University of Southern Indiana
Pott College of Science, Engineering, and Education
Engineering Department

8600 University Boulevard
Evansville, Indiana 47712

The Creation of Battle Bots: iRobot Roomba Conversion

Joshua M Whaley, Paul B. Hart, Clifton Campbell, Frederick H. Buehl II, Patrick N Bruner
ENGR 491 – Senior Design
Fall 2021

Approved by: ____
Faculty Advisor: Brandon S. Field, Ph.D. Date

Approved by: ____
Department Chair: Paul Kuban, Ph.D. Date

ACKNOWLEDGEMENTS

In this section, we would like to acknowledge and thank our advisor Dr. Brandon Field for providing feedback and keeping the design and build process on course, Dr. Paul Kuban and the University of Southern Indiana (USI) Engineering department for organizing and funding the competition, Dr. Todd Nelson and the entire USI Engineering staff that helped answer questions and provide feedback, Justin Amos and USI's Applied Engineering Center for assistance in the water-jet cutting of the frame and chassis components of our robots, Westgate Academy for hosting the iRobot Roomba tech-to-engineer cohort competition, and NSWC Crane and Crane Division University for developing the Technician-to-Engineer Program. Finally, a special thanks to the families and friends of the technician-to-engineering cohort members that have been understanding and supportive throughout the four and a half years in the tech-to-engineer program.

ABSTRACT

The purpose of this project was to provide an opportunity for the seniors in the USI/NSWC Crane technician-to-engineer cohort to work in a small team setting and manage to completion, a project tasking of creating user controlled, battle type robots from an iRobot Roomba Vacuum platform. This project covered a wide variety of aspects including the design, construction, modification, and programming of the battle-bot. The ten members of the cohort were split in two, five member teams that were required to create at least one battle-bot per team. The bots would have to pass a speed test and a maneuverability test as required by the rules set forth by USI. After passing qualifying tests, both teams would battle in a three round main event, until a team's bot expires or the three rounds end. The teams will be judged on the overall design, creativity, quality, and competition results.

Early in the process, "DOOMBA" was decided as a team name. The name represents the roots of an iRobot Roomba vacuum system and the certain doom that would be brought to the bot's opponents. With many tasks and a variety of areas from programming, frame construction, assembly, lab tests, interfacing, and proper teamwork, communication was vital to the success of the project. Our team quickly setup a file share drive, text chain, weekly electronic conferences and secured a workshop facility to utilize in the battle bot development. The team originally focused on creating a single battle-bot due to worries of budget constraints. Once budgetary requirements for a bot were calculated, it became evident that a secondary bot was financially feasible. Discussion then ensued about offensive weaponry and defensive tactics the bots should have. The team quickly decided that the two bots should have different weaponry and exterior makeup. This decision was driven in order to diversify skills and risks. If a bot's weapons were ineffective against the opponent, the hope would be that the second bot would have a more effective set of attack. Defensively, if one of the bots were susceptible to the opposing team's bot, then possibly the second bot would be less affected by their weaponry and mode of attack. The process began with the team studying champion level bots frequently seen on the television series Battle-Bots, along with You-Tube videos. After much thought and brainstorming ideas, it was decided to go with a wedge style bot (this became DOOMBA Dozer) and a second bot with spinning blades (this became DOOMBA Saw). The wedge style bot would be constructed of heavy material and its offense would be ramming or pushing the opponent. The defense would be the

capability of taking repeated hits with heavy construction. The spinning blade bot offense would focus on hitting or cutting the opponents housing. The defense would be agility, due to the lighter weight construction.

Through various question and answer sessions with the team's advisor Dr. Brandon Field, it was determined that using the original wheel assemblies of the Roomba platform, was sufficient enough to be considered a "Roomba Platform". The body and other components were not a criterion for the project, thus providing a large amount of freedom with respect to restrictions on the bot's construction and makeup. Work began with scrapping the internals of the Roomba and replacing the electronics with an Arduino focused on controlling payload and another Arduino focused on controlling wheel motor speed and direction for each bot. A reduced C\C+ language was used to program the Arduino Uno's. The Uno's were used to control the weapons and steering capabilities for DOOMBA Saw as well as DOOMA Dozer. Once the programming proved functional with all components integrated, the housing design process was initiated. At that time, the team made the decision to split into two teams. One team made up of three working on the Dozer and the other team made up of two people working on the Saw. This was enacted with the understanding that all personnel's skill sets and capabilities were available to both subgroups at any time, to assist in completion of tasks. In the end, team DOOMBA created two different battle-bots named "DOOMBA Dozer" and "DOOMBA Saw".

TABLE OF CONTENTS

2021 Making of a Battle Bot: iRobot Roomba Design, Conversion, and Implementation

TABLE OF CONTENTS: Cont.

2021 MAKING OF A BATTLE BOT:

IROBOT ROOMBA DESIGN, CONVERSION, AND IMPLEMENTATION

1.0 Introduction

The purpose of this project was to provide an opportunity for the seniors in the USI/NSWC Crane technician-to-engineer cohort to work in a small team setting and to manage to completion, a project tasking of creating user controlled, battle type robots from an iRobot Roomba Vacuum platform. This project covered a wide variety of aspects including the design, construction, modification, and programming of the battle-bots. The ten members of the cohort were split into two, five member teams that were required to create at least one battle-bot per team. The bots must pass a speed test and a maneuverability test. After passing the qualifying tests, they would be allowed to battle in a three round main event against the other team's bot(s). The teams would be judged on the overall design, creativity, quality, and competition results. Our team chose the moniker "DOOMBA", representing the certain doom that would be caste upon our opponent's bot(s), as well as representing the bot's Roomba roots.

2.0 Competition Background

The 2021 USI / Crane Cohort Battle Bot Competition is the first USI/Crane sponsored battle bot competition. This competition will take place on December 14th, 2021 in the "big room" at the WestGate Academy located at Crane Technology Park. It is located in southwestern Indiana next to Naval Surface Warfare Center Crane Division. The competition allows for the ability to highlight the first graduating cohort of the USI/Crane Technician-to-Engineer program. The program facilitates the ability of NSWC Crane technicians, programmers, drafters, and the like to attend the University of Southern Indiana in the pursuit of an engineering degree.

3.0 Battle Area Description

The main arena of the competition is set to be a raised platform of a length 24' x 12'. This platform will be used for 3 separate stages of the competition. Specifications and layout can be viewed in Figure 1.

1. Speed Challenge
2. Maneuverability Challenge
3. Battle Royal

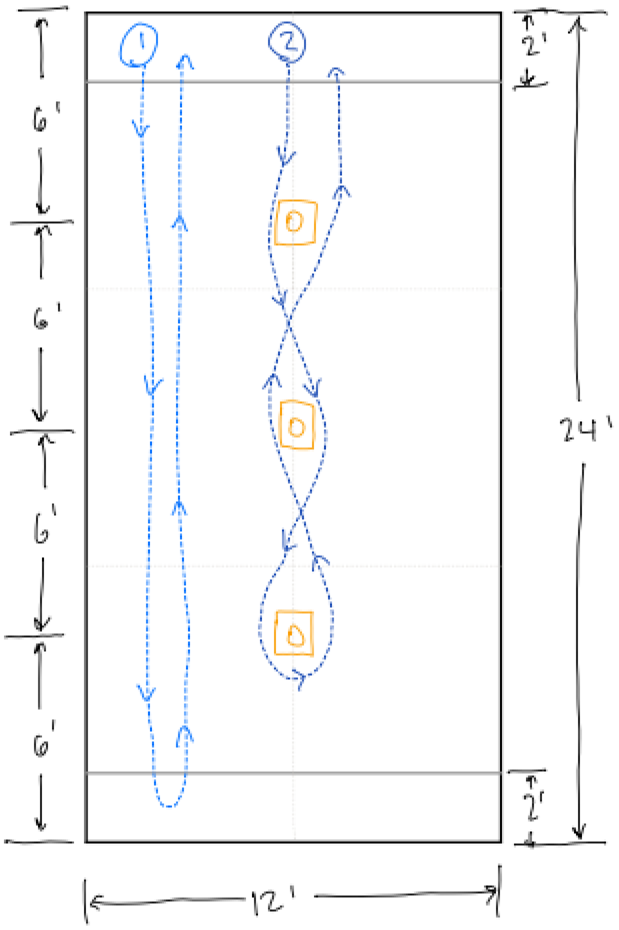


Figure 1: Specifications & Layout for the Competition Platform

4.0 Competition Rules

Below is a summary of competition rules that directly influenced the robot design and the strategy used.

4.1 Robots

The robot shall be based off an iRobot Roomba platform. It shall not use fire, explosives, untethered projectiles or signal jamming.

4.2 Procedures

1. Speed Challenge – The robot should traverse a 20-foot distance, then back 20 feet to its origination, all in under three minutes.
2. Maneuverability Challenge – The robot should traverse a slalom course of obstacles in under 5 minutes.

3. Battle Challenge – The robot should be remote controlled. Battle bots must fight until disqualification happens or until three rounds have ended. The team with the most points wins the competition, unless bots become unresponsive, which is automatic disqualification.

4.3 Scoring

Points will be awarded for design functions, completing certain challenges, and for performing certain tasks during the 3-round competition. Deductions will be awarded if certain tasks are incorrectly performed or rules are violated. A full list of the point breakdown can be viewed in APPENDIX A**Error! Reference source not found.**.

4.4 Format

1. Each team will have three 10-minute rounds.
2. Each team's score will be the total of their 3 rounds.
3. Each team will have three 2-minute time outs.

5.0 Robot Design and Construction

Being an open-ended project, many options were available when choosing the design and components for the robot. The dozer robot was designed in Solid Edge modeling software produced by Siemens and a Finite Element Analysis (FEA) was performed on a custom blade using Fusion 360 by Autodesk. Using modeling software allowed for easier modifications to the robot design, without compromising the accuracy of custom parts. This also allowed any future modifications to be virtually fitted with the current chosen parts before any construction was completed.

5.1 Roomba Teardown

The green and grey circular device seen in Figure 2 depicts the initial Roomba that was purchased through USI. Figure 2 also depicts the Roomba iRobot tear down process. The housing, harness, sensors, control board, and battery from the bot were discarded. The team felt as if the plastic housing would not withstand repeated attacks from the opposition. Since the wheel assemblies were a weight and speed limiting aspect of the Roomba, the wheel assemblies were utilized on both bots, to ensure the bots were based on a Roomba platform.

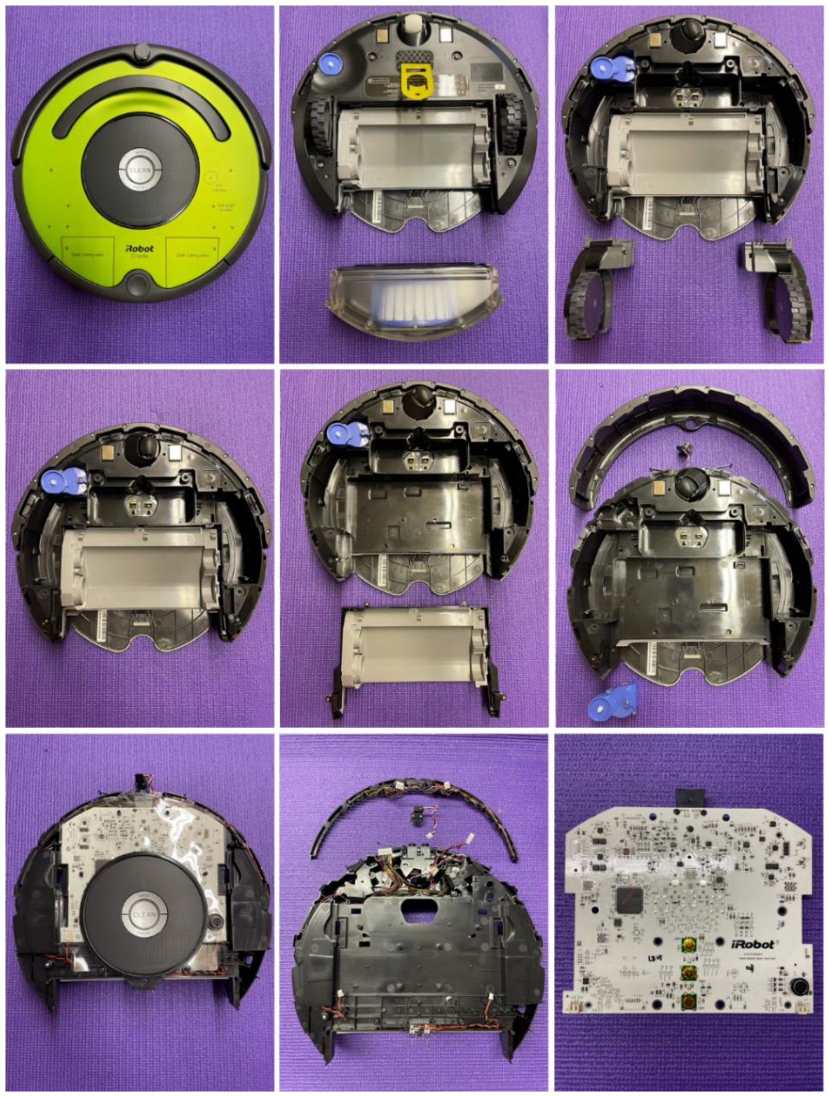


Figure 2: iRobot Roomba Teardown

5.2 *Electronic Control Systems*

The electronic control system for each bot is shown in Figure 3 and Figure 4. The common parts of each system are highlighted in blue. The unique components for each system are highlighted in white. Common components include a 5-volt regulator, an Arduino for payload control, an Arduino for drive control, H-bridge motor driver, 20-volt power source, rear wheel assembly, one or two sets of 4 AA battery packs (6-volt source), 2.4 Giga Hertz (GHz) transmitter/receiver, and an E-stop for safety. The Dozer has unique parts as shown in Figure 3. They are a sound board, a speaker amp board, an extra front wheel assembly, a 940 Nano-meter light emitting diode, and a left and right set of speakers. The Saw has unique parts as shown in Figure 4. It consists of two relays, two angle grinders, and a 21-volt power source. Each electronic control system has 20 volts, 6-volts, and 5-volt connections, as shown in each block diagram. Full wiring diagrams can be seen in APPENDIX and APPENDIX .

Thanks to team member Patrick Bruner being certified, all soldered electrical wiring and soldered electronic assemblies on the DOOMBA saw were soldered and inspected in accordance with J-STD-001: Requirements for Soldered Electrical and Electronic Assemblies. All assemblies on the DOOMBA saw were also soldered and inspected by a certified IPC specialist (CIS).

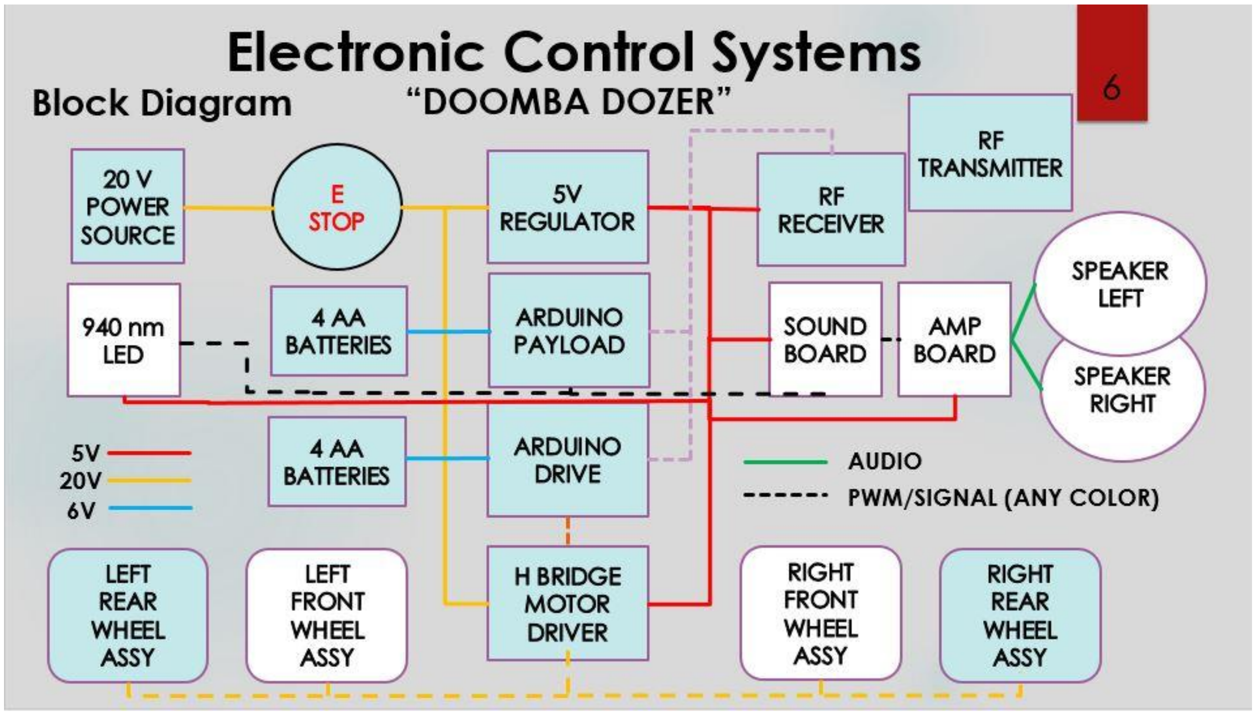


Figure 3: DOOMBA Dozer Electronic Control System Block Diagram

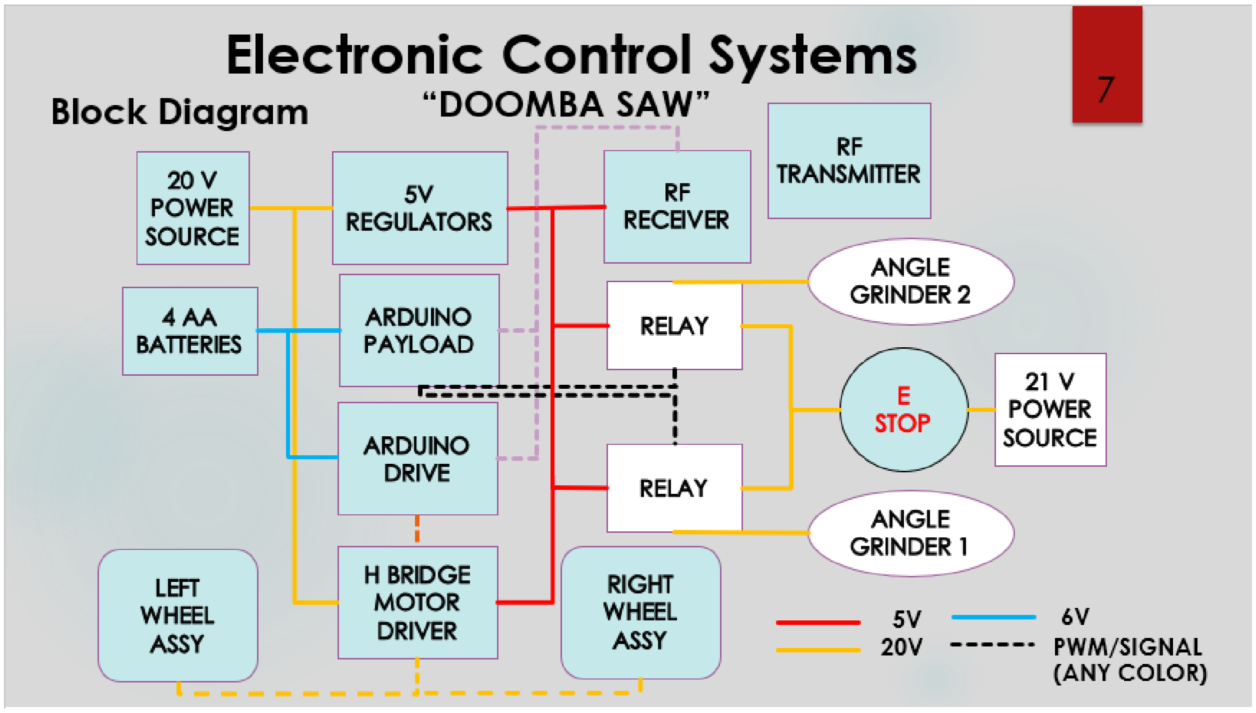


Figure 4: DOOMBA Saw Electronic Control System Block Diagram

5.3 *Common Components*

A clone microcontroller similar in construction to an Arduino Uno R3 was chosen as the DOOMBA's microcontroller for drive as well as payload control. This selection was made based on cost and availability. It was also chosen because of the team's familiarity with C programming due to multiple USI programming classes that the members had taken. The Arduino used a reduced C/C++ programming language. Two microcontrollers per bot were chosen to separate the drive logic from the payload logic, to ensure top responsiveness in both aspects. The board has an Atmel ATmega328P chip for processing. For peripherals, the board includes 14 digital I/O pins, 6 of which can be used for pulse width modulation (PWM) and 6 analog pins that can be used as digital pins. The microcontroller's timing is kept using a 16 MHz ceramic resonator. The microcontroller that was used can be seen in Figure 5.

Once received the ELEGOO Uno R3 boards were inspected in accordance with IPC-A610: Acceptability of Electronics Assemblies. The inspection was checked for completeness since the micro-controllers were Arduino substitutes. All clone Uno boards passed inspection.

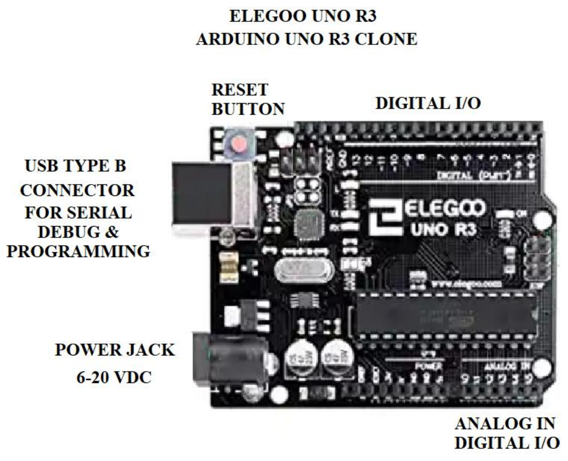


Figure 5: ELEGOO UNO R3 ARDUINO CLONE

A 6-volt battery pack was used to supply the Arduino payload as well as an additional 6-volt battery pack for the Arduino drive. Both bots used this pack, however, due to space restrictions, the DOOMBA Saw only used one pack split between two Arduinos. Figure 6 depicts the 6-volt LAMPVPATH battery pack. Each pack holds four AA size, 1.5 VDC alkaline batteries.

Figure 6: LAMPVPATH AA Battery Pack

The team discussed what type of controller would be used. There was a debate between using a Bluetooth game controller or an RC type car/helicopter controller. With a longer range and a proven history of adaptability, the RC type controller was selected. A trigger and wheel style remote were first researched but the lack of additional channels for the battlebot's offensive payloads were needed. Further searching led to airplane and helicopter type RC controllers which have 8 to 12 controllable channels. The initial expected cost of this style controller was sizeable, but after researching popular retail sites, it was found that controllers with 6 to 10 channels could be obtained for less than 7% of the budget.

A programmable FLYSKY FS-i6X RC transmitter controller was used to navigate and operate the weapons and navigation of each bot. An FS-iA10B receiver, with 10 controllable channels, were used to read the transmitter controller and send input signals to the Arduinos. The default output signals of the receiver are pulse width modulated signals. The Arduinos have the capability of measuring pulse widths, so the pairing seemed to be a natural fit. Figure 7 depicts the transmitter controller. Figure 8 depicts the receiver component.

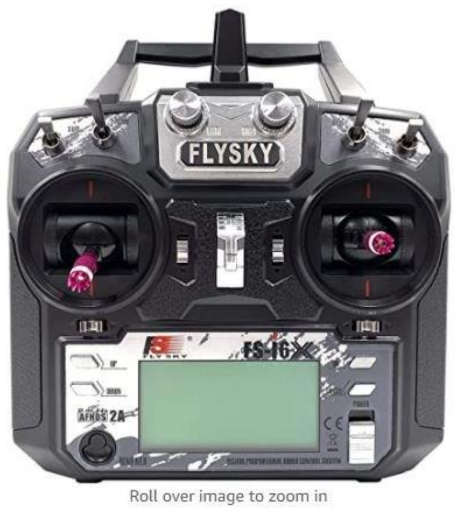


Figure 7: FLYSKY Transmitter Controller

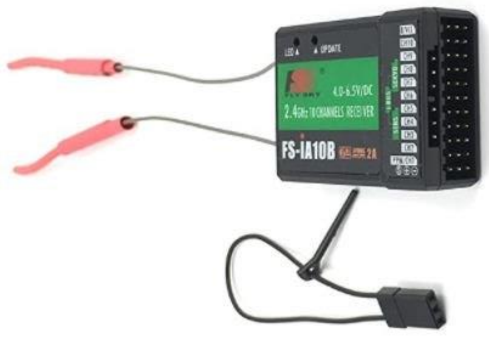


Figure 8: FS-iA10B Receiver

A few different motor controllers were discussed for use during the design stage. MC33926 based dual motor controllers had been used before by one of the team members for work applications. Further retail store searching revealed the DROK L298 Dual H Bridge Motor driver board. This board was much more economical for the project and had higher current output per channel (7 Amps) than the MC33926 board. The dual H bridge provides the capability of forward and reverse driving of the wheel assemblies as well as enabling graduated speeds by using pulse width modulation signals from the output pins from the microcontroller. The DROK L298 Dual H Bridge Motor driver can be seen in Figure 9.

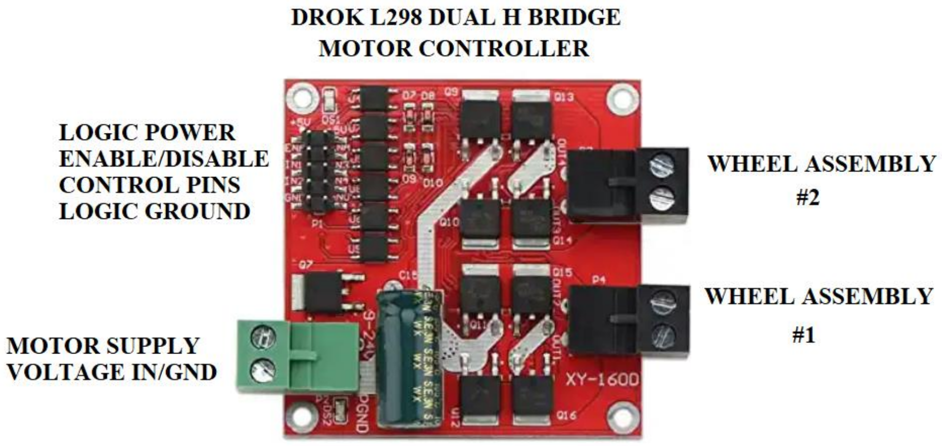


Figure 9: DROK L298 Motor Speed Controller

A DeWalt DCB204 lithium-ion battery was used for the 20-volt power source in each bot. Battery adapters for this style battery were easily found at online retailers due to the growing popularity of modifying children's motorized Power Wheels. The battery packs have a 4 Ah output rating. This would provide the capability of outputting 4 Amps for one hour. Considering the wheel assemblies consumed around 0.125 Amp each, the consumption rate of the bot was well under 4 Amps. The battery and its adapter can be seen in Figure 10.

Figure 10: DeWalt 20-Volt power source and adapter

In both DOOMBA configurations, the 20-volt source supplied a 5-volt regulator and H-Bridge motor driver, as referenced in Figure 3 and Figure 4. A DROK voltage regulator was used to convert the 20-volt DC battery power into a 5-volt DC power in this particular application. Multiple components on the bots required a 5 V logic voltage to be supplied to them besides the input signals. The motor driver board, the sound board and the speaker amplifier were the main components. The bots needed a method to down convert the 20 VDC from the main supply down to a 5 V logic level. Multiple online retailers were researched in order to find a buck 5 VDC regulator. After a fairly short search, the DROK step down or "buck" regulator was found at a low price that allowed for voltage inputs of 4.5-24 VDC, with the capability of multiple fixed outputs or an adjustable voltage. The units have 20 mV or less of ripple voltage and are switched at 500kHz. The regulator was used for a fixed 5.0 VDC output by shorting a couple pads on the back of the board with solder. The physical size of the regulator is very small taking up less footprint than a quarter. The regulator can be seen in Figure 11 below.

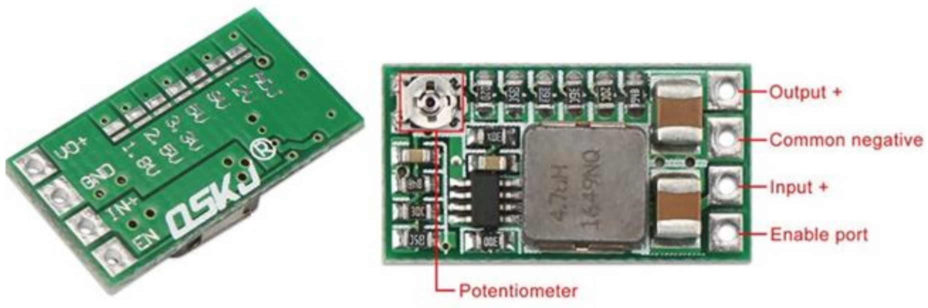


Figure 11: DROK Voltage Regulator

For safety concerns with the weaponry of the DOOMBA Saw in particular, an E-stop was added as a feature. A Mxuteuk HB2-ES544 normally closed red mushroom emergency stop push button switch was installed, as shown in Figure 12. It has an operating voltage of up to 600 volts and 10 amps. As shown in Figure 4, the E-stop was inserted between the 21-volt power source and angle grinders, thus disabling the weaponry by the push of a button.

The safety feature was added to the DOOMBA Dozer as well. Although the Dozer does not have the dangerous weaponry of saw blades, the decision was made to incorporate it into the Dozer as a matter of overall safety. Because the weaponry system on the Dozer has no lethal capability, the safety feature was incorporated in a different location. As shown in Figure 3, the E-stop is located after the 20-volt power source, before the 5-volt regulator, and the H-Bridge Motor Controller. With a push of the emergency stop, the Dozer affectively loses power to the wheel assembly and to the audio amp, sound board, and LED. It effectively has no function once the emergency stop is engaged.

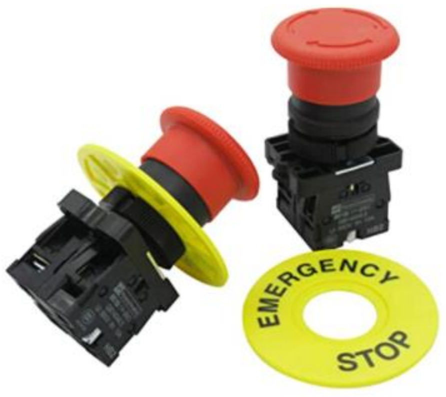


Figure 12: Mxuteuk HB2-ES544 NC Emergency Stop

5.4 Unique Components – DOZER

5.4.1 Chassis and Housing

For the chassis of DOOMBA Dozer, accuracy of wheel location became more critical with the use of four drive wheels instead of the standard two. To achieve this accuracy, 1/8” steel was cut using a water jet into the design specifications that were created in the modeling software. The model drawings that the waterjet pieces were derived from can be seen in APPENDIX . The frame being cut and the water jet machine located at the USI Applied Engineering Center can be seen in Figure 13.

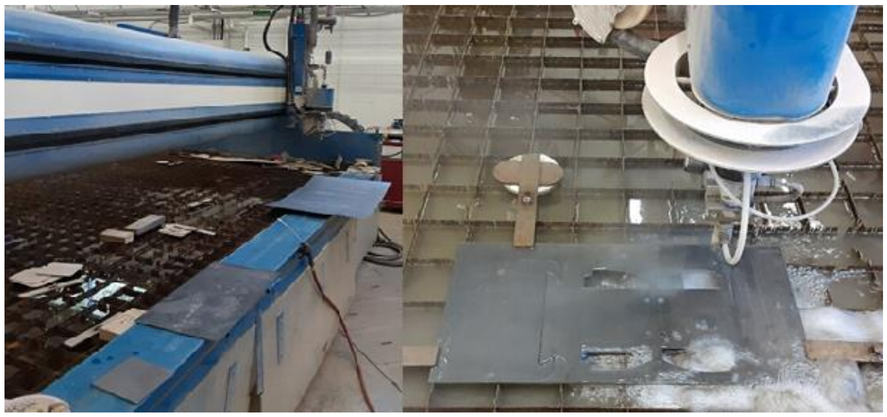


Figure 13: Water jet machine and DOOMBA Dozer chassis

The wheel cut outs were initially cut in the same direction, as seen on the left of Figure 14, but during prototyping it was noticed that the mobility could be improved by rotating the front wheels 180 degrees as seen on the right of Figure 14.

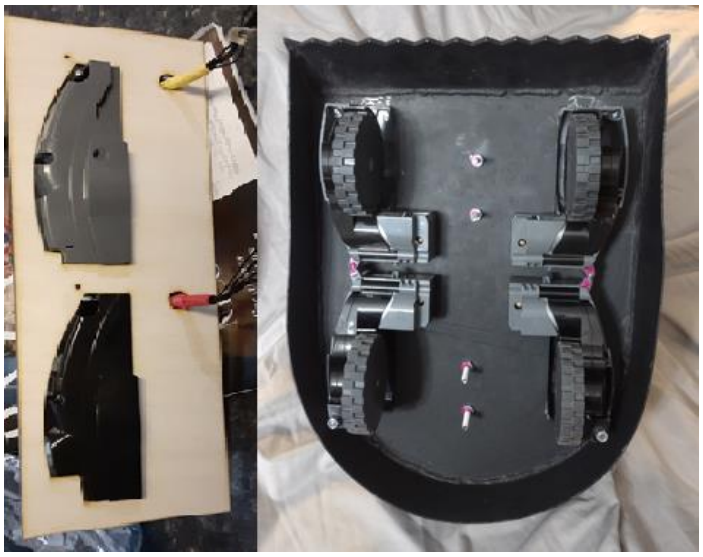


Figure 14: Prototype wheel mounting configuration

This moved the center of the turning radius closer to the center of the bot. The frame of the DOOMBA Dozer determined the overall size of the bot. Since steel was to be used, it was determined that the size of the bot needed to be as small as possible to reduce the overall weight of the frame. This would help with speed and maneuverability yet maintain a solid platform. It was decided to mount all the sensitive electronics in its own enclosure and in the center of the frame.

The battery was mounted on the top for easy access, thus making it easier to change the battery if needed during the competition.

The housing was made with 1/16” steel. DOOMBA Dozer utilizes speakers with sounds that can be activated by the operator. To make the speakers easier to hear, the speaker grills were cut into the body along with the required mounting hole locations. The water jet machine was used for precision.

It was found during the use of the waterjet that slots may have been a better choice for the speaker grills. The many small holes added considerable amount of time to the cutout. The top cover of the bot and the battery housing walls were also cut using the water jet. The top cover was mounted on top of the main body with tabs folded down with pilot holes added to aid in securing the lid with self-tapping screws. The battery compartment was welded together and then welded on top of the lid. The battery cover was designed to be secured the same way as the lid of the bot. With the use of the water jet, all parts to be welded together were cut to the precise size and angle needed for assembly. This made everything line up much easier for assembly.

For the Dozer plow, 1/8” Steel was cut using the water jet. An online resource about bulldozer blades indicated the optimal angle for a push blade was up to 30 degrees (“Types of Bulldozers and Their Uses”). Since the goal of the dozer was only being a push bot, the plow angle was then set to 25 degrees. Once the housing was cut out, the pieces were welded together. All welds were made using a Lincoln Electric Power MIG 180 Dual MIG, Flux-Cored welder. This was utilizing the flux cored process with a E71T-GS electrode installed. All welds applied were fillet welds giving us the greatest strength for our joint configurations. The strength of these type welds was previously witnessed in our strength and materials class previously taken at USI. Figure 15 depicts the DOOMBA Dozer chassis and housing hull just after welding. Final weight of DOOMBA Dozer came up to 15.5 pounds.

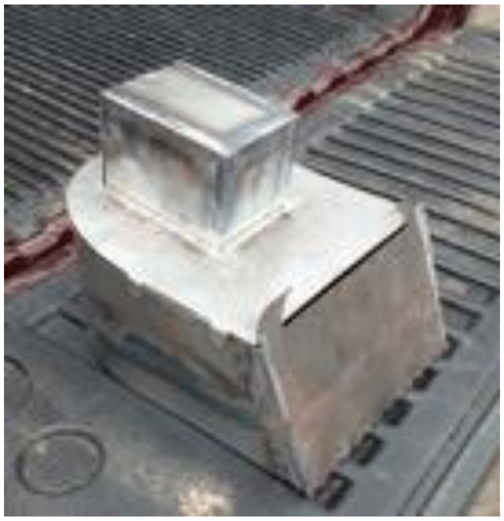


Figure 15: DOOMBA Dozer Body

5.4.2 Electronic Control System

An Audio FX Mini-Sound Board ADA2342 was used specifically for the Dozer as seen in Figure 16. The board has the capability of using eight WAV or OGG audio files up to a total size of 2MB. The DOOMBA Dozer used almost all of 2 MB of space for five different sound clips. The clips consisted of annoying sounds, popular video game audio bits and a robot introduction. Audio sounds were added to the system to taunt and distract the opponent as a form of psychological operations (PSYOP). Two 4 Ohm Gikfun EK1725 speakers were mounted to the Dozer body and are rated for 5 watts. Because the mini-sound board did not have built in amplification, an amplifier was required to boost the audio to an Audible level. A 5-volt, 3-Watt mini audio amplifier was used to get adequate volume through the small 2-inch audio speakers. Figure 17 depicts the speakers and Figure 18 depicts the audio amp.

.

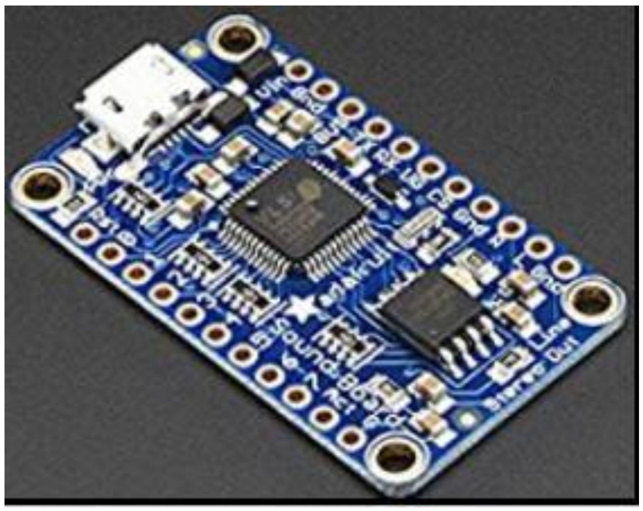


Figure 16: Audio FX Mini-Sound Board ADA2342

Figure 17: Gikfun 5 Watt 4 Ohm Speakers

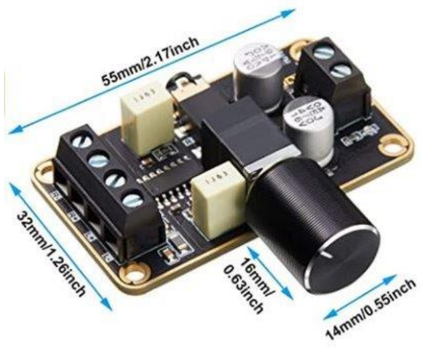


Figure 18: Audio Amp

A decision was later made to utilize a 940 nm Light Emitting Diode (LED) as an offensive weapon with the Dozer, as shown in Figure 19. The diode's anode was connected to a 220 Ohm resistor which was connected to the output of the 5 VDC regulator. The cathode of the diode was connected directly to one of the payload Arduino's pulse width modulation capable pins. Using a built-in remote library in the Arduino, a simulated wall signal could be produced that a Roomba with sensors would try to avoid.

Figure 19: Gikfun 5mm 940nm LEDs Infrared Emitter (part of EK8443)

5.5 Unique Components – SAW

5.5.1 Chassis and Housing

The decision was made to use aluminum sheet metal to upgrade the chassis and housing of DOOMBA Saw. Each piece was cut out specifically to balance two heavy duty saw blades on each end of the bot. For the Frame of DOOMBA Saw, a 14"x14" platform was cut from 1/16" aluminum plate stock. This general size was selected based on the size of the Roomba. Although the Roomba is circular in shape, a square platform was chosen for simplicity. The body of DOOMBA Saw was made with 1/16" aluminum sheet stock. The sheet was bent and formed over the frame to accommodate the internal components and to provide protection against opponent's weaponry. The body was designed to be low profile and keep low ground clearance for stability during attack and mobility for defense. When mounting and testing the first angle grinder, it was determined that additional stability was needed. An offset plate was cut using 1/16" aluminum plate stock and bent on both long edges at 90 degrees to improve the stability of the platform and provide a rigid support for both grinders to mount to.

This aluminum stock was used for a strength of materials project in a previous class and proved to be a durable light material which played a key role in our decision to use this for the majority of the housing. Final bot weight was 15.5 pounds. See Figure 20 for the chassis and housing of the DOOMBA Saw.

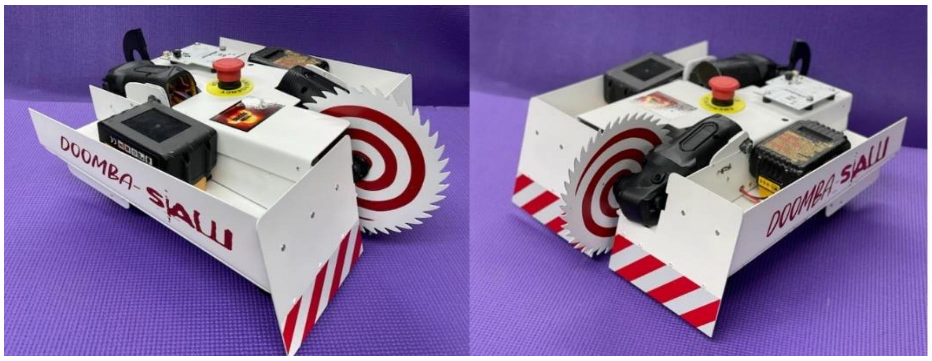


Figure 20: DOOMBA Saw Body

5.5.2 Electronic Control System

The DOOMBA Saw consisted of two 21-volt angle grinder tools that can be seen in Figure 21. The grinder provided the torque and speed for the two weapons attached to the end of the tool.

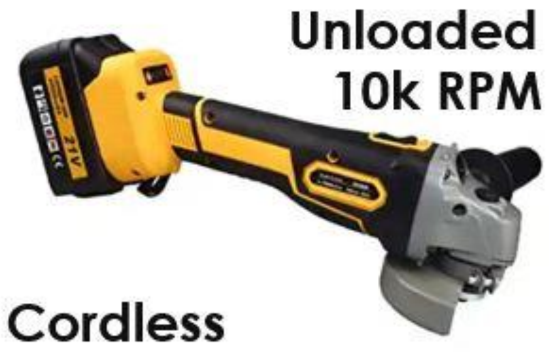


Figure 21: 21-Volt Cordless Angle Grinder

Two Songhe single channel DC optocoupler relay modules for Arduino were used and can be seen in Figure 22. Refer to Figure 3 and Figure 4 for configurations in the two bots.

Figure 22: Songhe Single Channel Optocoupler Relay

5.6 Wheel Assembly

The Roomba came with a set of two standard wheels, and upon initial inspection, the team saw them as inadequate. It was suggested during design meetings to utilize the wheels from a scrap hover board to increase stability, speed, and durability. Analysis and testing of the hover board wheels proved challenging to control from the microcontroller. To properly control this type of wheel an additional motor controller would need to be procured for which the team had not planned in the budget. During early conversations with our advisor, it was also found that the wheel assemblies represented the base of the Roomba platform due to its speed and weight bearing restrictions. With the discussed design and attack strategy differences between the two bots, it was decided that the Dozer bot would utilize two sets of Roomba wheel assemblies and the Saw bot would use a single set of wheel assemblies. The Saw would also use free spinning caster wheels on the front of the bot. See Figure 23 for the wheels used for the DOOMBA Dozer. See Figure 24 for the DOOMBA Saw wheel configuration.

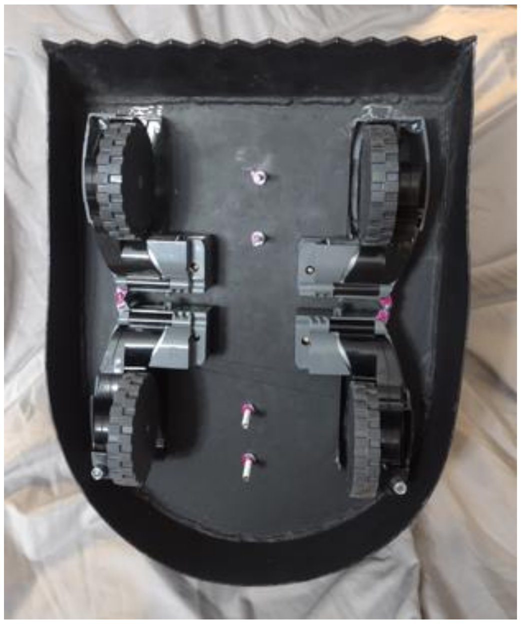


Figure 23: Wheel Assembly on DOOMBA Dozer

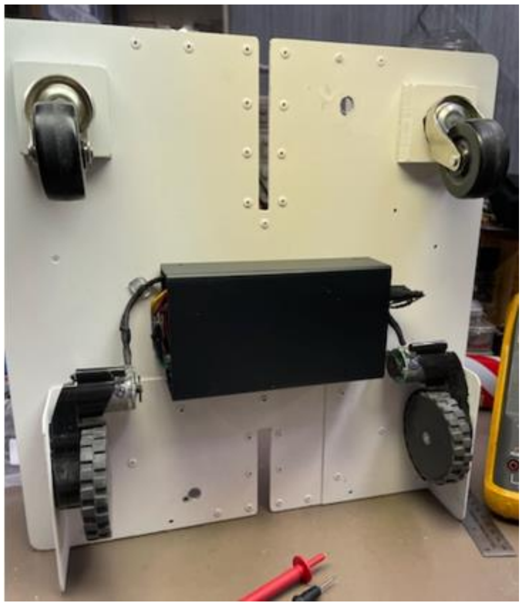


Figure 24: Wheel Assembly on DOOMBA Saw

5.7 Offensive & Defensive Strategy

5.7.1 Offensive & Defensive Strategy - DOZER

A decision was made late in the design phase of the Dozer to utilize a 940 nm Light Emitting Diode (LED) as an offensive weapon. The thought behind this was to use an IR transmitter as a beacon where an Arduino program would generate specific IR signals that would create a “simulated wall” and confuse the other bot if it was using the built-in board logic of the standard Roombas. This would only work in the case that the other team left their IR sensor intact for a “home” function to utilize the recharging station. An existing IR remote.h Arduino library would be used to create the fake wall pulse pattern. The Roomba would sense with its upper IR sensor and would naturally turn to avoid, thus creating confusion. This could be used to effectively make the opposing teams bot react and become temporarily uncontrollable due to the automated Roomba response. This possibility is due to the fact that the serial port that is used to command the Roomba’s motions does not offer complete control of the robot and some basic programming and responses are still functioning from the base robot. The other offensive strategy of the Dozer is to utilize the four drive wheels along with the heavy construction to push the opponent’s bot off the battle platform. A standard Roomba only has two drive wheels and utilizes a caster wheel to stabilize and roll. Using two sets of drive wheels against a bot with only two drive wheels will provide the Dozer with a wheel to ground torque advantage over its opponent.

5.7.2 Offensive & Defensive Strategy – SAW

During the team design process, it was decided to incorporate angle grinders into one of the battle bots. The grinders were small, powerful, and easy to trigger from the microcomputer. The team discussed the advantages & disadvantages of using regular saw blades. One advantage is that there are many different blades readily available with many different applications. These types of blades can be purchased at low cost. A disadvantage with these types of saw blades is that they are designed to cut a specific material and not designed for impact with various materials. To use the angle grinders with a more robust blade, a thick custom blade was designed and fabricated to be more durable and reek more havoc. An FEA analysis was done on the blade design using Fusion 360 by Autodesk. Fusion 360 is based on Nastran that was originally developed by NASA. The custom blade would be well suited for major damage and used to flip the opponent. The team decided to utilize one custom blade in a vertical orientation, and one commercially purchased saw blade in a vertical direction. See Figure 25 for the commercial blade. See Figure 26 for the custom blade.

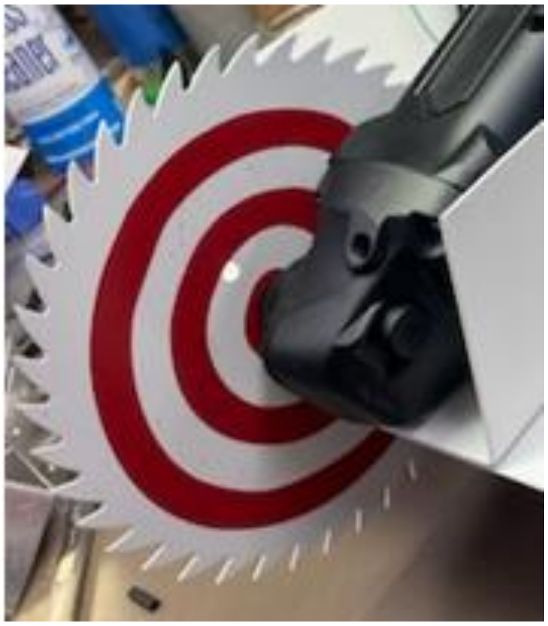


Figure 25: Commercial saw blade

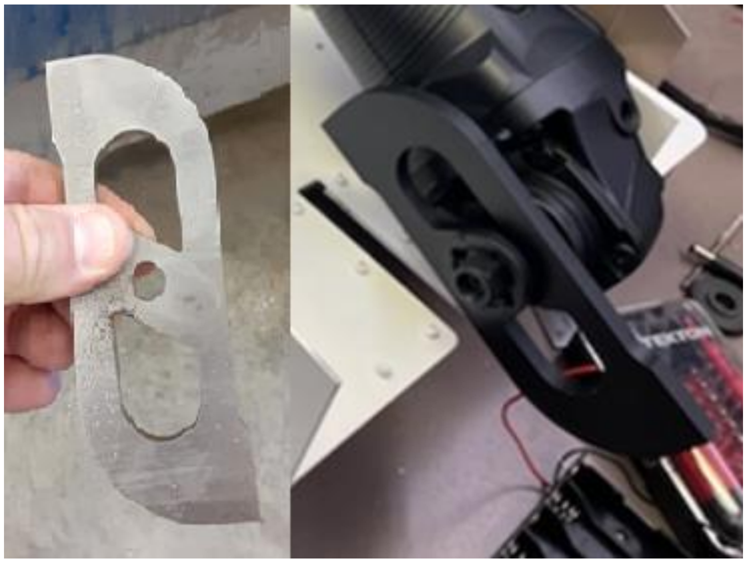


Figure 26: Custom 1/4" steel blade

6.0 Code Structure

Since the drive and payload logic were chosen to be separated, there are three separate program files. All programs are almost identical with respect to core functionality. Both direct the Arduino to measure the pulse width of the signal that is on the receiver's output pins. That number is then compared to a set noise factor. If the number is within the noise range it is turned to zero. If not, then the program forms them into a usable number from -255 to +255. That number is subjected to multiple "if then" statements to set Arduino pins to the appropriate settings. The main difference between the three programs is the number of channels that the board is reading from and if the output pins are going to a motor driver board (MDB), sound board, relay boards, or a light emitting diode. The similarity in function can be seen in the flowcharts for the two programs seen below in Figure 27. The full code can be seen in APPENDIX D.

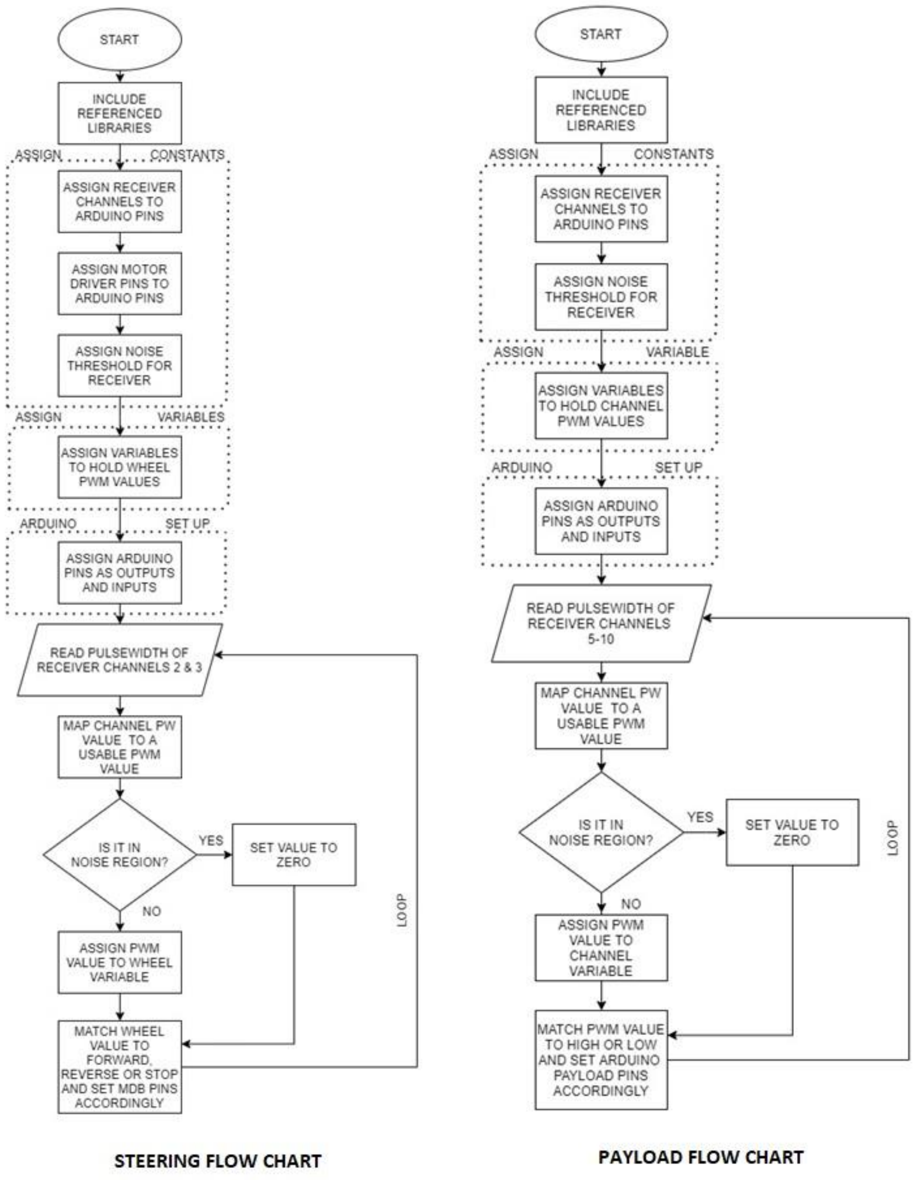

START
INCLUDE REFERENCED LIBRARIES
ASSIGN CONSTANTS
ASSIGN RECEIVER CHANNELS TO ARDUINO PINS
ASSIGN MOTOR DRIVER PINS TO ARDUINO PINS
ASSIGN NOISE THRESHOLD FOR RECEIVER
ASSIGN VARIABLES
ASSIGN VARIABLES TO HOLD WHEEL PWM VALUES
ARDUINO SET UP
ASSIGN ARDUINO PINS AS OUTPUTS AND INPUTS
READ PULSEWIDTH OF RECEIVER CHANNELS 2 & 3
MAP CHANNEL PW VALUE TO A USABLE PWM VALUE
IS IT IN NOISE REGION?
YES
SET VALUE TO ZERO
NO
ASSIGN PWM VALUE TO WHEEL VARIABLE
MATCH WHEEL VALUE TO FORWARD, REVERSE OR STOP AND SET MDB PINS ACCORDINGLY
LOOP
STEERING FLOW CHART
START
INCLUDE REFERENCED LIBRARIES
ASSIGN CONSTANTS
ASSIGN RECEIVER CHANNELS TO ARDUINO PINS
ASSIGN NOISE THRESHOLD FOR RECEIVER
ASSIGN VARIABLE
ASSIGN VARIABLES TO HOLD CHANNEL PWM VALUES
ARDUINO SET UP
ASSIGN ARDUINO PINS AS OUTPUTS AND INPUTS
READ PULSEWIDTH OF RECEIVER CHANNELS 5-10
MAP CHANNEL PW VALUE TO A USABLE PWM VALUE
IS IT IN NOISE REGION?
YES
SET VALUE TO ZERO
NO
ASSIGN PWM VALUE TO CHANNEL VARIABLE
MATCH PWM VALUE TO HIGH OR LOW AND SET ARDUINO PAYLOAD PINS ACCORDINGLY
LOOP
PAYLOAD FLOW CHART

Figure 27: Programming Flow Charts for Steering and Payload

7.0 EXPENDITURES

When the idea of creating battle bots was formed as the cohort's senior project and the teams were decided, a discussion was then started about what an appropriate budget would be for the project. After multiple discussions, the final decision was made to cap the project at a total budget of $1,000. This included the first purchased bot, $250 in donated items and $500 more for another Roomba, or necessary parts. The team's decision to create two bots hinged heavily on if it was feasible to have two bots within this budget. After the initial survey of material requirements for one bot, it was quickly realized that it was feasible to pursue the building of two bots. The budgeted items and their totals can be viewed in APPENDIX C. The project cost summary can be seen in Table 1.

Table 1: Budget Summary

	AMOUNT	% OF BUDGET
SAW COSTS	$ 528.57	52.86%
DOZER COSTS	$ 423.08	42.31%
PROJECT TOTAL	$ 951.65	95.17%
ALLOWED BUDGET	$1,000.00	100.00%

Most of the items used for the project were purchased through Amazon or in a manner that shipping was not a factor. The pursuit of the creation of two separate battle bots was successfully executed within the provided budgetary restraints.

8.0 TEAMWORK EXPERIENCE

The team experience in the iRobot Roomba to battle-bot conversion project, saw many aspects of group decision making, and also lessons learned. When the project was first presented to our team members, we began to meet on Mondays on a weekly basis, the majority by Zoom session. One of the first decisions made was the team moniker "DOOMBA". Knowing that any good product or campaign requires a memorable logo to be memorable, side discussions started about making a DOOMBA logo to match the team's name. Eventually the multiple logo ideas formed into one final logo that focuses on a screaming goat in awe of the destructive power of

Team DOOMBA. The logo can be seen in Figure 28. The screaming goat was later used as a payload on the Dozer.

Figure 28: Team DOOMBA Logo

Early serious discussion evolved around the future battle-bot build and attack system, and if one or two battle-bots was feasible and/or affordable. After multiple discussions, all members were in favor of working on one bot, to ensure the project stayed within budgetary constraints.

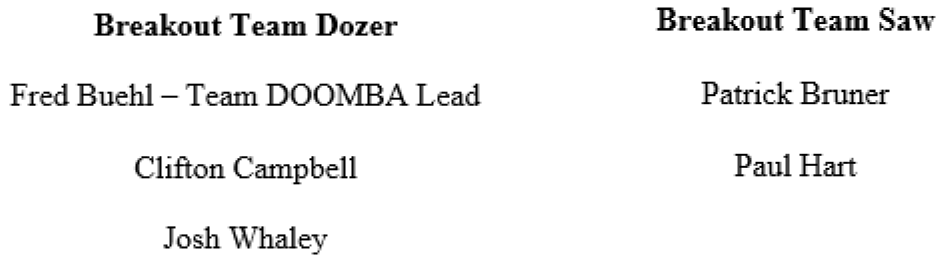
With many tasks and a variety of areas from programming, frame construction, assembly, lab tests, interfacing, and proper teamwork, communication was vital to the success of the project. To encourage robust communication between team members, the team quickly setup a file share drive, text chain, and secured a workshop facility to utilize in the battle-bot development. Weekly electronic conferences with our project advisor were set on Thursdays. This kept all members on schedule with tasking. For a full schedule of meetings and requirement dates, see APPENDIX B. The file share drive allowed for each member to participate by uploading new files for others to view. It was also an avenue to share ideas and build concepts.

Researching battle-bots became the focus for the group. It was important to understand the possibilities as to the proper weaponry and defenses of the bot with respect to budget constraints. Discussion during the team meetings centered around the type of bot the team wanted. Once all team members agreed, the design process started by making a budget list of parts. It became apparent the team could stay well under the budget granted, therefore the decision for a second bot was made.

One hurdle that presented itself was the fact that all the team members had some distance between one another. Because of this, sometimes meeting as a group posed a difficult task. Another hurdle arose once a team decision was made that two bots would be designed and constructed. For a short period, the team was not getting very far with the project. Everyone was focused on the first bot, but little focus was on the forward progress of the second bot. The team leader decided to split up the five-man team, so that resources would be utilized for both bots. This proved to be a critical move in the design and development of both DOOMBA bots. This also alleviated some of the distance issues. Two of the members lived within proximity and could meet more regularly for the bot assigned to them. The other three members worked accordingly to get the other bot in development. Throughout the changes, weekly team meeting via Zoom were utilized.

From beginning to end, team DOOMBA faced many questions with programming, build capability, design, attack strategy, and defensive strategy. Throughout the process, questions were faced with brainstorming of ideas, and then a general consensus based on the best solutions. The team worked as a unit, stayed on course, designed, programmed, and developed two DOOMBA battle-bots that met all the criteria of the project.

<u>Team DOOMBA Makeup</u>



Breakout Team Dozer	**Breakout Team Saw**
Fred Buehl – Team DOOMBA Lead	Patrick Bruner
Clifton Campbell	Paul Hart
Josh Whaley	

9.0 PROJECT CONSIDERATIONS

During the process of the project development, the team was highly encouraged to keep in mind various aspects that were learned through videos and discussions, while attending the ENGR491 Senior Design lectures. Besides just creation of a product while engineering, we were to keep in mind public health, safety and welfare, global and cultural perspectives, social, environmental, economic, ethical and professional impacts.

With respects to public safety and welfare, the robots have an emergency stop installed that allows for the disabling of the robot's primary offensive weapons. In the case of the Dozer, the stop turns off all power to the drive, audio and led systems. In the case of the Saw, the stop turns off power to both grinders powering the blades. Early in the discussion of how the battle arena would be laid out, there were discussions on how to protect the audience from possible airborne debris created from the robots interacting. It was decided that camera systems and or plexiglass shields would be used to isolate the public, or audience from harm.

With respect to environmental concerns, our team decided to take a slow approach on purchasing a secondary Roomba system until we had time to inspect and decide what would be usable and non-usable parts for our final robots. This allowed the team to come to the conclusion that we should not buy a secondary Roomba platform and have a second number of materials to dispose of. We decided to purchase Roomba wheel assemblies by themselves to save money, but also, we could purchase them used through various retailers. The reuse of four separate wheel assemblies allowed for the reduction in a waste footprint from the project while supporting a retailer that sells reclaimed parts. When possible, the wiring from the Roomba's outer sensors was also used in an attempt to reduce the amount of initial purchased product that would become waste.

10.0 LESSONS LEARNED

Over the course of any project, there are always decisions that could have been done differently, in order to make something better or easier. This project was no different. Hind sight provides perspective and focus. With respect to electronic control systems, it would have been better served to use a buck voltage regulator to feed the Arduino input power. This would have eliminated the requirement of AA batteries and would have made all power requirements center on the 20 VDC rechargeable battery or the 21 VDC grinder batteries. This would have also reduced weight and reduced required space. With respect to the Dozer housing, the battery enclosure would be bigger in order to fit the 940 nm LED higher on the platform, giving the output more chance to spread and increase effectiveness. With respect to both housings, a raised metal wall would be added surrounding the emergency stop, that way it would not be as accessible to the other team during battle. By adding containing walls, the stop would have to be accessed from directly above. This would not affect the usage of the stop since during the competition, a broomstick was to be used in the case that one needed shut down. This would prevent someone from cutting their hands by rotating blades. The decision to split the teams into two sub groups was a bit too slow and lagged the team pursuing two bots. This should have been integrated the moment that the two bots were conceptualized. It would have sped up production and completion of the two bots considerably. Building two bots in a committee style forum is not as productive as focused teams. While there are many other improvements that could be made, these represent some major ones that would have been very beneficial.

11.0 PROJECT COMPLETION

Upon securing the components, attaching the outer housing, and testing each bot for complete functionality, the DOOMBA Dozer and the DOOMBA Saw, were finished products of a modified Roomba iRobot vacuum. The finished DOOMA Dozer can be viewed in Figure 29 and the finished DOOMBA Saw can be viewed in Figure 30**.**

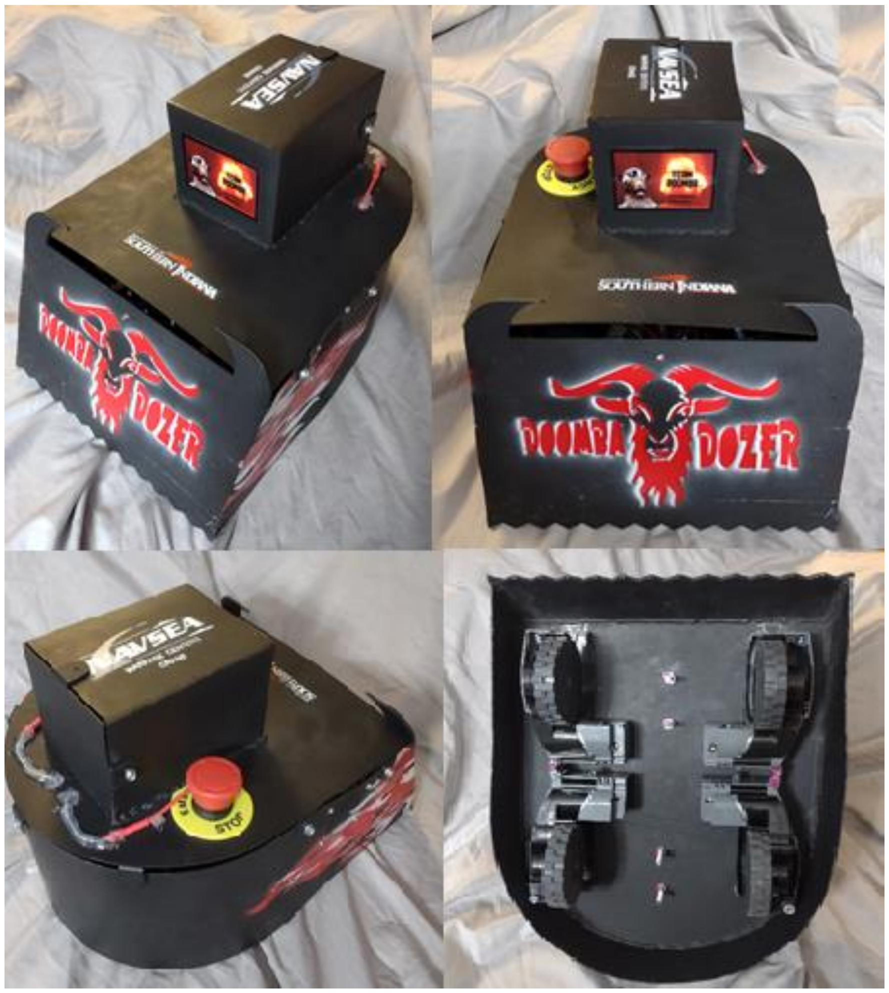


Figure 29: DOOMBA Dozer

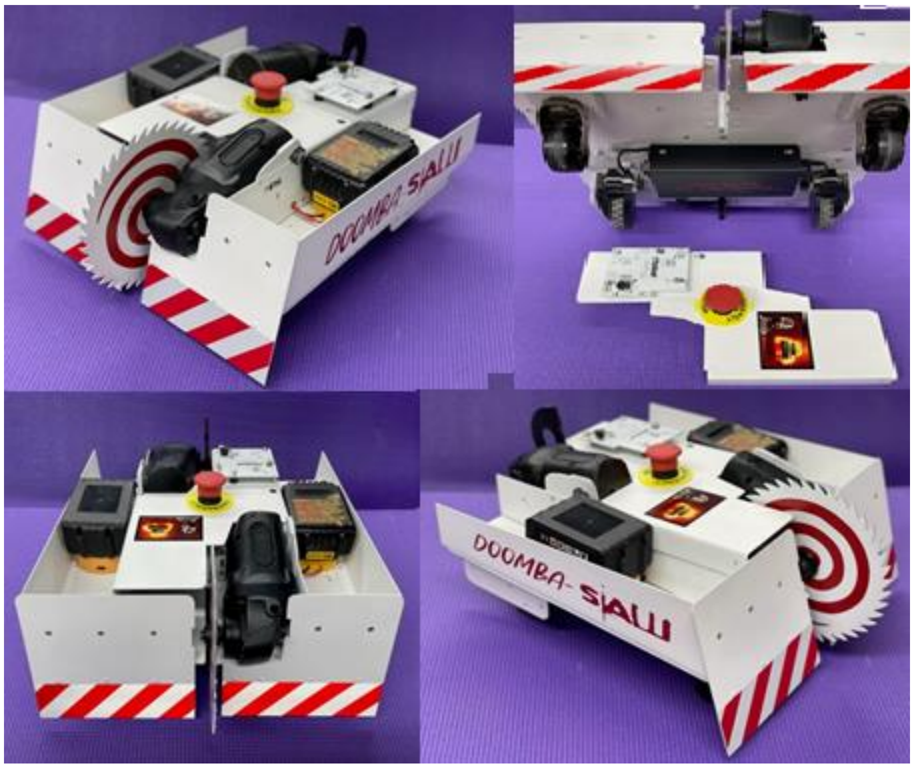


Figure 30: DOOMBA Saw

12.0 CONCLUSION

The purpose of this senior design task was to provide students with a project to use both knowledge and skills learned during the course of the engineering program to create a battle-bot from an iRobot Roomba platform. As our team met and pondered through our ideas it was clear that we could not fit them all into one bot and complete our project on time and under budget. This brought on our decision to split our team into two groups which worked very well. The two-bot team allowed us to pool our resources and knowledge, yet split our tasking geographically. We chose cheap micro-controllers to keep cost down. This allowed the team to spend wisely on other items, such as a ten-channel radio controller, for example. We divided tasking amongst the team members based on individual strengths which benefited the team's "just in time" strategy. As we progressed, our strategy allowed us to make improvements on the original design and discuss possible defense strategies. Initially the Saw concept was our primary offensive strategy but also brought healthy discussion about the opposition's attack. This discussion led to the creation of the second bot which maximized physical defense, relying on a push type or “dozer” offense. This concept was brought forth as a strategic attack utilizing the elevated platform in which the battle will take place. The team environment led to many ideas and discussions which no one team member could have accomplished on their own. The programming used for both communication and movement was a reduced C\C++ programming, which was learned in our many programming courses. This foundation was critical in writing the language these bots speak. The base platform each bot is built upon was designed and constructed using structural concepts learned in mechanical courses. The materials used were selected based solely on materials used during our strength of materials class. During construction, fasteners and adhesives were used in a way to strengthen the chassis for impact from all sides. All aspects of construction were decided and performed with only victory in mind. The team is confident in the design and construction of each bot as well as the battle strategy chosen. In conclusion, the raw purpose of this project was to construct a battle-bot, however this team’s purpose was to construct a bot worthy of battle and that has been accomplished twice over. [DOOMBA goat’s victory bleat here]

REFERENCES

1 “Types of bulldozers and their uses.” GoToYard, www.gotoyard.com/en/types-of-bulldozers-and-their-uses/. Accessed 12 October 2021

APPENDIX

APPENDIX A

ENGR 491 BattleBot Point breakdown

Design points:

- E-Stop functional: +20 points with competition not allowed until functional.
- Judging by independent, unbiased observers:
    - Aesthetics/build quality of robot: +0-20 points
    - "Coolness" of design: +0-20 points
    - Autonomous elements of design: +0-20 points (By demonstration)
- Speed challenge: robot should traverse a 16 foot distance under a minute: +20 points
- Maneuverability challenge: robot should traverse a slalom course of obstacles and back again in under 5 minutes: +20 points. (You may not touch the opposing robot during this round.)

Round scoring:

- Three, 10-minute rounds are played separated by 5-minute intermissions. Robots can be repaired/adjusted/refueled during the intermissions. Team members may not touch the robots or enter the platform during the rounds.
- Each team has three 2-minute Time Outs for the whole battle (3 rounds) Time Outs can be called at any time that the robots are not engaged with each other. During the 2-minute Time Out, both teams can repair/adjust their robots. The robots must be returned to the platform in time to restart the round.
- If the robot falls off the platform, a Time Out can be called without penalty and the robot replaced on the platform at the end of the timeout. If no Time Outs are remaining and/or if the team replaces the robot quickly without making any adjustments to the robot, a 5 point penalty is given to that robot.
- Each successful "attack" that is landed on the opposing robot is awarded 2 points. The "attack" strategies for your robot must be defined during the presentation.
- If one robot fails to move/respond in 30 seconds, the round is concluded and the other team is awarded 25 points. The 5-minute intermission begins immediately. If that robot can not be made to respond at the end of the intermission, the subsequent round is forfeited and the 25 points are awarded to the other team. If that was Round 2, the team can have another 5 minutes to repair the robot for Round 3.
- Deductions of 50 points will be given for each projectile that enters the audience. Injuring an audience member will result in forfeiture of the battle. Injuring or killing a faculty member may result in failing the course.

APPENDIX B

Description	Date
Weekly Team Meeting (Zoom)	August 20, 2021
Advisor Led Meeting (Zoom)	September 2, 2021
Order Deadline for Roomba	September 2, 2021
Weekly Team Meeting (Zoom)	September 6, 2021
Instructor Led Weekly Meeting (Zoom)	September 9, 2021
Weekly Team Meeting (Zoom)	September 13, 2021
Advisor Led Weekly Meeting (Zoom)	September 16, 2021
Weekly Team Meeting (Zoom)	September 20, 2021
Advisor Led Weekly Meeting (Zoom)	September 23, 2021
Weekly Team Meeting (Zoom)	September 27, 2021
Advisor Led Weekly Meeting (Zoom)	September 30, 2021
Weekly Team Meeting (Zoom)	October 4, 2021
Advisor Led Weekly Meeting (Zoom)	October 7, 2021
Weekly Team Meeting (Zoom)	October 11, 2021
Advisor Led Weekly Meeting (Zoom)	October 14, 2021
Weekly Team Meeting (Zoom)	October 18, 2021
Advisor Led Weekly Meeting (Zoom)	October 21, 2021
Weekly Team Meeting (Zoom)	October 25, 2021
Advisor Led Weekly Meeting (Zoom)	October 28, 2021
Weekly Team Meeting (Zoom)	November 1, 2021
Program Info to Shared Drive	**November 3, 2021**
Names, Emails on Invite List in Drive	**November 3, 2021**
Advisor Led Weekly Meeting (Zoom)	November 4, 2021
Weekly Team Meeting (Zoom)	November 8, 2021
Advisor Led Weekly Meeting (Zoom)	November 11, 2021
Design Presentation Review Complete	**November 12, 2021**
Description	**Date**

Weekly Team Meeting (Zoom)	November 15, 2021
Draft Report to Advisor	November 17, 2021
Advisor Led Weekly Meeting (Zoom)	November 18, 2021
Weekly Team Meeting (Zoom)	November 22, 2021
Thanksgiving (No Meeting)	November 25, 2021
Weekly Team Meeting (Zoom)	November 29, 2021
Advisor Led Weekly Meeting (Zoom)	December 2, 2021
Final Presentation day, PPTs to drive	**December 3, 2021**
Weekly Team Meeting (Zoom)	December 6, 2021
Advisor Led Weekly Meeting (Zoom)	December 9, 2021
Poster Presentation	December 9, 2021
Final Report, to Advisor, shared drive	**December 10, 2021**
Final Report Submitted to SOAR	**December 17, 2021**

*** Note: Major Milestones in bold and holidays are highlighted in yellow.**

APPENDIX C

Battle Bot Project Bill of Materials & Cost Accounting

DOOMBA-SAW				
NOMENCLATURE	VENDOR	QTY	COST WITH TAX	TOTAL ORDER
iRobot Create 2 Programmable Robot	iRobot	1	$213.99	213.99
DROK L298 H Bridge Motor Driver	Amazon	1	$16.90	16.90
FLYSKY FS-iA10B Receiver	Amazon	1	$24.48	24.48
ELEGOO Arduino UNO R3 Clone Kit	Amazon	2	$32.08	32.08
Songhe 1 Channel Relay Module (5 pack)	Amazon	1	$13.39	13.39
SMAUP Cordless Angle Grinder	Amazon	2	$53.48	53.48
Dewalt 20V Battery Adapter	Amazon	1	$12.39	12.39
2 Pack Front Wheel Caster Assembly	Amazon	1	$12.12	12.12
Used Dewalt 20V max 4 Ah Lion Battery	Pat Bruner	1	$25.00	25.00
Aluminum .060" Sheet folded 6" x 4.5"	Pat Bruner	1	10.00	10.00
Aluminum Chassis/Exterior Estimate	Pat Bruner	1	50.00	50.00
4 AA Battery Holder	Amazon	1	$7.04	7.04
Mxuteuk NC Red Mushroom E-Stop Push Button	Amazon	1	$6.41	6.41
5v Regulator, DROK (5pcs)	Amazon	2	$3.85	3.85
Miscellaneous- Paint, Epoxy, Solder, etc.	Pat Bruner	1	$20.00	20.00
6" x 6" x 0.250" Cold Rolled Steel (Custom Blade)	West Metal Sales	1	7.45	7.45
Small Saw Blade	Pat Bruner	1	20.00	20.00
			SAW TOTAL	528.57

DOOMBA-DOZER				
NOMENCLATURE	VENDOR	QTY	COST WITH TAX	TOTAL ORDER
4 AA Battery Holder	Amazon	2	$14.08	$7.04
DROK L298 H Bridge Motor Driver	Amazon	1	$16.90	$16.90
FLYSKY FS-i6X 10CH Transmitter/Receiver	Amazon	1	$58.63	$58.63
ELEGOO Arduino UNO R3 Clone Kit	Amazon	2	$32.08	$32.08
Adafruit Audio FX Mini Sound Board	Amazon	1	$16.50	$16.50
Irobot Roomba L/R Wheel Assemblies	eBay	1	$33.19	$33.19
Irobot Roomba L/R Wheel Assemblies	eBay	1	$34.34	$34.34
Gikfun 940nm LEDs (PACK)	Amazon	1	$5.65	$5.65
24" x 48" Plain Sheet Metal Steel- 16 gauge	Menards	1	$58.84	$58.84
12" x 24" x 0.125" Cold Rolled Steel	Raven Operations	1	$50.00	$50.00
Used Dewalt 20V max 4 Ah Lion Battery	Pat Bruner	1	$30.00	$30.00
Aluminum .060" Sheet folded 6" x 4.5"	Pat Bruner	1	$10.00	$10.00
6" x 6" x 0.250" Cold Rolled Steel	West Metal Sales	1	$7.45	$7.45
Dewalt 20V Battery Adapter	Amazon	1	$12.39	$12.39
5v Regulator, DROK (5pcs)	Amazon	1	$1.92	$1.92
Miscellaneous- Paint, Epoxy,zipties, velcro, solder etc.	Fred Buehl	1	$30.00	$30.00
Gikfun 2" 4Ohm 3W Audio Speaker EK1725 (2)	Amazon	1	$10.14	$10.14
PAM8406 Mini Audio Amplifier Board DC 5V	Amazon	1	$8.02	$8.02
			DOZER TOTAL	423.08

APPENDIX D

Code for Steering Control (Tank Mode)

```
/*----------------------ARDUINO CONTROL FOR STEERING-----------------------
--------------------------------------


THIS CODE READS CHANNELS 2 & 3 ON THE FS-iA10B 2.4G 10 CHANNEL RECEIVER
FROM THIS INPUT IT ASSIGNS A -255 TO 255 NUMBER AND THEN CHANGES
MOTOR DRIVER BOARD INPUTS ACCORDINGLY FOR MOVEMENTS

CODE WRITTEN BY TEAM DOOMBA USI SENIOR DESIGN TEAM -CLIFTON CAMPBELL, PAUL
HART, FRED BUEHL, PATRICK BRUNER, JOSHUA WHALEY
GUIDENCE FOR CODE WAS FOUND AT
https://gist.github.com/ShawnHymel/520c3dda8293ff4f55b330d277acd89c
SHAWN HYMEL FROM SPARKFUN-https://www.youtube.com/watch?v=u0Ft8SB3pkw&t=13s

-----------------------------------------------------------------------------
-------------------------*/

#include <SoftwareSerial.h>  //ALLOWS FOR USING SERIAL COMS AND MONTITOR FOR
TROUBLESHOOTING

//-------------------------CONSTANTS-------------------------------------

//ASSIGNING CHANNELS FROM RECEIVER TO DRIVER ARDUINO PINS
const int CH_2_PIN = 2;
const int CH_3_PIN = 8;

//ASSIGNING ARDUINO PINS TO MOTOR DRIVER BOARD ENABLE/PWM AND CONTROL PINS
const int EN_A_PIN = 3;
const int IN_1_PIN = 4;
const int IN_2_PIN = 5;
const int EN_B_PIN = 9;
const int IN_3_PIN = 10;
const int IN_4_PIN = 11;

//ASSIGNING NOISE LEVEL IN RECEIVER THAT WILL BE VIEWED AS ZERO
const int noiserange = 20;


//-------------------------END CONSTANTS-------------------------------------


//-------------------------VARIABLES----------------------------------------

//ASSIGNING VARIABLES THAT WILL BE USED TO HOLD CONVERTED RECEIVER CHANNEL
DATA (CHANNELS 2 & 3)
int  FrontnBackRW = 0;
int  FrontnBackLW = 0;

//-------------------------END VARIABLES------------------------------------
-

void setup() {
```

```
//SET THE FOLLOWING ARDUINO PIN AS INPUTS FOR RECEIVER CHANNEL 2
pinMode(CH_2_PIN, INPUT);

//SET THE FOLLOWING ARDUINO PINS AS OUTPUTS FOR MOTOR DRIVER BOARD
pinMode(EN_A_PIN, OUTPUT);
pinMode(IN_1_PIN, OUTPUT);
pinMode(IN_2_PIN, OUTPUT);

//SET THE FOLLOWING ARDUINO PIN AS INPUTS FOR RECEIVER CHANNEL 3
pinMode(CH_3_PIN, INPUT);

//SET THE FOLLOWING ARDUINO PINS AS OUTPUTS FOR MOTOR DRIVER BOARD
pinMode(EN_B_PIN, OUTPUT);
pinMode(IN_3_PIN, OUTPUT);
pinMode(IN_4_PIN, OUTPUT);

//Serial.begin(9600);//START SERIAL COMMS SO SERIAL MONITOR CAN BE USED
}

void loop() {

 // READ PULSE WIDTH FROM CHANNELS 2 AND 3 FROM RECEIVER
    int ch_2 = pulseIn(CH_2_PIN, HIGH, 250000);
    int ch_3 = pulseIn(CH_3_PIN, HIGH, 250000);

 // Serial.println((String)" "+ch_2+","+ch_3);
  delay(5);

  FrontnBackRW = RxCHPWValueToPWMValue(ch_2);
  FrontnBackLW = RxCHPWValueToPWMValue(ch_3);
  //Serial.println((String)"RW FB "+FrontnBackRW);
  //Serial.println((String)"LW FB "+FrontnBackLW);
  MDBControl(FrontnBackLW, FrontnBackRW);


}
//// CONVERT RC PULSE WIDTH VALUE TO A MOTOR DRIVER PWM VALUE
int RxCHPWValueToPWMValue(int RxCHPWValue) {
//
//  // If we're receiving numbers, convert them to motor PWM
  if ( RxCHPWValue > 900 ) {
    RxCHPWValue = map(RxCHPWValue, 900, 2100, -500, 500);
    RxCHPWValue = constrain(RxCHPWValue, -255, 255);
  } else {
    RxCHPWValue = 0;
  }
//
  // Anything in noiserange should stop the motor
  if ( abs(RxCHPWValue) <= noiserange ) {
    RxCHPWValue = 0;
  }

  return RxCHPWValue;
}
//---------------------------------------------------------------------------
```

```
//READ RIGHT AND LEFT WHEEL PWM VALUE AND SET ARDUINO PINS ACCORDINGLY FOR
MOTOR DRIVER BOARD DROK L298

void MDBControl(int LW, int RW){
  if (LW==0){
      digitalWrite(IN_3_PIN,HIGH);
      digitalWrite(IN_4_PIN,HIGH);
      digitalWrite(EN_B_PIN,LOW);
//    Serial.println((String)"LW LOW LOW");
      }
  else if (LW>0){
      digitalWrite(IN_3_PIN,HIGH);
      digitalWrite(IN_4_PIN,LOW);
      analogWrite(EN_B_PIN,abs(LW));
//    Serial.println((String)"LW HIGH LOW");
      }
  else {
      digitalWrite(IN_3_PIN,LOW);
      digitalWrite(IN_4_PIN,HIGH);
      analogWrite(EN_B_PIN,abs(LW));
//    Serial.println((String)"LW LOW HIGH");
      }

  if (RW==0){
      digitalWrite(IN_1_PIN,HIGH);
      digitalWrite(IN_2_PIN,HIGH);
      analogWrite(EN_B_PIN,abs(LW));
//    Serial.println((String)"RW LOW LOW");
      }
  else if (RW>0){
      digitalWrite(IN_1_PIN,HIGH);
      digitalWrite(IN_2_PIN,LOW);
      analogWrite(EN_A_PIN,abs(RW));
//    Serial.println((String)"RW HIGH LOW");
      }
  else {
      digitalWrite(IN_1_PIN,LOW);
      digitalWrite(IN_2_PIN,HIGH);
      analogWrite(EN_A_PIN,abs(RW));
//    Serial.println((String)"RW LOW HIGH");
      }

}
```

Code for Six Channel Controls

```
/*------------------------ARDUINO CONTROL FOR PAYLOAD-------------------------
-------------------------------------

THIS CODE READS CHANNELS 5-10 ON THE FS-iA10B 2.4G 10 CHANNEL RECEIVER
FROM THIS INPUT IT ASSIGNS AN "ON" OR "OFF" STATUS AND ACTIVATES PINS
ACCORDINGLY
CHANNELS 5 AND 6 CURRENTLY RUN RELAY, 7-10 ARE OPEN FOR USE
```

```
CODE WRITTEN BY TEAM DOOMBA USI SENIOR DESIGN TEAM -CLIFTON CAMPBELL, PAUL
HART, FRED BUEHL, PATRICK BRUNER, JOSHUA WHALEY
GUIDENCE FOR CODE WAS FOUND AT
https://gist.github.com/ShawnHymel/520c3dda8293ff4f55b330d277acd89c
SHAWN HYMEL FROM SPARKFUN-https://www.youtube.com/watch?v=u0Ft8SB3pkw&t=13s

--------------------------------------------------------------------------------
-------------------------*/

#include <SoftwareSerial.h> //ALLOWS FOR USING SERIAL COMS AND MONTITOR FOR
TROUBLESHOOTING

//--------------------------CONSTANTS--------------------------------------


//ASSIGNING CHANNELS FROM RECEIVER TO PAYLOAD ARDUINO PINS
const int CH_5_PIN = 2; //RCVR CHANNEL 5 ASSIGNED TO PIN 2 ON ARDUINO
const int CH_6_PIN = 3; //RCVR CHANNEL 6 ASSIGNED TO PIN 3 ON ARDUINO
const int CH_7_PIN = 4; //RCVR CHANNEL 7 ASSIGNED TO PIN 4 ON ARDUINO
const int CH_8_PIN = 5; //RCVR CHANNEL 8 ASSIGNED TO PIN 5 ON ARDUINO
const int CH_9_PIN = 6; //RCVR CHANNEL 9 ASSIGNED TO PIN 6 ON ARDUINO
const int CH_10_PIN = 7; //RCVR CHANNEL 10 ASSIGNED TO PIN 7 ON ARDUINO

//ASSIGNING NOISE LEVEL IN RECEIVER THAT WILL BE VIEWED AS ZERO
const int noiserange = 20;

//--------------------------END CONSTANTS----------------------------------

//--------------------------VARIABLES--------------------------------------

//ASSIGNING VARIABLES THAT WILL BE USED TO HOLD CONVERTED RECEIVER CHANNEL
DATA (CHANNELS 5-10)
int RelayOne = 0;
int RelayTwo = 0;
int FreeChannelOne = 0;
int FreeChannelTwo = 0;
int FreeChannelThree =0;
int FreeChannelFour =0;

//--------------------------END VARIABLES----------------------------------
-

void setup() {
//SET FOLLOWING ARDUINO PINS TO OUTPUTS THAT ARE HOOKED TO RELAYONE AND
RELAYTWO
pinMode(8, OUTPUT);
pinMode(9, OUTPUT);
//SET FOLLOWING ARDUINO PINS TO OUTPUTS THAT ARE HOOKED TO ....OTHER
PAYLOAD/SOUNDBOARD/ETC
pinMode(10, OUTPUT);
pinMode(11, OUTPUT);
pinMode(12, OUTPUT);
pinMode(13, OUTPUT);
//SET FOLLOWING ARDUINO PINS TO INPUTS TO READ FROM THE RECEIVER
pinMode(CH_10_PIN, INPUT);
pinMode(CH_9_PIN, INPUT);
pinMode(CH_8_PIN, INPUT);
pinMode(CH_7_PIN, INPUT);
pinMode(CH_6_PIN, INPUT);
pinMode(CH_5_PIN, INPUT);
Serial.begin(9600);
}

void loop() {
// main code here, to run repeatedly:
// READ PULSE WIDTH FROM RECEIVER CHANNELS AND ASSIGN TO VARIABLE
int ch_5 = pulseIn(CH_5_PIN, HIGH, 250000); //pulseIn(pin, value, timeout)
int ch_6 = pulseIn(CH_6_PIN, HIGH, 250000); //pulseIn(pin, value, timeout)
int ch_7 = pulseIn(CH_7_PIN, HIGH, 250000); //pulseIn(pin, value, timeout)
int ch_8 = pulseIn(CH_8_PIN, HIGH, 250000); //pulseIn(pin, value, timeout)
int ch_9 = pulseIn(CH_9_PIN, HIGH, 250000); //pulseIn(pin, value, timeout)
int ch_10 = pulseIn(CH_10_PIN, HIGH, 250000); //pulseIn(pin, value, timeout)

delay(5);

RelayOne = RxCHPWValueToPWMValue(ch_5);
RelayTwo = RxCHPWValueToPWMValue(ch_6);
FreeChannelOne = RxCHPWValueToPWMValue(ch_7);
FreeChannelTwo = RxCHPWValueToPWMValue(ch_8);
FreeChannelThree = RxCHPWValueToPWMValue(ch_9);
FreeChannelFour = RxCHPWValueToPWMValue(ch_10);


PayloadControl(RelayOne,RelayTwo,FreeChannelOne,FreeChannelTwo,FreeChannelThr
ee,FreeChannelFour);


}
//// Convert RC RECEIVER pulse width value to PWM value
int RxCHPWValueToPWMValue(int RxCHPWValue) {
//
//  // If we're receiving numbers, convert them to PWM values between -255 to
255
  if ( RxCHPWValue > 900 ) {
    RxCHPWValue = map(RxCHPWValue, 900, 2100, -500, 500);
    RxCHPWValue = constrain(RxCHPWValue, -255, 255);
  } else {
    RxCHPWValue = 0;
  }
//
  // Anything in noiserange should stop the motor
  if ( abs(RxCHPWValue) <= noiserange ) {
    RxCHPWValue = 0;
  }

  return RxCHPWValue;
}

//---------------------READ CHANNEL VALUE AND ASSIGN PIN HIGH OR LOW
ACCORDINGLY-----------------------------------


void PayloadControl(int CH5,int CH6, int CH7,int CH8, int CH9, int CH10){
if (CH5<225){
      digitalWrite(8,LOW);}
else {digitalWrite(8,HIGH);}

if (CH6<225){
      digitalWrite(9,LOW);
//      Serial.println((String)"low ");
}
else {digitalWrite(9,HIGH);}

if (CH7<225){
      digitalWrite(10,HIGH);}
else {digitalWrite(10,LOW);}


if (CH8<225){
      digitalWrite(11,HIGH);}
else {digitalWrite(11,LOW);}


if (CH9<225){
      digitalWrite(12,HIGH);}
else {digitalWrite(12,LOW);}


if (CH10<225){
      digitalWrite(13,HIGH);}
else {digitalWrite(13,LOW);}

//SERIAL OUPUT CHANNEL VALUE
Serial.println((String)"FREECH1 "+FreeChannelOne);
Serial.println((String)"FREECH2 "+FreeChannelTwo);
Serial.println((String)"FREECH3 "+FreeChannelThree);
Serial.println((String)"FREECH4 "+FreeChannelFour);
}
```

Code for Six Channel Controls with LED

```
/*------------------------ARDUINO CONTROL FOR PAYLOAD-------------------------
-------------------------------------

//active code in paylode for dozer

THIS CODE READS CHANNELS 5-10 ON THE FS-iA10B 2.4G 10 CHANNEL RECEIVER

FROM THIS INPUT IT ASSIGNS AN "ON" OR "OFF" STATUS AND ACTIVATES PINS
ACCORDINGLY

CHANNELS 5 AND 6 CURRENTLY RUN RELAY, 7-10 ARE OPEN FOR USE



CODE WRITTEN BY TEAM DOOMBA USI SENIOR DESIGN TEAM -CLIFTON CAMPBELL, PAUL
HART, FRED BUEHL, PATRICK BRUNER, JOSHUA WHALEY
```

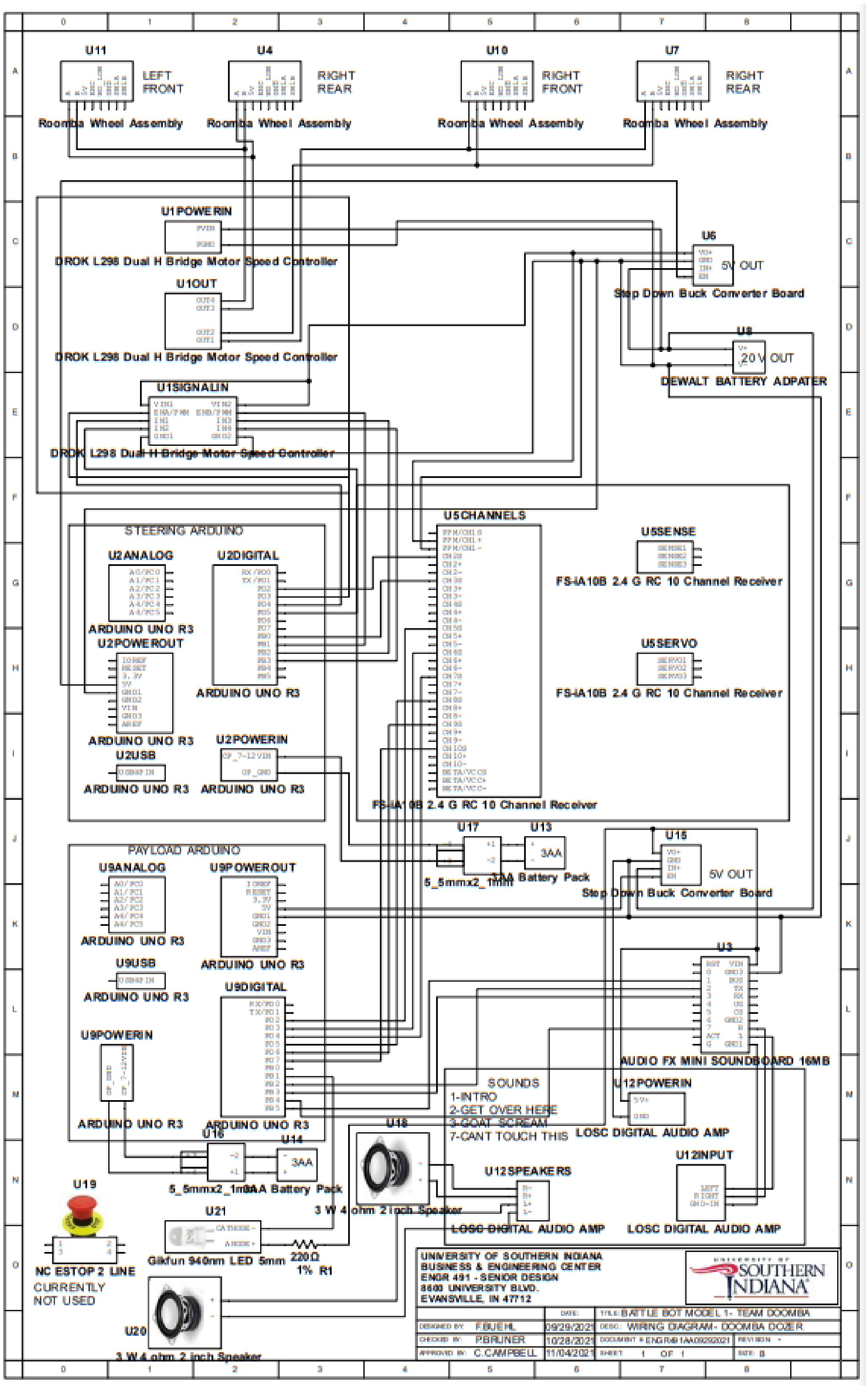
```
GUIDENCE FOR CODE WAS FOUND AT
https://gist.github.com/ShawnHymel/520c3dda8293ff4f55b330d277acd89c
SHAWN HYMEL FROM SPARKFUN-https://www.youtube.com/watch?v=u0Ft8SB3pkw&t=13s

-----------------------------------------------------------------------------
-------------------------*/

//#include <SoftwareSerial.h> //ALLOWS FOR USING SERIAL COMS AND MONTITOR FOR
TROUBLESHOOTING
#include <IRremote.h>//ALLOWS FOR THE USE OF IRREMOTE LIBRARY
IRsend irsend(9);//SETS UP IR SEND ON PIN 9
//---------------------------CONSTANTS---------------------------------------



//ASSIGNING CHANNELS FROM RECEIVER TO PAYLOAD ARDUINO PINS
const int CH_5_PIN = 2; //RCVR CHANNEL 5 ASSIGNED TO PIN 2 ON ARDUINO
const int CH_6_PIN = 3; //RCVR CHANNEL 6 ASSIGNED TO PIN 3 ON ARDUINO
const int CH_7_PIN = 4; //RCVR CHANNEL 7 ASSIGNED TO PIN 4 ON ARDUINO
const int CH_8_PIN = 5; //RCVR CHANNEL 8 ASSIGNED TO PIN 5 ON ARDUINO
const int CH_9_PIN = 6; //RCVR CHANNEL 9 ASSIGNED TO PIN 6 ON ARDUINO
const int CH_10_PIN = 7; //RCVR CHANNEL 10 ASSIGNED TO PIN 7 ON ARDUINO

//ASSIGNING NOISE LEVEL IN RECEIVER THAT WILL BE VIEWED AS ZERO
const int noiserange = 20;

//---------------------------END CONSTANTS-------------------------------------

//---------------------------VARIABLES-----------------------------------------
```

```
//ASSIGNING VARIABLES THAT WILL BE USED TO HOLD CONVERTED RECEIVER CHANNEL
DATA (CHANNELS 5-10)
int RelayOne = 0;
int RelayTwo = 0;
int FreeChannelOne = 0;
int FreeChannelTwo = 0;
int FreeChannelThree =0;
int FreeChannelFour =0;

//--------------------------END VARIABLES------------------------------------
-

void setup() {
//SET FOLLOWING ARDUINO PINS TO OUTPUTS THAT ARE HOOKED TO RELAYONE AND
IRTransmitter (replaced RELAYTWO)
pinMode(8, OUTPUT);
//pinMode(9, OUTPUT);
//SET FOLLOWING ARDUINO PINS TO OUTPUTS THAT ARE HOOKED TO ....OTHER
PAYLOAD/SOUNDBOARD/ETC
pinMode(10, OUTPUT);
pinMode(11, OUTPUT);
pinMode(12, OUTPUT);
pinMode(13, OUTPUT);
//SET FOLLOWING ARDUINO PINS TO INPUTS TO READ FROM THE RECEIVER
pinMode(CH_10_PIN, INPUT);
pinMode(CH_9_PIN, INPUT);
pinMode(CH_8_PIN, INPUT);
pinMode(CH_7_PIN, INPUT);
pinMode(CH_6_PIN, INPUT);
pinMode(CH_5_PIN, INPUT);

//Serial.begin(9600);ALLOWS FOR FEEDBACK

irsend.enableIROut(38);//Lib function ENABLED


}


void loop() {

// main code here, to run repeatedly:

// READ PULSE WIDTH FROM RECEIVER CHANNELS AND ASSIGN TO VARIABLE

int ch_5 = pulseIn(CH_5_PIN, HIGH, 250000); //pulseIn(pin, value, timeout)

int ch_6 = pulseIn(CH_6_PIN, HIGH, 250000); //pulseIn(pin, value, timeout)

int ch_7 = pulseIn(CH_7_PIN, HIGH, 250000); //pulseIn(pin, value, timeout)

int ch_8 = pulseIn(CH_8_PIN, HIGH, 250000); //pulseIn(pin, value, timeout)

int ch_9 = pulseIn(CH_9_PIN, HIGH, 250000); //pulseIn(pin, value, timeout)

int ch_10 = pulseIn(CH_10_PIN, HIGH, 250000); //pulseIn(pin, value, timeout)


delay(5);


RelayOne = RxCHPWValueToPWMValue(ch_5);

RelayTwo = RxCHPWValueToPWMValue(ch_6);

FreeChannelOne = RxCHPWValueToPWMValue(ch_7);

FreeChannelTwo = RxCHPWValueToPWMValue(ch_8);

FreeChannelThree = RxCHPWValueToPWMValue(ch_9);

FreeChannelFour = RxCHPWValueToPWMValue(ch_10);



PayloadControl(RelayOne,RelayTwo,FreeChannelOne,FreeChannelTwo,FreeChannelThr
ee,FreeChannelFour);
```

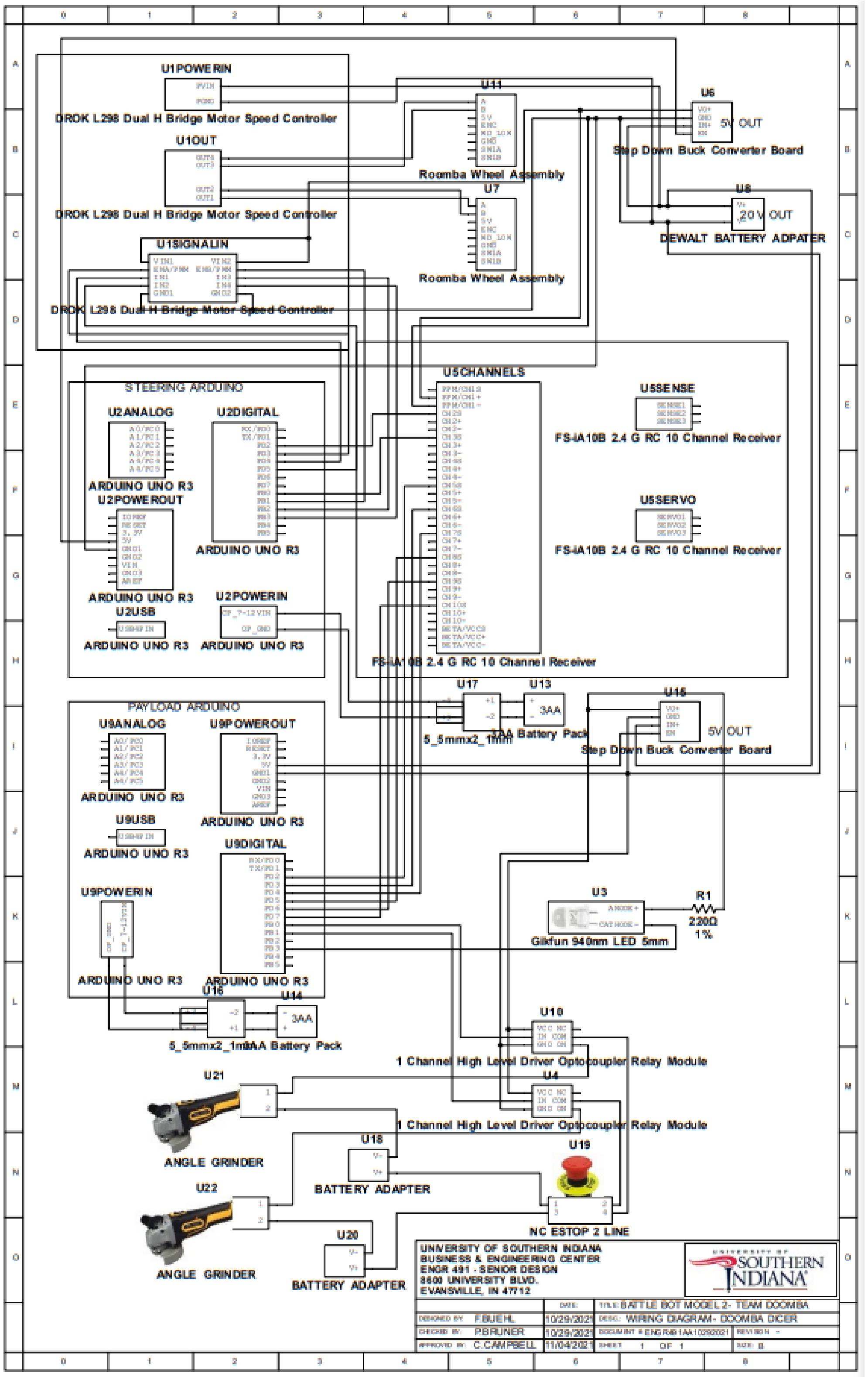
```
}

//// Convert RC RECEIVER pulse width value to PWM value

int RxCHPWValueToPWMValue(int RxCHPWValue) {

//

//  // If we're receiving numbers, convert them to PWM values between -255 to
255

  if ( RxCHPWValue > 900 ) {

    RxCHPWValue = map(RxCHPWValue, 900, 2100, -500, 500);

    RxCHPWValue = constrain(RxCHPWValue, -255, 255);

  } else {

    RxCHPWValue = 0;

  }

//

  // Anything in noiserange should stop the motor

  if ( abs(RxCHPWValue) <= noiserange ) {

    RxCHPWValue = 0;

  }


  return RxCHPWValue;

}


//---------------------READ CHANNEL VALUE AND ASSIGN PIN HIGH OR LOW
ACCORDINGLY-----------------------------------



void PayloadControl(int CH5,int CH6, int CH7,int CH8, int CH9, int CH10){

if (CH5<225){
      digitalWrite(8,HIGH);}
else {digitalWrite(8,LOW);}


if (CH6<225){
      digitalWrite(9,HIGH);
}
else {irblocker();}


if (CH7<225){
      digitalWrite(10,HIGH);}
else {digitalWrite(10,LOW);}



if (CH8<225){
      digitalWrite(11,HIGH);}
else {digitalWrite(11,LOW);}



if (CH9<225){
      digitalWrite(12,HIGH);}
else {digitalWrite(12,LOW);}



if (CH10<225){
      digitalWrite(13,HIGH);}
else {digitalWrite(13,LOW);}


//SERIAL OUPUT CHANNEL VALUE
```

```
//Serial.println((String)"FREECH1 "+FreeChannelOne);

//Serial.println((String)"FREECH2 "+FreeChannelTwo);

//Serial.println((String)"FREECH3 "+FreeChannelThree);

//Serial.println((String)"FREECH4 "+FreeChannelFour);

}

void irblocker()

{

//   /* for loop execution */

for( int a = 0; a < 5; a = a + 1 ){

  irsend.mark(1000);

  irsend.space(1000);

}

}
```

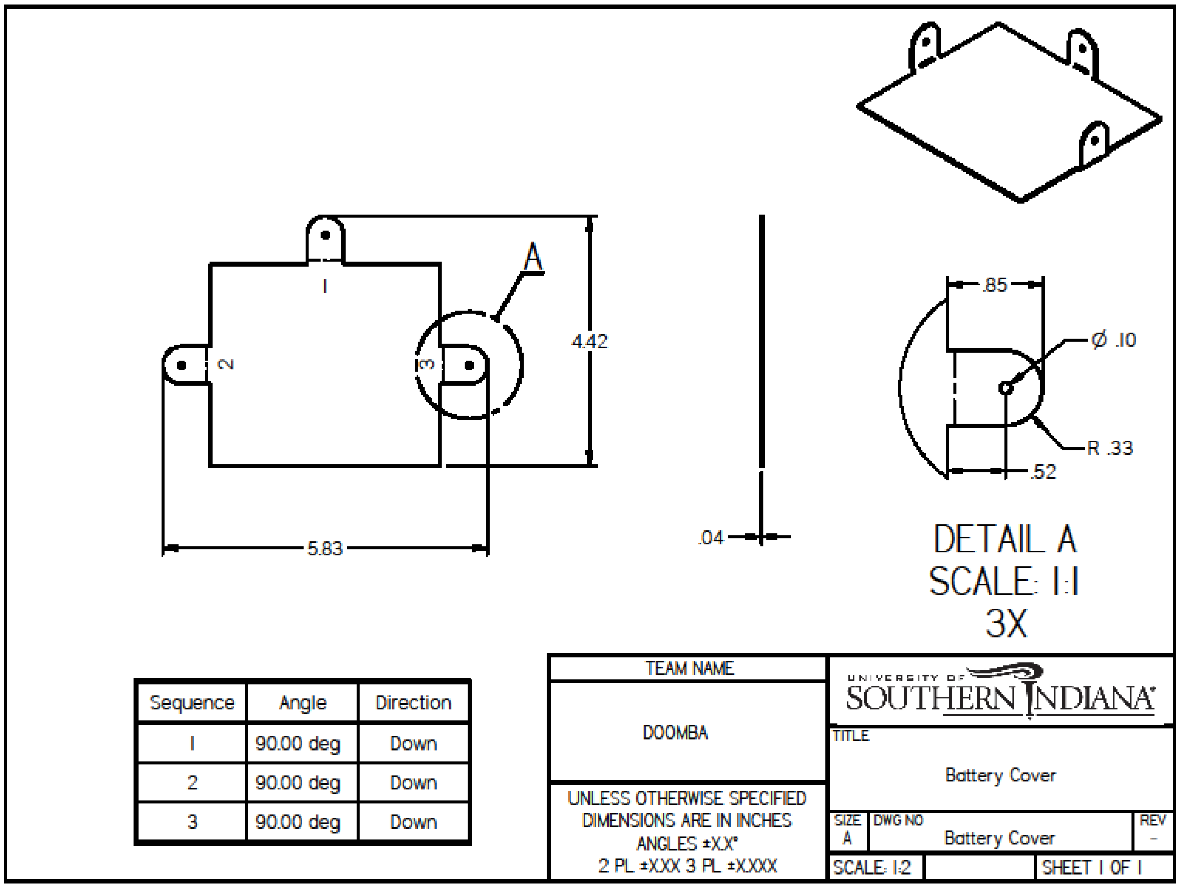
APPENDIX E


U11
LEFT FRONT
Roomba Wheel Assembly
U4
RIGHT REAR
Roomba Wheel Assembly
U10
RIGHT FRONT
Roomba Wheel Assembly
U7
RIGHT REAR
Roomba Wheel Assembly
U1POWERIN
DROK L298 Dual H Bridge Motor Speed Controller
U1OUT
DROK L298 Dual H Bridge Motor Speed Controller
U1SIGNALIN
DROK L298 Dual H Bridge Motor Speed Controller
U6
5V OUT
Step Down Buck Converter Board
U8
20 V OUT
DEWALT BATTERY ADPATER
STEERING ARDUINO
U2ANALOG
U2DIGITAL
ARDUINO UNO R3
U2POWEROUT
ARDUINO UNO R3
ARDUINO UNO R3
U2USB
U2POWERIN
ARDUINO UNO R3
ARDUINO UNO R3
U5CHANNELS
U5SENSE
FS-iA10B 2.4 G RC 10 Channel Receiver
U5SERVO
FS-iA10B 2.4 G RC 10 Channel Receiver
FS-iA10B 2.4 G RC 10 Channel Receiver
U17
U13
3AA
5_5mmx2_1mm
3AA Battery Pack
U15
5V OUT
Step Down Buck Converter Board
PAYLOAD ARDUINO
U9ANALOG
U9POWEROUT
ARDUINO UNO R3
U9USB
ARDUINO UNO R3
ARDUINO UNO R3
U9DIGITAL
U9POWERIN
ARDUINO UNO R3
ARDUINO UNO R3
U3
AUDIO FX MINI SOUNDBOARD 16MB
SOUNDS
1-INTRO
2-GET OVER HERE
3-GOAT SCREAM
7-CANT TOUCH THIS
U12POWERIN
LOSC DIGITAL AUDIO AMP
U18
U16
U14
3AA
5_5mmx2_1mm
3AA Battery Pack
U12SPEAKERS
U12INPUT
3 W 4 ohm 2 inch Speaker
LOSC DIGITAL AUDIO AMP
LOSC DIGITAL AUDIO AMP
U19
U21
CATHODE -
ANODE +
Gikfun 940nm LED 5mm
220Ω 1% R1
NC ESTOP 2 LINE
CURRENTLY NOT USED
U20
3 W 4 ohm 2 inch Speaker
UNIVERSITY OF SOUTHERN INDIANA
BUSINESS & ENGINEERING CENTER
ENGR 491 - SENIOR DESIGN
8600 UNIVERSITY BLVD.
EVANSVILLE, IN 47712
UNIVERSITY OF SOUTHERN INDIANA
DATE:
TITLE: BATTLE BOT MODEL 1- TEAM DOOMBA
DESIGNED BY: F.BUEHL 09/29/2021
DESC: WIRING DIAGRAM- DOOMBA DOZER
CHECKED BY: P.BRUNER 10/28/2021
DOCUMENT # ENGR491AA09292021
REVISION -
APPROVED BY: C.CAMPBELL 11/04/2021
SHEET 1 OF 1
SIZE B

APPENDIX F


U1POWERIN
DROK L298 Dual H Bridge Motor Speed Controller
U1OUT
DROK L298 Dual H Bridge Motor Speed Controller
U1SIGNALIN
DROK L298 Dual H Bridge Motor Speed Controller
U11
Roomba Wheel Assembly
U7
Roomba Wheel Assembly
U6
5V OUT
Step Down Buck Converter Board
U8
20 V OUT
DEWALT BATTERY ADPATER
STEERING ARDUINO
U2ANALOG
U2DIGITAL
ARDUINO UNO R3
U2POWEROUT
ARDUINO UNO R3
ARDUINO UNO R3
U2USB
U2POWERIN
ARDUINO UNO R3
ARDUINO UNO R3
U5CHANNELS
FS-iA10B 2.4 G RC 10 Channel Receiver
U5SENSE
FS-iA10B 2.4 G RC 10 Channel Receiver
U5SERVO
FS-iA10B 2.4 G RC 10 Channel Receiver
U17
5_5mmx2_1mm
U13
3AA
3AA Battery Pack
U15
5V OUT
Step Down Buck Converter Board
PAYLOAD ARDUINO
U9ANALOG
ARDUINO UNO R3
U9POWEROUT
ARDUINO UNO R3
U9USB
ARDUINO UNO R3
U9DIGITAL
U9POWERIN
ARDUINO UNO R3
ARDUINO UNO R3
U3
Gikfun 940nm LED 5mm
R1
220Ω
1%
U16
5_5mmx2_1mm
U14
3AA
3AA Battery Pack
U10
1 Channel High Level Driver Optocoupler Relay Module
U4
1 Channel High Level Driver Optocoupler Relay Module
U21
ANGLE GRINDER
U18
BATTERY ADAPTER
U19
NC ESTOP 2 LINE
U22
ANGLE GRINDER
U20
BATTERY ADAPTER
UNIVERSITY OF SOUTHERN INDIANA
BUSINESS & ENGINEERING CENTER
ENGR 491 - SENIOR DESIGN
8600 UNIVERSITY BLVD.
EVANSVILLE, IN 47712
DATE:
TITLE: BATTLE BOT MODEL 2 - TEAM DOOMBA
DESIGNED BY: F.BUEHL 10/29/2021
DESC: WIRING DIAGRAM - DOOMBA DICER
CHECKED BY: P.BRUNER 10/29/2021
DOCUMENT #: ENGR491AA10292021
REVISION: -
APPROVED BY: C.CAMPBELL 11/04/2021
SHEET 1 OF 1
SIZE: B

APPENDIX G



Sequence	Angle	Direction
1	90.00 deg	Down
2	90.00 deg	Down
3	90.00 deg	Down

TEAM NAME

DOOMBA

UNLESS OTHERWISE SPECIFIED
DIMENSIONS ARE IN INCHES
ANGLES ±X.X°
2 PL ±X.XX 3 PL ±X.XXX

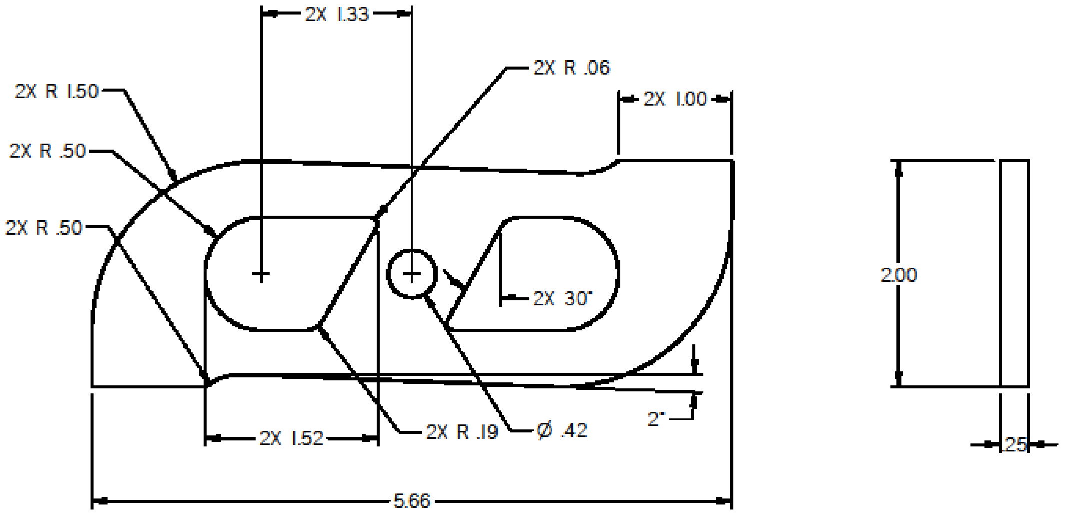
UNIVERSITY OF SOUTHERN INDIANA

TITLE

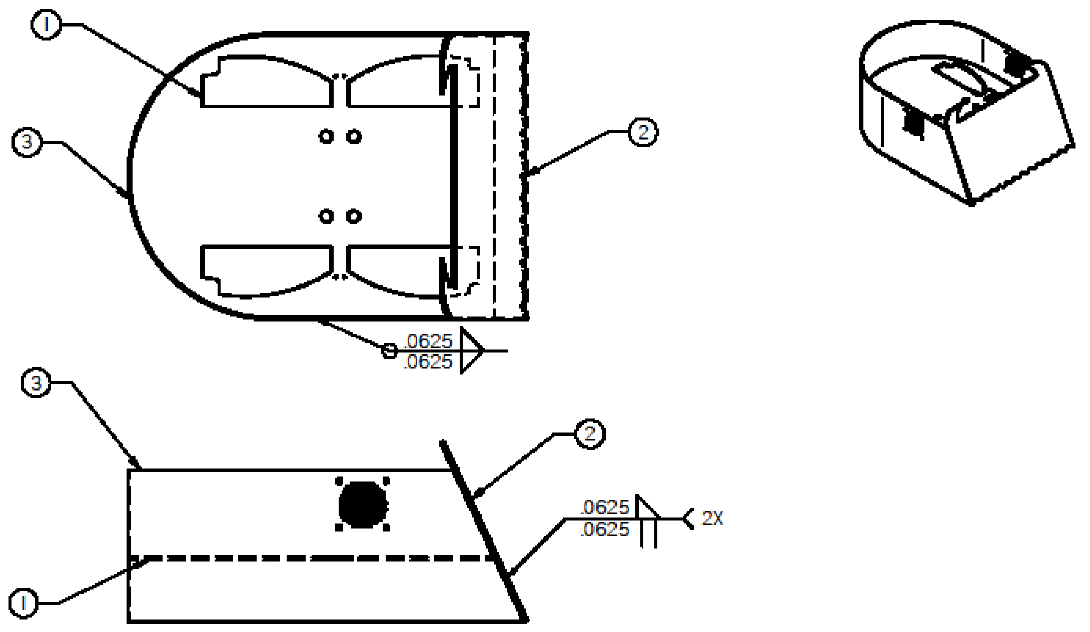
Battery Cover

SIZE	DWG NO	REV
A	Battery Cover	–

SCALE: 1:2 | SHEET 1 OF 1



TEAM NAME	UNIVERSITY OF SOUTHERN INDIANA		
DOOMBA	TITLE Blade		
UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES ANGLES ±X.X° 2 PL ±X.XX 3 PL ±X.XXX	SIZE A	DWG NO Blade	REV –
	SCALE: 1:1		SHEET 1 OF 1



Item Number	File Name (no extension)	Quantity
1	Dozer Base Plate	1
2	Push Plate	1
3	Skirt	1

TEAM NAME

DOOMBA

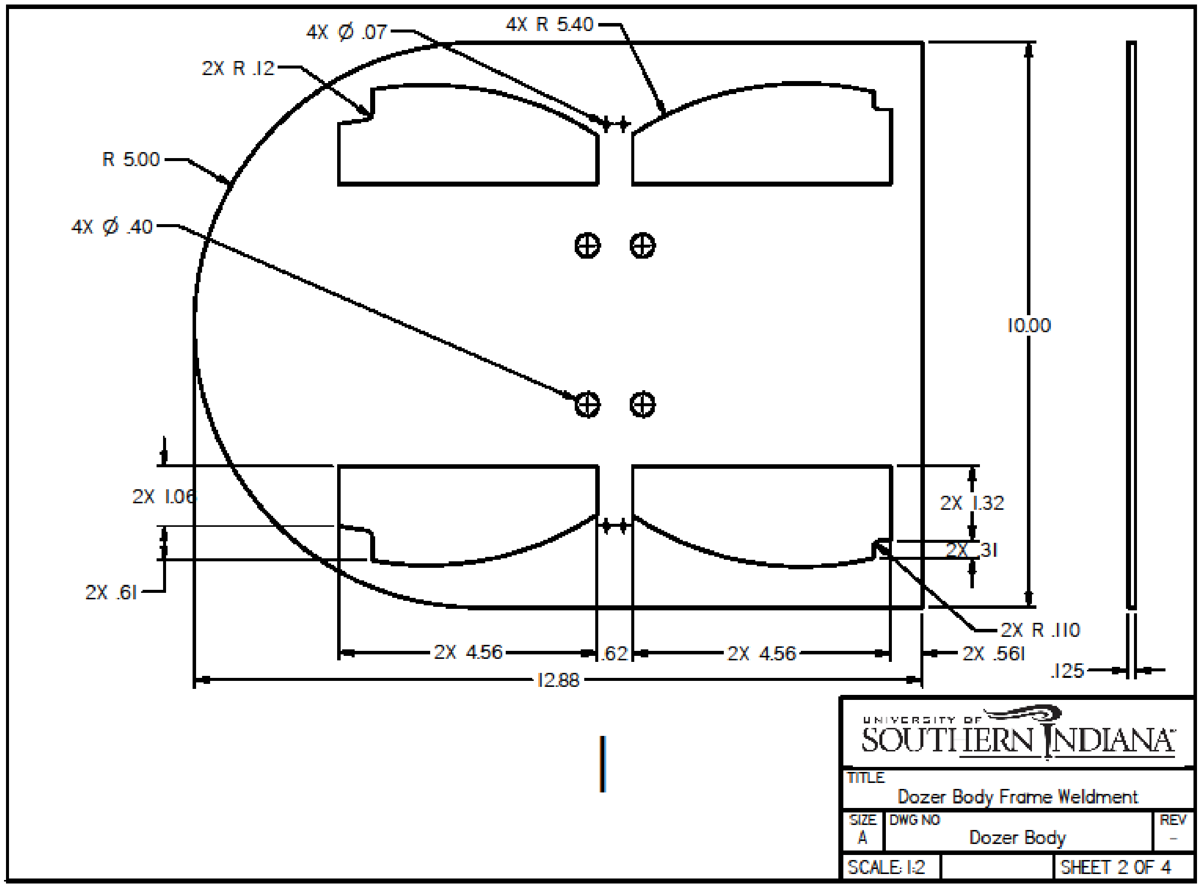
UNLESS OTHERWISE SPECIFIED
DIMENSIONS ARE IN INCHES
ANGLES ±X.X°
2 PL ±X.XX 3 PL ±X.XXX

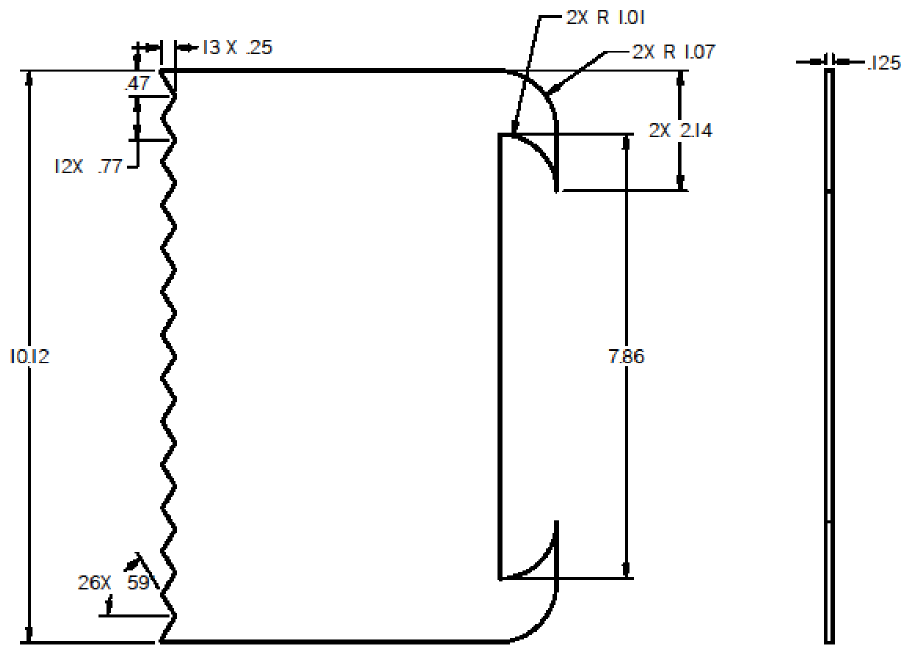
UNIVERSITY OF SOUTHERN INDIANA®

TITLE
Dozer Body Frame Weldment

SIZE	DWG NO	REV
A	Dozer Body	-

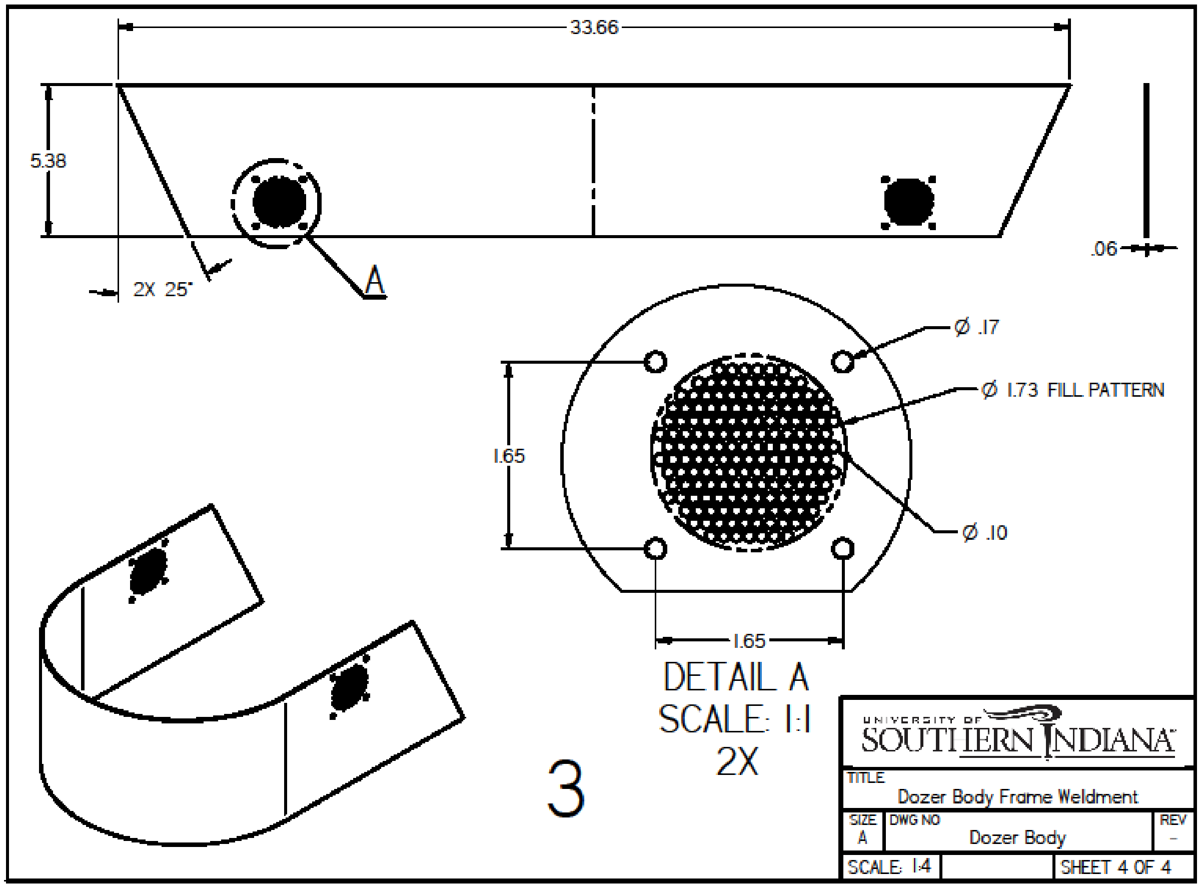
SCALE: 1:4 | SHEET 1 OF 4

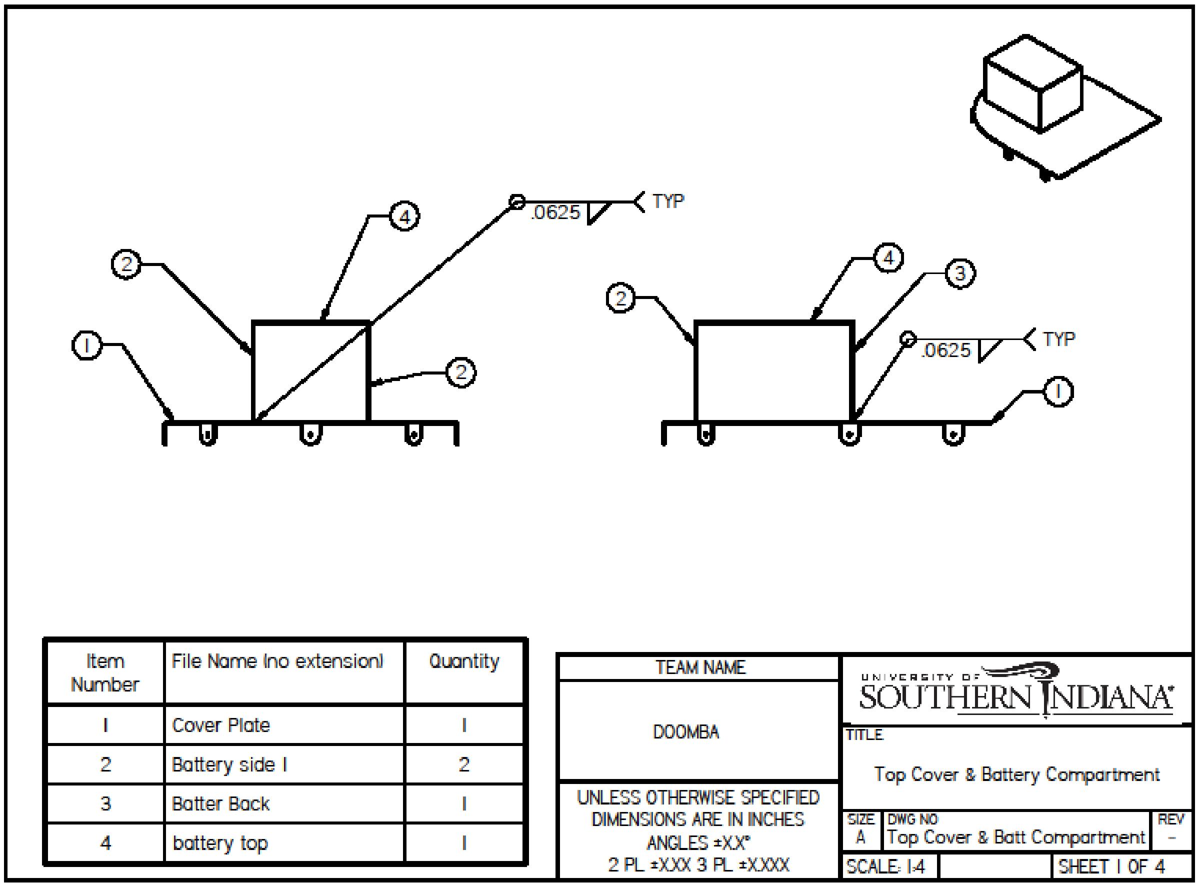

4X Ø .07
4X R 5.40
2X R .12
R 5.00
4X Ø .40
10.00
2X 1.06
2X .61
2X 1.32
2X .31
2X R .110
2X 4.56
.62
2X 4.56
2X .561
12.88
.125
UNIVERSITY OF SOUTHERN INDIANA
TITLE
Dozer Body Frame Weldment
SIZE
A
DWG NO
Dozer Body
REV
–
SCALE: 1:2
SHEET 2 OF 4



2

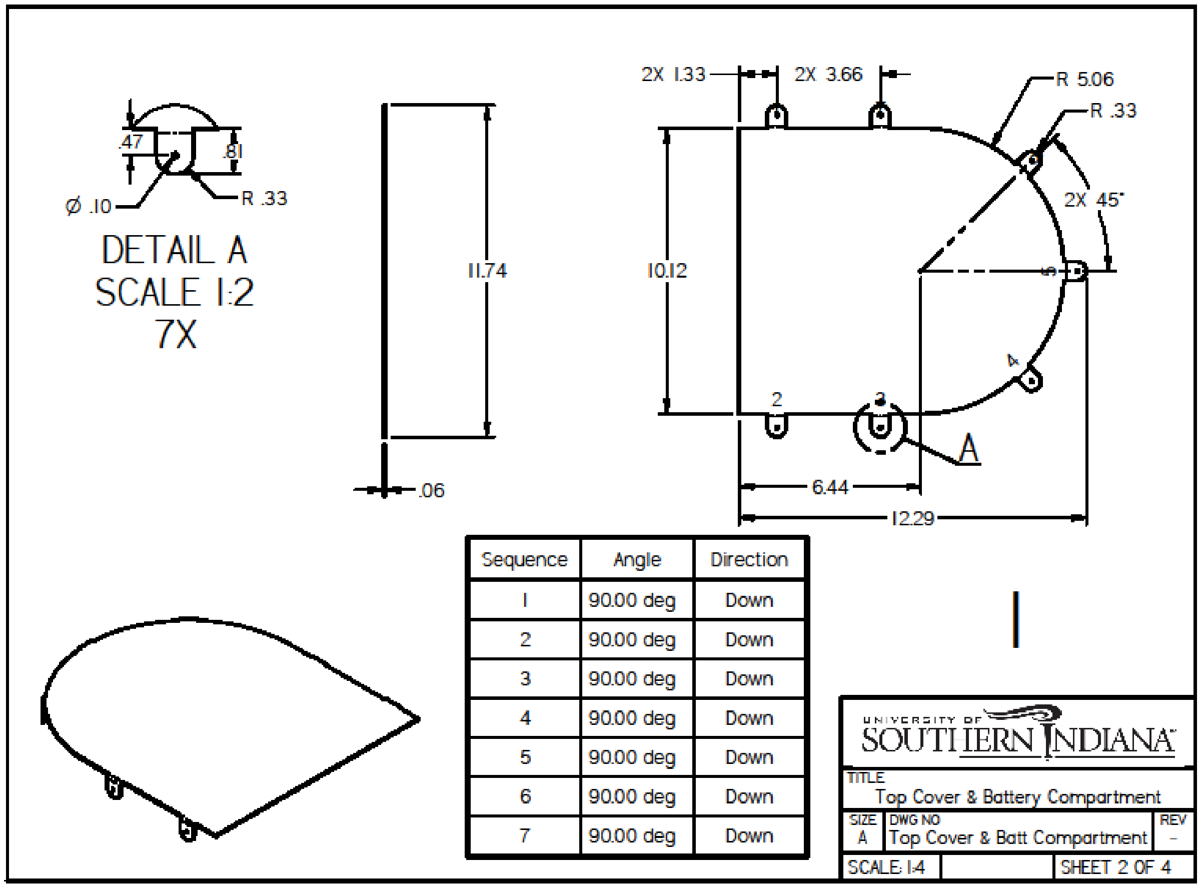
UNIVERSITY OF SOUTHERN INDIANA		
TITLE Dozer Body Frame Weldment		
SIZE A	DWG NO Dozer Body	REV –
SCALE: 1:2		SHEET 3 OF 4


33.66
5.38
2X 25°
A
.06
Ø .17
Ø 1.73 FILL PATTERN
1.65
Ø .10
1.65
DETAIL A
SCALE: 1:1
2X
3
UNIVERSITY OF SOUTHERN INDIANA
TITLE
Dozer Body Frame Weldment
SIZE
A
DWG NO
Dozer Body
REV
-
SCALE: 1:4
SHEET 4 OF 4



Item Number	File Name (no extension)	Quantity
1	Cover Plate	1
2	Battery side 1	2
3	Batter Back	1
4	battery top	1

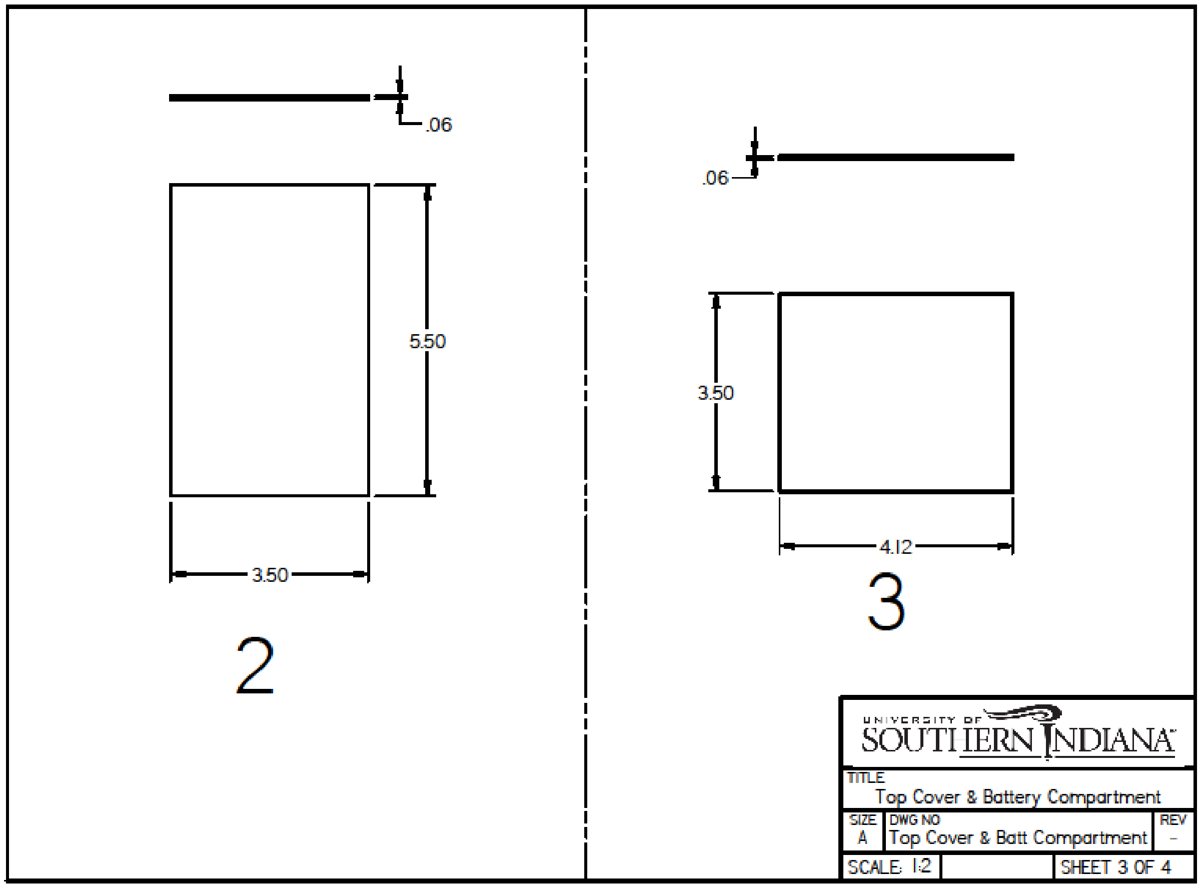
TEAM NAME	UNIVERSITY OF SOUTHERN INDIANA
DOOMBA	TITLE Top Cover & Battery Compartment
UNLESS OTHERWISE SPECIFIED DIMENSIONS ARE IN INCHES ANGLES ±X.X° 2 PL ±X.XX 3 PL ±X.XXX	SIZE A · DWG NO Top Cover & Batt Compartment · REV -
	SCALE: 1:4 · SHEET 1 OF 4

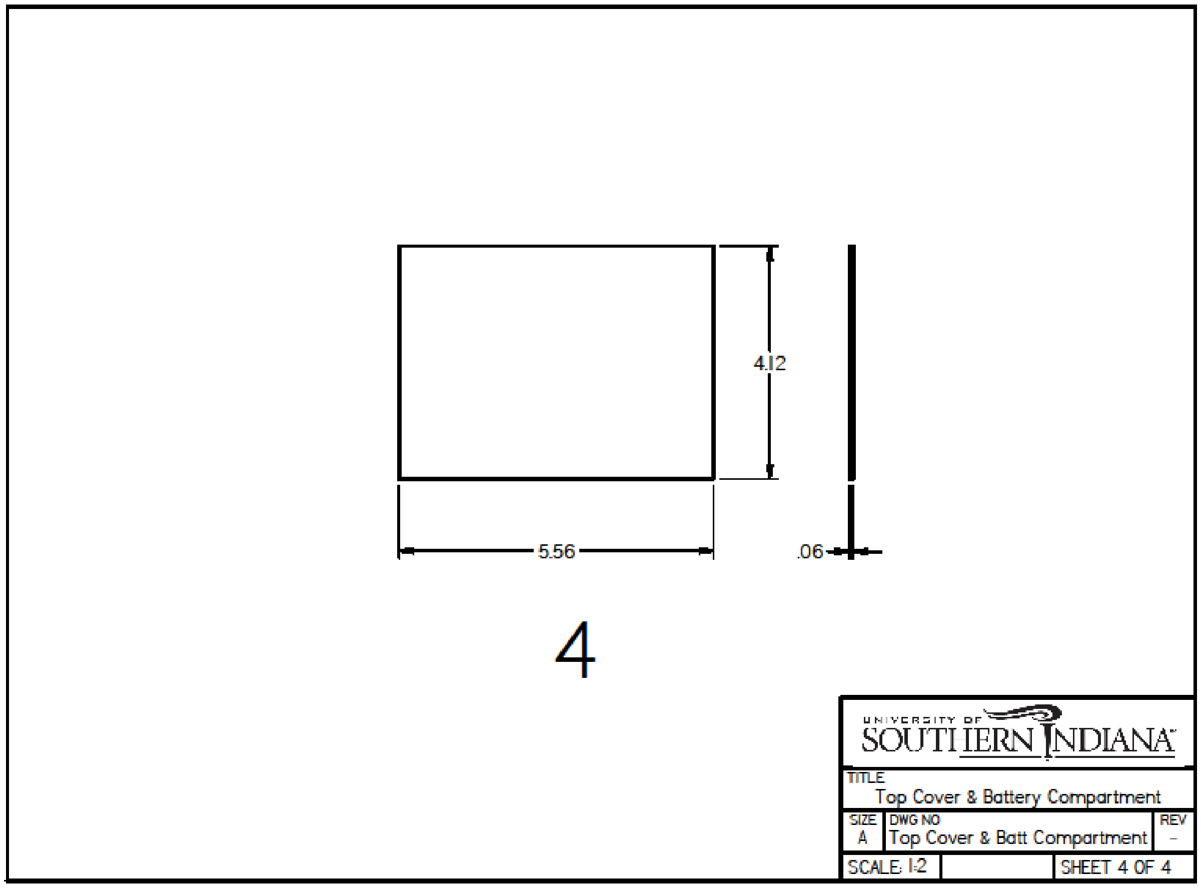


Sequence	Angle	Direction
1	90.00 deg	Down
2	90.00 deg	Down
3	90.00 deg	Down
4	90.00 deg	Down
5	90.00 deg	Down
6	90.00 deg	Down
7	90.00 deg	Down

UNIVERSITY OF SOUTHERN INDIANA

TITLE Top Cover & Battery Compartment		
SIZE A	DWG NO Top Cover & Batt Compartment	REV –
SCALE: 1:4		SHEET 2 OF 4


.06
5.50
3.50
2
.06
3.50
4.12
3
UNIVERSITY OF SOUTHERN INDIANA
TITLE
Top Cover & Battery Compartment
SIZE
A
DWG NO
Top Cover & Batt Compartment
REV
-
SCALE: 1:2
SHEET 3 OF 4



4.12
5.56
.06
4
UNIVERSITY OF SOUTHERN INDIANA
TITLE
Top Cover & Battery Compartment
SIZE
A
DWG NO
Top Cover & Batt Compartment
REV
-
SCALE: 1:2
SHEET 4 OF 4