noise
- Implement thermal management by placing heat sinks near heat-sensitive components
- Use a multilayer PCB to isolate power, ground, and signal planes to reduce cross-talk

Final Integrated PCB

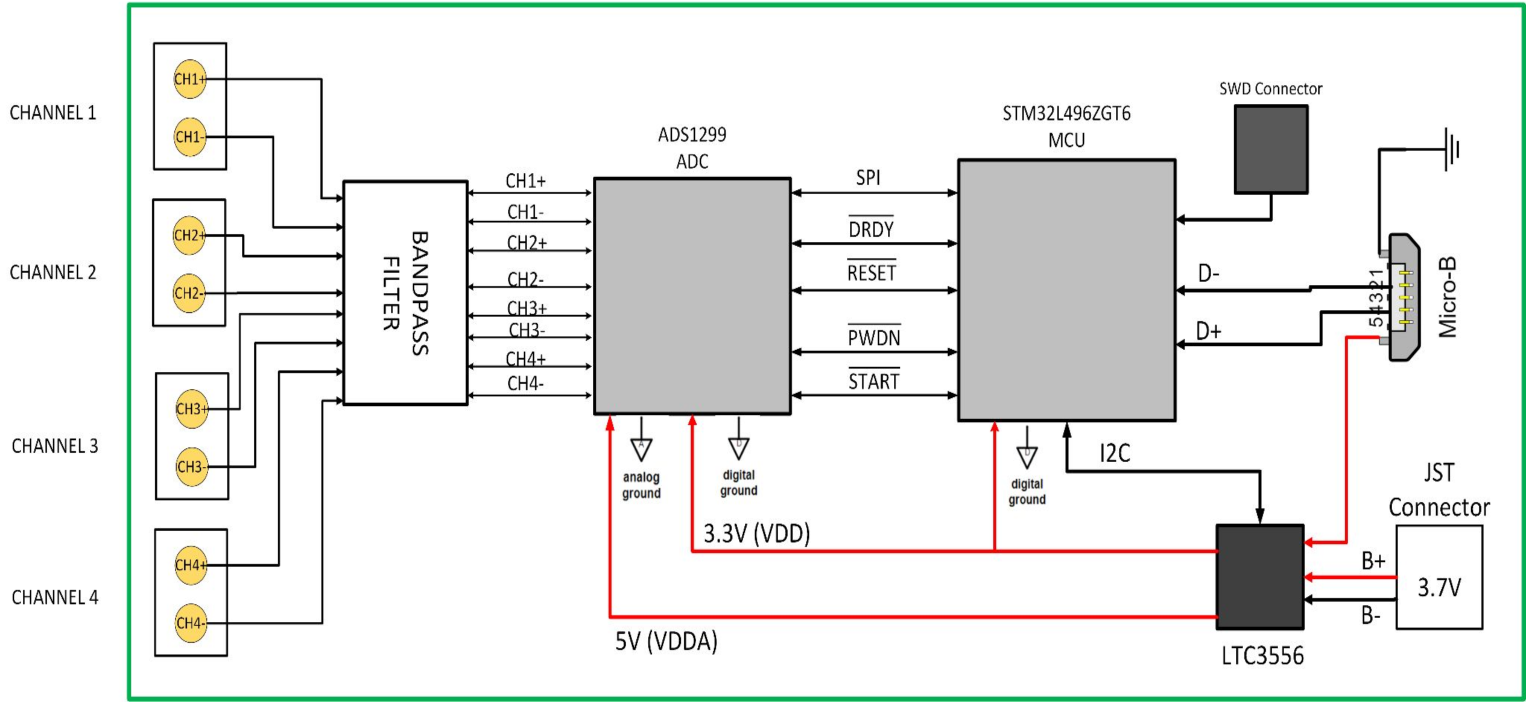


Figure 10:

The three separate subsystems are integrated onto a single custom PCB. Each subsystem are precisely routed with consideration for noise and heat. Layers are added to PCB in order to support complex routing and efficient power distribution.