AUTOMATIC FIRE TRUCK:  
LIL PUMP

DIY MANUAL

Final Project

Marc Fricker

May 10, 2019

By: Neal Evans

Table of Contents

[TABLE OF FIGURES 3](#_Toc8421094)

[INTRODUCTION 5](#_Toc8421095)

[Description of Prototype 5](#_Toc8421096)

[Before Beginning 5](#_Toc8421097)

[THEORY & DESIGN 6](#_Toc8421098)

[Basic Functionality 6](#_Toc8421099)

[Design Choices – Components 6](#_Toc8421100)

[Design Choices – PCB 7](#_Toc8421101)

[BILL OF MATERIALS/TOOLS 8](#_Toc8421102)

[Electrical Components and Parts 8](#_Toc8421103)

[PCB 9](#_Toc8421104)

[PCB ASSEMBLY 10](#_Toc8421105)

[Important Tools 10](#_Toc8421106)

[Other Tools And Resources 10](#_Toc8421107)

[PCB Soldering 10](#_Toc8421108)

[Extra MOSFET Switch 13](#_Toc8421109)

[ROBOT CHASSIS ASSEMBLY 15](#_Toc8421110)

[Chassis Instruction Manual 15](#_Toc8421111)

[Wheel Tracks 16](#_Toc8421112)

[Inside of the Chassis 18](#_Toc8421113)

[3D PRINTED ENCLOSURE 23](#_Toc8421114)

[Autodesk Fusion 360 23](#_Toc8421115)

[WIRE CONNECTIONS 23](#_Toc8421116)

[How to Connect Wires 24](#_Toc8421117)

[Pump and Sensor Suite Placement 36](#_Toc8421118)

[SOFTWARE AND FIRMWARE 39](#_Toc8421119)

[Firmware 39](#_Toc8421120)

[Software 39](#_Toc8421121)

[GUI 39](#_Toc8421122)

[USER GUIDE/TESTING 41](#_Toc8421123)

[Turning it On 41](#_Toc8421124)

[Test Cases 41](#_Toc8421125)

[Troubleshoot Cases 42](#_Toc8421126)

[Mis-use Cases 42](#_Toc8421127)

[FUTURE CONSIDERATIONS 43](#_Toc8421128)

[APPENDIX 44](#_Toc8421129)

[Definitions 44](#_Toc8421130)

[Pertinent Documents/Links 44](#_Toc8421131)

# TABLE OF FIGURES

[Figure 1 PCB Layout and 3D PCB Layout 9](#_Toc8421175)

[Figure 2 PCB Component Placements 11](#_Toc8421176)

[Figure 3 Populated PCB 12](#_Toc8421177)

[Figure 4 Microcontroller on PCB 13](#_Toc8421178)

[Figure 5 Extra MOSFET Piece 14](#_Toc8421179)

[Figure 6 Tracked Wheels and Bottom Half of Chassis 15](#_Toc8421180)

[Figure 7 Top Half of Chassis and Gears Added to Chassis 15](#_Toc8421181)

[Figure 8 Constructed Chassis 16](#_Toc8421182)

[Figure 9 Inserting SIM Card Ejection Pin 16](#_Toc8421183)

[Figure 10 Removing Metal Rod and Excess Tracks 17](#_Toc8421184)

[Figure 11 Wheel Tracks Perfect Fit 17](#_Toc8421185)

[Figure 12 Inside of Chassis 18](#_Toc8421186)

[Figure 13 Soldering Toggle Switch 19](#_Toc8421187)

[Figure 14 Toggle Switch Completed 19](#_Toc8421188)

[Figure 15 H. Bridge Jumpers and Removing the Jumpers 20](#_Toc8421189)

[Figure 16 H. Bridge Digital Wires and Power Wires 20](#_Toc8421190)

[Figure 17 Toggle Switch to Battery 21](#_Toc8421191)

[Figure 18 Completed Chassis 22](#_Toc8421192)

[Figure 19 PCB Hanging Out of Enclosure 24](#_Toc8421193)

[Figure 20 Connecting the ENA/ENB Pins 25](#_Toc8421194)

[Figure 21 Connecting Ultrasonic Sensors 26](#_Toc8421195)

[Figure 22 Connecting VCC/GND Power Wires 27](#_Toc8421196)

[Figure 23 Connecting the Servo Motor 28](#_Toc8421197)

[Figure 24 Connecting L298N Power Wires 29](#_Toc8421198)

[Figure 25 Connecting MOSFET Switch to Digital Pin 30](#_Toc8421199)

[Figure 26 Connecting the Thermal Sensor 31](#_Toc8421200)

[Figure 27 Connecting PWM Pins 32](#_Toc8421201)

[Figure 28 Connecting VIN Power Wires 33](#_Toc8421202)

[Figure 29 Connecting Pump to MOSFET Switch 34](#_Toc8421203)

[Figure 30 Connecting Water Level Sensor 35](#_Toc8421204)

[Figure 31 Water Pump and Water Level Sensor 36](#_Toc8421205)

[Figure 32 Servo Motor, Thermal Sensor and Tube 37](#_Toc8421206)

[Figure 33 Ultrasonic Sensors Orientation 38](#_Toc8421207)

[Figure 34 Arduino IDE 40](#_Toc8421208)

[Figure 35 Robot GUI 40](#_Toc8421209)

# INTRODUCTION

## Description of Prototype

The Automatic Fire Truck, named “*Lil Pump”*, is an automated robot prototype that shoots out water when there’s a fire present. It is completely independent, so human input is not needed (except for turning it on and using the GUI, which will be covered later in the manual). It will search and scan its surroundings until it locates a flame to be put out, otherwise it roams around the room.

This prototype uses water to function, so make sure that there is water available to test the functionality. Also, be careful not to get any of the components wet, since they might fail if exposed to considerable amounts of moisture.

This manual will show how to construct the prototype, as well as explain the theory and design choices made to develop it. It will list all the parts/components and tools necessary to build it. Additionally, it will show the [PCB](#_Definitions) design of the [daughterboard](#_Definitions) and instruct how to populate it. Also, it will explain how the enclosure was made and it will show how to 3D print it.

***WARNING: THIS IS A PROTOTYPE. DO NOT DEPEND ON THE ROBOT FOR ACTUAL FIRE PREVENTION.***

## Before Beginning

* Read through the theory to learn about the prototype.
* Allocate enough working space.
* Plan ahead to save enough time to build.
* Find a professional or watch a tutorial video for knowledge on how to solder (a link for a tutorial can be found [here](https://www.youtube.com/watch?v=Qps9woUGkvI).)
* Sort and organize the screws accordingly.
* For testing purposes, it will be necessary to light a fire. **DISCLAIMER: Exert caution when dealing with an open flame. Do not play with it or do anything else unnecessary!**
* Most importantly, remember to have fun!



# THEORY & DESIGN

## Basic Functionality

The functionality of the prototype isn’t very complex. Simply put, the robot will continuously roam forward until an obstacle appears in front of it. If it detects an obstacle, it will stop moving and it will activate the servo motor. On top of the servo motor is a thermal sensor that reads temperature and a tube attached to a water pump. The servo motor will move in a counter-clockwise sweeping motion to start scanning for a target, in this case: a flame. If a flame is found, it will use the pump to shoot water from its tank. It will sweep a second time clockwise to make sure the flame has been put out. Finally, the robot will turn away from the obstacle and begin to roam again in another direction to repeat this process.

## Design Choices – Components

For the development of the prototype, certain design choices were made. The first was to find a suitable platform/chassis that would be able to carry its weight. It also needed to be able to drive on many different types of terrain. The solution was the “DFRobot Devastator Tank Mobile Platform”. Its sturdiness, size, included DC motors and tracked wheels satisfied the characteristics needed to build the prototype. It has the strength to carry the rest of the components on top of it and the ability to drive on several terrain types.

Another consideration was power. The chassis had the appropriate strength and size, but it also needed an appropriate amount of power. However, the power that the wheels needed was too much for the other components to handle. So, instead of using just one power source, the robot has two: one specifically for the DC motors and one for the rest of the robot. A LiPo battery was used as the power source for the DC motors. It has a high enough voltage to drive the motors and enough current to keep it running for a considerable amount of time. The rest of the robot could be powered by AA batteries.

The third and fourth consideration were the water tank and water pump. The latter had to be strong enough to shoot out a powerful stream of water. Luckily, a 3V water pump is enough for the task. It’s powerful enough to shoot out a jet of water and it draws a small amount of current, so it can use the AA battery power source. The former was more difficult to design. It needed to be big enough to hold a substantial amount of water, but also small enough to not take up too much room on the chassis. The solution was to create an enclosure that would house all the components on top of the chassis and that would also act as a tank that could fill a reasonable amount of water.

The next consideration was to determine a good [sensor suite](#_Definitions) to detect the fire and any obstacle or wall that appears in front of it. The sensor suite would need to be comprised of two different sensor components: a sensor that reads temperature and a sensor that reads the distance between itself and another object. Because the robot only needs to avoid getting too close to another object, the sensor measuring distance didn’t need to be powerful. So, the cheapest option was picked: the HC-SR04 ultrasonic sensor. It emits an echo that bounces off objects and back to the sensor to read/measure distances. More than one of these sensors are used in the prototype to increase accuracy.  
The temperature reading sensor needed to be sensitive enough to accurately find fires. So, the D6T-1A-01 was purchased. It can calculate the average temperature of what’s in front of it, and fortunately, it isn’t expensive. For it to be accurate when finding fires, it will work alongside a servo motor. The servo motor will move the thermal sensor to different positions to search for a flame.

Another consideration was measuring the water volume inside of the tank. Since this is a prototype of a miniaturized version of an actual fire truck, it wouldn’t be sensible to not be able to monitor the amount of water left in the tank. So, another ultrasonic sensor was used. It measures the distance between itself and the water to calculate the amount of volume left.   
Another design choice was to use this water sensor to determine if the robot should operate or not. If the volume of water is too low, then the robot will not run.

Also, a [microcontroller](#_Definitions) was needed. The microcontroller will run the robot’s programming and administer all the components. The best option was to use an Arduino MEGA. The Arduino is a popular microcontroller used by many people, so there are many examples of software and support documents that made it easy to research how to code the prototype. Additionally, the Arduino MEGA had a large number of available pins: some to connect the components of the robot, and the rest for future use. The extra pins make it possible to upgrade the robot in the future if need be.

Lastly, an extra piece was added outside of the PCB. This was because of a mistake done on the PCB and unfortunately, there wasn’t enough to time to re-order a new one. So, to remedy the issue, the mistake had to be corrected by soldering an extra piece that connects from the outside of the PCB. It is a MOSFET switch that will allow the water pump to work. There is another MOSFET switch on the PCB that exists for the same purpose, but it’s only used for pumps that can handle a voltage of 12V or more. A 12V water pump was too complicated to implement. This new MOSFET switch works for water pumps that handle 5V or less, like the current water pump that’s being used for this prototype.

## Design Choices – PCB

The purpose of the PCB is to have one component that links all the other components together. For that reason, the PCB is mostly populated with headers and terminal blocks. There isn’t much else in terms of circuitry.

Because of this, the placement of the headers and terminal blocks mattered the most. All of them needed to be placed in areas that are accessible for easier wire connections. Meaning they were all placed around the edges of the PCB. The microcontroller would be plugged in at the middle, because it rarely needs to be accessed.

The rest of the PCB is comprised of placements for some resistors, a MOSFET switch and two LEDs: one to indicate that the power is on and the other to indicate that the pump has been activated. There are also extra microcontroller pin headers added onto the PCB for future upgrades.

# BILL OF MATERIALS/TOOLS

## Electrical Components and Parts

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Quantity** | **Item** | **Purchased from:** | PART NUMBER (DIGIKEY) | **Sub-total $** | **Shipping $** | **TOTAL $** |
| 1 | Arduino MEGA + Cable | Amazon Prime |  | 21.86 | 0 | 21.86 |
| 1 | L298N DC Motor Driver | GearBest |  | 3.92 | 4.8 | 8.72 |
| 1 | D6T-1A-01 Thermal Sensor | DigiKey | Z10315-ND | 24.5 | 8 | 32.5 |
| 1 | D6T-HARNESS-02 Cable | DigiKey | Z10407-ND | 18.89 | 4 | 22.89 |
| 10 | 10k Ohm Resistors | DigiKey | [CF14JT10K0CT-ND](https://www.digikey.ca/product-detail/en/stackpole-electronics-inc/CF14JT10K0/CF14JT10K0CT-ND/1830374) | 0.6 | 0 | 0.6 |
| 10 | 330 Ohm Resistors | DigiKey | [CF14JT330RCT-ND](https://www.digikey.ca/product-detail/en/stackpole-electronics-inc/CF14JT330R/CF14JT330RCT-ND/1830338) | 0.6 | 0 | 0.6 |
| 3 | Blue LEDs | DigiKey | [732-5019-ND](https://www.digikey.ca/product-detail/en/wurth-electronics-inc/151053BS04500/732-5019-ND/4490021) | 0.9 | 0 | 0.9 |
| 3 | Red LEDs | DigiKey | [732-5021-ND](https://www.digikey.ca/product-detail/en/wurth-electronics-inc/151053RS03000/732-5021-ND/4490027) | 0.9 | 0 | 0.9 |
| 1 | Devastator Tank Mobile Robot Platform | Robot Shop |  | 106.65 | 0 | 106.65 |
| 1 | 3V Vertical Water Pump | Robot Shop |  | 5.2 | 0 | 5.2 |
| 1 | Jumper Wires | Robot Shop |  | 7.87 | 0 | 7.87 |
| 3 | Screw Terminal Blocks | DigiKey | 1849-1111-ND | 6.84 | 8 | 14.84 |
| 3 | Push Down Terminal Blocks | DigiKey | 277-1667-ND | 1.83 | 0 | 1.83 |
| 1 | 14.8V LiPo Battery | Amazon Prime |  | 35.99 | 0 | 35.99 |
| 1 | AA Batteries Set | Amazon Prime |  | 22 | 0 | 22 |
| 5 | HC-SR04 Ultrasonic Emitter | Amazon Prime |  | 11.99 | 0 | 11.99 |
| 5 | PCB V2.1a | PCBWay |  | 37.15 | 0 | 37.15 |
| 1 | Servo Motor | Given | 1738-1300-ND | 0 | 0 | 0 |
| 2 | IRLU024N Transistors | Given | IRLU024NPBF-ND | 0 | 0 | 0 |
| 10 | 2-by-20 Male Pins Headers | Given | 609-3221-ND | 0 | 0 | 0 |
| 10 | 1-by-8 Male Pins Headers | Given | 952-1992-ND | 0 | 0 | 0 |
| 2 | 1N4001 Diodes | Given | 641-1310-1-ND | 0 | 0 | 0 |
| 1 | Pair of XT60 Connectors (Male & Female) | Amazon Prime |  | 9.6 | 0 | 9.6 |
|  | **TOTAL** |  |  | **0** | **24.8** | **342.09** |

Table 1 BOM

## PCB

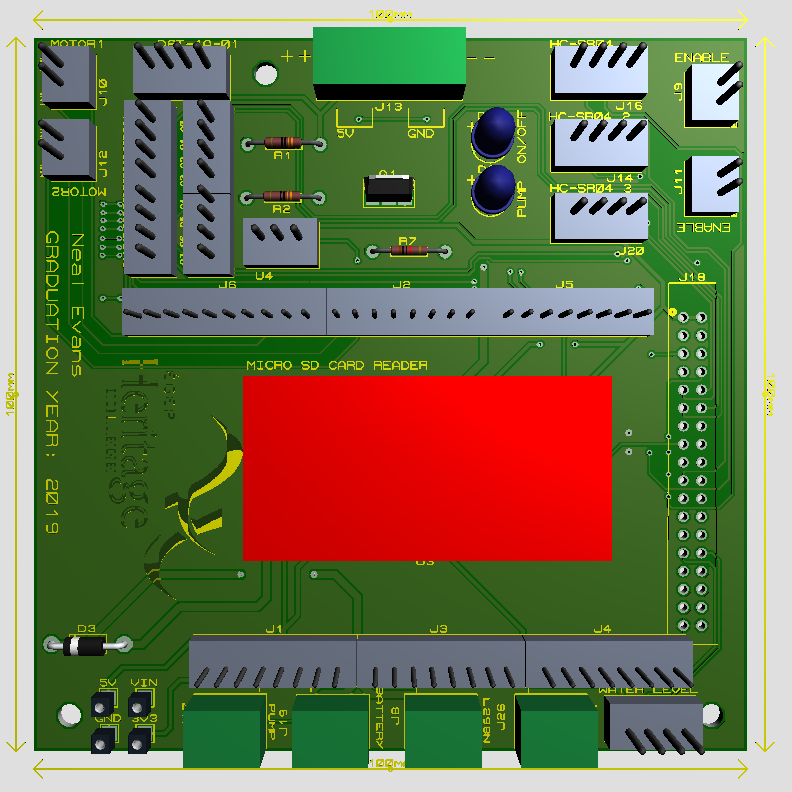
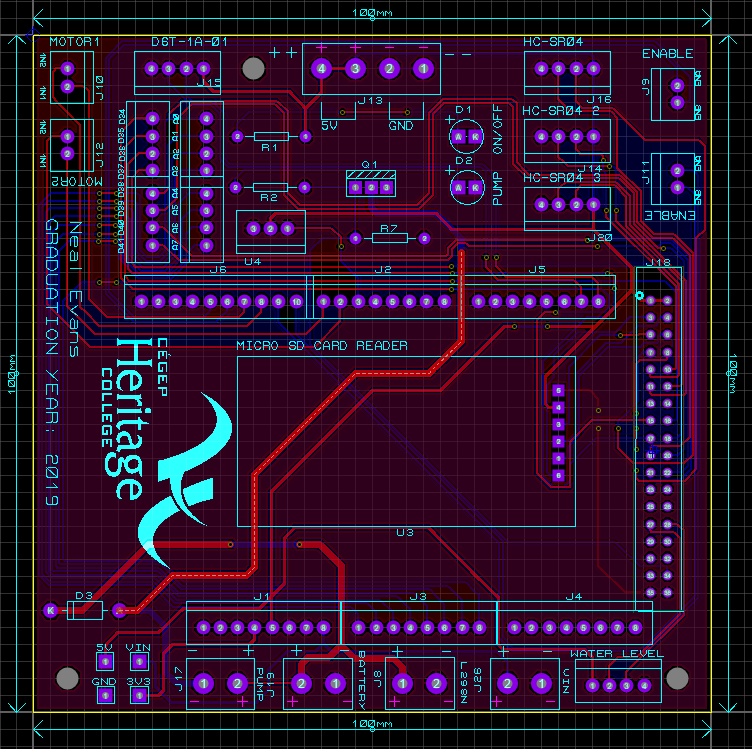


Figure 1 PCB Layout and 3D PCB Layout

The PCB circuitry looks like the image above (Figure 1). The design can be sent to a manufacturer and delivered in about 8 to 15 business days approximately. It is recommended to purchase from “PCBWay”. The first order is free. It will only ask to pay for shipping! Referrals can also provide more coupons and rebates. The GERBER files for the PCB design can be found [here](https://github.com/Chadesky/GERBER-Files).

The link to order a PCB from “PCBWay” can be found [here](https://www.pcbway.com/).

“PCBWay” Parameters/Settings:  
(*DIMENSIONS: 100 mm X 100 mm; QUANTITY: 10; Solder Mask colour can be changed to preference; Keep the rest of the settings to default*)

# PCB ASSEMBLY

## Important Tools

* Philips & Flathead Screwdriver
* Zip Ties
* Wire Stripper/Cutter
* A bunch of wires (male-to-male, male-to-female, female-to-female)
* Soldering Iron & Solder
* Type-B USB Cable
* A computer
* A 3D Printer

## Other Tools and Resources

* Hot Glue Gun
* Masking Tape
* SIM Card Ejection Pin
* Tweezers
* Spray Sealant

*[Some of the tools can be purchased in one convenient product: the “iFixit Essential Electronics Toolkit”! Included in it are a screwdriver with the required bits, a SIM card ejection pin and some tweezers. The link to purchase this can be found* [*here*](https://www.amazon.ca/iFixit-IF145-348-2-Essential-Electronics-Toolkit/dp/B01MRNIFR6/ref=pd_lpo_vtph_469_tr_t_2?_encoding=UTF8&psc=1&refRID=2YWD975PX694R5YA183G)*.]*

## PCB Soldering

To start, solder the components onto the daughterboard, AKA the PCB.

The components required are simple. The locations of where each component sits will be colour coded. The following is needed: pin headers (red), resistors (yellow), LEDs (blue), a transistor (green), a diode (pink), and terminal blocks (orange).

[Any remaining exposed pads by the end of the assembly are intended to be used for testing.]

**[Black rings = GND; White Rings = VCC (5V); Beige Ring = VIN (6V+)]**

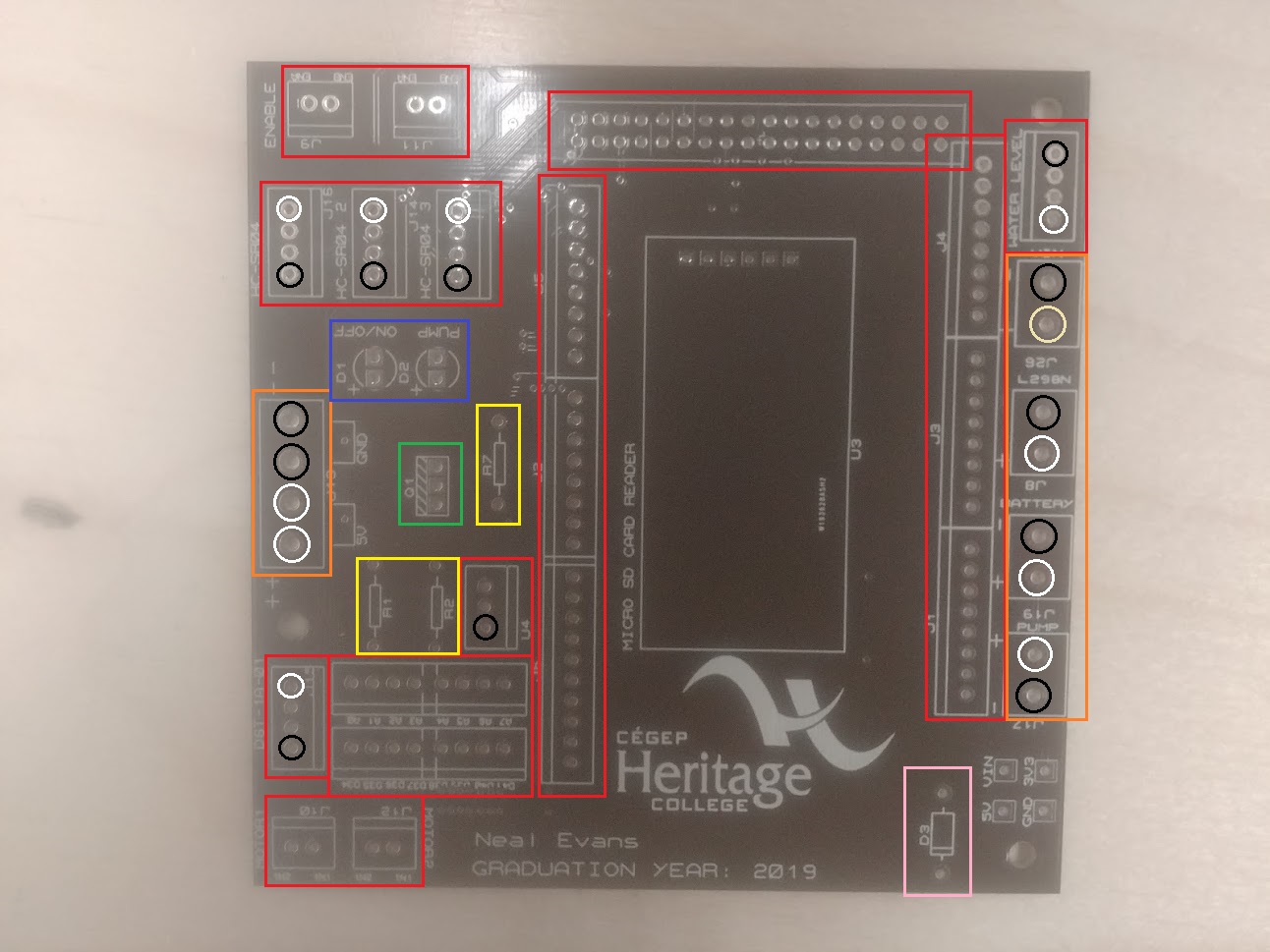


Figure 2 PCB Component Placements

The datasheet for the HC-SR04 can be found [here](https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf). The datasheet for the D6T-1A-01 can be found [here](https://omronfs.omron.com/en_US/ecb/products/pdf/en-d6t.pdf). The datasheet for the servo motor can be found [here](http://www.ee.ic.ac.uk/pcheung/teaching/DE1_EE/stores/sg90_datasheet.pdf). The datasheet for the IRLU024N MOSFET can be found [here](https://www.vishay.com/docs/91322/sihlr024.pdf).

1. First, solder the pin headers at the middle of the motherboard. It will be easier this way because they won’t be blocked by anything (leaves enough clearance). The microcontroller headers go first. Make sure that the longer ends of the pins are facing above the PCB. Solder the shorter ends to the PCB pads. **(J1, J2, J3, J4, J5, J6, J18)**
2. Next, solder the LEDs, the transistor and the resistors to the board. The blue LED goes to “D1” and the red LED goes to “D2”. The 330-ohm resistors go to “R1” and “R2”, and a 10k-ohm resistor goes to “R7”. **(D1, D2, R1, R2, R7)**
3. Followed by the headers for the extra pins. **(D34-D41/A0-A7)**
4. Next, solder the remaining pin headers around the edges of the board, the MOSFET and the diode. The diode goes to “D3” and the IRLU024N MOSFET goes to “U4”. Refer to Figure 2 for the polarities. **(D3, J9, J10, J11, J12, U4, HC-SR04, HC-SR04 2, HC-SR04 3, WATER LEVEL)**
5. Next, solder the terminal blocks to the PCB. Make sure that the insertion holes are facing away from the PCB. The push down terminal blocks are soldered to “VIN” and “PUMP”. **(J17, J29)** The screw terminal blocks fill in the remaining spots. **(J8, J13, J19)**
6. Lastly, screw in the XT60 female connector to the “BATTERY” terminal block (terminal blocks are labelled; be cautious about the polarity, which is also labelled).

The final result should look like Figure 3.

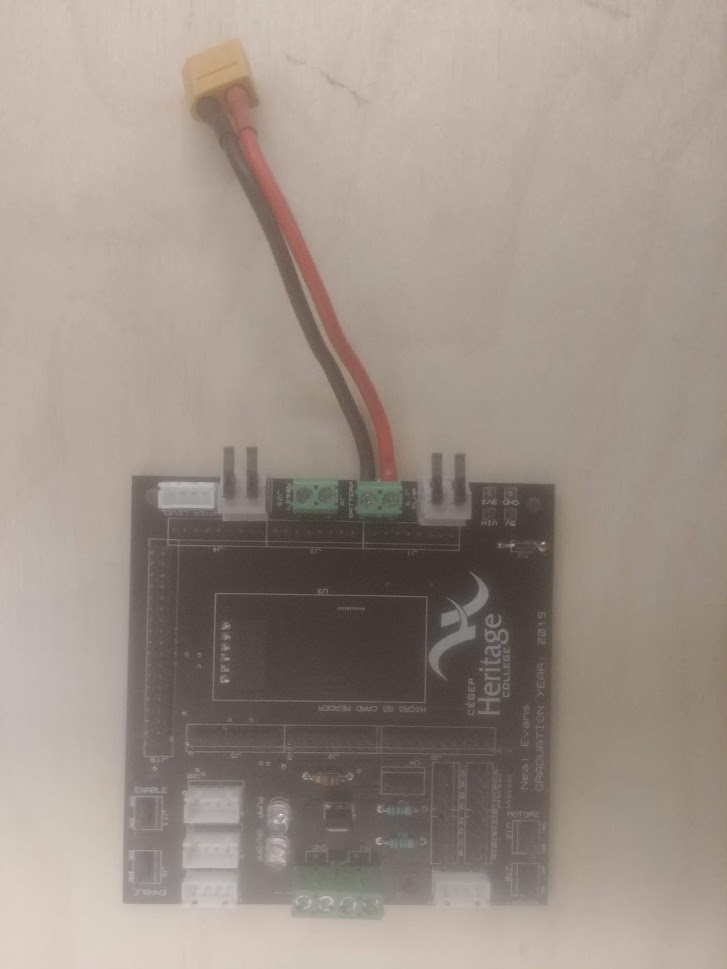


Figure 3 Populated PCB

Next, remove and re-solder the headers on the microcontroller. Instead of them pointing above, they should be pointing downwards by being re-soldered beneath the microcontroller so that it can fit into the pin headers on the PCB.

So that it looks like Figure 4 when you add the microcontroller to the PCB.

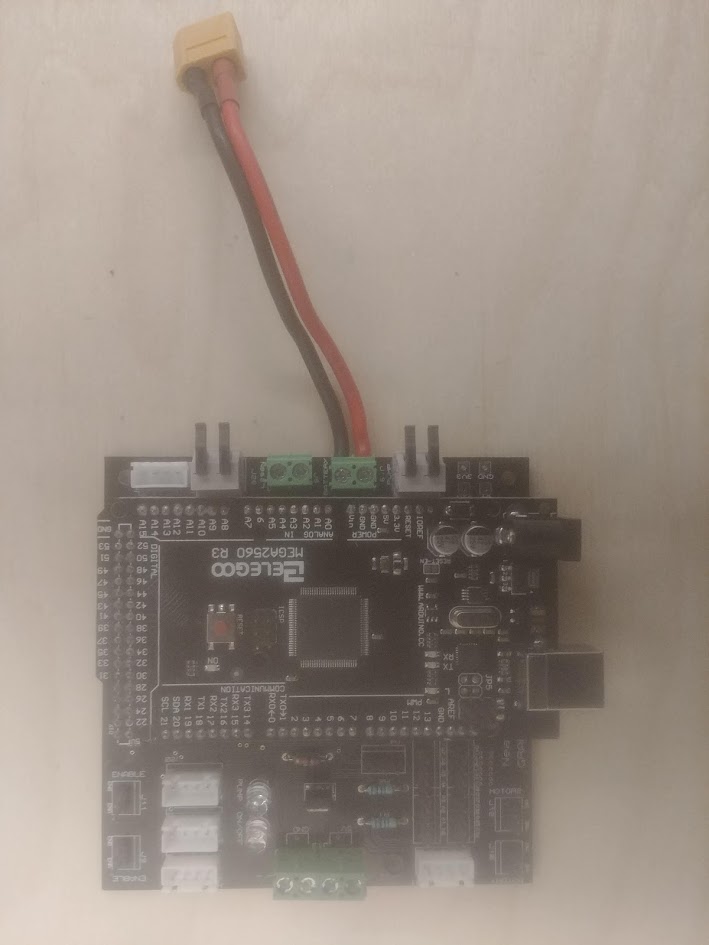


Figure 4 Microcontroller on PCB

## Extra MOSFET Switch

Next, an extra MOSFET switch must be soldered and added to the outside of the chassis. It is comprised of the following: a 10k-ohm resistor, a IRLU024N transistor, some male-to-male wires and some male-to-female wires. Remember to colour code your wires!

Here’s how to assemble this final piece (The datasheet for the MOSFET can be found [here](https://www.vishay.com/docs/91322/sihlr024.pdf)):

1. Solder a male pin wire to one end of a 10k-ohm resistor. This will connect to common GND, later.
2. Solder the other end of the resistor and a female pin wire to the ‘G’ pin of the MOSFET. The female pin wire will connect to a digital pin on the microcontroller, later.
3. Solder a male pin wire and a female pin wire to the negative end of a 1N4001 diode. The male pin wire will connect to VCC and the female pin wire will connect to the positive of the pump, later.
4. Solder two male pin wires (one longer than the other) to the positive end of the diode.
5. Solder the shorter wire of the previous step to the ‘D’ pin of the MOSFET. Keep the other wire free, because it will connect to the negative wire of the pump, later.
6. Lastly, attach a male-to-female wire to the ‘S’ pin of the MOSFET. This will be attached to common GND, later.

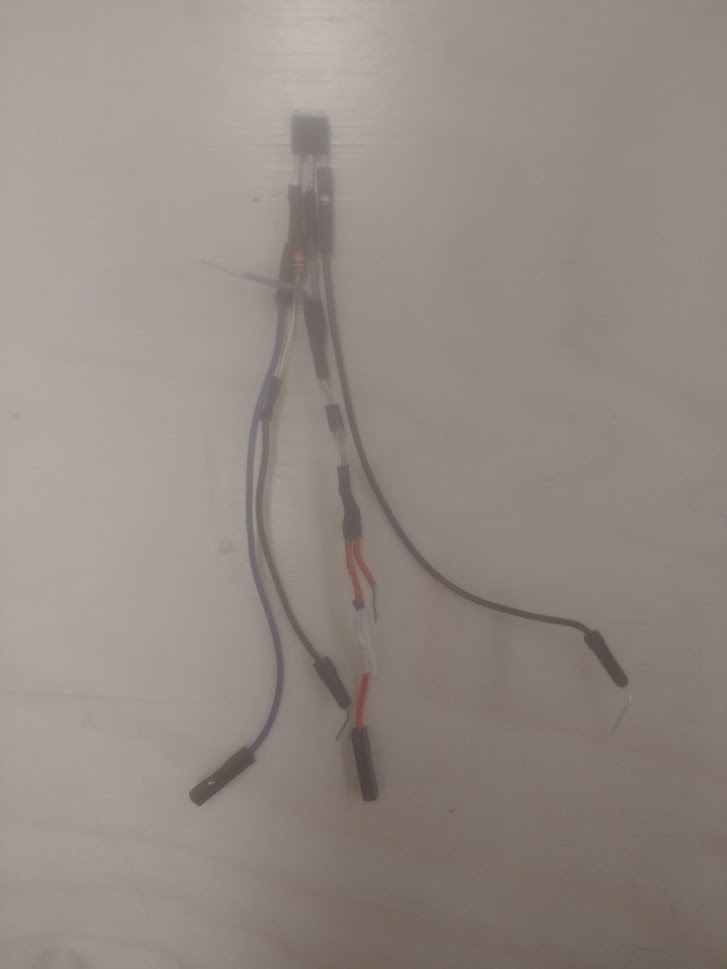


Figure 5 Extra MOSFET Piece

ASSEMBLY OF PCB COMPLETE

# ROBOT CHASSIS ASSEMBLY

The next step is to assemble the “DFRobot Devastator Tank Mobile Platform”, the chassis for the prototype. The enclosure, along with the daughterboard, microcontroller and other components, will sit on top of the chassis. The inside of the chassis will be reserved for the DC motors, the AA battery pack, the LiPo battery and the [PWM](#_Pertinent_Documents/Links) H. Bridge.

## Chassis Instruction Manual

Instructions on how to assemble it can be found [here](https://github.com/Arduinolibrary/DFRobot_Devastator_Tank_Mobile_Platform/raw/master/Devastatoir%20Instruction%20Manual%20V2.pdf):

For visual guidance, here are some pictures that should showcase the progress of assembling it:

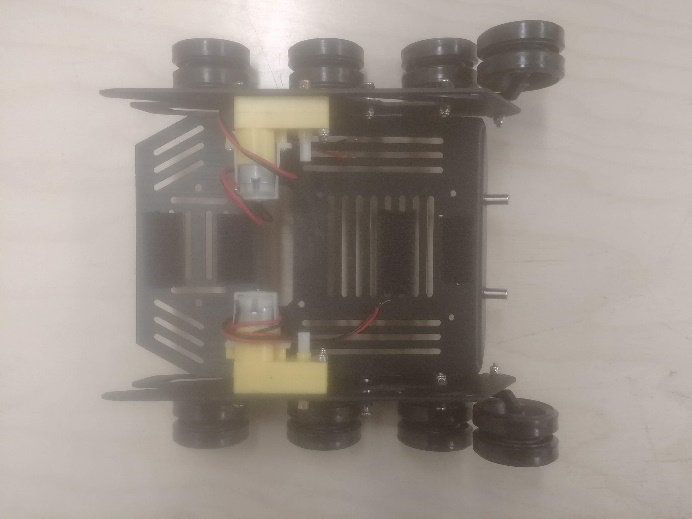
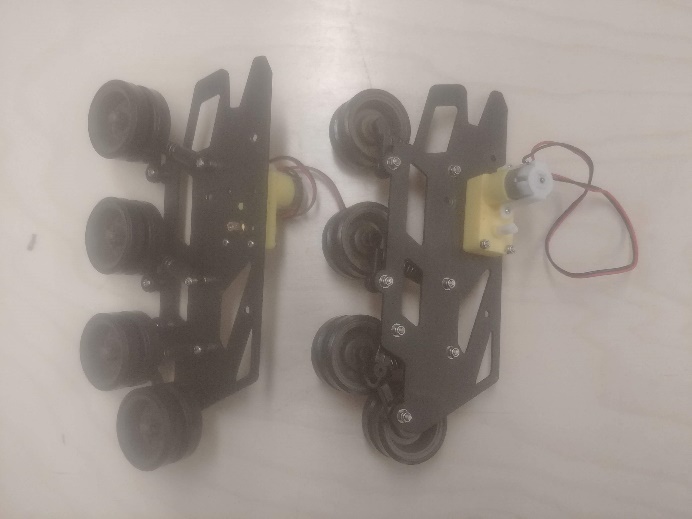


Figure 6 Tracked Wheels and Bottom Half of Chassis



Figure 7 Top Half of Chassis and Gears Added to Chassis



Figure 8 Constructed Chassis

**[Be prepared to take apart the chassis to add the components!]**

## Wheel Tracks

There’s a chance that the wheel tracks won’t fit perfectly on the wheels right out of the box. Fixing this is very simple. Get a SIM card ejection pin and follow these steps:



Figure 9 Inserting SIM Card Ejection Pin

1. Take the SIM card ejection pin and fit it into one of the small holes on the tracks. If it doesn’t want to budge, try from the other side. A metal rod should poke out of the tracks.



Figure 10 Removing Metal Rod and Excess Tracks

1. Push the metal rod all the way through. This will separate the tracks (as seen in Figure 10).
2. Repeat this process at another point on the tracks to remove the excess (as seen in Figure 10). Test the size of the tracks by wrapping it around the wheels on the chassis. Make sure it’s not too tight or not too loose (if the wheels are being squeezed, it’s too tight). Keep adjusting the tracks until it fits perfectly.
3. Re-insert the metal rod to join both ends of the tracks when the sizing is good. Dispose of the excess tracks and extra metal rod in a safe place.



Figure 11 Wheel Tracks Perfect Fit

Now, it should have no problem properly fitting around the wheels!

## Inside of the Chassis

The Devastator Tank Mobile Platform comes packaged with some Velcro strips. Use them to attach the LiPo battery and AA battery pack to the chassis. Here is the preferred placement for the inside of the chassis:



Figure 12 Inside of Chassis

*(From left to right: LiPo battery, H. Bridge (L298N), AA battery pack)*

First, insert the included toggle switch to the chassis. The included red and black wires will be soldered to the toggle switch pins so that the robot can be turned on and off with it (solder before screwing it into the chassis).

***NOTE:*** **Colour-code the wires for better organization! [Black = GND, Red = 5V, Green = 12V, etc.]**

Here’s how to solder the wires for the toggle switch:

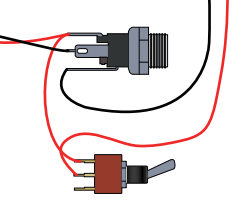


Figure 13 Soldering Toggle Switch



Figure 14 Toggle Switch Completed

When the soldering is done, add the toggle switch to the chassis. The final result should look like Figure 14.

The next set of steps is to plug in all the components inside of the chassis. The DC motors are first.

Plug in the wires from the DC motors into the H. Bridge “OUT1, OUT2, OUT3 and OUT4” terminal blocks.

Add 6 AA batteries to the AA battery pack.

Using some tweezers, remove the jumpers on the H. Bridge. There are three: two connected to the enable pins (blue) and one connected to the built-in 5V voltage regulator (green). Dispose of them for safe keeping.

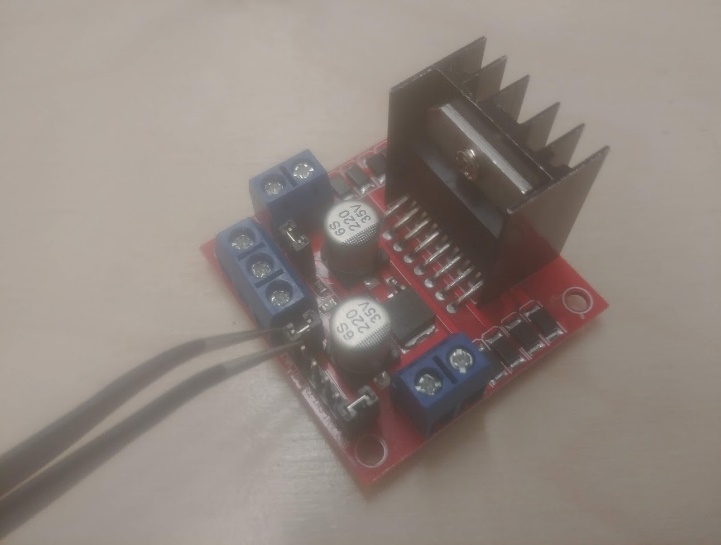
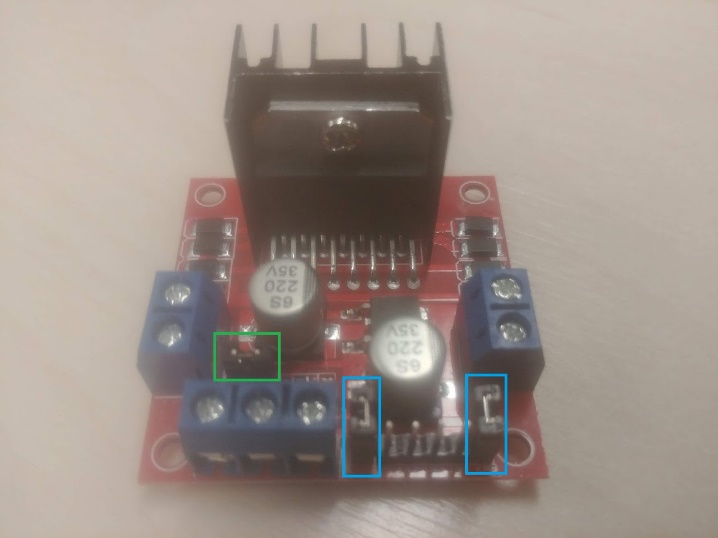


Figure 15 H. Bridge Jumpers and Removing the Jumpers

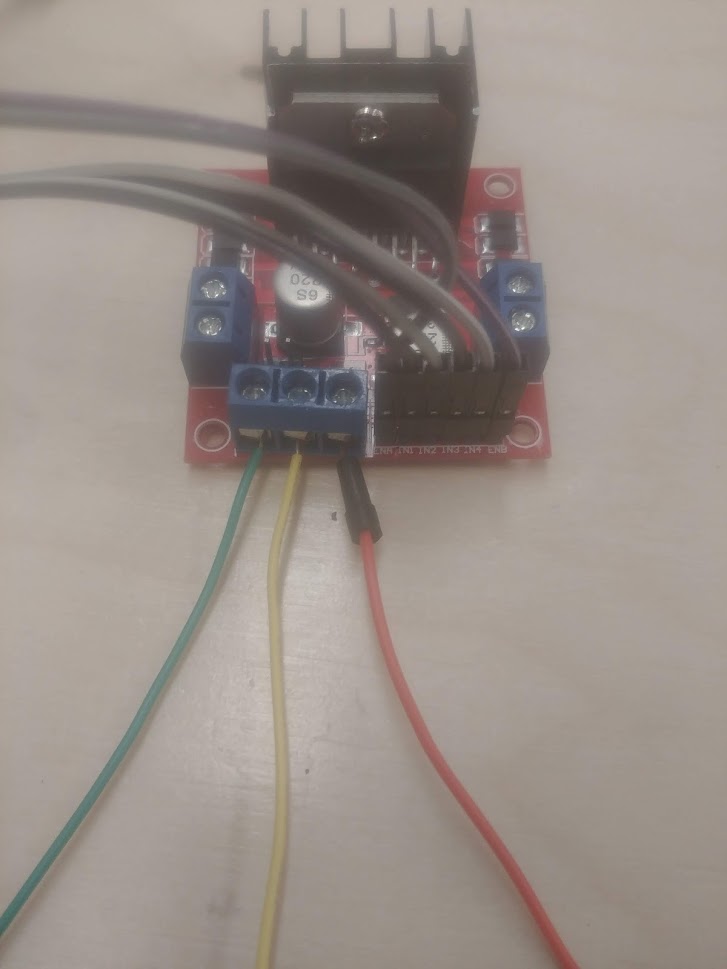
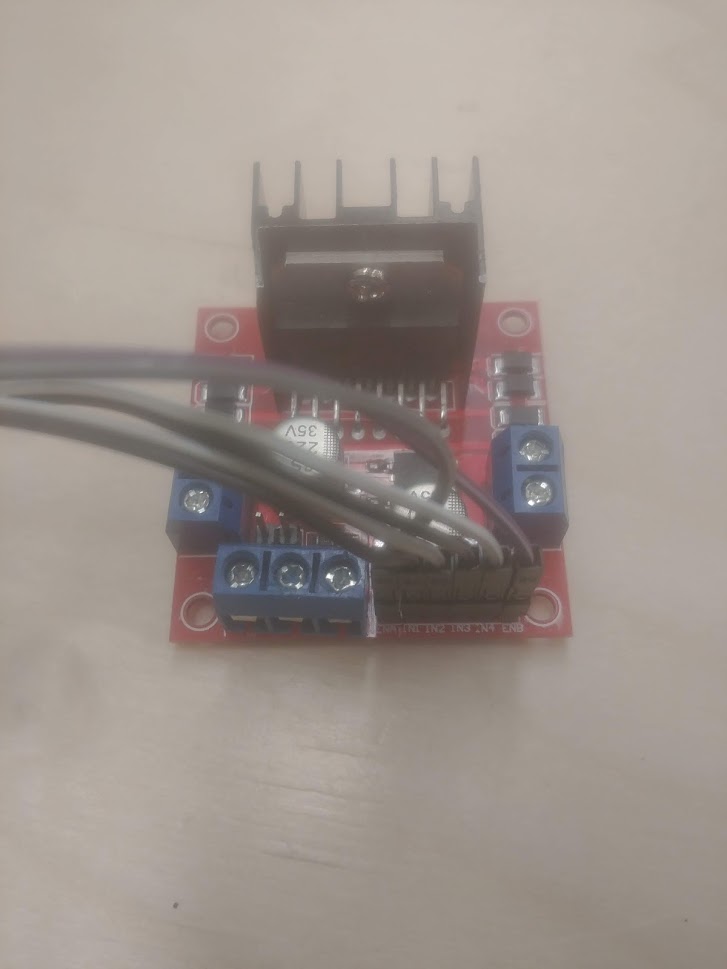


Figure 16 H. Bridge Digital Wires and Power Wires

Connect the 6 following pins with female-to-female wires: “ENA, ENB, IN1, IN2, IN3 and IN4” (as seen in Figure 16).

Next, connect male-to-male wires to the remaining terminal blocks: one wire for GND, one for 12V+, and one for 5V (as seen in Figure 16).

After, connect the AA battery pack to the toggle switch. Included in the box is a terminal block to DC barrel jack connector. Connect the toggle switch wires to the terminal block end, and the DC barrel jack end to the battery pack (watch out for polarity!)

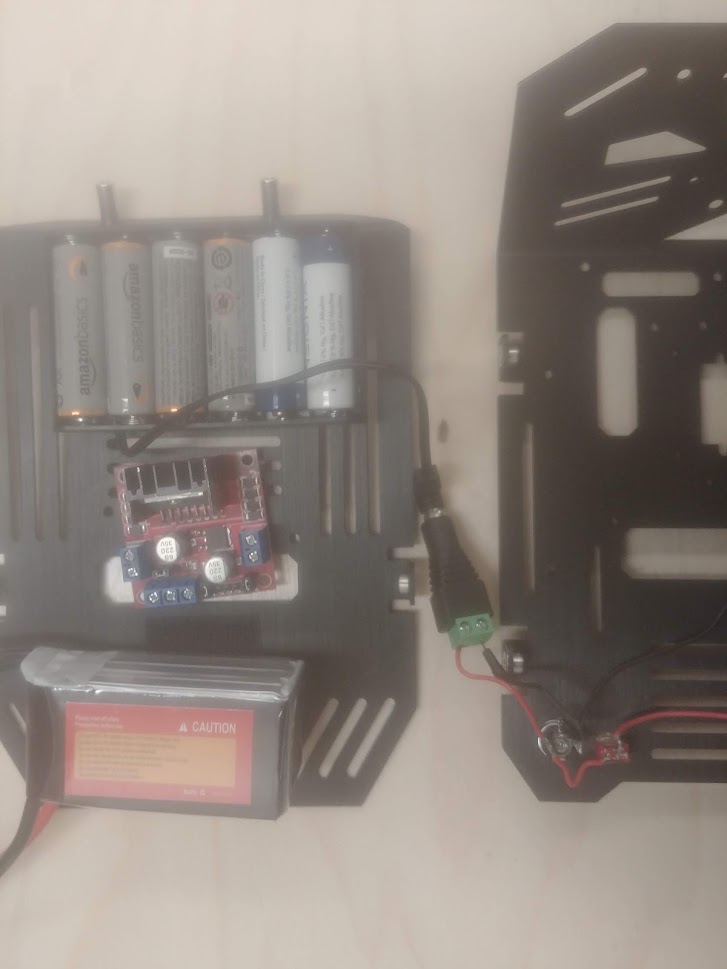


Figure 17 Toggle Switch to Battery

Next, attach the LiPo battery to the inside of the chassis.

Now that the chassis is ready, make sure to route/loop all the loose wires through the side openings. These will be connected to the PCB later.

And then, add the top layer to complete the chassis. The final result should look like Figure 18:

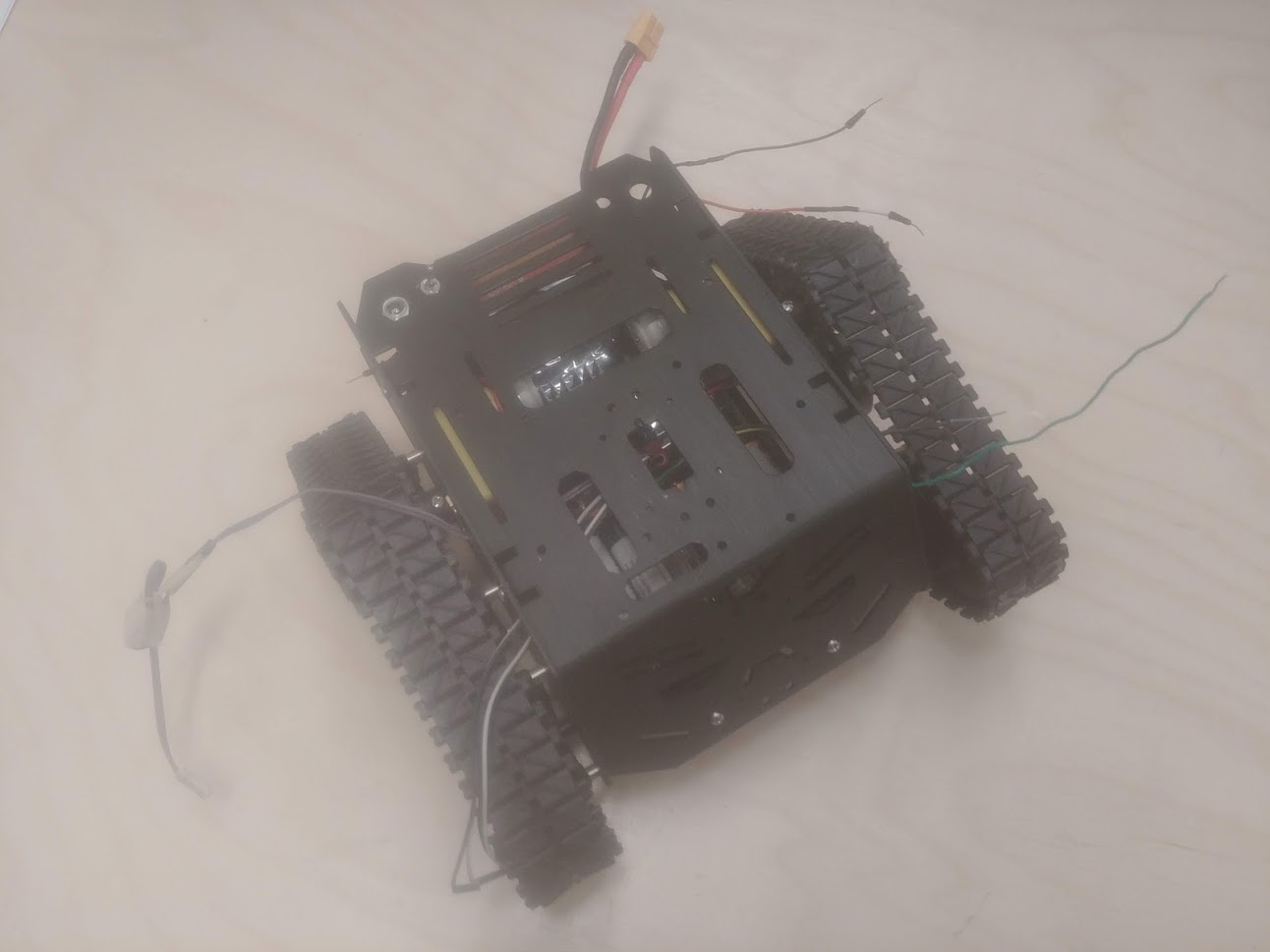


Figure 18 Completed Chassis

ASSEMBLY OF CHASSIS COMPLETE

# 3D PRINTED ENCLOSURE

## Autodesk Fusion 360

Autodesk Fusion 360 is a famous CAD software used for modeling and rendering 3D objects. This program was used to model the enclosure for the prototype. It also doubles as a water tank to hold all the water.

The enclosure is made up of two parts: the actual enclosure and the enclosure lid. The models for these two parts were printed using a 3D printer. A 3D printer is necessary to obtain the enclosure, so find one that is accessible. The models need to be printed sperately, so it will take some time before they’ll be ready, because 3D printing is a long and meticulous process. Make sure that enough time has been allocated to aacount for this or let the 3D printer run overnight. The (.stl) files for the 3D printed enclosure can be found [here](https://github.com/Chadesky/3D-Printed-Enclosure). These files are what any 3D printer needs to render the models for printing.

# WIRE CONNECTIONS

This step explains how and where all the connections are made. It’s rather simple, but because the enclosure gets in the way, the process will require patience and steady hands to make the proper connections, so that the daughterboard can fit comfortably into the enclosure after it’s done.

## How to Connect Wires

Here are the steps to connecting all the wires:

1. Hang the PCB out of the enclosure like in Figure 19 (this will help for easy access to the pins):



Figure 19 PCB Hanging Out of Enclosure

1. Start with the pins at the back: the enable pins. Choose the left most pins. Connect the wires attached to “ENA” and “ENB” on the H. Bridge to their respective pins on the PCB.

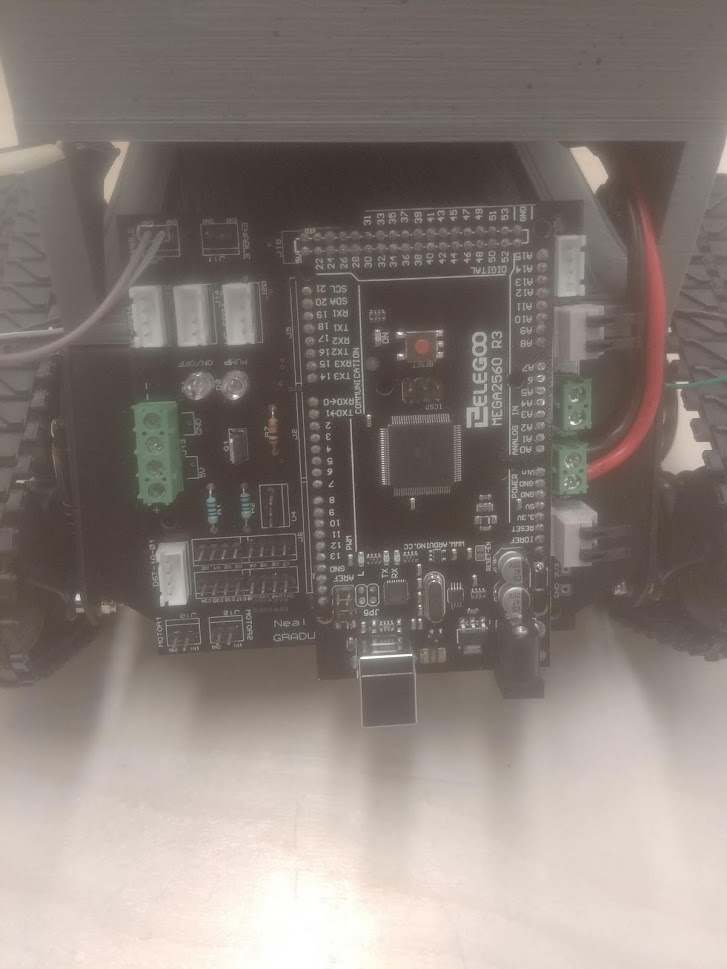


Figure 20 Connecting the ENA/ENB Pins

1. Next, connect the wires for the ultrasonic sensors to ‘HCSR04’, ‘HCSR04 2’ and ‘HCSR04 3’. There are four pins. The order of the pins is: VCC, TRIG, ECHO and GND (the VCC pin is closest to the enable pins; the GND pin is the farthest). Match that up with the sensors and then connect them.
2. Do the same for the “WATER LEVEL” header on the other side.

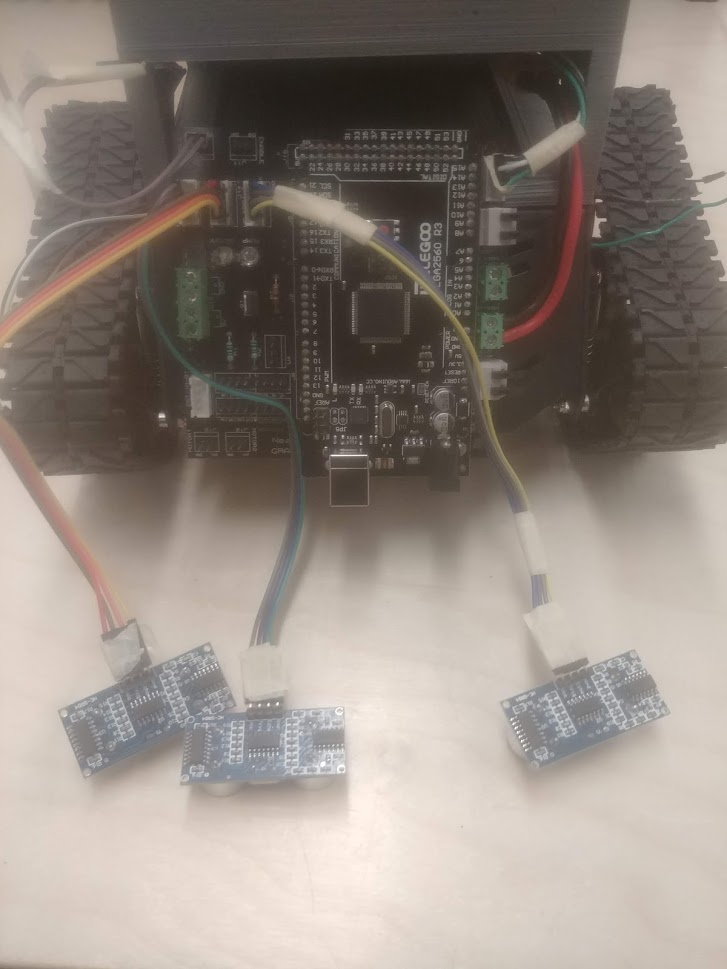


Figure 21 Connecting Ultrasonic Sensors

1. Next up, grab the extra MOSFET switch. Take the wires that connect to common GND and screw them into the GND terminal blocks. (Reminder for which wires these are can be found [here](#_Extra_Mosfet_Swicth).)
2. Connect the wire from the MOSFET switch that goes to VCC into one of the VCC terminal blocks.
3. Connect the 5V wire from the H. Bridge to the remaining VCC terminal block.



Figure 22 Connecting VCC/GND Power Wires

1. Next, connect the servo motor plug into its pin headers. The brown wire connects to GND, which is closest to the extra pin headers. It should look like Figure 23:

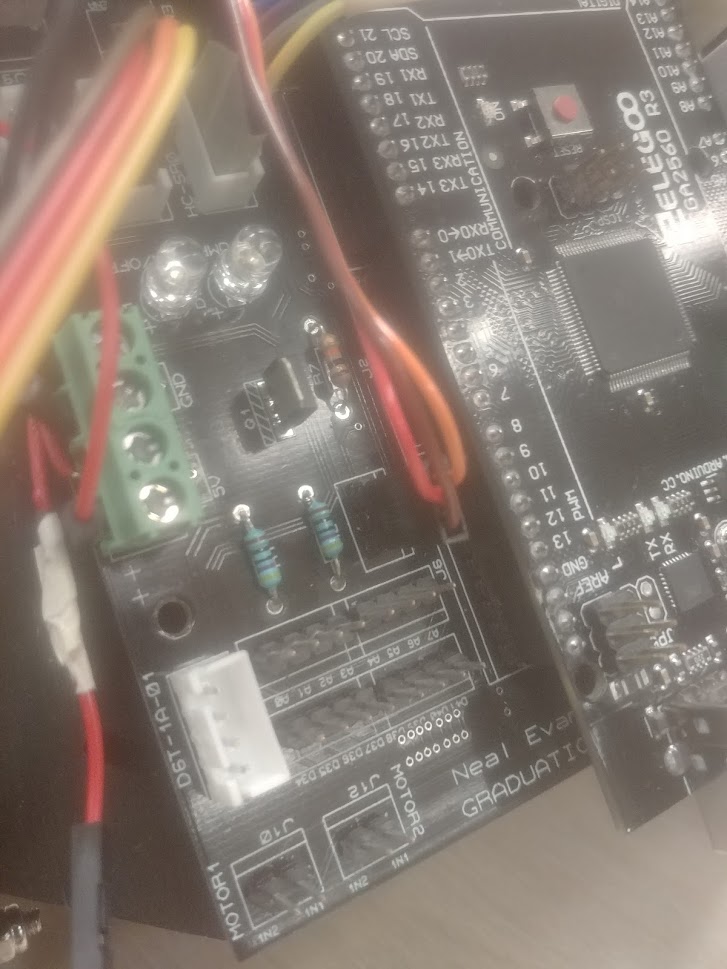


Figure 23 Connecting the Servo Motor

1. Head on over to the other side of the PCB.
2. Connect the 12V+ wire from the H. Bridge into the positive side of the “L298N” terminal block, and the GND wire from the H. Bridge into the negative side of the “L298N” terminal block.

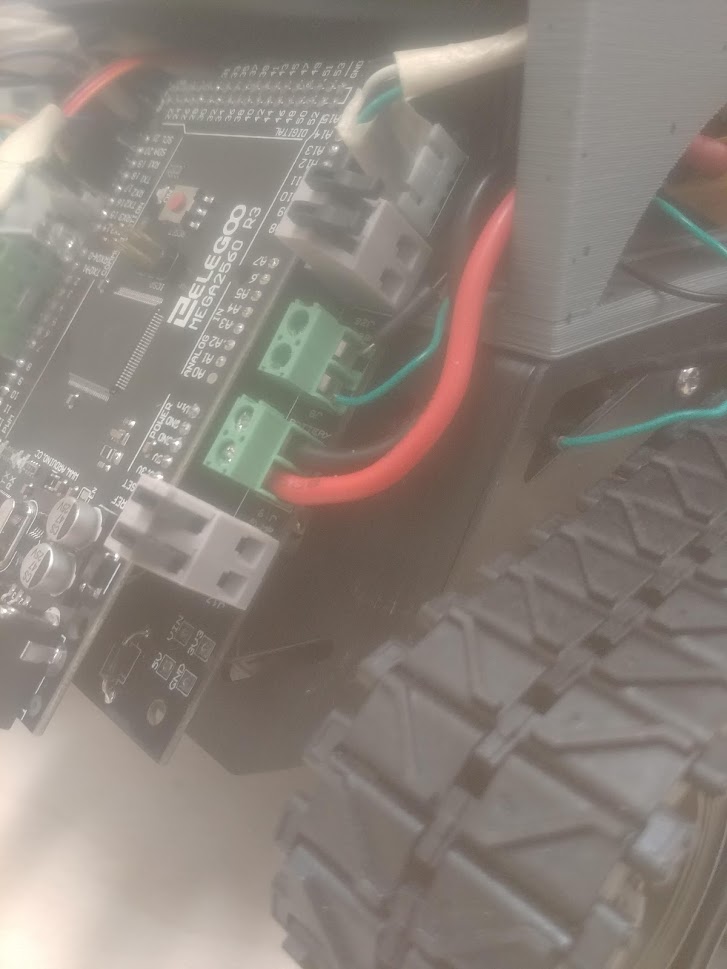


Figure 24 Connecting L298N Power Wires

1. Go back to the other side of the PCB.
2. Connect the female pin wire soldered to the ‘G’ pin on the extra MOSFET switch to the D34 pin from the extra pin headers.

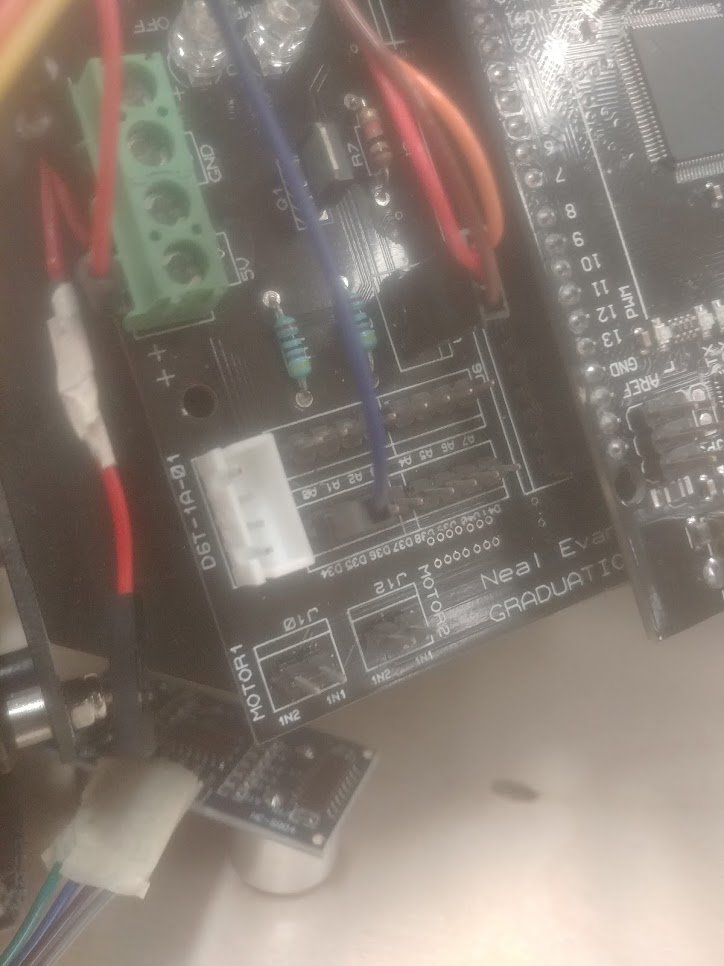


Figure 25 Connecting MOSFET Switch to Digital Pin

1. Now, push in the PCB a little bit into the enclosure like in Figure 26. Connect the thermal sensor cable into its “D6T-1A-01” header.

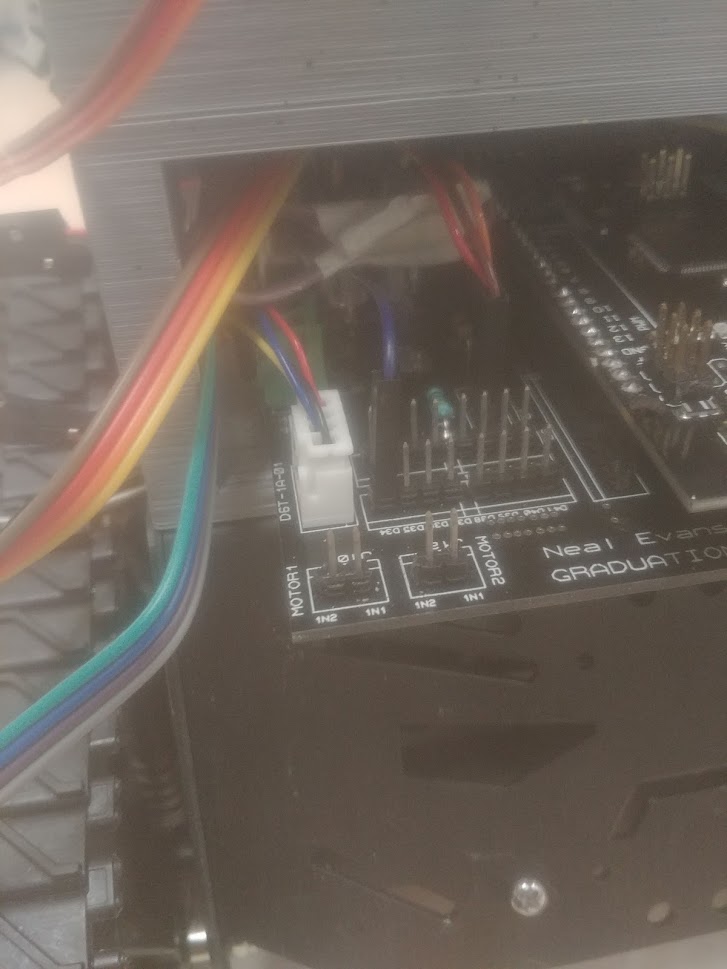


Figure 26 Connecting the Thermal Sensor

1. Next, connect the IN1 and IN2 wires from the H. Bridge into their respective pins at “MOTOR2”. Connect the IN3 wire from the H. Bridge to IN1 at “MOTOR1”. Connect the IN4 wire from the H. Bridge to IN2 at “MOTOR1”.

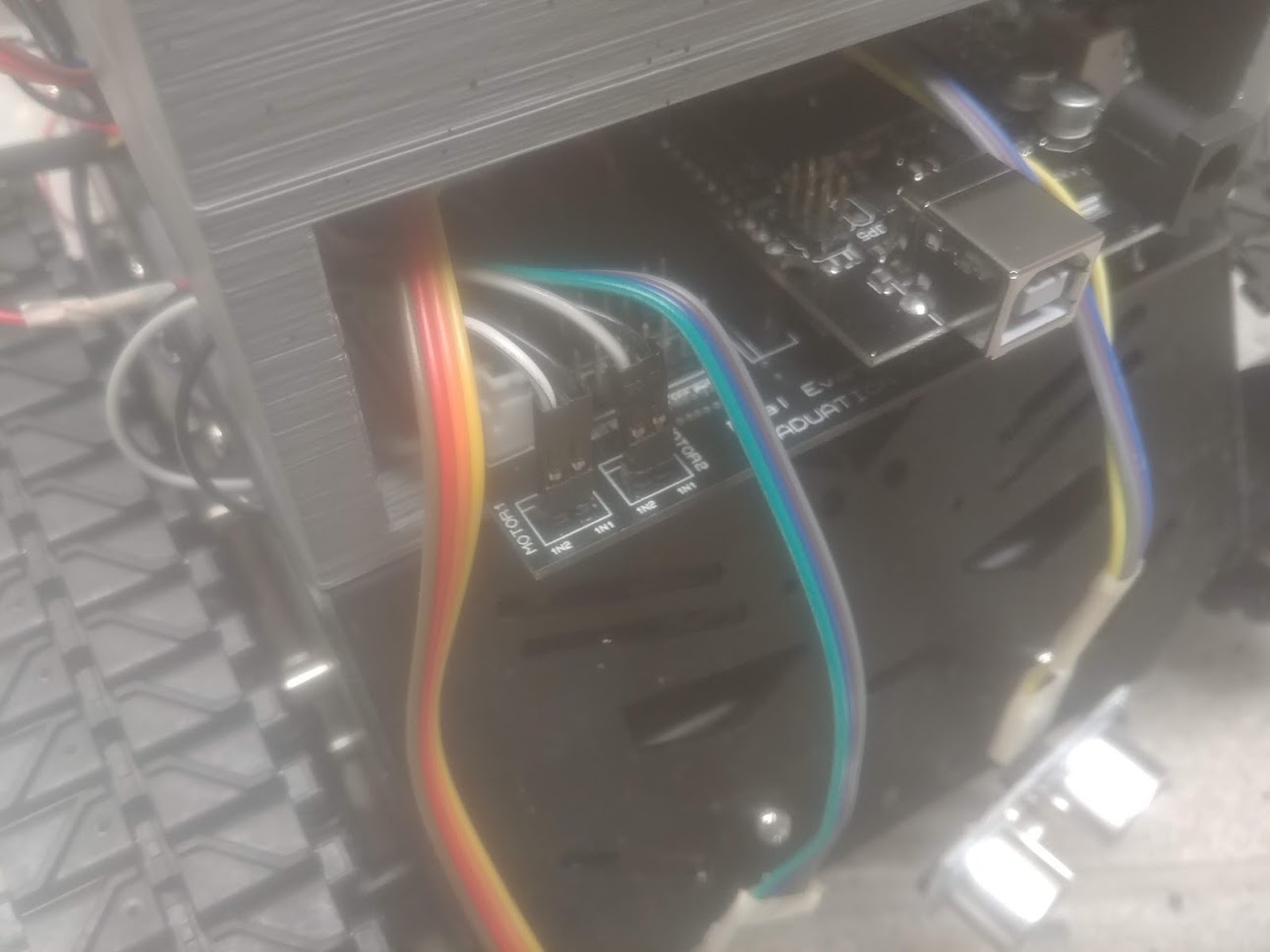


Figure 27 Connecting PWM Pins

1. Go ahead and push in the PCB all the way through. Make sure none of the wire connections are loose.
2. Next, move to the other side of the PCB again. Connect the toggle switch male pin wires to the “VIN” terminal block (pay attention to polarity!)

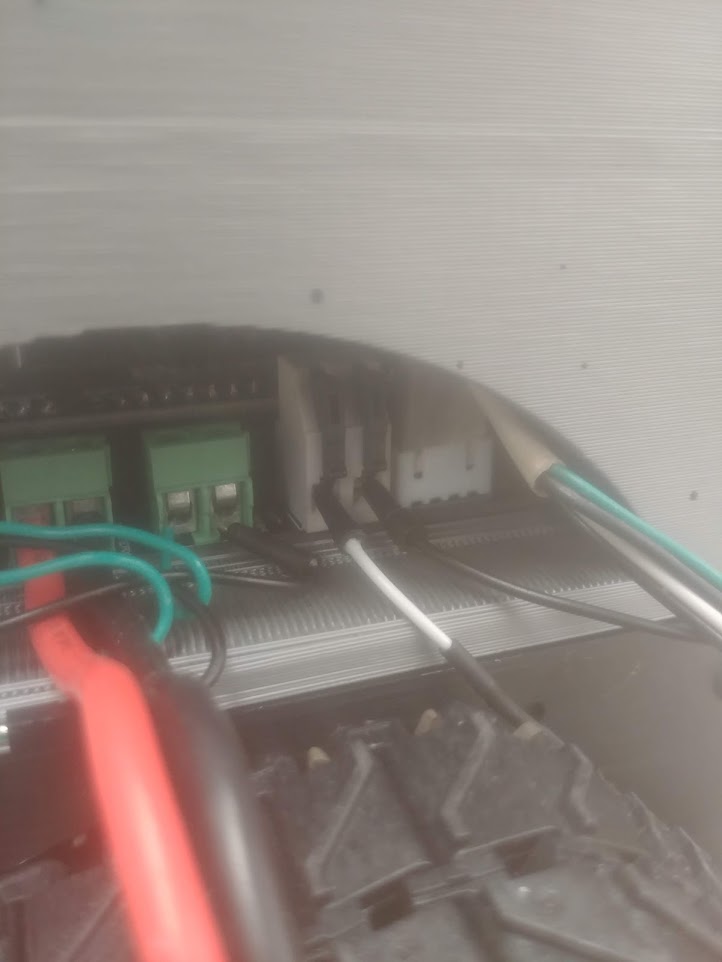


Figure 28 Connecting VIN Power Wires

1. Connect the wire that connects to the negative wire of the pump and connect the wire that connects to the positive wire of the pump.



Figure 29 Connecting Pump to MOSFET Switch

1. Look to the back of the enclosure and grab the wires at the “WATER LEVEL SENSOR” header that was connected earlier. Connect those wires with another ultrasonic sensor and remember to match them with the right pins (in the order of closest to farthest from the “VIN” terminal blocks: VCC, TRIG, ECHO, GND).

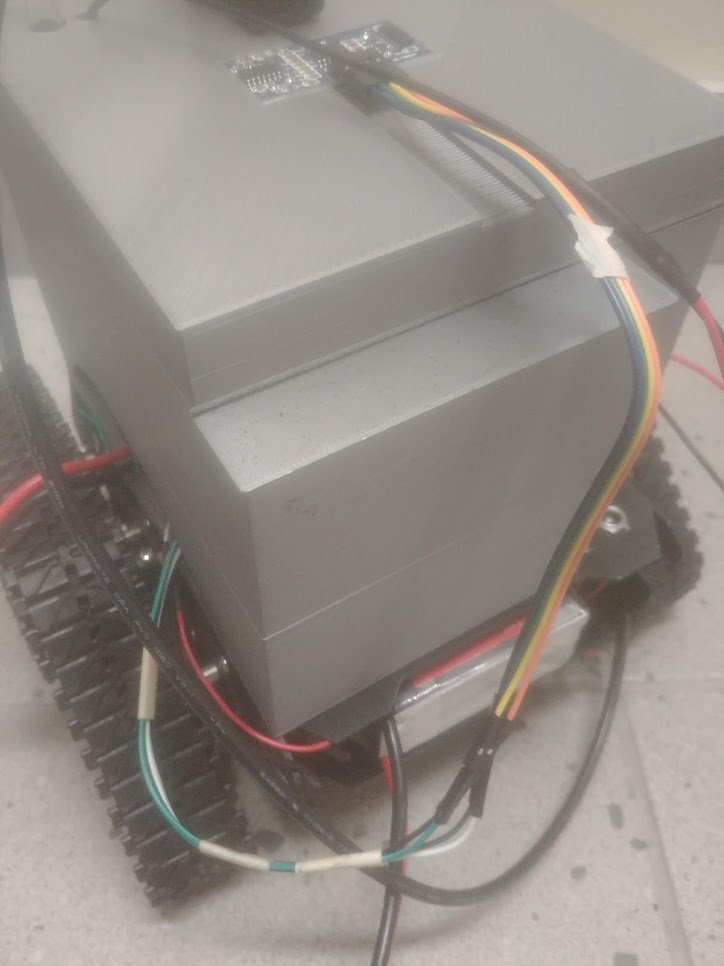


Figure 30 Connecting Water Level Sensor

1. The last connection is the XT60 cable and the LiPo battery.

**Congratulations, all the wire connections have been made!**

## Pump and Sensor Suite Placement

Now that the wire connections have been made, it is time to place the sensors, water pump and servo motor in the right orientation.

Here is how it is done:

1. Start with the water level sensor and the water pump. Push the water level sensor into its designated hole in the enclosure lid. Keep the water pump upright on the underside of the lid and route its wires and tube through the square and circle holes in the lid.

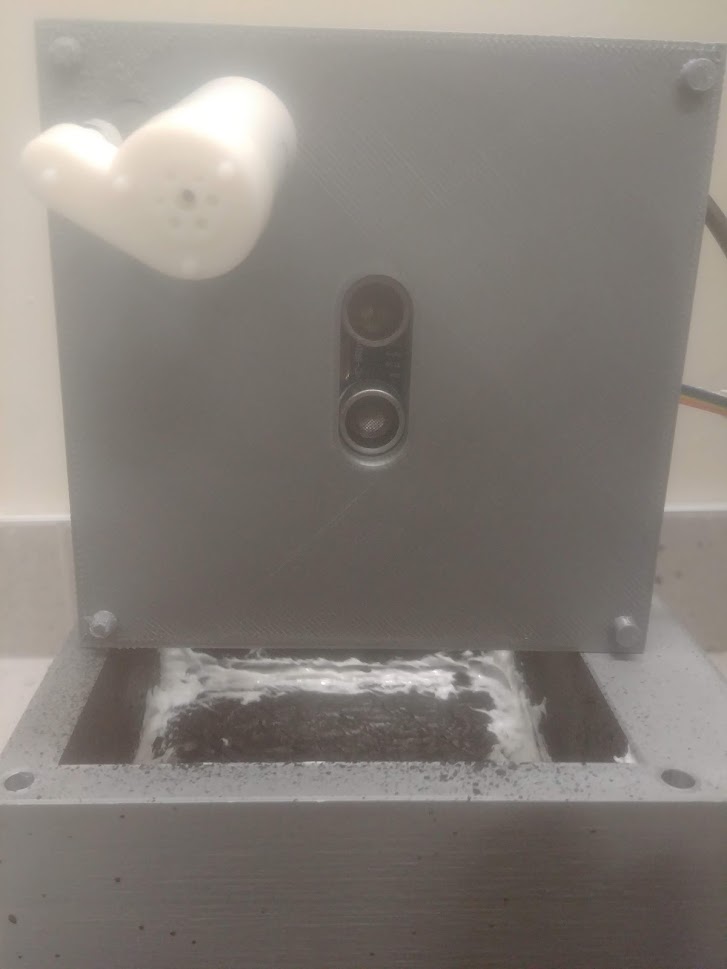
 

Figure 31 Water Pump and Water Level Sensor

1. Next, place the lid down on the enclosure. Grab the servo motor and place it on top of the lid. Point it towards the front of the robot (as seen in Figure 32). The wire should be sticking out forward. Once properly positioned, hot glue the servo motor to the lid to secure it.
2. After, take the thermal sensor cable and tube and join them together using masking tape. The thermal sensor and tube should be pointing in the same direction (as seen in Figure 32).
3. Next, use zip ties to attach the thermal sensor cable and tube to the servo motor (as seen in Figure 32).  
   **NOTE:** Have the tube and thermal sensor extend further out from the front of the robot. This will prevent any water from leaking or dripping onto the sensors.

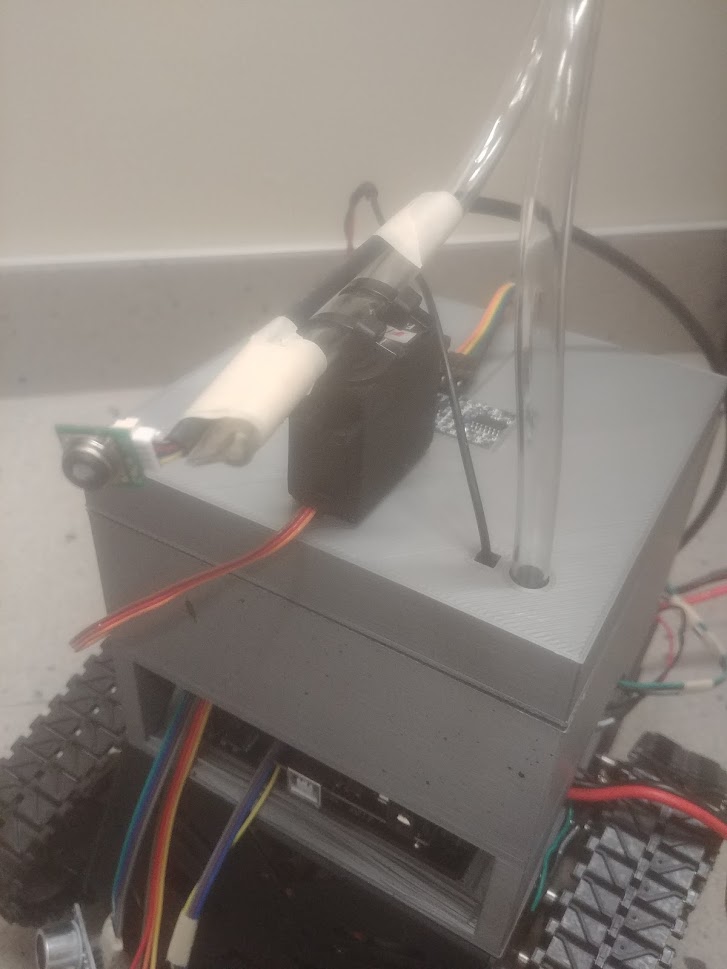


Figure 32 Servo Motor, Thermal Sensor and Tube

1. Lastly, adjust the orientation of the three ultrasonic sensors at the front of the chassis. It is best to use masking tape because it makes it easier to make changes, if necessary. The recommended orientation is: one sensor above the PCB and the remaining two on the lower front of the chassis (as seen in Figure 33).

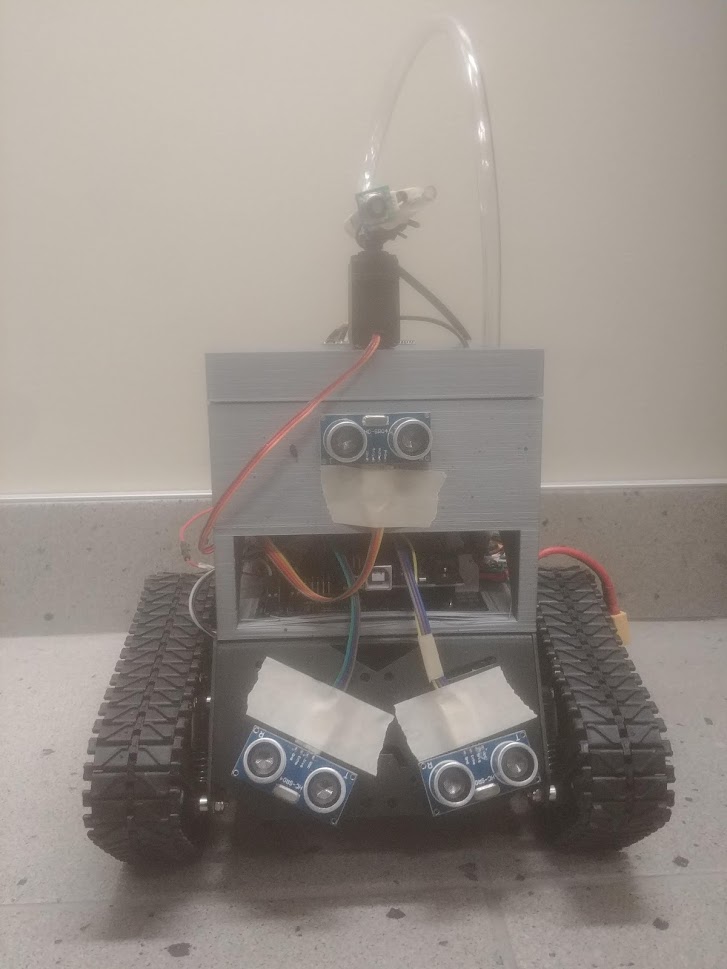


Figure 33 Ultrasonic Sensors Orientation

WIRE CONNECTIONS AND COMPONENT PLACEMENT COMPLETE

# SOFTWARE AND FIRMWARE

## Firmware

The microcontroller should be pre-installed with the proper firmware to be immediately recognized by a computer when plugged in. But, if not, the link for the firmware can be found [here](https://www.arduino.cc/en/Hacking/DFUProgramming8U2).

## Software



The software used to code the microcontroller was Arduino IDE. (The download link can be found [here](https://www.arduino.cc/en/Main/Software).) The programming language used was C++. Arduino IDE contains some pre-installed libraries that makes life easier, such as the servo motor library. But a third-party library is needed to run the thermal sensor. The code for the robot in its entirety is already provided through this link [here](https://github.com/Chadesky/Robot-Code).

Simply, open the Arduino file inside of the “pwm\_no\_en\_jumper” folder and plug in the microcontroller. Click on “Tools” to change “Board” to “Arduino/Genuino Mega or Mega 2560” and “Port” to the proper COM port. The proper COM port can be found in Windows Device Manager. Finally, click on “Upload” on the main window (top left). This will program the microcontroller with the code.

While plugged into the computer, it is possible to view what the robot is doing. Click the top right icon on the main window. This will pull up the serial monitor. While the robot is plugged in, the serial monitor will print out certain statements depending on what the robot is doing.

## GUI

The robot doesn’t need human interaction to work properly. However, interaction with the robot is needed when it comes to performing a diagnosis on it. The GUI of the prototype displays the information from the robot, such as the amount of water left in its tank, the amount of power left in both batteries, etc. It will warn the user when the tank is low on water or if the batteries need to be replaced/recharged.

Connecting it to the robot is similar to using the serial monitor. Plug in the microcontroller to a computer and run the GUI application. Click on “Change COM Port” to select the proper COM port and click on “Connect”. Once connected, this will make any of the remaining buttons in the GUI clickable to update the values. It is also possible to choose to make the robot non-operational with one of the buttons.

The software for the GUI can be found [here](https://github.com/Chadesky/Robot-GUI). It only works when opened in a software called “Qt”. The link to download it can be found [here](https://www.qt.io/download).



Figure 34 Arduino IDE

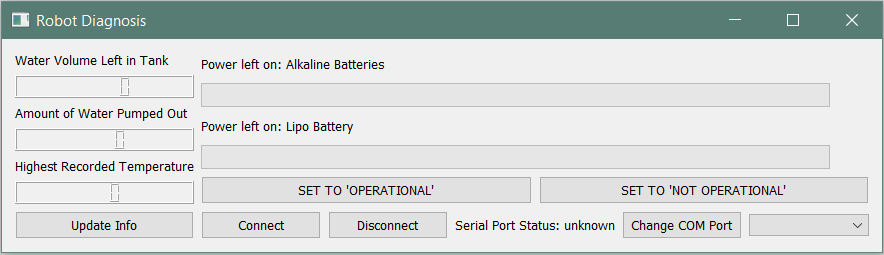


Figure 35 Robot GUI

# USER GUIDE/TESTING

## Turning it On

Now that the robot has been constructed and the software has been uploaded to the microcontroller, it’s time to try it out!

All that’s left is to make sure that the LiPo battery is connected to the XT60 cable on your PCB, and then flip the toggle switch. An LED on the PCB should light up indicating that the robot has been turned on. After a couple of seconds, the wheels will begin turning and the robot will start moving forward. And, that’s it! The robot should take care of the rest.

## Test Cases

#### Testing the PCB

1. Apply power to the PCB by plugging in the microcontroller to a computer with it connected to the daughterboard.
2. Use a multimeter to measure the voltage coming from the test points.
3. Measure the voltage at the “5V” testing pad and confirm that it is 5V.
4. Measure the voltage at the “GND” testing pad and confirm that there is no voltage applied through it.
5. Measure the voltage at the “3v3” testing pad and confirm that it is 3.3V.
6. Plug in the wires from the AA battery pack into the “VIN” terminal block. Measure that the voltage at the “VIN” testing pad and confirm that it is between 5-9V.

#### Testing the Robot

1. Flip the toggle switch to turn it on. The robot should start moving.
2. Look to see if the robot is moving forward and straight.
3. While the robot is moving, place an object in front of it. The robot should see that it’s getting close to something, and it will stop.
4. The servo motor should then start moving. It will stop and check in several positions to look for a fire. Every time that the servo motor stops at one of these positions, the thermal sensor will check for a fire. Use a lighter or candle and hold a flame near the thermal sensor. It should activate the water pump and shoot water out of the tube.

#### Testing the GUI

1. Plug in the microcontroller to a computer and open the GUI.
2. Click on “Change COM Port” to make a list of available COM ports appear. Select the proper COM port.
3. Click on “Connect” and look at “Serial Port Status”. If it updates and the baud rate is 9600, then the connection was successful.
4. Click on “Disconnect” to see if “Serial Port Status” goes back to “unknown”.
5. Click on “Connect” again. Check to see that the progress bar for “Power Left on: Alkaline Batteries” and “Power Left on: Lipo Battery” have been updated.
6. Click on “Update Info” and check if the values for “Water Volume Left in Tank”, “Amount of Water Pumped Out” and “Highest Recorded Temperature” have been updated.
7. Click on “SET TO ‘NOT OPERATIONAL’” and check to see if the robot isn’t running.
8. Click on “SET TO ‘OPERATIONAL’” and check to see if the robot is running again.

## Troubleshoot Cases

To make sure that the robot is operating properly, follow these troubleshooting cases:

#### Troubleshoot Case #1

If the robot isn’t moving, there might not be any water inside of the water tank. Turn off the robot and add some water to fix the issue.

#### Troubleshoot Case #2

If the robot is moving backwards instead of forward, check that the “IN1, IN2, IN3 and IN4” wires are connected to the right pins (refer to Figure 27).

#### Troubleshoot Case #3

If the robot isn’t stopping, turn off your robot, unplug the LiPo from the XT60 cable and plug in the microcontroller to a computer. Use the serial monitor on Arduino IDE to view the printed distance measurements. If the readings are incorrect, check that the ultrasonic sensors are plugged in all the way.

#### Troubleshoot Case #4

If no water is being pumped out, listen for a noise coming from the pump. If it doesn’t make a noise, make sure that the MOSFET switch connections are securely fitted on their respective pins and terminal blocks (refer to [this](#_Extra_Mosfet_Swicth)). Also, make sure that the pump’s negative and positive wires are properly connected to the MOSFET switch. If it is making a noise, open the lid and make sure that pump is oriented in the correct way. The tube should be directly above it and the intake of the pump should face the bottom of the water tank (refer to Figure 31).

## Mis-use Cases

1. The polarity of the components is important to ensure that the robot functions properly. There are no safety measures to mitigate the risk flipping the polarity of the power sources. If that were to happen, the power sources can be damaged of even combust.
2. The GUI will connect to any COM port that is listed in its COM port list. There are no mitigations keeping it from connecting to COM ports that aren’t associated with the robot. Using any of the buttons when connected to the incorrect COM port might have unintended effects on the computer.
3. There is no safety measure to keep the microcontroller from being shorted or overloaded with too much voltage or current. If a high voltage is source to either of the primary power sources, this can cause permanent damage to all the components of the robot, including the power sources themselves.

# FUTURE CONSIDERATIONS

In the future, the robot could be improved. Currently, there are things that could be better. For example, the extra MOSFET switch shouldn’t be separate from the daughterboard. A future upgrade will include it into the PCB. Also, the robot could benefit from a reset button on the PCB as well. Other than that, the mis-use cases would be addressed.

In terms of making the robot more robust, a new enclosure can be designed. The new one could hold a higher quantity of water. Also, it could be made of a better material that could withstand much higher temperatures. This would keep it from melting when around very hot fires. Additionally, the servo motor could be replaced with a better component to improve the robot’s overall accuracy. A component that could move up and down, as well as left and right, will expand the mobility and range of the thermal sensor and tube.

# APPENDIX

## Definitions

PCB (Printed Circuit Board): a board, or plate, with conductive copper paths imbedded into it; used to link electronic/electrical components together

PWM (Pulse Width Modulation): changes the value of the DC voltage based on the duty cycle

Microcontroller: a circuit board and the main component of the prototype; similar to a computer and executes all the functions of the robot

Daughterboard: an extra circuit board that extends the functionality of the primary circuit board by connecting to it

Sensor Suite: a collection of components that are used for making certain measurements; comprised of the ultrasonic sensors and the thermal sensor

## Pertinent Documents/Links

1. *DFRobot Devastator Tank Mobile Platform Instruction Manual*  
   <https://github.com/Arduinolibrary/DFRobot_Devastator_Tank_Mobile_Platform/raw/master/Devastatoir%20Instruction%20Manual%20V2.pdf>
2. *Arduino MEGA Documentation (PINOUT, SCHEMATICS, ETC…)*  
   <https://store.arduino.cc/usa/mega-2560-r3>
3. *PCB Design GERBER Files*<https://github.com/Chadesky/GERBER-Files>
4. *PCBWay Website Link*<https://www.pcbway.com/>
5. *MOSFET Datasheet*<https://www.vishay.com/docs/91322/sihlr024.pdf>
6. *D6T-1A-01 (Thermal Sensor) Datasheet*<https://omronfs.omron.com/en_US/ecb/products/pdf/en-d6t.pdf>
7. *HC-SR04 (Ultrasonic Sensor) Datasheet*<https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf>
8. *Servo Motor Datasheet*<http://www.ee.ic.ac.uk/pcheung/teaching/DE1_EE/stores/sg90_datasheet.pdf>
9. *iFixit Essential Electronics Toolkit Amazon link*<https://www.amazon.ca/iFixit-IF145-348-2-Essential-Electronics-Toolkit/dp/B01MRNIFR6/ref=pd_lpo_vtph_469_tr_t_2?_encoding=UTF8&psc=1&refRID=2YWD975PX694R5YA183G>
10. *Arduino MEGA Firmware*<https://www.arduino.cc/en/Hacking/DFUProgramming8U2>
11. *Arduino IDE Download Link*<https://www.arduino.cc/en/Main/Software>
12. *Robot Code*  
    <https://github.com/Chadesky/Robot-Code>
13. *Robot GUI Application*  
    <https://github.com/Chadesky/Robot-GUI>
14. *3D Printed Enclosure Files*  
    <https://github.com/Chadesky/3D-Printed-Enclosure>
15. *Water Pump Datasheet*<https://www.robotshop.com/media/files/content/e/elf/pdf/3v_vertical_water_pump_-_elecfreaks_learn.pdf>
16. *Soldering Tutorial*<https://www.youtube.com/watch?v=Qps9woUGkvI>
17. *Autodesk Fusion 360 Tutorial – Part 1*<https://www.youtube.com/watch?v=A5bc9c3S12g>
18. *Autodesk Fusion 360 Tutorial – Part 2*<https://www.youtube.com/watch?v=HXRMzJWo0-Q&t=21s>
19. *Autodesk Fusion 360 Tutorial – Part 3*<https://www.youtube.com/watch?v=zS8dYA_Iluc>
20. *Qt Download Link*<https://www.qt.io/download>