[Group 11 ECE445 Design Document]

By

[Matthew Jin]

[Timothy Fong]

[Ved Tiwari]

Proposal Document for ECE 445, Senior Design, [Fall/Spring 2026]

TA: [Zhuoer Zhang]

[24] [February] 2026

Project No. 11

[1] Introduction

[1] Problem and Solution:

The all-encompassing challenge of this Battlebot project is to weigh engineering decisions that allow for tactical and functional advantages over other Battlebot participants. This is encapsulated by aspects of high reliability, structural integrity, ideal component selection, and countermeasures sculpted by foresight. To attain this objective, the drivetrain, weapon mechanisms, and power systems have all been sharpened through numerous design iterations and modifications. With the culmination of knowledge attained through firsthand research and consulting Machine Shop personnel/ECE Alumni, our Durian Battlebot was born. Where other teams have had their wheels damaged, we've decided to place them within the armored shell. Where some Battlebots struggled with effective motion, we've selected motor and wheel pairings that work optimally for the competition. Where other groups have over-engineered their active weaponry to be difficult-to-replace, our motor-powered drill remains effective yet simple to replace. Not only does this solution answer the challenge of a true R&D test in a competitive environment, but it builds on culminated knowledge of those who have failed AND succeeded.

[2] Visual Aid:

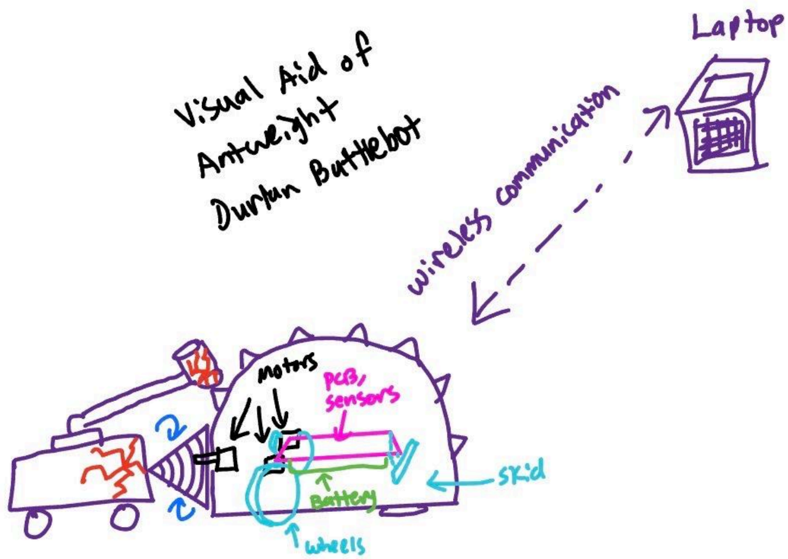


Figure 1: This is the visual aid depicting the Antweight Durian Battlebot's full functionality (wireless communication, passive and active weaponry capabilities, and high-level mechanical system layout).

A pictorial representation of your project that puts your solution in context. Not necessarily restricted to your design. Include other external systems relevant to your project (e.g. if your solution connects to a phone via Bluetooth, draw a dotted line between your device and the phone). Note that this is not a block diagram and should explain how the solution is used, not a breakdown of inner components.

[3] High-Level Requirements List:

The most important characteristics for our Battlebot are as follows, all commonly defined by their non-negotiable presence in answering the project scope:

- The BattleBot, as a holistic system (comprising all its subsystems), must be extremely reliable for its use case, encompassing all possibilities within the ring.
-The BattleBot must have a high degree of lethality such that it bends opponent strategy through its potency.
-The BattleBot must be optimized enough to be easily repairable in the unfortunate event a component is damaged beyond reusability.

[2] Design

[1] Block Diagram:

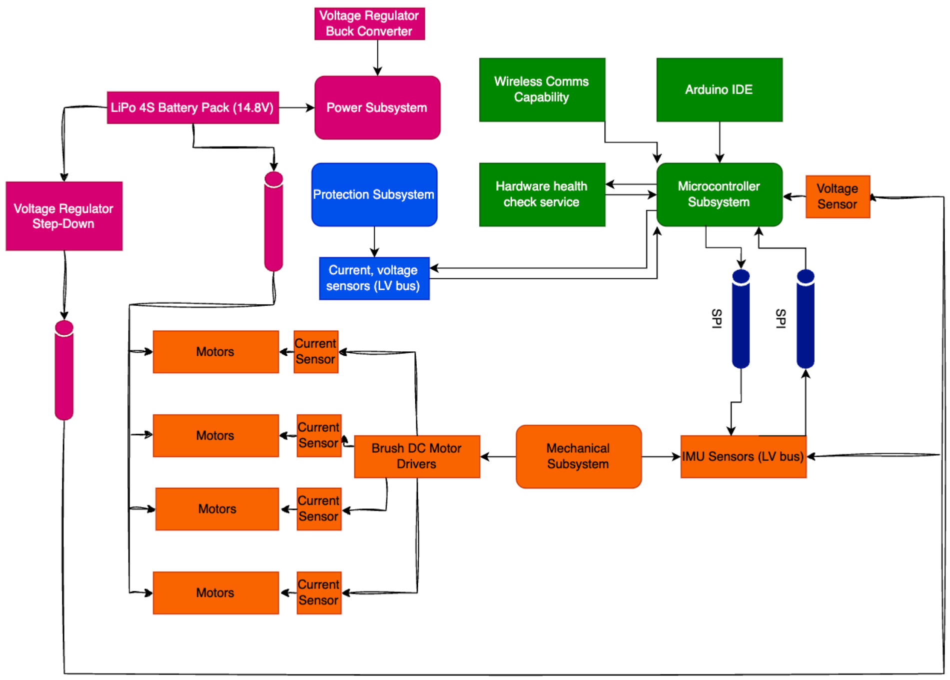
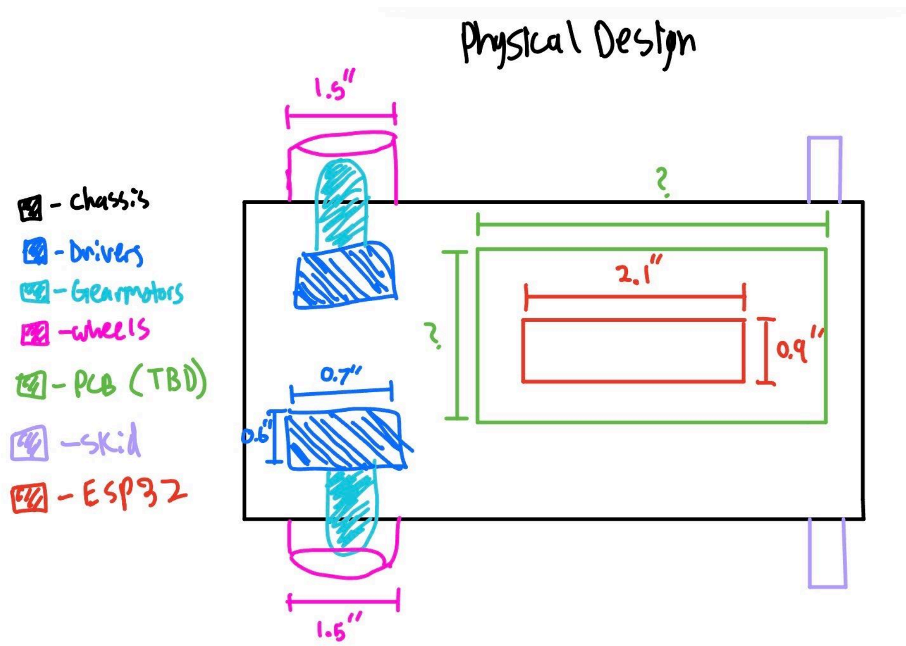


Figure 2: This is a high-level block diagram that captures the four subsystems of the robot, along with its components and interconnections.

This block diagram consists of four subsystems. The first one is the mechanical, which encapsulates all drivetrain components and the active weaponry system. The two drivetrain motors and single weapon motor will communicate to motor drivers located on the centralized chassis, which then form feedback loops between the ESP32 microcontroller. The second subsystem is power– it is the energy delivery system supplying components within all other subsystems. The battery directly or indirectly (goes through step-down voltage regulators first) supplies power to motors, drivers, the ESP32 microcontroller, the voltage/current sensors, and the PCB. The protection subsystem is intrinsically tied to the power one: it is made up of the numerous sensors that protect the BattleBot from a mechanical and electrical perspective. Voltage, current, and IMU sensors are all placed at critical points of interest and then fed into the ESP32 microcontroller for closed-loop control. Finally, the microcontroller subsystem has several layers to it. It will communicate to the motor drivers and other components using the Arduino IDE, and a wireless communications modem will be initialized on the microcontroller. Finally, the numerous aforementioned feedback loops will feed into it, and in turn, responsive feedback control will be outputted.

[2] Physical Design:



This diagram depicts the mechanical dimensioning and placements of components to be placed in proximity to/inside the chassis. Certain dimensions are provided through their datasheets, while the PCB requires more R&D time for its specific dimension values. This visual aid should also bring attention to the additional space left in the chassis for future additions/room for further complexity.

[3] Power/Electrical Subsystem:

The power subsystem oversees the energization of all other subsystems (Microcontroller, sensors, and drivetrain) involved in this project. The LiPo 4S battery was specifically chosen to provide a clear balance between power and charge capacity. This way, any concerns of non-idealities in capacity storage capabilities and/or not lasting the entire match duration are null possibilities. Additionally, the loads (primarily the motors) are expected to have high current ramp rates due to the battlebot movements' necessity to be nimble and responsive to threats. This battery configuration (4S) achieves this purpose. A step down regulator will be needed to achieve the desired 3.3v to power the microcontroller, IMU, and current sensors. Another stepdown will be used to step down the voltage to around 4-5 volts to power the DC Brushed Motor Drivers for the drivetrain as well as the weapon systems.

Requirements	**Verification**
A stop button that can be pressed to stop the robot from operating and the robot must be shut off in 60 seconds or less.	Perform trial runs with the robot and during it perform a safe shutdown of the robot by pressing the button on the robot and record with a timer how long it takes for the weapon and motors to stop rapidly spinning for safe handling.
Voltage Regulator can supply a stable input voltage of 3.3v with a ±2% tolerance for microcontroller, IMU, and current sensor voltage input.	Measure test points at voltage inputs and observe under an oscilloscope that the voltages will be stable while components are running.
Voltage Regulator can supply a stable input voltage of 5v with a ±2% tolerance for the motor drivers.	Measure test points at the motor driver's voltage input pin with an oscilloscope and see if the supplied voltage will be stable while motors are running.

[4] Mechanical Subsystem:

There were several considerations to be conscious of when designing the mechanical subsystem. Given the 2-lb weight limit, there were several mechanical-related components that needed to be incorporated and/or optimized for project use. The selected motors (including their drivers) were chosen due to their lightweight, reliable, and effective capabilities. The Repeat Robotics Brushed Drive Gearmotor (3mm shaft variant) checked all of the boxes for this project. It is compatible with 2S-4S voltage (perfect for our 4S battery), weighs approximately 0.05lb per gearmotor, offers ~1.2kg/cm of torque, and is rated for 1220 RPM. This is a Brushed DC motor, which is arguably the most commonly used gearmotor in BattleBot context given its effectiveness. Brushless DC motors have been increasing in use cases, yet they are scarcer to come by, and even harder to find when searching for particular mechanical and/or electrical specs. Given that the gearmotor variant chosen was the 3mm shaft variant, the wheels naturally needed to have a 3mm axle. Several prospective wheel types were chosen, and polyurethane seemed to provide the most grip and durability for what is requisite of BattleBots. In comparison, 3D printed wheels are easily replaceable, but wear out significantly and do not absorb shock very well. Furthermore, because a 2WD system is being utilized, the wheels are required to be larger than within a 4WD use case. The wheel diameter is to be 1.5”, or 38.1mm. A third motor will additionally be employed – not for the wheels, but for the active weaponry (a drill). The drill will have screws threaded into it from the top and bottom in order to pinch the axle protruding into the drill. A centralized chassis will hold the motors, PCB, microcontroller, and sensors. This chassis will be connected to the outer shell via mounting holes. Then, a bar-like skid will sit at the back of the chassis. A bar was chosen over a sphere or other shape due to its easy replaceability and because it behaves as a stabilizer when the BattleBot becomes inertially volatile. Finally, a spike-covered chassis ends the mechanical subsystem, providing ample protection to all internal components while extruding both passive (spike) and active (drill) weaponry.

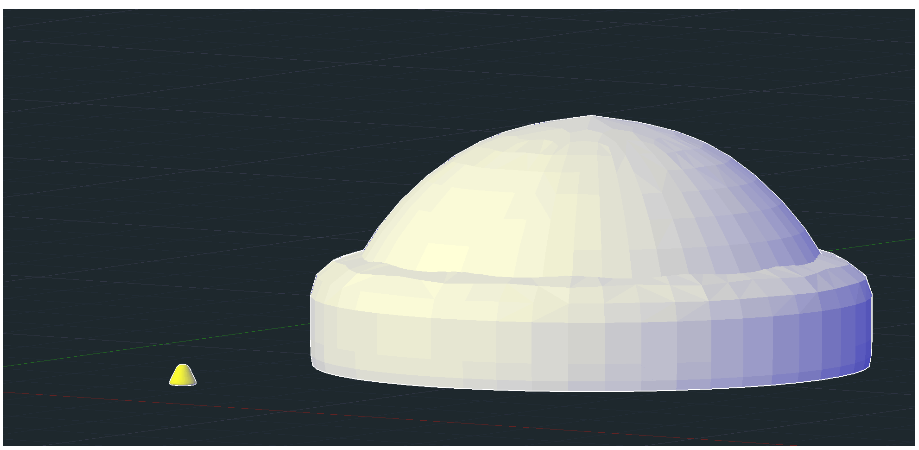


Figure 2: This depicts the rudimentary 3D modeling prototypes for the shell and spike. To be modified further with additional components (drill and chassis) included.

Requirements	**Verification**
The motors must not experience overloading conditions. Current limitations will be set at 90% of the rated stall current value, and voltage limitations will be set at 16.8V (fully charged 4S LiPo battery) such that	DMMs and feedback from the installed voltage, current sensors will verify whether this requirement is met.

Requirements	Verification
the motor has ample wiggle room for functioning OCP/OVP without false positives triggers.	
The drill, skid, and wheels must be easily replaceable.	The drill, skid, and wheels will be repeatedly attached/deattached to verify that they can be easily replaced in a non-damaged state. Then, deformities will either be theorized (or physically applied when components become damaged during trial runs) on these components, and whether they remain easily attachable/detachable will be verified.
All components are designed to operate in a room-temperature environment with a plus-minus tolerance of around 5 Celsius.	Trial runs will be run in room temperatures for extended periods of time (to thermal temper components), and functionality of each and every component, namely electrical, will be validated through the microcontroller's numerous feedback loops. If closed-loop response/wireless communication fails, it may suggest issues with the microcontroller itself.

[5] Microcontroller Subsystem:

The microcontroller subsystem serves as the main control center of the BattleBot. It owns all I/O operations that the user intends to operate on the device and is responsible for important algorithmic functions to detect and prevent many critical physical complications that *could* arise during the competition. To aid our decision on which specific microcontroller configuration to pick, we had to consider many tradeoffs and benefits that were available. Firstly, we had to decide between using WiFi or Bluetooth to connect wirelessly to the laptop to offer control to the bot. In this design, we decided to use WiFi as it offers lower latency and better range as opposed to bluetooth. Furthermore as it was a product requirement to use wireless communication, we also opted to use an ESP32 configuration as opposed to the STM32 configuration as the ESP32 offered native support for WiFi or Bluetooth as opposed to the STM32, where we had to purchase and configure a separate WiFi MCU (Microcontroller Unit), placing additional pressure on overall performance, budget, footprint and weight. For the development, the Arduino IDE will be used to write Arduino code. From the client side (wireless laptop), we will need a way to transmit our control inputs wirelessly to the ESP32, and vice versa. The best approach for this would be to configure the ESP32 to host a webserver that's connected to a shared router. We can then open a websocket connection from the client to the board to offer input requests and responses over one continuous connection. This server can be accessed through WIFI in a browser on an agreed-upon port (think :8000). The payload will be transmitted through this websocket that is established upon boot time containing the values for the keys that are pressed along with an acknowledge bit from the server to ensure that our keystroke has been received all and well.

All I/O operations on board will be communicated and transmitted using the SPI communication protocol. For the communication protocol, our two main considerations were between SPI and I2C. We chose to use the SPI communication protocol as opposed to the I2C protocol because there are two different lines for sending and receiving. This allows for us to send and receive signals through our server simultaneously.

Requirements	**Verification**
Latency from input to output via the websocket should be under 1 second. With a tolerance of 20ms	This can be measured by tracking the elapsed time between sending the payload through the websocket and receiving the acknowledge bit from the microcontroller

Requirements	Verification
Memory and CPU power should consistently be well under threshold for microcontrollers. Specifically, RAM usage should be < 2MB with tolerance of 30kb	Can track critical system benchmarks like the stack/heap and report back to the host device. Upon exceeding these numbers potentially can reset the system as a whole to reset all cache.
Code must have >80% test coverage for system critical functions.	Tests can be written using a framework natively supported in C such as µnit. Running the test suite summary yields the percentage of main function code that's covered.

[6] Safety/Reliability Subsystem:

Due to the extreme nature of a battlebot competition, ramp currents and collisions are expected to be quite extreme. The system will also need to step-down voltage from 14.8V to signal/comms voltage levels. It is imperative to not bank on the notion that our components will function perfectly as intended when sewn together with step-down voltage regulators. Additionally, contests between two opponents are commonly decided once one robot capsizes. To prevent this, an additional layer of preventative design is employed through a closed-loop feedback loop between the IMU and the ESP32. This protection subsystem was deemed necessary to uphold overall system reliability. In lieu of the motors becoming overloaded and/or the presence of electrical faults, current and voltage sensors will be scattered at critical PCCs (points of common coupling) to monitor any long-term threshold violations that cannot be classified as transients/noise. As explored in the RV table below, current and voltage limitations for all relevant components have been explicitly explored.

Requirements	Verification
All components with electrical input must not experience overloading conditions. Current limitations will be set at 90% of the rated stall current value, and voltage limitations will be set at 16.8V (fully charged 4S LiPo battery) such that the motor	The current and voltage sensor communication to the ESP32 should yield derating responses in the event these OVP/OCP limitations are encroached. This is how the safety/reliability system components will be validated.

Requirements	Verification
has ample wiggle room for functioning OCP/OVP without false positives triggers. Per the ESP32 microcontroller datasheet, the measured voltage must be between 3V and 3.6V. OVP will be set at 3.6V, which is defined by the manufacturer as “unsuitable for long-term operation” and “may cause permanent damage to component if exceeded”. Fortunately, for the BDC Motor Drivers, they have internally-built current sensing and regulation limits, consequently not requiring redundant sensors.	
The BattleBot must have an improved response to inertial distress with the assistance of the IMU sensors.	The IMU sensor communication to the ESP32 should coerce a response command to the motor that reduces the chance of flipping over. Different approach speeds and angles of collisions will be repeatedly tested with the IMU sensors enabled + disabled to qualitatively assess their usefulness.

[7] Tolerance Analysis:

After discussion with Zhuoer, it has been mutually agreed that the wireless communications interface is the most critical requirement for this project. Without the functionality of this aspect of the project, the BattleBot is deemed useless. From a mathematical standpoint, the connectivity capabilities can be tested by sending commands to the Battlebot from a connected laptop with set distances. Distances of 2-3 inch increments will be measured, and connectivity strength will be assessed through any trends in latency and/or signal receival failure rate.

[3] Cost and Schedule

[1] Cost Analysis:

Component/Service	Cost ($)	Quantity	Manufacturer	Part Number
Adafruit ESP32 Feather V2	$19.95/per	1	Adafruit	ESP32-PICO-V3-02
BRUSHED Drive Gearmotor	$19/per	3	Repeat Robotics	rr-brushed-3mm
Brush DC Motor Driver	$8.15/per unit	3	Polulu	DRV8876
Polyurethane Wheels	$2.49/per unit	4	BenchCraft	BCT5016-055
4S (14.8V) 750mAh LiPo Battery	$18.59/per unit	2	Tattu	B097BY6Z43
PETG 3D Printing Costs	$10-15/per lb (let's say 15$)	~1 to 1.5lb		
Current Sensor	$4.42/per unit	5	Texas Instruments	TMCS1123C4AQDVGR
IMU Sensor	$19.95	1	Adafruit	ISM330DHCX
3.3V output Voltage Regulator Step Down	$6.95 per unit	2	Pololu	APM81815
5V output Voltage Regulator Step Down	$6.95 per unit	2	Pololu	APM81815

	TOTAL: $233.39			

In the event this project were to be repeated, the financial aspects of it would be changed. Namely, labor would be included. Assuming minimum wage ($15/hr for Illinois) and a work week of ~8 hours, this semester-long project (16 weeks) would yield a $1,920 payout to each individual working on this. With three partners, $5,760 becomes the grand total for labor. There are numerous other aspects that must be considered for cost analysis, specifically the cost to 3D print PETG and component costs. The above table has these covered.

Week of	Task	Persons
March 6, 2026	Receive requested parts Continue working on the 3D CAD models (chassis, drill, shell).	All members of group
March 13, 2026	Gather microcontroller, and start prototyping code for microcontroller. - Set up websocket - Experiment with WiFi connectivity and study latency thresholds as defined in the requirements Solder PCB components and begin testing PCB. Begin 3D printing copies of the 3D CAD models. Start mechanical tolerance testing.	Ved (Microcontroller) Tim(PCB/Hardware) Matt(Mechanical & Hardware Support)
Spring Break	Get motors and sensors to start collecting data and sending it back to client through WiFi Determine if PCB needs another order and work on ordering another if necessary. Continue mechanical tolerance testing.	Ved (Microcontroller) Tim(PCB/Hardware) Matt(Mechanical & Hardware Support)

March 27, 2026	Design GUI and setup payload skeleton for interpreting keystrokes and sending through wires	Ved (Microcontroller) Tim(PCB/Hardware) Matt(Mechanical & Hardware Support)
April 6, 2026	Progress Demo	Everyone
April 13, 2026	Last Minute Adjustments	Everyone
April 20, 2026	Mock Demo	Everyone
April 27, 2026	Final Demo	Everyone
May 4, 2026	Work on Final Paper	Everyone

[4] Discussion of Societal Impact, Engineering Standards, Ethics, and Safety Considerations:

Our project makes a positive impact to public health, safety and welfare by considering the practical applications of the technologies and strategies that we used to integrate all of our separate independent subsystems and how they can relate to the real world. Some examples of how the core functionalities of our battlebot can be used is in the health sector. As technology advances, robotics plays an increasing role in performing crucial surgeries and powerful operations. We follow the IEEE Code of Ethics and Engineering Standards with respect to safety and having respect with each other. IEEE code of ethics l.1 prioritizes the safety of all individuals involved. The addition of a turn off button and a proper procedure in turn on and off the robot to protect people from accidental harm from the robot weapon. There is a true risk of unintentional harm to competitors during competition, which is the central ethical issue with regards to this project as many robots can be capable of seriously injuring people. Another aspect of the code of ethics that has been obliged has been the open design discussions that have been had between individuals in the group in respectful manners. We've also been documenting individually and as a group the design discussions as well as any major issues that have come up as a group. As stated in our team contract we strive to have empathy with each other in turns of what's going on in all of our busy schedules and that shows respect for each other as individuals in accordance with IEEE Code of Ethics l.5.

Electrical and Mechanical safety included in the project feature current sensors that will connect with the microcontroller to limit the motors in the event that it is drawing too much current which would be detrimental to the robot as there are risks of electrical fires that could risk the safety of all participants involved. If the motors overdraw current that will also result in a mechanical risk of the motors suddenly stopping which could cause the robot to spin out of control and act irrationally in movement. We want to minimize irrational movement of the robot as that could increase the risk of injury due to unexpected collusions with the competing robot and parts may fly out of the arena. In addition, we have a stop button that will allow for easier handling of the robot during set up to avoid any injury before and after the competition time.

[5] Citations:

[1] *Repeat Robotics Brushed Drive Gearmotor*. ItGresa Products: Robotics and Small Business Support. (2026, February 23). https://itgresa.com/product/repeat-robotics-brushed-drive-gearmotor/

[2] IEEE code of ethics | IEEE. (n.d.). https://www.ieee.org/about/corporate/governance/p7-8