Ant-weight, 3D Printed Battlebot

By
Justin Leong(jyleong2)
Yuxuan Nan(yuxuann2)
Zilong Jiang(zjian4)

Final Report for ECE 445, Senior Design, Spring 2025
TA: Bill Yang

07 December 2025
Project No. 1

Abstract

This report describes the process of designing, building, and testing a 3D-printed Battlebot for the Ant-weight class of robotic combat with specific guidelines and requirements. The goal and what we ultimately created was a lightweight, durable robot capable of performing quick and precise movements while delivering powerful attacks using a pneumatic flipping mechanism. Our 3D-printed Battlebot was developed utilizing an ESP32 microcontroller, contains a pneumatic scooping weapon, and moves on two “sticky” wheels powered by high torque motors, all wirelessly controlled by our controller app on our phones. After extensive testing, the robot met all key performance goals, including responsive control, reliable flipping power, structural durability, and enough battery life to last through a full match.

Contents

1. Introduction

Robotic combat presents a unique challenge at the intersection of mechanical design, embedded systems, and power management. In our case, we wanted to take part in the Ant-weight 3D printed Battlebot division competition that would be hosted by Professor Gruev at the end of the Spring 2025 semester. For our case, the problems include the Ant-weight division limiting the robots' weights to approximately two pounds. As such, this demands trade-offs between durability, agility, and weight. Furthermore, the Battlebot must also meet requirements of having enough battery power and the correct battery type to move quickly and use its weapon effectively. Finally, the Battlebot must also be able to be controlled wirelessly while also following certain structure material guidelines.

This project addresses those challenges through the design and construction of a 3D-printed Battlebot equipped with a pneumatic flipping weapon and wireless control via Bluetooth. Beyond mechanical and electrical design, the project also required thoughtful consideration of system integration, power regulation, and verification through testing. A 11.1V LiPo battery supplies power to all subsystems, while a buck converter steps down voltage for logic-level electronics. Each design decision was evaluated based on its contribution to performance, safety, and compliance with Ant-weight class rules.

Our goal and solution was to create a Battlebot that could hold its own in the final Ant-weight Battlebot competition. We wanted to ensure that it could take hits, move with agility, and react to commands instantly. To make that happen, we built a custom chassis utilizing PLA+ plastic. This material has the strength we need without exceeding our weight requirement for the chassis. Additionally, the Battlebot was powered by two high-torque motors, which would allow the Battlebot to drive and turn smoothly during matches. Furthermore, these motors also allowed for better offense capabilities. In terms of offensive capabilities, a pneumatic flipping mechanism was mounted that could reliably destabilize opponents. To control the entire system, the whole system is centered around the ESP32-WROOM-32E, which receives Bluetooth signals from the user and translates them into quick, responsive actions.

Ultimately, this report outlines our design and implementation process, and also provides the results of our Battlebot's performance based upon the details of each subsystem. Our report also discusses technical challenges we encountered, how we developed solutions to counteract them, and future improvements that could enhance performance in a competitive setting.

1.1 Top Level Block Diagram (Original)

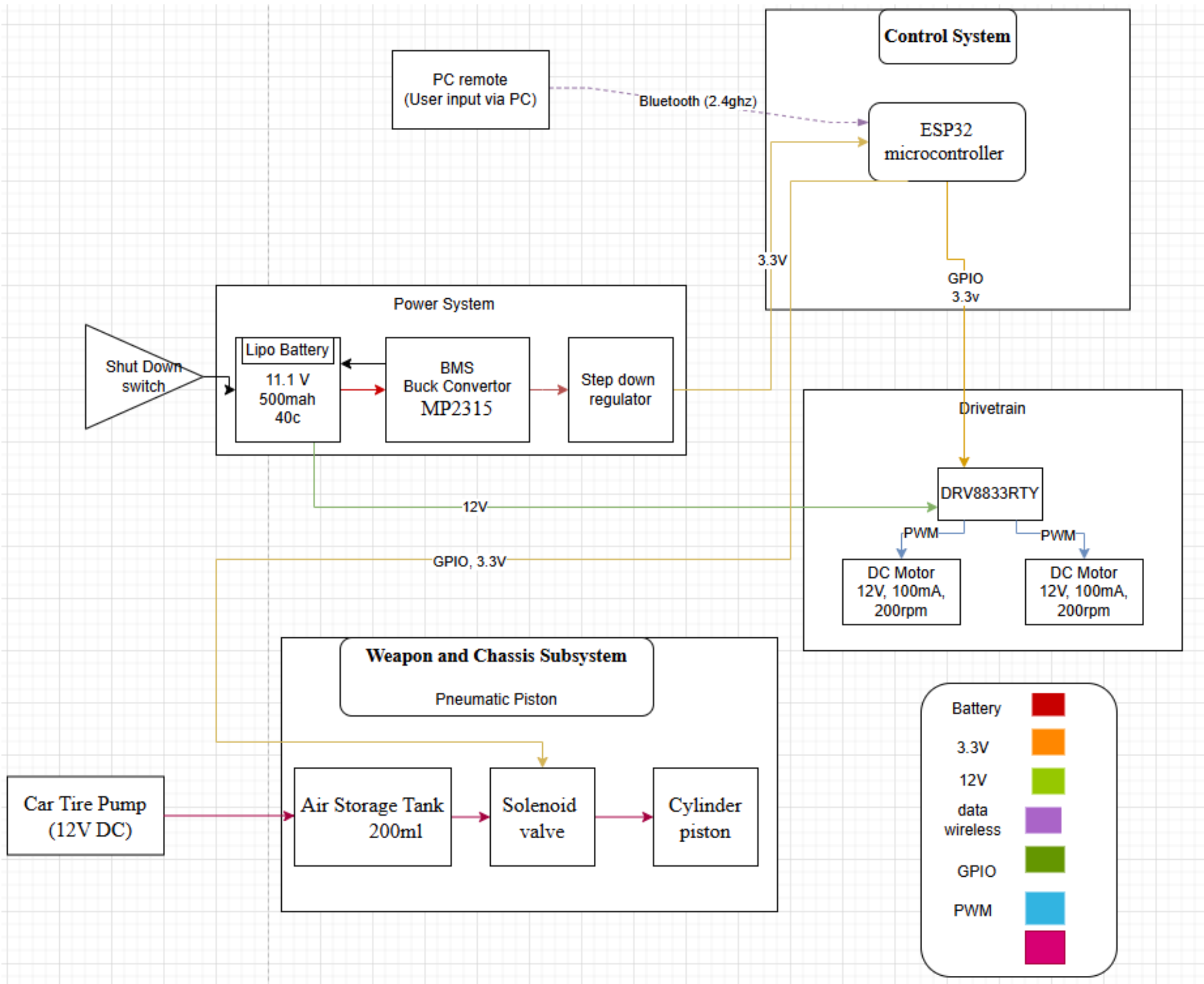

Figure 1. The original block diagram for our battlebot. This was the block diagram mostly followed in our design. However, it should be noted that in the power system, we never used a step down regulator and we also used a different buck converter. Furthermore, for our drivetrain, we ended up using a different H-Bridge chip due to complications in our circuit.

For the most part, this was the block diagram used in the creation for our final design. However, there were a few changes we had to make for the final design. Similar to what *Figure 1* states, we never used a step down regulator in the power system. We had also used different chips for both the buck converter and H-Bridge (drivetrain). The H-Bridge change in chip was due to incorrectly wiring the circuit, which will be talked about in *Section 2.2 Drivetrain*. Additionally, it should be noted that we utilized a phone as our control device instead of a PC.

To be specific in what our top level block diagram contains, it contains the four subsystems of drivetrain, weapon and chassis, power, and control subsystems. The control system is the heart of our robot, giving signals to the rest of the Battlebot on what actions the user is inputting through a remote control via phone app. The signals may be sent to the weapon and chassis subsystem or the drivetrain subsystem. For the weapon and chassis subsystem, it will receive a single signal using GPIO from the microcontroller on when to activate and deactivate the weapon controlled via solenoid, the weapon being a pneumatic piston connected to a scooper. The other GPIO signal sent out from the microcontroller controls the drivetrain. The drivetrain receives a total of four signals that assist with controlling the motors through an H-Bridge module. Finally, the entire Battlebot is powered by our power subsystem. This power subsystem utilizes a 11.1V LiPo battery and a buck converter to output a 3.3V signal. The 11.1V signal is used to power the motor subsystem and trigger our solenoid. The 3.3V signal is used to power our ESP32 which can then send its own 3.3V signals to the drivetrain subsystem and the weapon and chassis subsystem.

2. Design

Our Battlebot was designed around four core subsystems: the drivetrain, weapon and chassis, power, and control subsystems. Each of these subsystems played a critical role in ensuring the Battlebot is fast, durable, and capable of delivering solid offensive strikes. Throughout the design process, we balanced performance, weight, and reliability by choosing components that were both effective and efficient.

2.1 Drivetrain

Figure 2. This is a picture of how our actual Battlebot was laid out. There are two motors towards the backside of the internal layout of our Battlebot. To control these two motors, we mounted our H-Bridge on the front wall of our Battlebot. The reasoning of using a different H-Bridge will be discussed later in *Section 2.1 Drivetrain*.

The drivetrain of our system served as the main component of how the Battlebot was able to move and maneuver around opponents. For choice of motors, the Greartisan high-torque DC motors propelled our Battlebot. Each of these motors provided roughly 2.2 kg·cm of torque, with a no-load speed of 200 RPM. Our motors could displace enemies up to 4.4kg (1)(2) under ideal conditions where coefficient of friction is 1 with both motors. These were seated at the back of our robot to ensure that our front scooper was as close to the ground as possible at a steeper angle, which can be seen in *Figure 2*. Additionally, we utilized "sticky" tires that would best compliment these high torque motors. Tires with a maximum amount of grip would allow our robot to utilize the maximum amount of torque and be able to drive well in the arena without slipping. In our original design, we wanted to put omni-directional balls at the front or an omni-directional wheel, but due to the height increase this would elevate our Battlebot by, we decided against implementing this in our final design. Instead, we wanted our scooper to be close to the ground, thus removing these front omni-wheels or omni-ball would best compliment the weapon.

$$F_{torque} = \text{torque} / \text{radius} = (2.2 \text{ kg*cm} * 9.81 \text{ N*m / kg*cm}) / 0.01\text{m} = 21.582\text{N} \quad (1)$$
$$\text{Total Weight Displaceable} = F_{torque} / (\mu_k * g) = 21.582\text{N} / (1 * 9.81) = 2.2 \text{ kg} \quad (2)$$

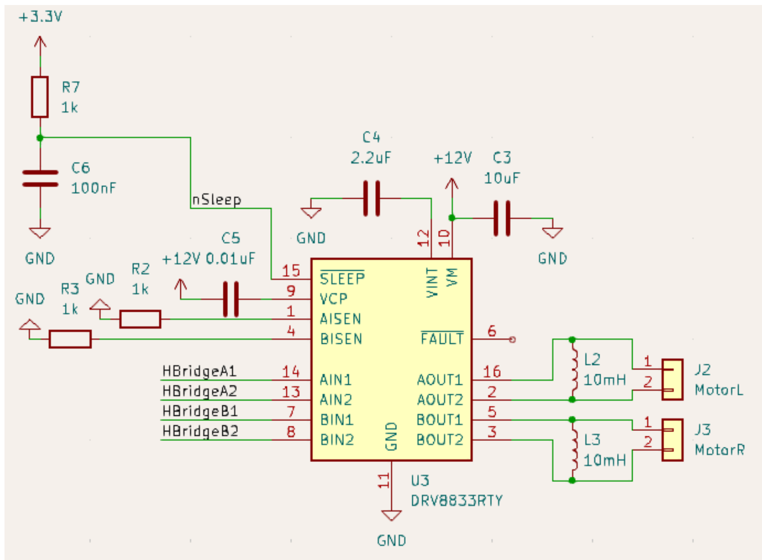

Figure 3. This is a schematic of how we wired our original drivetrain subsystem. The drivetrain subsystem is centered around the DRV8833RTY chip from Texas Instruments.

Based on *Figure 3*, the schematic of how the drivetrain's H-Bridge was wired in the PCB is demonstrated. In this original layout from the PCB, it should be noted that an incorrect wiring was made. Due to how the DRV8833RTY chip should inherently be wired, the VM pin was incorrectly given access to the 12V rail from the battery. This is an issue that essentially shorted or fried the H-Bridge chips we had on hand for this specific module due to the fact that the max voltage the VM pin is able to handle is 10.8V [7]. We were able to detect this issue by checking the nFault pin which would display an output logic level zero if the chip has incurred a fault condition. The fault conditions that we had deduced this to were either burning or over-volting to the chip.

To solve the issue of having fried our DRV8833RTY H-Bridge modules, we decided to use an external H-Bridge module – the L9110s DC motor driver module. This new H-Bridge module functioned similar to our original module, with it accepting four inputs to directionally control the outputs. The H-Bridge logic of the DRV8833RTY and L9110s H-Bridge modules can be seen in *Table 1*. However, this new L9110s H-Bridge module, on the other hand, accepted up to a maximum of 12V without short circuiting in comparison to the maximum 10.8V the DRV8833RTY can handle.

Table 1. H-Bridge Logic

xIN1	xIN2	xOUT1	xOUT2	Function
0	0	Z	Z	Coast/fast decay
0	1	L	H	Reverse
1	0	H	L	Forward
1	1	L	L	Brake/slow decay

2.2 Weapon and Chassis

The primary weapon we designed is a Pneumatic Ramp Flipper, specifically designed to flip or destabilize opponents with precision and efficiency. It features a low-profile, angled ramp that smoothly slides underneath for effective lifting. The system originally was designed to operate using a pneumatic actuator connected to a 200ml gas tank pressurized at 120 PSI, enabling controlled solenoid valve activation for each flip. However, we had actually opted to buy CO_2 cartridges that held above the pressurized 120 PSI, which were more cost effective and easier to source. This also made our design more easily swappable between fights instead of having to find a location to refill our cartridges.

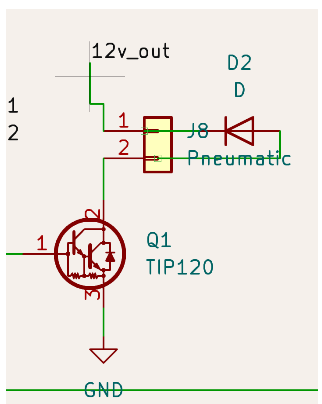

Figure 4. The actual schematic of our Pneumatic system and how it would be activated. The signal for the transistor comes from our control system through a 10kΩ resistor.

Originally, we had wanted to activate our weapon system utilizing a NMOS transistor, however, the voltage outputted by our ESP32 would not consistently meet the requirement of 3.3V-3.6V to trigger and close the gate. As such, we had to transition to a BJT gate which would be controlled by current. This way, there would be no reliance on our ESP32 to output above 3.3V consistently as we were only able to get a consistent output of 3.25V. Additionally, this meant the solenoid could also be reliably opened and closed utilizing the BJT transistor. Furthermore, this new portion of the circuit was wired the way it was to have a 12V to trigger the solenoid as well as have a flyback diode to give the inductive load spike a safe path to circulate. Finally, we also wired a 10kΩ resistor between the BJT and the microcontroller to minimize current draw from the microcontroller.

In terms of chassis, we still continued with using PLA+ plastic for our 3D prints. This is because of PLA+'s offering a balance between durability and weight efficiency. Furthermore, it is also less brittle in comparison to PLA, meaning an increased impact resistance. Finally, for our final design, we had opted to mount heavy components at the bottom of the chassis to lower the center of gravity (*refer to Figure 2*). Additionally, we had opted for two layers of a ramp and scooper to be able to more easily scoop under opponents.

2.3 Power System

Our Battlebot is powered by a 3-cell (3S) 11.1V LiPo battery rated at 500mAh. We had selected the LiPo variant battery as this was one of the provided battery types safe to use in the Antweight Battlebot competition. Additionally, while the battery was labeled as 11.1V, we could charge it to voltages around 11.8V or even 12V at full charge. This provides a strong and consistent power source for both the motors and the pneumatic weapon system, which are wired directly to the battery and operate comfortably within that voltage range. However, the ESP32 microcontroller requires a much lower operating voltage – specifically 3.3V. To supply the ESP32 safely, we used an LM2596S-3.3 buck converter to step down the battery voltage. This converter takes the ~12V input from the battery and reliably outputs 3.3V to power the logic side of our control system. Without this step-down circuit, the ESP32 would be damaged due to overvoltage as it has a max voltage of 3.6V [6].

By splitting our power delivery this way—sending the full battery voltage to high-power components like the motors and solenoid, while using the buck converter for the ESP32 – we were able to simplify our wiring and avoid unnecessary regulators or converters for the rest of the system. We tested the battery under full load and confirmed that it could consistently power all components for more than 2 minutes, which exceeds the length of a typical match and meets the project's performance goals. In actuality, the battery could last up to ~10 minutes under specific conditions and loads.

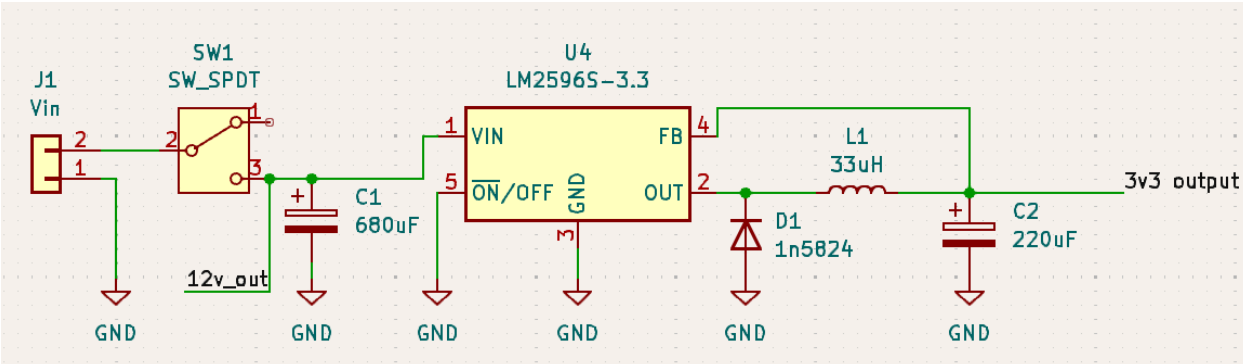

Figure 5. A schematic of our power subsystem and what it would output. There are two points of shutoff to ensure our system has a safe shutoff – the switch and connector to battery.

This buck converter circuit steps down a 12V input to 3.3V for powering low-voltage components like the ESP32. The capacitors help smooth voltage fluctuations – C1 at the input and C2 at the output – while the inductor and diode ensure efficient energy transfer during switching. The diode also provides a return path for current when the switch is off, preventing voltage spikes and improving overall circuit stability and efficiency.

It should be noted that contradictory to our original block diagram in *Figure 1*, we do not use a step down regulator. This is because our buck converter – LM2596S-3.3 – safely steps down our voltage from 12V to 3.3V safely without too much heat, removing the need for a step down regulator.

2.3.1 Tolerance Analysis

In our original tolerance analysis, we calculated the type of battery required based upon the factor that competition rounds were approximately two minutes long.

The motor we selected (at 12V) has a max efficiency of 0.1A draw with a 2A stall current.

Average and Peak Current Draw of both motors:

$$I_{motors} = N_{motors} * I_{motor} = 2 * 0.1A = 0.2A \quad (3)$$

$$I_{peakM} = N_{motors} * I_{motor\_peak} = 2 * 2A = 4A \quad (4)$$

Our Pneumatic system has a rough average current draw of 3A with a peak current draw of 5A. Additionally, our ESP32 and related components have an average current draw of 0.15A and a peak current draw of 0.2A.

The calculation of how much our battery capacity need as well as tolerance can be calculated as follows:

Total and Peak Current Draws:

$$I_{total} = I_{motors} + I_{pneumatic} + I_{IC} = 0.2A + 3A + 0.15A = 3.35A \quad (5)$$

$$I_{peak} = I_{motors\_peak} + I_{pneumatic\_peak} + I_{IC\_peak} = 4A + 5A + 0.2A = 9.2A \quad (6)$$

$$\text{Capacity of Battery Wanted} = I_{total} * \text{Total Time} = 3.35A * 2min = 3.35 * 2/60\ h = 0.1116\ Ah \quad (7)$$

$$\text{Peak current of battery} = 9.2A \quad (8)$$

$$3.35A/500mAh = 6.7C \quad (9)$$

$$9.2A/500mAh = 18.4C \quad (10)$$

Because actual consumptions with batteries will vary, we need to make sure that the battery can handle 9.2A of peak load draw as well as have a total capacity of at least 0.1116Ah (7). This will ensure that the battery can handle the components we are using as well as last the total battlebot round of 2 minutes. The battery we have opted for has a capacity of 500mAh - which satisfies the capacity of battery wanted by a lot to ensure even with not optimal conditions it will work - as well as discharge rate of 35C and max discharge rate of 70C. To ensure that the peak current and current ratings are within this discharge rate, we take the amperage divided by the total capacity of the battery (9)(10), both of which are within limits: 6.7C < 35C and 18.4C < 70C.

While these calculations were all estimations, they were pretty close to our actual draws and thus satisfied our requirements. It should be noted that with both motors running, there is approximately 0.3A of current draw with a peak of 1A. The pneumatic system also draws approximately 0.5A of current for triggering, meaning that the minimum and maximum peaks of current draw range between 0.3A and 1.5A.

2.4 Control System

To serve as our central processing unit, the ESP32-WROOM-32E served as our microcontroller. This ESP32 offers both WiFi and Bluetooth, perfect for our design to wirelessly control our Battlebot while also having enough pins for the Drivetrain and Weapon/Chassis subsystems. Originally, our design opted to utilize a PC as our main controller. However, we found this very “clunky” in a sense and opted to wirelessly control it from our phones. Additionally, we had swapped from Bluetooth, to WiFi (using hotspot), back to Bluetooth. This was due to the fact that we originally thought that Bluetooth would have too much input delay, however, this was not the case. The reason why we swapped from WiFi back to Bluetooth as well is the fact that WiFi utilizing a hotspot introduces more room for error. There would be more points of failure, that being the hotspot we would need to connect to, as well as both our Battlebot and PC controller needing to connect to that. As such, we had decided to transition back to Bluetooth, which offers individual private connections as well as fast response times, under 100ms, for low-latency executions.

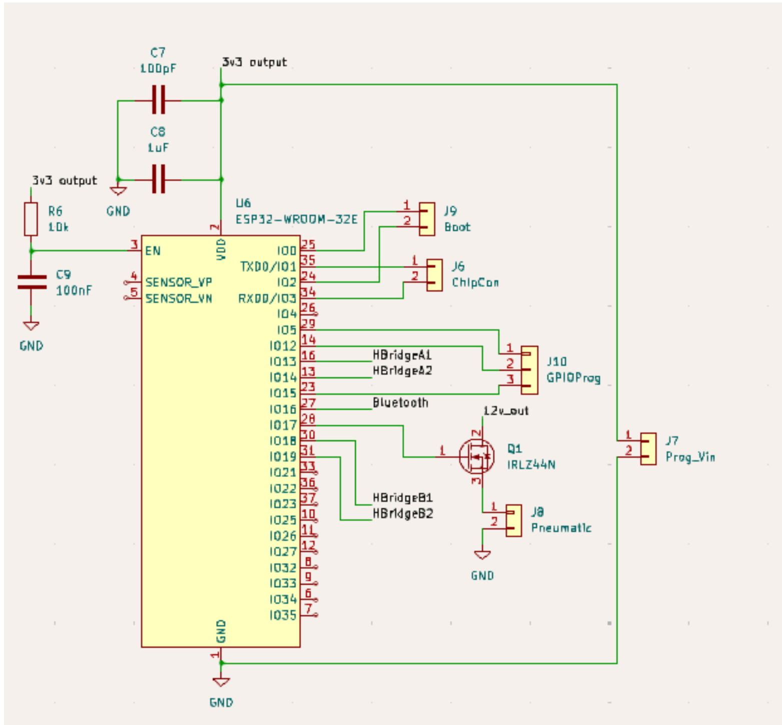

Figure 6. Our schematic for the control subsystem. It should be noted that the weapon subsystem is included but is also outdated, more can be found in the section *2.2 Weapon and Chassis*.

Once again from *Figure 6*, it should be noted that the weapon subsystem portion of this schematic is incorrect, and an updated version can be found in the previous Weapon and Chassis subsystem. While there aren't many deviations from this design for our microcontroller schematic, it should be noted that to actually wire our new drivetrain, we had directly connected wires to the pins labelled with "HBridge". This was because the layout for the DRV8833RTY was actually too small for connections.

Finally, in terms of code, we had utilized an Arduino library that we modified for our purpose. That library is called "Dabble", as it provides a phone app with a controller, perfect for our Battlebot use case. This library was able to send signals to our microcontroller to control the directions as well as have buttons for triggering our pneumatic system. Furthermore, we were able to connect LEDs to show that we had powered up and had Bluetooth connections enabled for our system. Additionally, when wirelessly controlling our Battlebot, the connection timings when inputting our controls were all 100ms or less – with an average of ~90-100ms during one of our test sessions (utilizing a timer) – meaning that the testing had been verified for low latency connection.

3. Design Verification

Table 2. Drivetrain Requirements and Verification

Requirements	Verification
Can produce a maximum of 2.2kg*cm of torque under no load	● Tested by pushing something of 2.2kg for approx. 1cm ● Ended up using the battlebot to push roughly 4lbs (close to 2.2kg) for over 1 meter with our final battlebot
Can reach a maximum of 210 rpm under no load	● We did not have a tachometer, counted revolutions of a spinning tape and approximated/calculated the rpm ● Roughly 210 rpm (3.5 revolutions per second)

Table 3. Weapon and Chassis Requirements and Verification

Requirements	Verification
1.5A battery current draw requirement	● Measured current draw of solenoid trigger at 0.5A
Solid build structure using PLA+ plastic - Able to move and withstand ramming 2 lbs	● Our robot can move/push over 4 lbs of weight (ramming robot into the 4lbs)
Gas tank that can withhold 120 psi for pneumatic weapon	● We utilized pre-filled CO2 cartridge that holds over 120psi; tube was approximately 8in long, ¼ in diameter ● Able to launch over 7 fully power flips

Table 4. Power Subsystem Requirements and Verification

Requirements	Verification
Can supply continuous 3A of current to all subsystems when underload	● Measured current, varied from 0.3A to 1.5A when all subsystems are underload, meaning our design works very efficiently compared to planned maximum current
Has enough capacity to last 2 minutes of continuous usage	● Our full design under load lasts longer than 2 minutes ● Ran a timer, roughly ~10 minutes under load

Table 5. Power Subsystem Requirements and Verification

Requirements	Verification
Connect via Bluetooth	● Able to use phone (Dabble library) to connect to Battlebot; ~100ms signal transmission times (timer) ● LED to display bluetooth is in use
Be able to control all 4 directional movements of battlebot with given signals - left, right, forward, backwards	● Able to move in the 4 directions properly using the directional movements ● H Bridge A1,A2,B1,B2

4. Costs and Schedule

4.1 Parts

Table 6. Final Design Parts Costs

Part	Manufacturer	Retail Cost ($)	Bulk Purchase Cost ($)	Actual Cost ($)
PLA+ Filament	Amazon	$16.99	$16.99	$16.99
DC 12V 200RPM Gear Motor	Amazon	$14.99	$29.98	$29.98
RCBattery	rcbattery.com	$10.99	$10.99	$10.99
220uf capacitor	Mouser Electronics	$1.19	$1.19	$1.19
680uf capacitor	Mouser Electronics	$1.08	$1.08	$1.08
diodes	ebay	$2.21	$2.21	$2.21
10mH inductor	Switch Electronics	$2.03	$4.06	$4.06
33uH inductor	Digi-Key Corporation	$0.18	$0.54	$0.54
solenoid valve	Amazon	$9.65	$9.65	$9.65
single return cylinder	Amazon	$18.28	$18.28	$18.28
co2 gas	Amazon	$23.90	$23.90	$23.90
gas hose	Amazon	$14.69	$14.69	$14.69
buck converter	Digi-Key Corporation	$7.89	$23.67	$23.67
H bridge	Digi-Key Corporation	$1.87	$5.61	$5.61
Esp32	Digi-Key Corporation	$4.36	$13.08	$13.08
switch	Digi-Key Corporation	$2.68	$2.68	$2.68
wheels	ebay	$2.59	$2.59	$2.59
Total				181.18

It should be noted that some parts were bought ourselves, but were factored into the total costs.

4.2 Labor

The total labor cost is calculated as $40 × 60 = $2,400.
For the EE student and $38 × 60 = $2,280 per CE student,
resulting in a total labor cost of 2*$2280+$2400= $6,960 for the team.

4.3 Schedule

Table 7. Schedule

Week (date)	Group Members	Tasks
3/3	Justin, Yuxuan	Able to send signal by using esp32 through wifi
3/10	Zilong, Justin	Finish the pcb design
3/24	Zilong	Figure out how to use the correct mosfet for our design
3/31	Zilong, Justin, Yuxuan	Fix the pcb design and order in the third round.
4/7	Zilong, Justin, Yuxuan	We keep soldering the pcb board, and test the pneumatic weapon in breadboard
4/14	Zilong, Justin, Yuxuan	Keep testing pcb board and resoldering another PCB board
4/21	Zilong, Justin, Yuxuan	Combine all the designs together and print the chassis

5. Safety and Ethical Considerations

5.1 Ethical Considerations

Our project adheres to the IEEE Code of Ethics [1] and the ACM Code of Ethics [2] to ensure ethical responsibility in both design and development. Key ethical concerns include fair competition, safety of participants and bystanders, and transparency in development.

1. Fair Competition: We will ensure compliance with competition rules, avoiding unauthorized modifications that may provide an unfair advantage. All components used will comply with the competition's technical regulations, and no deceptive practices will be employed.
2. Safety of Participants and Bystanders: The Battlebot will be tested in a controlled environment, ensuring it does not pose accidental risks to participants or observers.
3. Transparency in Development: We will maintain proper documentation of hardware and software to ensure integrity and reproducibility. Design files and testing reports will be kept up to date for reporting purposes.

5.2 Safety Concerns and Strategies

To ensure the safety of developers, users, and bystanders, we will implement the following safety measures in accordance with Occupational Safety and Health Administration [3] regulations.

5.2.1 Electrical Safety

Our project involves a 11.1V LiPo battery and high-torque motors, necessitating robust safety measures to prevent electrical hazards:

To reduce the risk of short circuits, we took several practical steps during both assembly and testing. One specific measure involved taping over any exposed metal parts, such as screw heads or frame components, that were close to the PCB. This prevented accidental contact between conductive surfaces and the circuit board, which could otherwise result in a short. In addition, we used heat-shrink tubing on all soldered wires, properly insulated power lines, and organized cables to avoid tangling or friction. We also performed continuity checks before powering the system to ensure that no unexpected connections were present.

5.2.2 Mechanical Safety

The weapon system and chassis structure will be designed to minimize risk to operators and audience:

- Finite Element Analysis (FEA): The chassis and weapon system will undergo stress and impact simulations to ensure they can withstand combat forces without unexpected failure.
- Material Safety: The bot will be 3D-printed with PLA+ plastic, selected for impact resistance and durability.

5.2.3 Pneumatic Safety

Pressure Limits Compliance

- Do **not exceed 120 PSI** during normal operation.
- Safety buffer limit: **140 PSI maximum** before automatic pressure release is triggered.

Pre-Operation System Check

- Inspect all air hoses, connections, and fittings for leaks or damage before every use.

Safe Charging & Pressure Release

- Always charge the system in a well-ventilated area.
- Use only approved pressure sources with a regulator to prevent over-pressurization.
- Before maintenance, fully depressurize the system using the manual release valve.

5.3 Lab Safety Compliance

Development and testing will follow University of Illinois Lab Safety Guidelines, including:

1. Personal Protective Equipment (PPE): Team members will wear safety goggles, gloves, and anti-static wristbands when handling batteries, electrical circuits, and pneumatics.
2. Safe Work Practices: All testing will occur in designated safety zones, with emergency response plans in place.

5.4 Required Documentation

Given the high-risk factors associated with pneumatics, high voltage, and combat interactions, we will develop a comprehensive Safety Manual that includes:

- Emergency Procedures: Clear steps to handle system failures, air leaks, electrical shorts, and unintended solenoid activation.
- Risk Mitigation Strategies: Guidelines for safe handling of pressurized air systems, battery management, and ESD protection for electronic components.
- Safe Handling Guidelines: Standard operating procedures (SOPs) for system assembly, maintenance, and combat deployment.

5.5 Demonstration of Safety Compliance

Adherence to Best Practices: During the final demo, we will showcase strict compliance with our Safety Manual, demonstrating that all team members follow safety protocols.

- Pneumatic System Integrity Test: Before operation, we will verify that all connections hold pressure without leaks, ensuring the gas tank, solenoid, and actuator function correctly.
- Electrical Safety Measures: We will implement proper insulation, fuse protection, and emergency cutoff switches to prevent accidental system overloads.

6. Conclusion

Over the course of the semester, we successfully designed, built, and tested a Bluetooth controlled 3D Printed Ant-weight Battlebot. The robot integrated a pneumatic flipping weapon, dual-motor drivetrain, custom PLA+ chassis, and reliable power and control systems. In our final tests, it met key performance goals: it was responsive to wireless commands, drove in the four directions required, activated its weapon, and was structurally durable during collisions. By precisely opening the cartridge, we confirmed that the weapon system did not leak any gas. During testing and competition, triggering the system allowed the piston to successfully flip the ramp at least 5 times without failure. The control interface worked reliably with under 100ms latency, and the power system remained stable during extended operation.

One area where the design did not fully meet competition requirements was weight. Our chosen motors provided excellent torque but were heavier than anticipated, causing the bot to slightly exceed the weight limit by approximately ~½ pounds over the Ant-weight classification. While the robot performed effectively in practice, this made it non-compliant for official competitive entry. This issue was identified during testing and verified by scale measurements, highlighting the trade-off between torque output and weight constraints. Another issue we would like to address was utilizing a separate H-Bridge module due to the incorrect wiring of the 12V to the max voltage of 10.8V input on our original H-Bridge module which inevitably sent through our measured 11.1V+ battery voltage to the H-Bridge VM pin which can only accept the max 10.8V. This can be fixed in the future by finding either a compatible H-Bridge module with our specifications, or having a way to step down the voltage to below 10.8V.

In future iterations, we plan to address the weight issue by selecting more compact and lightweight motors that still provide adequate torque. There are numerous lighter motors on the market that we could utilize and still be able to accomplish our goals, meaning that there can be further improvements to optimizing the weight of the motors while also maintaining a high amount of torque and speed. We also see potential for optimizing the chassis geometry further, possibly reducing material usage while maintaining structural integrity. Having a more sleekly designed chassis will ensure our Battlebot cannot be flipped from behind like what occurred in round 2 of the competition. Furthermore, our wheels on our chassis were placed quite high, making it really easy to slide a scooping mechanism underneath our chassis. Changing this in the future will also ensure that the Battlebot will not be as easily flipped from behind.

References

[1] IEEE, "IEEE Code of Ethics," 2023. https://www.ieee.org/about/corporate/governance/p7-8.html

[2] ACM, "ACM Code of Ethics and Professional Conduct," 2023. https://www.acm.org/code-of-ethics

[3] OSHA, "Occupational Safety and Health Administration Regulations," 2023.https://www.osha.gov/

[4] NFPA, "National Fire Protection Association Electrical Safety Standards," 2023. https://www.nfpa.org/

[5] University of Illinois, "Division of Research Safety," 2023. https://www.drs.illinois.edu/

[6] Espressif Systems, "ESP32-WROOM-32E/ESP32-WROOM-32UE Datasheet," version 1.3, Jan. 20, 2023. https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e_esp32-wroom-32ue_datasheet_en.pdf

[7] Texas Instruments, "DRV8833 Dual H-Bridge Motor Driver Datasheet," Nov. 2013. https://www.ti.com/lit/ds/symlink/drv8833.pdf

Appendix A Requirement and Verification Table

Table 8. System Requirements and Verifications

Requirements	Verification	Verification Status (Y/N)
Drivetrain Requirements 1. Can produce a maximum of 2.2kg*cm of torque under no load 2. Can reach a maximum of 210 rpm under no load	Drivetrain Verifications 1. — a. Tested by pushing something of 2.2kg for approx. 1cm b. Ended up using the battlebot to push something of roughly 4lbs (close to 2.2kg) with our final battlebot 2. — a. We did not have a tachometer, counted revolutions and approximated the rpm b. Roughly 210 rpm (3.5 revolutions per second)	1. Y 2. Y
Weapon and Chassis Requirements 1. 1.5A battery current draw requirement 2. Solid build structure using PLA+ plastic a. Able to move and withstand ramming 2 lbs 3. Gas tank that can withhold 120 psi for pneumatic weapon	Weapon and Chassis Verification 1. — a. Measured current draw with the solenoid trigger is 0.5A 2. — a. Our robot can move/push over 4 lbs of weight (ramming robot into the 4lbs) 3. — a. We utilized pre-filled CO2 cartridge that holds over 120psi; tube was approximately 8in long, ¼ in diameter b. Able to launch over 7 fully power flips	1. Y 2. Y 3. Y
Power Requirements 1. Can supply continuous 3A of current to all subsystems when underload 2. Has enough capacity to last 2 minutes of continuous usage	Power Verification 1. — a. Measured currents, actual current amounts vary from 0.3A to 1.5A when all subsystems are underload, meaning our design works very efficiently compared to planned maximum current 2. — a. Our full design under load lasts longer than 2 minutes b. Ran a timer, roughly ~10 minutes under load	1. Y 2. Y
Control Subsystem 1. Connect via Bluetooth	Control Verification 1. —	1. Y 2. Y

2. Be able to control all 4 directional movements of battlebot with given signals - left, right, forward, backwards	a. Able to use phone (Dabble library) to connect to Battlebot; ~100ms signal transmission times b. LED to display bluetooth is in use 2. — a. Able to move in the 4 directions properly pushing the directional buttons b. H Bridge A1,A2,B1,B2	