# EEL 4915 Senior Design II

Final Documentation

Fire Extinguishing Robot

University of Central Florida
Department of Electrical and Computer Engineering

Dr. Lei Wei
Group C

Gabriel Martinho Bizuti, Electrical Engineering
Sydney Brown, Electrical Engineering
Juan Cuervo, Computer Engineering
Noah Schrock, Computer Engineering

December 6, 2022

# Contents

# List of Figures

# List of Tables

# 1. Executive Summary

Robots and autonomous devices are making their way into the life of everyone more and more as time passes. They are part of an attempt to ease the difficulties of the daily life of most people. There are areas where robots and autonomous devices are being integrated into society. An example is kitchen assistant robots. In Japan, one of the leading countries in robot technology, they use robots as assistants in the kitchen to make sushi and chop vegetables. They are also used in earlier steps of food production such as planting rice and growing crops. Another example would be the use of vacuum cleaner robots that are programmed to go around the house keeping it clean on their own. Our project involves building a fire extinguisher robot that would help with the security and protection of people, since all people, regardless of whether they work in a kitchen or not, are exposed to the dangers of fires. Among the three major causes of house fires, two of them can happen due to heating equipment problems and electrical malfunction. These problems can happen when no one is paying attention and that might lead to several damages, injuries, and even the loss of lives. In those cases, firefighters are the ones responsible for putting out the fire as soon as possible to save as many lives as they can and to minimize the damage caused. Because of this, every second is important when dealing with home fires.

For example, if a possible malfunction happens and a fire starts, it will take some steps for the firefighters to be notified of what is happening. First, the fire will have to produce enough smoke to activate the sprinklers in a building or a fire smoke detector in a home which will then notify the authorities. In the case that either of these fail, the responsibility to notify the authorities about the situation would shift to the people near the incident. This delay is enough to have major damage caused by the fire and maybe even the loss of the lives of those who were unable to escape the fire, should the fire get big enough to pose a life-threatening danger. Our project is a robot that is reliable enough to put out a small or medium sized flame on its own with little to no user interaction. Later, the project could be the basis for a more ambitious product that could be placed in every home and help to keep the fire to a minimum until the firefighters arrive at the place. Since almost 50% of the causes of home fires are not related to the action of people and can happen at any time, the presence of a robot that helps to mitigate the spread of fire would be very welcome.

Some fire departments have their own fire extinguisher robot. Those robots usually are designed to work on larger fires where the fire has already spread to multiple places. Our motivation for working on this project was to develop a product that will help put out small fires on its own. We also want to make it simple to use and to set up. Our goals and objectives with this project are to develop an accurate, easy to use, portable and simple robot that would track down small to medium flames and try to put them out.

The robot that we proposed to design has multiple sensors that would be responsible for constantly being on the lookout for possible flames. The sensors are connected to an ADC that would translate the information to the digital domain for the MCU to use it. We also wanted a way of notifying people in the immediate area in case a flame is spotted so that they can take necessary action. It also has a container that is used to put out the flames when detected.

# 2. Project Description

We are proposing what we created for this project under specific conditions and how we conceptually developed it with a large array of details considered. Before diving into the specifics of the project – which will include the detailed parts chosen as well as thought out reasonings as to why those parts were chosen over the other competitors – this chapter will first address a large number of sections that will shed greater light as to what the overarching goals are for this project.

The subheadings contained are as follows:

1. Motivations
2. Goals and Objectives
3. Functionality
4. Requirement Specifications

First and foremost, this chapter will address the underlying motivation behind why this project was considered and eventually chosen by all group members. In that subheading, the many reasons behind choosing this project will be considered and explored.

Exploring the motivations behind taking on a certain project is a wonderful first step in any development because it becomes the focusing agent throughout the entire designing and building process. As will be discussed, the primary motivation was the safety of the users, and this idea is able to drive the project forward with that specific goal in mind beyond any other secondary or tertiary motivations.

Following this, the chapter will contain a subheading that addresses the multiple goals and objectives of the project. Beyond the motivation, outlining the goals and objectives in mind can help to zero in on the specifics that are primarily required in order to meet the demand that was initially envisioned.

In this subheading, we discuss not only the initial goals and objectives that were proposed and intended by the group, but also the potential goals and objectives that are considered possibilities, given an abundance of resources. These types of goals will be referred to as “stretch goals” that may or not be met, depending on how far the project progresses and how many resources are available.

Next, the functionality section will address what the robot is able to do at a very high level. The point of this section is to simply outline the very basic idea of what the group intends the robot to be able to accomplish.

Finally, the subheading of Requirement Specifications contains within itself three sub-subheadings, and each of these subheadings will contain at least one graph or chart that adds specific information on the given heading.
The first subheading will outline the different design specifications of the robot. These specifications will lead the rest of the project going forward in determining details such as size, weight, time requirements, power supplies, etc.

The second subheading will contain a House of Quality matrix. The House of Quality is a product planning matrix that is used in order to communicate to the customer how their requirements are directly related to the methods that the group will employ in order to meet those requirements.

Lastly, the final section is one that contains multiple Block Diagrams. The first of the four diagrams is a general hardware diagram that outlines each group members' responsibilities when it comes to the hardware design. The second diagram describes the embedded system and defines how each group member took charge of different aspects of the project, from sensor designing to micro controller programming and interface design. The third diagram is a general software diagram that describes the basic algorithm that the robot runs under. The last diagram is a power design that describes how the system is powered and how the power is distributed.

These are the sections of this second chapter and their main purpose is to begin focusing the project to something more precise and achievable. The charts provide useful visuals that describe the written information above them. They help synthesize the goals, specifications, and requirements while maintaining the objectives, goals, and underlying motivation as the guiding principle of the overall project.

## 2.1 Motivation

The motivation for creating a fire extinguishing robot came from our desire to innovate a device that could help keep humans safe without the risk of putting others in harm's way. House fires as well as fires occurring outside have been an issue for many years. The creation of a fire extinguishing robot could become particularly useful and beneficial to the safety of others. Having a device in the house that could detect a flame and put it out within seconds could result in a much less devastating incident.

Fires can escalate very quickly and a fire extinguishing robot could be a first line of protection for many homes. This robot could begin as a simple deterrent from fires developing into bigger fires. Eventually, this robot could be upgraded into one that can take the place of a firefighter from them having to put their lives at risk and entering houses on fire. Instead, they could send a robot that would be able to withstand the heat and fumes inside to put out the fire. In building this type of robot, the hope is that it could spark conversations and potential innovations for more powerful and durable fire extinguishing robots in the future.

## 2.2 Goals and Objectives

The goal of this project was to design a functional robot that could detect a flame using an IR Flame Sensor and maneuver its way to put the fire out. Our primary focus was to make sure the robot we designed was accurate and precise with every detection of a flame. Another objective was to make sure the robot was well designed in order for it to be able to move around and to get to specific places without having any issues or damage caused to the robot. We wanted the design to be simple but effective.

Our stretch goal is to create another model that switches out the water-based fire suppression system with one that uses fire extinguishers. There are tradeoffs to this system of course. While the price of fire extinguishers is much more than the price of water the firefighting capabilities are exponentially better. By switching to an extinguisher-based system the robot is able to fight a greater range of fires, since water is not only ineffective but can also be dangerous when used on the wrong type of fire. Of the five classes of fires, water is only effective in combatting one of them, class A fires. Class A fires are the easiest fires to fight, and can easily be put out using a fire extinguisher as well. By switching out the water-based system for a fire extinguisher based one the robot can get around this problem by using the right type of extinguisher to combat the fire class that it intends to fight. The size and weight of these extinguishers varies based on the class type; this is the main hurdle of this goal.

## 2.3 Functionality

Our fire extinguishing robot is able to move towards a fire when a flame is detected and put it out. This was be done by using an IR flame sensor to detect any amount of infra-red light emitted by the fire. The robot then maneuvers its way to the flame and pumps out water to put the fire out. This is all done without human intervention. The robot is driven by a set of two DC motors which are driven by a motor driver, backed by a portable battery, and controlled through a microcontroller unit. The three sensors, the driver, the pump and the servo motor are all controlled by the pre-programmed microcontroller unit.

## 2.4 Requirement Specifications

This section of requirement specifications will cover three basic subsections that will each deal with the specifics of the project and break it down further into more understandable sections. The three subsections will tackle the specifications of the project in three different ways. The first one will have a table that will outline the group’s selected specifications and it will provide descriptions of each one. The second subsection will be a house of quality figure that will provide an associative visual on how the requirements will be met through the specifications. Lastly, the third subsection will have a set of diagrams that each describe how the hardware, software, and overall system will be engineered and how each individual in the group will take care of specific sections.

### 2.4.1 Specifications

This section will include a table that provides the robot design specifications. It will provide more information and focus in on the specifics that have been chosen for the robot's overall design. The table will provide a list of nine different specifications chosen in order to begin closing in on the final idea for the project, going from a more abstract and high level idea to a more specific, concrete, and low level design. As the documentation continues, the design will understandably become more and more specific and thought out.

With that said, the nine different specifications chosen will range from more generic choices such as the overall size of the robot, to more specific selections such as the speed of the robot as well as the extinguishing and response times. Although some of these specifications may seem relatively unnecessary to determine at this point in the documentation, they will provide a stable framework upon which the team will build on and alter, given that the current specifications are not met.

It is to be noted that these specifications are the ideal construction under the high level paradigm the team has started with as a foundation. As such, many of these specifications will thus be generalized to ranges as opposed to strict values, in order to give the team more flexibility in deciding what the best specific values will be for the final project overall.

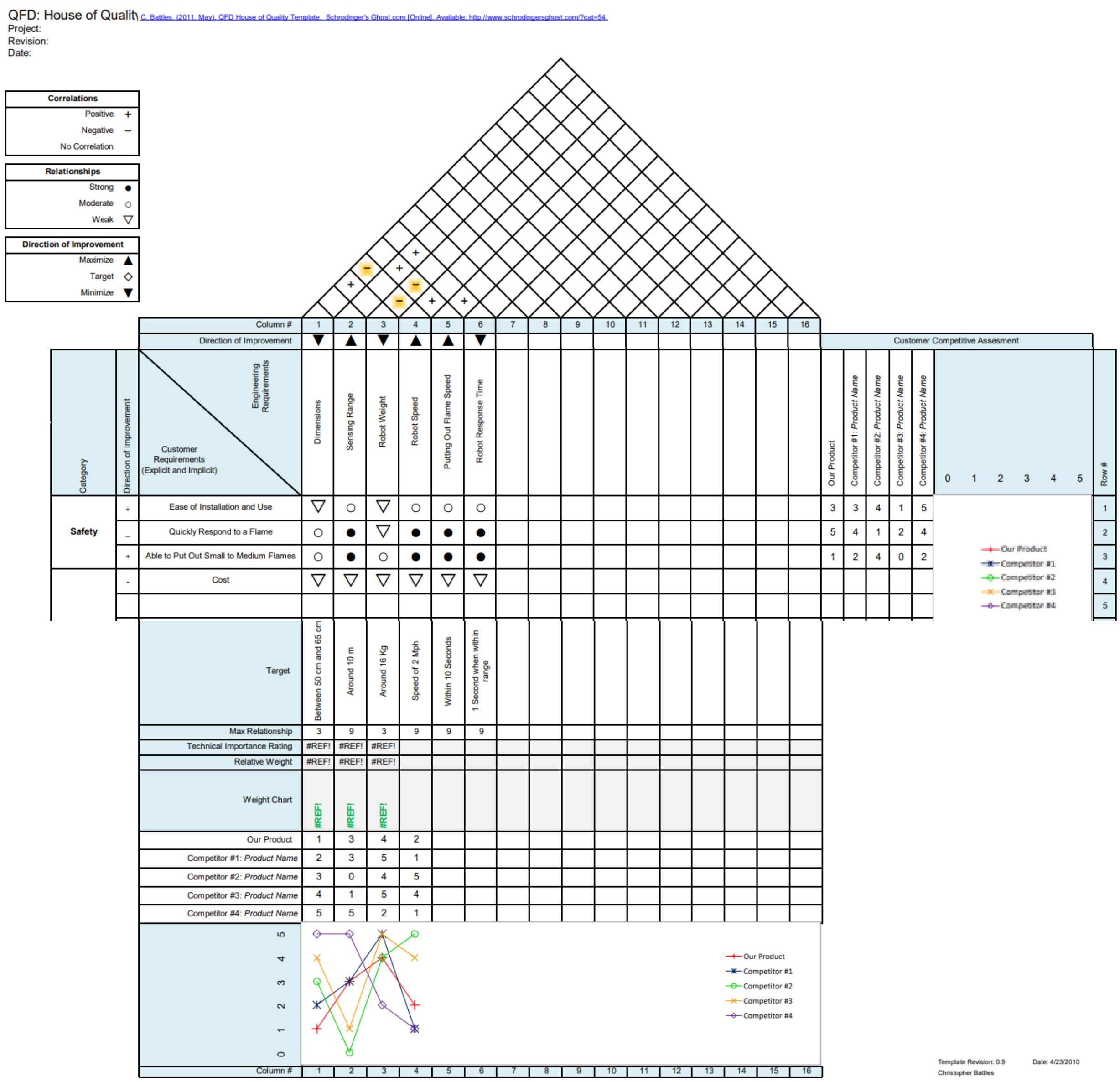
Table 1: Robot Design Specifications

| Specification | Description (Value) |
|---|---|
| Robot Size | 25 cm – 30 cm |
| Number of sensors | 3 |
| Robot Weight | Less than 2 Kg |
| Sensing Range | 30 cm<br>60 degrees |
| Extinguish time | 3 seconds |
| Robot movement speed | 2 Km/H |
| Robot power supply | 5 V |
| Response time | Instantaneous once within range |

# 2.4.2 House of Quality

This section will contain the House of Quality figure, which, as was previously described, is a product planning matrix. It was used to communicate how the customer's requirements were related to the methods that were employed in order to meet those requirements.

Figure 1: House of Quality



## 2.4.3 Block Diagrams

This next section will have four block diagrams that will describe how the team will break up the work among the four members. The four block diagrams that will be included are a general hardware diagram, an overall embedded system diagram, a general software diagram, and a power design diagram.

The general hardware diagram is a visual representation of how each group member will approach the hardware of the robot. As can be seen below, the hardware will be broken up into four parts: the first part will be the sensors and power supply, the second will be the microcontroller unit, the third will be the motor drive and pump, and lastly, the fourth will be the motor-specific power supply as well as the actual left and right motors that will power the whole robot.

Figure 2: General Hardware Diagram for Design

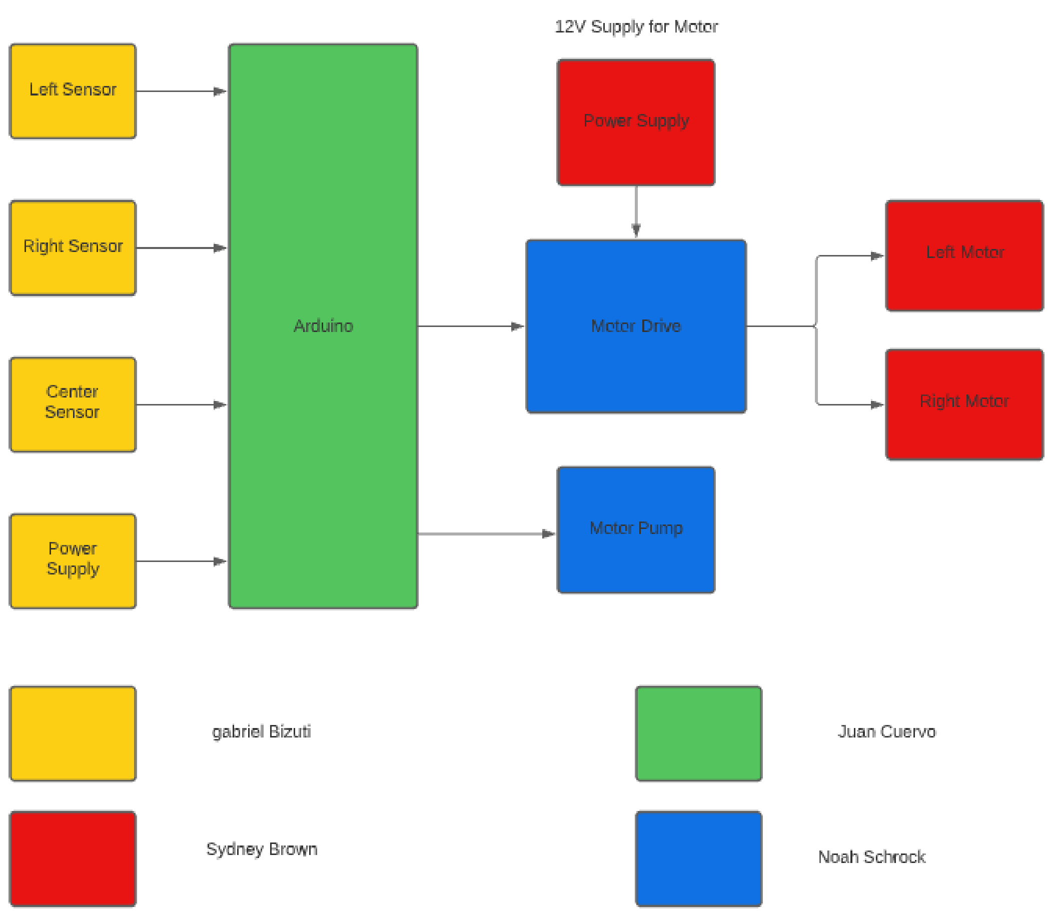
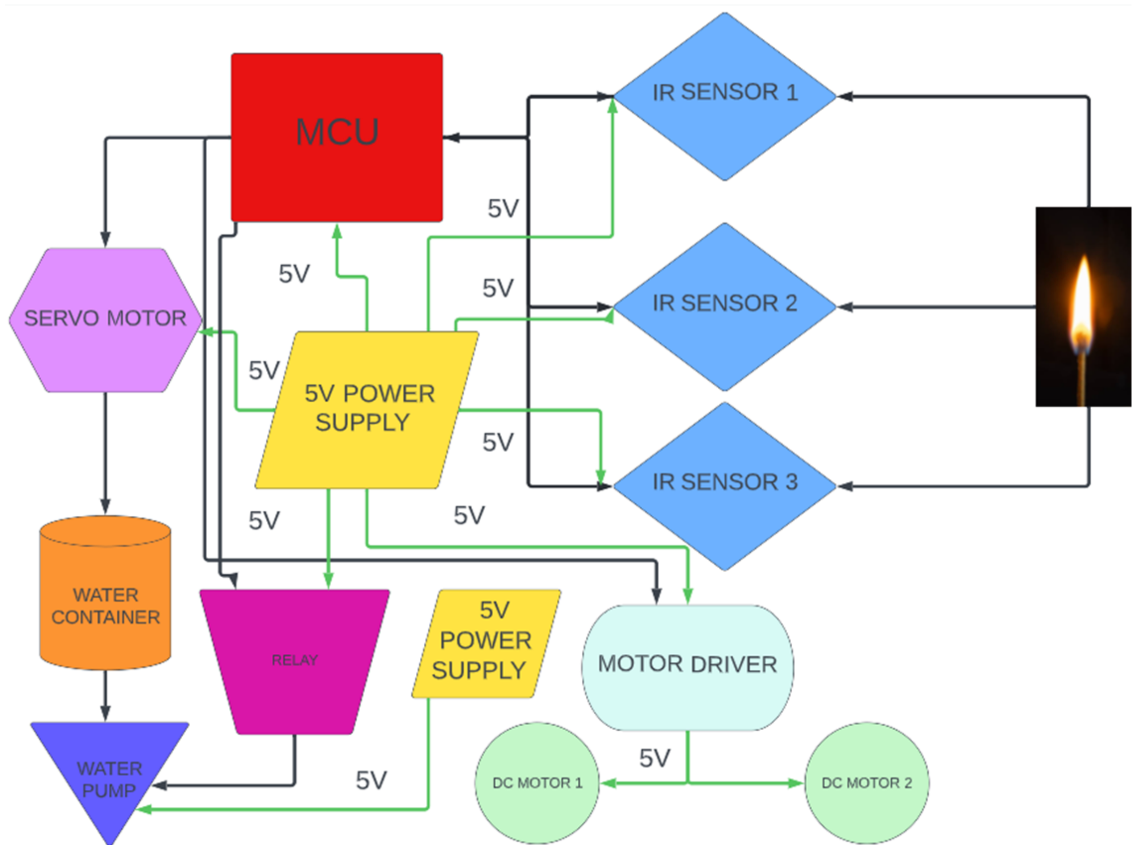


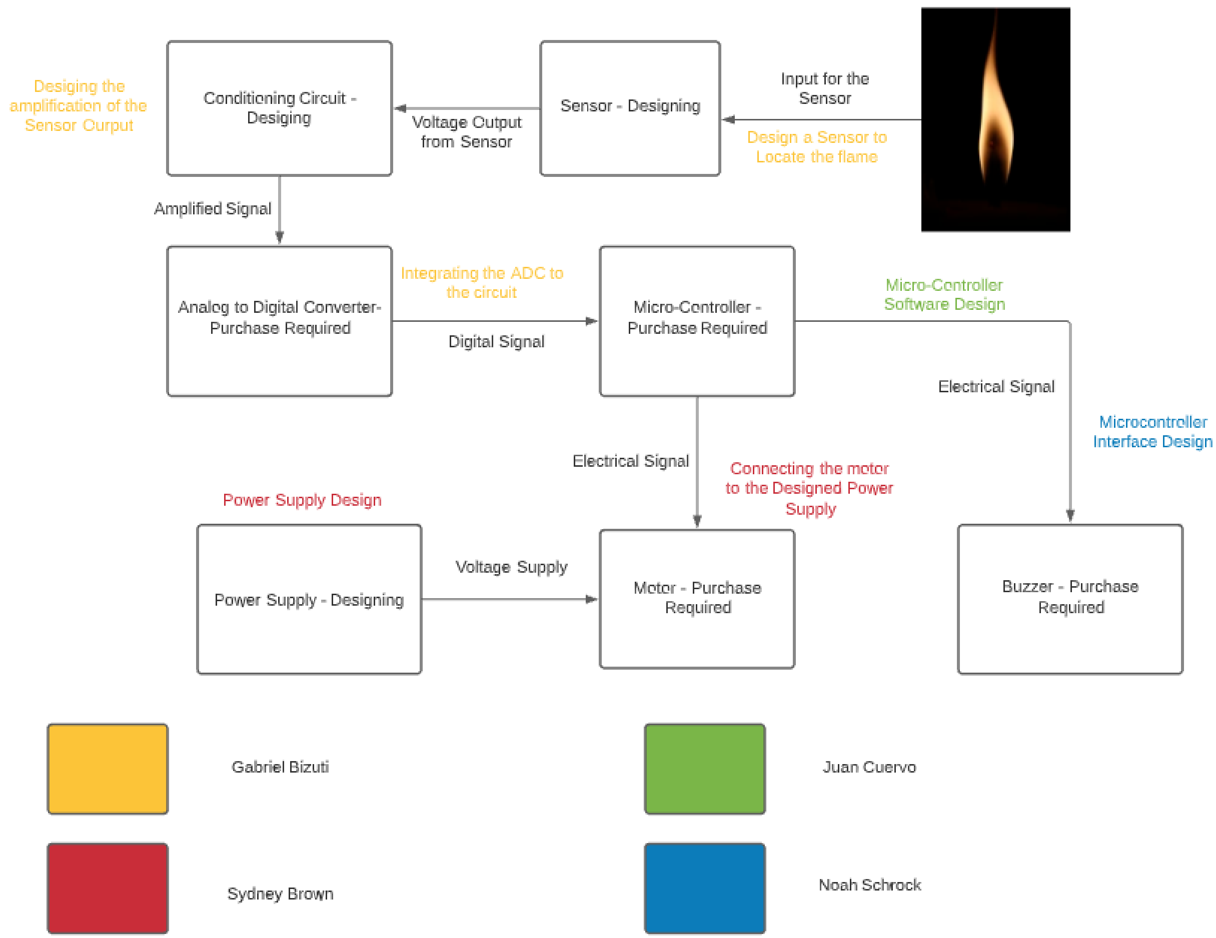
Figure 3: Updated Hardware Diagram After SD 2 Changes




The next figure is a generalized block diagram that breaks up the robot's embedded system into seven separate blocks, all of which are described. Each block is related to each other through a series of arrows which are given a particular team member in order to break up the work.

The block diagram provided, similarly to the previous block diagram, is broken up into sections that start from a given source and works its way up to the rest of the components. In this case, the diagram begins with the input from the flame in the given sensor, which corresponds to the right, center, or left sensors in the previous diagram. After this, it works its way through the embedded system until it reaches the software in the MCU, and the source of power in the power supply and motors.

Figure 4: Embedded System

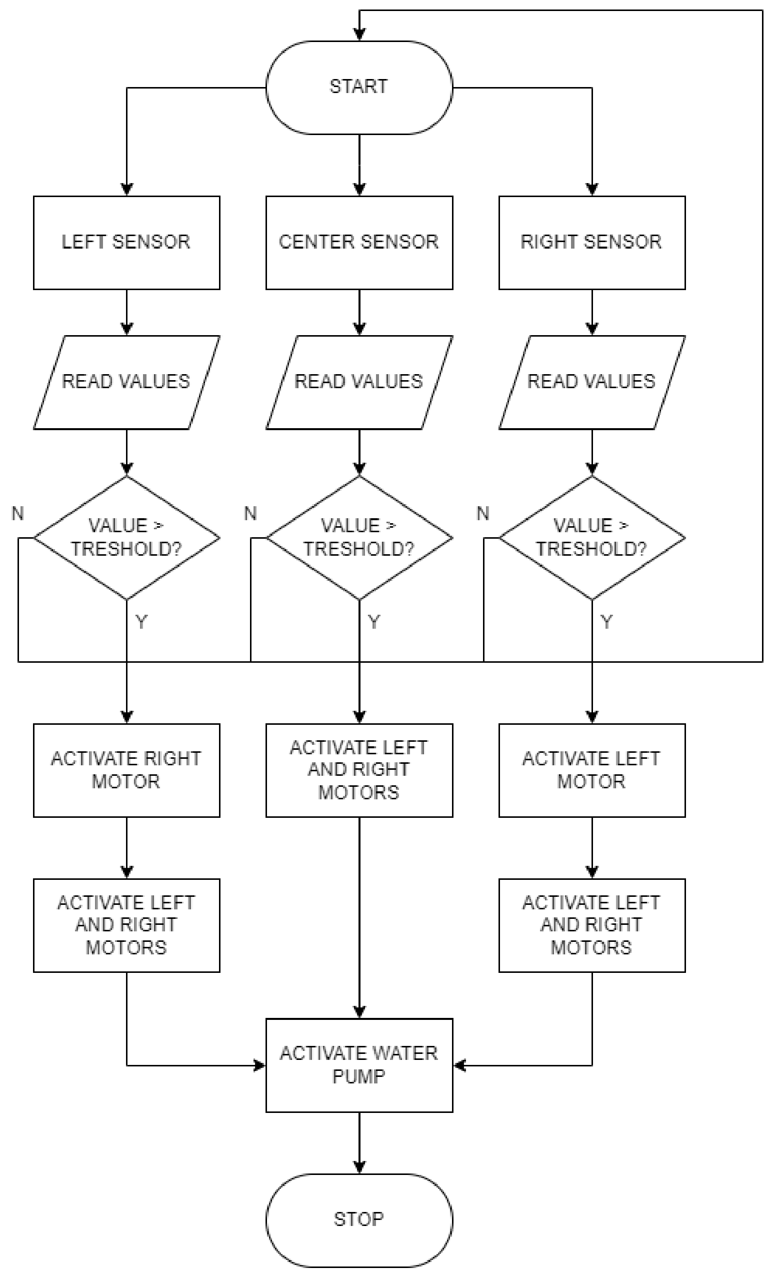



This diagram is a simple software diagram that expresses the basic algorithm that the robot operates under. It is based on the sensors that continually send signals, and depending on the given signal, the motors and water pump will be activated.

As shown in the color coded diagram, each member had different roles when it came to designing the embedded system for our firefighting robot. Gabriel and Sydney were more hardware focused as they are the electrical engineers on the team, and Noah and Juan were more software based as they are the computer engineers on the team.

When designing each part, the members had to come together to make sure everything coincided in the design process.

Figure 5: Software Diagram




This final diagram is a straight forward power design of the whole system. It describes the power transference between the main battery to all the components of the robot and shows how the battery voltage is transformed through a regulator to feed the MCU and the sensors along the servo and pump. Also, the motor driver serves as its own regulator to run the motors.

Figure 6: Power Design

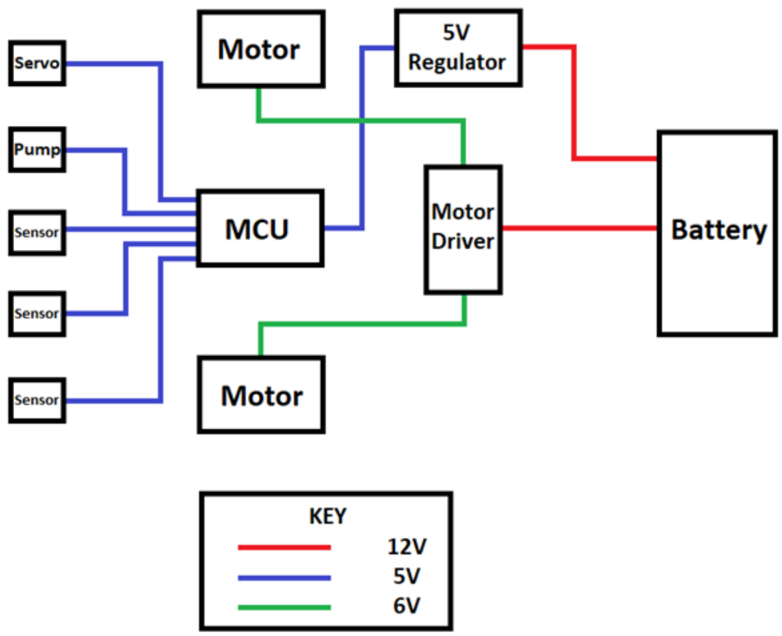
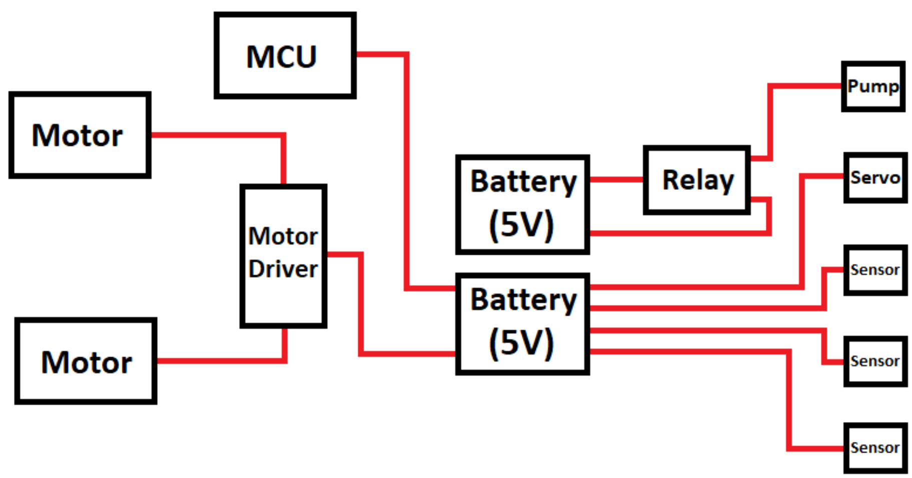



Figure 7: Updated Power Design




Note: the relay is connected on both sides to show that one end shows the relay itself being powered, whereas the other side is connected to power the pump when the relay is activated.

# 3. Research Related to Project Definition

In order to properly approach the problem of unintended fires, there is a fair amount of research that was done before a better solution could be addressed. Also, the research allows you to see blind spots that other projects haven't already addressed, or it allows you to focus on improving weaknesses of other projects that have come before.

This chapter will deal with all of the research that went into the project definition before anything else was considered. There will be six subheadings in this chapter that each focus on one specific form of research. As this project combines many different areas of research, there is a large amount of content in this chapter that helps familiarize the reader with the different areas that needed to be synthesized before approaching the project.

The subheadings contained are as follows:

1. Fire
2. Existing Projects and Products
3. Applications in the Field
4. Relevant Technologies
5. Hardware Components Overview
6. Part Research and Selection

The first subheading deals primarily with information regarding the different types of fires and fire extinguishers. Not all fires are alike and this section discusses the different ways that a fire can form and sustain itself, and with that, how not every type of fire extinguisher can safely and effectively extinguish every type of fire. The different types of fire extinguishers are all discussed and considered in order to find the best method of putting out the greatest number of fires.

Next, the subheading discusses the different kinds of existing projects that already address some form of the same issue. Since fire has always present some kind of danger, it is no surprise that multiple projects have already been developed in order to combat this danger. There are four projects that are considered and explored, and these projects helped the group find a direction in which to travel, in order to make the project unique, and also to help focus on a particular need that may not have already been addressed.

Following this, multiple applications in the field are considered. Specifically, the application of robots in the field of firefighting are looked at and discussed at length through five different developed technologies that have helped fire departments fight fires in a safer, more technologically advanced manner. These applications also helped shape the future of the project.

In order to begin developing a robot, different kinds of technology need to be understood and regarded. This subheading contains the different kinds of technologies that would be necessary in order to develop this project, ranging from smoke and heat detectors and sensors. Considering all of these technologies helped to focus the project even more, as it allows the methods through which the problem will be solved become more apparent, until one is selected as the most convenient and efficient.

The next subheading is the largest in the chapter and it deals with the different hardware components of the project. Since this robot leans very heavy on the hardware aspect, a lot of different components are discussed, from the Microcontroller Unit to the robot chassis, wheels, and electronic components like sensors and motors. This is where the project starts to get more specific, as the particular components are selected and described.

Finally, this subheading focuses on research done on the specific kinds of components that were previously discussed. In particular, three kinds of parts are considered in this section. First, the different kinds of fire sensors are described in detail. Pros and cons of each are also taken into account. Second, the different kinds of motor drivers are weighed and compared. Lastly, the different kinds of microcontroller units are considered and outlined. Pictures, charts, and figures are provided in order to illustrate the choices and illuminate why the parts were chosen as they were.

# 3.1 Fire

In order to properly approach the problem of putting out unwanted fires, the different types of fire extinguishers have to be properly understood. Most people are unaware that there are multiple types of fire extinguishers, and this difference from the fact that there are different kinds of fires. Since fires can start through different means, these different kinds of fires have different properties, and so not every type of extinguisher can extinguish every type of fire.

Before one can begin to talk about the different kinds of fire extinguishers, one has to first understand the different kinds of fires that exist. For the purposes of this research, there will be five different "classes" of fires that will be examined. These are the fires that will be addressed by the different kinds of fire extinguishers.

The different types of classes range from Class A to Class K. A brief outline and definition of each class is provided below:

- Class A: these fires are freely burning that come from combustible materials such as wood or paper.
- Class B: these fires burn from all kinds of flammable liquids and gases such as gasoline, propane, etc.
- Class C: these are fires that ignite from an energized electrical source. This class is specifically about the source of the ignition, since if the electrical source is not present, these fires would be either Class A or Class B.
- Class D: these fires burn from all kinds of metals such as titanium, zirconium, magnesium, sodium, etc.
- Class K: these fires burn from oils such as animal or vegetable oils or fats. These are fires that happen while cooking.

Every fire has the same four elements present: fuel, heat, oxygen, and chain reaction. In order to put out any of these fires, the theory behind portable fire extinguishers posits that removing at least one of these four elements will end up extinguishing the fire. In the following section, different kinds of extinguishers will be considered, where each one removes at least one of the four necessary elements present in a fire.

Every fire is different in ignition and source of fuel. In order to effectively extinguish a fire, one must be prepared with different kinds of extinguishers that depend on the class of fire. For example, a Class A fire can be put out with water, since the wood or paper that fuels the fire will become wet and the fire can no longer burn. However, a Class C fire cannot be put out with water because throwing water on an electrical fire could cause the electricity to be conducted through the water and this could cause injury to whoever is trying to put out the fire.

Properly understanding the different kinds of fires is beneficial to the project because it provides us with a better grasp on how to extinguish the fires the robot is presented with, but it also gave us a better narrowing down to what kind of problem this project attempts to address. The robot had different potential uses: it could've been specified to only fires that can be put out with water, or more broadly it could've carried multiple kinds of extinguishers as a far-reaching goal.

### 3.1.1 Types of Fire Extinguishers

The following section will be addressing six different types of fire extinguishers that all have different properties and sources of extinguishing fluids. Each will be explained in detail and lastly all compared and contrasted for benefits and drawbacks.

### ABC Powder Fire Extinguisher

The ABC powder fire extinguisher is one of the most common extinguishers, as it is multi-purpose, being able to put out numerous kinds of fires. It is the most versatile type of fire extinguisher and so it is commonly found in most places.

This fire extinguisher uses a powder of monoammonium phosphate as its extinguishing element, which suffocates the fire when sprayed over it, not allowing it to receive the oxygen it needs to continue burning. In this case, oxygen is the one element out of the four that will be removed in order to put out the fire.

Monoammonium phosphate is especially convenient in that it allows for the extinguisher to be especially versatile, as it is not a conductor. This means that this type of extinguisher can put out fires of Class A, Class B, and Class C. The lack of water in this extinguisher allows it to be used for more than one type of fire.

### Carbon Dioxide Fire Extinguisher

The carbon dioxide fire extinguisher is a convenient fire extinguisher since it leaves no residue on the fire, unlike the ABC powder fire extinguisher which leaves a layer of monoammonium phosphate powder. This is also a common type of extinguisher.

This fire extinguisher uses a liquid form of carbon dioxide that, when released into the air, displaces the oxygen from the fire, which takes the necessary oxygen away from the burning fire, and so extinguishes it. Also, the carbon dioxide in the chamber is cold, which when sprayed at the fire helps to cool down the fuel.

This fire extinguisher is perfect for Class B fires that are specifically flammable liquid fires and Class C fires. Since it uses carbon dioxide, the gaseous nature of this extinguisher allows it to be used safely on electrical equipment. However, it cannot be used with Class A fires and so it is less versatile than the ABC powder fire extinguisher, but still remains more convenient for clean-up purposes.

# Wet Chemical Fire Extinguisher

The wet chemical extinguisher is designed specifically for cooking fires, or Class K fires, that ignite through animal and vegetable fats and oils. This makes this extinguisher not particularly versatile. However, it can also be used on Class A fires. Considering that Class A and Class K are two of the most common fires, the lack of versatility doesn’t injure the usefulness of this type of extinguisher.

This type of extinguisher uses a potassium solution that addresses the fire in a two-step process that involves cooling and sealing the fire. This addresses all of the four necessary elements of a fire: fuel, heat, oxygen, and chain reaction.

The potassium solution sprays as a liquid mist which first cools the fire and the fuel, thus reducing the heat. Second, the potassium solution enters into a chemical reaction with the animal or vegetable oil or fat and creates a soapy element that creates a seal on the oil or fat, thus preventing the chain reaction from continuing, and removes the possibility of oxygen entering the oil and feeding the fire.

This makes the wet chemical fire extinguisher also quite common, since even though it can really only be used on two kinds of fires, these two classes (A and K) are two of the most common, so one can typically expect to find wet chemical fire extinguishers in kitchens, since all kitchen fires are usually Class K.

# Water Mist Fire Extinguisher

The water mist fire extinguisher is by far the most versatile fire extinguisher out of the ones discussed in this section. This type of extinguisher can be used in four out of the five classes of fire, and even though it uses water, it can still be safely used on fires that involve electrical equipment (Class C).

This fire extinguisher uses a misty spray of water that suppresses the fire by removing three out of the four elements needed in a fire: fuel, heat, and oxygen. The benefit of using a mist on the fire is that it takes up a lot of surface area. This causes the oxygen to be displaced from the fire and so helps it stifle the fire.

As a secondary effect on the fire, the water droplets are attracted to the fire, and so when it reaches the fire it cools down the fuel, which removes the necessary element of heat and it also addresses the fuel.

As previously stated, this type of fire extinguisher can be used on electrical (Class C) fires even though it uses water as the means of extinguishing. This is because this type of fire extinguisher holds water that has had its minerals removed, effectively de-ionizing the water and removing the possibility of conduction from the electrical source. Not only does this allow this extinguisher to be used on electrical fires, but it can also be used on fires that ignite through flammable liquids and gases.

As previously stated, this makes the water mist fire extinguisher the most versatile by far, as it can be used on Class A, Class B, Class C, and Class K fires. The only type of fire that it cannot address is Class D, which, being fires ignited through metals, is one of the most uncommon types of fires anyway.

# Foam Fire Extinguisher

The foam fire extinguishers are one of the less versatile fire extinguishers. This extinguisher, along with the ABC powder fire extinguisher, are the only two extinguishers

These fire extinguishers, as suggested by their name, spray foam onto the fire. This foam expands when it comes into contact with the air and covers the fire. This foam cover works to extinguish the fire in two ways. First, it prevents vapor from the fuel from reaching the fire, thus removing the element of fuel. Secondarily, the foam is water based, and so the foam addresses the element of heat by reducing the temperature of the fire.

The foam fire extinguisher may not be extremely versatile, but it is the best fire extinguisher specifically for fires that ignite from flammable liquids, such as gasoline. The drawback is that they cannot be used on fires that ignite from flammable gases, so it can’t be used for all types of Class B fires. However, this fire extinguisher can also be used on Class A fires involving wood or paper.

# Clean Agent Fire Extinguisher

The clean agent fire extinguisher is the best type of extinguisher for the environment. As an eco-friendly fire extinguisher, it very quickly dissipates in the atmosphere, and so it gets the job done without a negative effect on the environment.

This is one of the fire extinguishers that uses gas as its form of extinguishing. The gas, commonly Halon gas, is sprayed into the air from the chamber, where it’s kept in a liquid form that vaporizes when sprayed. The gas that is used is specifically non-conductive so that it can be used on multiple kinds of fires. This makes it not only safer, but also versatile, and safe for the environment.

This fire extinguisher addresses the fire by removing two of the four elements: oxygen and the chain reaction. The Halon gas sprayed on the fire displaces the oxygen from the fire, thus suffocating the fire. Also, the gas prevents the reaction from continuing, so over a short period of time the fire gets smaller until it is fully extinguished. On top of all of this, they require no cleaning after use.

The clean agent fire extinguisher is very convenient due to its non-conductive, eco-friendly, and clean nature. This makes this fire extinguisher common in rooms that are susceptible to electrical fires, such as computer centers, laboratories, data storage centers, museums, etc. This extinguisher can be used on Class A, Class B, and Class C fires, though they are most commonly used for Class B and Class C fires, as they are specifically designed for businesses, commercial facilities, and industrial facilities that have a lot of electrical components or flammable gasses or liquids.

Figure 8: The Six Fire Extinguishers in Order From Top Left to Bottom Right


# 3.2 Existing Projects and Products

This section will explore a variety of different existing projects and products that already attempt to answer the problem that the group is also trying to address. There are a great number of different approaches to this issue of undesirable and uncontrolled fires, but in order to avoid exploring every avenue that has been pursued, this section will focus primarily on four separate projects that approach the problem in a very similar way and style. These four projects are also robots that enter into situations that could be relatively dangerous for humans to undertake, and as such they represent more industrial level solutions to the more domestic audience that this project is trying to achieve. The four robots that will be considered in this section will be the Tactical Hazardous Operations Robot, the Thermite Robot, the Turbine Aided Firefighting Machine, and lastly, the Fire Ox firefighting vehicle.

# 3.2.1 THOR/SAFFiR

This autonomous firefighting robot was developed for the U.S. Navy's Shipboard Autonomous Firefighting Robot program. The Tactical Hazardous Operations Robot (THOR) is a humanoid robot capable of traveling across unstable floors on ships and utilizing hoses as well as opening doors. With hazardous materials aboard ships, it is a necessity to have the capability to extinguish fires. The robot uses stereoscopic thermal imaging and LIDAR (light detection and ranging) sensors to navigate its way to put out compartment fires. The goal of THOR is to walk and work semi-autonomously with the help of a remote operator. There are still improvements to be made as the machine is slow and susceptible to fire and water damage. These issues are being monitored so that eventually it will be able to extinguish fires that are too dangerous for people to get near.

Figure 9: The Tactical Hazardous Operations Robot (THOR)

# 3.2.2 Thermite Robot

This remote-controlled vehicle has the capability of pumping 500 gallons of water per mile. It was originally a small tank created for the U.S. Army by Howe and HOWE Technologies. It uses mounted cameras to travel into extremely hazardous situations while being controlled from up to a quarter mile away. It was designed as an IED (Improvised Explosive Devices) neutralizer but with some firefighting modifications, it can also be used as a fire neutralizer. With its ability to enter hazardous areas without putting people in harms way, it could be used to help extinguish fires in the future.

For the sake of comparing this robot with the tentative project, the Thermite Robot would undoubtedly be too big to solve the issue of domestic fires that form either as electrical fires or cooking fires. However, this robot could be a good basic template with the goal of duplicating it, albeit in a smaller scale, in order to approach the problem we are trying to solve.

Figure 10: Thermite Robot

## 3.2.3 Turbine Aided Firefighting Machine

This vehicle was created by Emicontrols to use turbines as an innovative method in firefighting. It is meant for small spaces like tunnels and it has the ability to move obstacles with its bulldozer blade. It can also clear smoke with its turbine and focus its water spray from mist to jet. While operators can be up to 500 meters away, this vehicle has its limitations because of its connection with a water hose.

A smaller version of this machine could be beneficial for an outdoor domestic machine, but it would not be feasible to have it as a possible solution for an indoor domestic fire such as an electrical or cooking fire. As will be explained in more detail in the further sections, some of these machines and robots will be used as inspiration for the building of our project, and this specific machine is one that provided some insights as to how the robot could be designed and built. One specific feature that we looked at was the turbine and how it aided the water jet. Eventually, the idea was scrapped, but the idea of using a non-conventional pump was retained.

Figure 11: Turbine Aided Firefighting Machine (TAF20)


## 3.2.4 Fire Ox

The Fire Ox was designed as a first response unit. It is one of the few robotic firefighting vehicles that carries its own water tank. The Fire OX suppresses fires, assists in search and rescue and can handle dangerous materials. It was originally created by Lockheed Martin as a Squad Mission Support System (SMSS) for helping soldiers with their gear in the field. It was then modified with a water tank and hose for distribution. The Fire Ox can be controlled from up to 200 miles away. This semi-autonomous robot has the ability to traverse situations unsafe for people, minimizing casualties and rescue time. The Fire Ox can also be useful for situations such as wildfires and structure fires.

Figure 12: Fire Ox

## 3.3 Applications in the Field

The fire department has incorporated some new technologies in the fire service and has made several advances in the industry. The following technologies are just a few of the many advances the fire department has developed.

## 3.3.1 Robotics Systems 3 (RS3)

The Los Angeles Fire Department revealed its Robotics Systems 3 (RS3) firefighting droid in October 2020. The hose-equipped tank-like vehicle can pump up to 2,500 gpm of foam or water into fires that are too dangerous for humans to attack because they could potentially be explosive. The RS3 proved its value during a textile warehouse fire that occurred unexpectedly, using its front-mounted plow to push its way into the burning structure after fire crews had to pull back for safety reasons.

Figure 13: RS3 Extinguishing Fire that Occurred in Los Angeles


## 3.3.2 Dragon Eggs

'Dragon Eggs' are a new technology advancement made for igniting fire breaks in hard-to-reach areas. Each Dragon Egg, which is about the size of a ping-pong ball, is filled with potassium permanganate. They are controlled by drones and flown remotely over an advancing fire and then dropped. The eggs receive a small injection of anti-freeze just before they are dropped, which causes them to ignite when they hit the ground. Aerial drops of Dragon Eggs helped stem wildfires in California and Colorado.

Figure 14: Dragon Eggs and Drone

## 3.3.3 EHang 216F Firefighting Drone

Ground-based equipment can only reach so far up a building, which is why the EHang 216F Firefighting drone was introduced in 2020 by a Chinese drone-maker EHang. The eight-rotor drone can carry up to 40 gallons of firefighting foams and six fire extinguisher bombs in a single trip. It uses a visible light zoom camera to quickly identify the location of a fire and extinguish it. Multiple 216Fs can be remotely dispatched before ground-based apparatus even arrives on scene. This design is slightly out of our range of potential inspiration since it is a flight based robot, but there are still some interesting design ideas that were considered from it.

Figure 15: EHang 216F Extinguishing Fire

### 3.3.4 WildFireSat System

Canada introduced a system known as the WildFireSat system, which consists of one or more space-based satellites equipped with infrared sensors to measure the energy emitted by wildfires. The WildFireSat will support smoke and air quality forecasting as well as carbon monoxide monitoring. The system is in the construction phase and plans to launch into orbit starting in 2025.

### 3.3.5 Conclusion of Advances and Applications in the Field

Advances are constantly being made in the firefighting industry. With fires being a very prominent issue not just in the United States but worldwide, it is a necessity to figure out how to improve the safety and well-being of humans and wildlife. The technology being incorporated into the system is greatly beneficial not only to the people in harm's way but also the firefighters who constantly put their life on the line to extinguish these dangerous fires.

## 3.4 Relevant Technologies

Automatic fire suppression devices are an important feature in many homes, offices, buildings, and other facilities. Knowing the advantages and specific uses of each of the different technologies can help a consumer decide what device is best for them. Below are several different technologies available for an automatic fire suppression device.

### 3.4.1 Smoke Detectors

Smoke detectors utilize smoke in the air to quickly detect fires. There are three subtypes of smoke detectors. The first is photoelectric alarms, which utilize a photoelectric sensor, or a light beam, to detect smoke particles and set the sensor off. The second type is an ionization alarm. These alarms use a small amount of radioactive material to keep an electric current running between two electrodes. When smoke enters the alarm, the ionization is disrupted and sets off the alarm. The last type of smoke detector is known as a combination alarm. This device uses both types of technologies mentioned in the first two and as a result, can detect both low and high energy fires.

### 3.4.2 Heat Detectors

Heat detectors are usually installed in the roof and inside them is a small heat sensitive element. When the heat from a fire rises and begins to heat up this element it will trigger an alarm letting those around know there is a fire in the area. There are two types of heat detectors, fixed temperature heat detectors and rate of rise heat detectors. The former, fixed temperature heat detectors, are the most common. They feature a heat sensitive alloy that transitions from a solid to a liquid when the surrounding air exceeds its melting point of around 58°C or 136°F. Rate of rise heat detectors use a different method; they instead measure the temperature change in a room to determine if there is a fire. Most rate of rise detectors will sound an alarm if there is an increase of 6.7° to 8.3°C or 12° to 15°F per minute.

### 3.4.3 Flame Detectors

Flame detectors respond to the presence of a flame or fire. There are many types of flame detectors but three are more widely used than the rest, the first of which is known as an optical detector. This device uses optical sensors to detect when flames are present. The second type is a UV detector, which detects the amount of UV radiation emitted from the flames at the instant they ignite. The moment the flame gives off the UV radiation, the flame detector is triggered. This method allows the device to detect a flame within four milliseconds, but has a time delay of about 2 to 3 seconds to reduce the number of false alarms. The third type is an IR (Infrared) detector, which uses thermal imaging cameras to monitor the infrared spectrum and trigger the alarm when it recognizes the specific pattern that hot gases produce by a fire. These are especially useful because they can monitor flames without being hindered by water. The alert time for IR detectors is three to five seconds.

### 3.4.4 Air Sampling Smoke Detectors

Air Sampling Smoke Detectors have the ability to detect low energy fires before they can damage equipment, which is why they are considered Very Early Warning Fire Detection Systems. These smoke detectors continuously sample and test particle levels in the air of a particular space, setting warning and alarms when the level of smoke particles rise above the typical amount. This type of detector system is more resistant to false alarms as it is constantly monitoring the particles in the air.

## 3.5 Hardware Components Overview

This section will cover an overview of the hardware components that will be used in the project. As the robot will be mainly hardware driven, it is absolutely imperative to make sure that the selected hardware parts are not only appropriately selected for their specific role, but that all selected parts can effectively work in tandem with one another for the required results.

The hardware components that will be considered in this section will cover a large quantity of parts over a large range of components. Each will be thoroughly explained and their overall job in the project will be outlined. There will be a total of ten parts that will be explored in this hardware component overview section, and the ten different parts will be the following:

<u>Arduino Uno:</u> This part is the brain of the entire robot. As the selected Microcontroller Unit, the Arduino serves as the centralized control of all of the robot's operations. Features of the Arduino Uno will be explained and later the paper will explain the reason for this choice.

<u>IR Flame Sensor:</u> The Infrared Flame Sensor plays the part of the eyes of the robot. This sensor is in charge of flame detection, and in this section, a basic overview of the sensor will be explored, as well as providing a general high level explanation of how it works.

Also, the paper will later on explain why this choice was made as compared to other sensors.

L293D Motor Driver: This specific motor driver serves as the robot's source of mechanical motion by being the component that controls the different motors that are used in the robot. Since we have two motors as opposed to one in order to control a two dimensional range of motion, the motor driver provides a mode of controlling this range of motion.

Servo Motor: The servo motor is the component that the robot uses in order to ascertain the necessary angles in the motion of the robot's spray hose. The section on the servo motor will provide greater detail and explain the reasons as to why the servo motor is a necessary component in the building of the robot.

Submersible Water Pump: Choosing an appropriate water pump is essential in determining how the robot effectively fights fires. For this project, the submersible water pump was chosen among many other options, and the reasons why this type of pump was chosen over the others will be discussed in more detail later on in the following section of the paper.

Relay Module: The relay module was an addition we made after performing hardware testing. The Arduino alone could not supply enough current to drive the pump so we used the relay module in order to provide power.

Bread Board: Another important component for the sake of testing all the parts of the project is a bread board. The bread board is invaluable in testing the components and how they work together.

Robot Chassis: The section on the robot chassis will include more in depth exploration of what the group will consider as an effective way to form a chassis to house the robot and the specifics of it.

Motors and Wheels: The motors and wheels will be discussed in greater length in the following section and will be explored in depth. The specific types of motors and wheels are also considered and weighed.

Connection Wires: Lastly, connection wires form the most basic hardware component that is essential for the robot to effectively communicate between all of its separate parts, and this section will give a brief explanation on types of wires and their intended use, as well as how the group planned to use them.

# 3.5.1. Arduino Uno

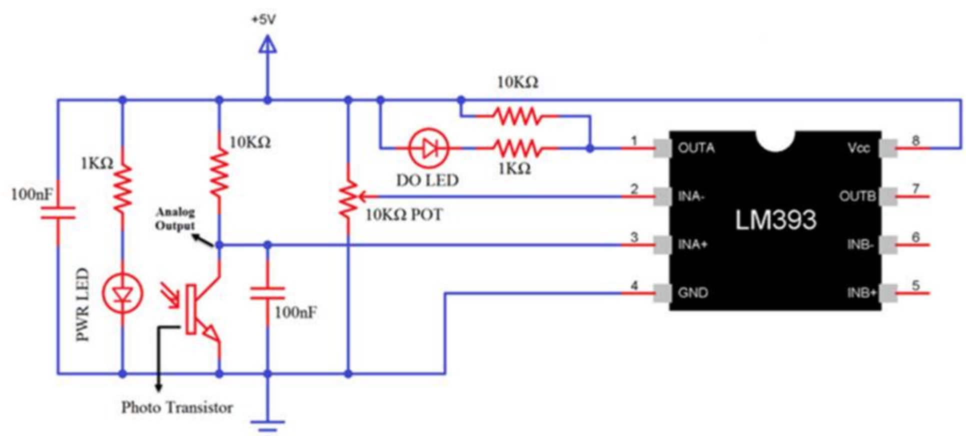
The Arduino Uno is a microcontroller board that can be implemented into several different electronic projects. It is easy to use and listed at a reasonable price, making it a very desirable choice for users. Features include the AVR microcontroller Atmega328, 6 analogue input pins, and 14 digital I/O pins in which 6 are used as PWM output.

The board contains a USB interface to connect it to a computer and program it using the Arduino IDE software. It is used in several applications including embedded systems, home automations, traffic controls, medical instruments and several others. The Atmega328 is a good platform for all sorts of robotic applications. The features and various applications that can be implemented using this board made it very desirable for our project.

## 3.5.2 IR Flame Sensor

The IR flame sensor detects the presence of fire or other infrared sources in the environment. It does this by detecting a flame or light source with a wavelength in the range of 760 nm to 1100 nm. Features of the IR flame sensor include an adjustable threshold value and a two-state binary output. The sensor is also small and compact and very easy to mount. The robot contains three flame sensors pointing in different directions in order to get a precise reading on where the flame is located.

Figure 16: Circuit Diagram of Flame Sensor




The working of this three pin sensor is made up of a few key components including a YG1006 NPN Phototransistor, LM393 Comparators IC, variable resistor, power LED, and an output LED. In using this flame sensor, infrared light can be detected up to 100 centimeters within 60 degrees of detection angle.

The YG1006 Phototransistor is coated by black epoxy, making it sensitive to infrared radiation. It is a 5mm NPN Transistor with only two terminals, unlike many other transistors. The long terminal is the emitter and the shorter terminal is the collector. When light is detected the current starts to flow between the emitter and collector.

The variable resistor (potentiometer) is set at 10 Kilohms and can be adjusted to change the sensitivity. When the knob is rotated clockwise, the sensitivity of the flame sensor will increase. When it is adjusted counterclockwise, the sensitivity is decreased.

The onboard LED indicates whether the flame sensor's power supply is on or off. When the power supply is connected to the flame sensor, the LED turns on.

The output LED turns red when the sensor detects a fire. If no fire is detected, the red LED is off.

How it works:

The sensor is connected to a 5 Volt power supply. The threshold is then set at the non-inverting input of the IC by adjusting the preset knob at the desired sensitivity. When the sensor detects a flame/fire with light in the range of 760 nm to 1100 nm wavelength the resistance of the Phototransistor decreases. The maximum amount of voltage will allocate across R2, resulting in a low amount of voltage from the Phototransistor to the inverting input of the IC. The Comparator IC compares the voltage with the threshold voltage and since the input voltage is less than the threshold, the sensor switches down to LOW, or to zero.

The opposite occurs when the flame sensor does not detect a fire or flame. The resistance of the Phototransistor is high, and the maximum amount of voltage is allocated to the Phototransistor. A high amount of voltage is given to the inverting input of the IC from the Phototransistor. Once the Comparator IC compares the voltage to the threshold voltage, the sensor goes HIGH, or to one, since the input voltage is greater than the threshold.

# 3.5.3 L293D Motor Driver

Motor drivers are essential if you are planning on creating a robot that moves, and one of the least expensive and easiest ways to create a robot that uses two motors is by interfacing L293D motor drivers with our Arduino. The L293D receives signals from the microprocessor and transmits the relative signal to the motors. It has two voltage pins, one of which draws current for the working of the L293D and the other of which is used to apply voltage to the motors. It contains 16 pins which are dedicated to controlling motors. There are two input pins, two output pins, and one enable pin for each motor. The primary use of the motor driver is to run the two DC motors simultaneously in any direction.

# 3.5.4 Servo Motor

Figure 17: Servo Motor

A servo motor is a device that allows for precise control of angles and/or linear positioning through the use of linear or rotary actuation. These devices can come in an array of sizes as well as quality and complexity. From large and complex like in industrial grade servo motors to a small one made to be mounted on remote controlled airplanes. Servo motors are often used in closed loop control systems and tend to require a dedicated module that is specifically designed for servo motors. However, with this trade off you usually get much better performance than something comparable like a stepper motor.

Given that the project at hand needs precise control of angles and linear positioning, the servo motor is likely the best candidate to accomplish the job required for the robot's maneuverability and accuracy. The purpose of the servo motor for the sake of the project is to make sure that the water pumped is aimed at the precise location that the fire is detected in order to avoid wasting water by pumping it to an incorrect, or imprecise, location. This issue is easily solved through the use of a servo motor that can be the determining agent in calculating and implementing the correct angles for the water pump to aim at the correct location.

## 3.5.5 Submersible Water Pump

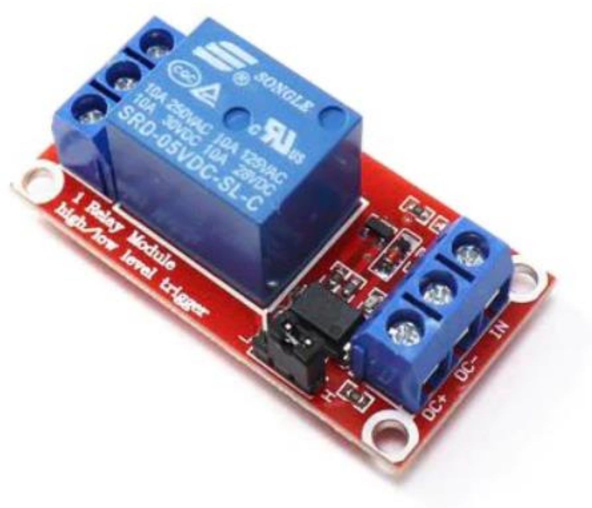
Figure 18: Submersible Water Pump

There are many types of pumps with various advantages and disadvantages. After looking at many different kinds, however, the conclusion drawn was that the submersible water pump seems to be the best pump for the firefighting application the group is attempting to address. These pumps, rather than producing a dedicated stream of fluids, focus rather on pushing fluids to the surface which creates a vacuum. Submersible pumps are sealed motors that are coupled to the pump body which is then submersed in the water reservoir. In later sections, different kinds of pumps will be discussed and the reasons as to why this specific kind was chosen will also be further explained alongside the drawbacks of the other competitors.

All else being equal, the group found that submersible pumps have one main advantage over other pumps overall. The advantage that tipped the scales was that these pumps do not suffer from pump cavitation. Pump cavitation is a problem that happens in a lot of pumps where when in an area of high pressure, a pocket of water vapor or bubbles can begin to form disrupting the flow and, in some cases, even damaging the machinery the pump is connected to. Considering the project at hand, since fires cause an increase in pressure in the area due to the rapid thermal expansion of gas in a closed area, the group concluded that a submersible pump would be the way to go in order to avoid the issue of pump cavitation. In an emergency, the robot cannot malfunction due to a cavitation issue that is likely to happen in a plurality of cases, since, as previously stated, pressure is something that the robot needs to take into account before trying to produce any kind of replicable solution. Considering the job that the water pump needs to play in any given domestic fire, it is clear that large quantities are not the main factor, at least not to the industrial scale of the previously explored projects. For the given task, the pump needs to be, first and foremost, reliable, and this pump was the best way to ensure that.

# Relay Module

Figure 19: Relay Module

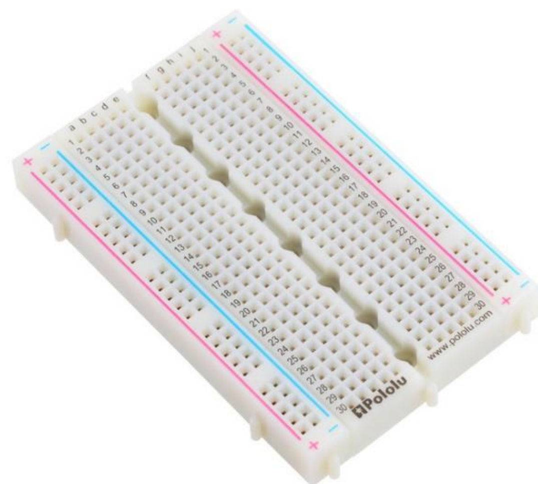



A relay module is an electrical switch that is operated by an electromagnet, which is activated by a low-power signal from a microcontroller. When it is activated, the electromagnet is either pulled open or closed.

This device is especially useful for our design as it is what drives our pump. The pump does not receive enough current from the Arduino alone, so the relay module was a necessary hardware component for our project.

## 3.5.6 Breadboard

Figure 20: Breadboard




Bread boards are blocks that have strips of metal in them which allows for quick circuit building and testing before finalizing your design. These are rather helpful as they allow you to create connections among multiple devices without having to solder or permanently connect any wires or connections.

The legs of components can be inserted through the top layer of plastic where they touch the metal strips inside the board creating connections. These are sometimes used in final products, but because the connections aren't permanent, they can sometimes come loose, and they often look sloppy and unprofessional. They are however invaluable to the testing stage of a project.

## 3.5.7 Robot Chassis

A robot chassis can be made out of anything including plastic, wood, metal, and even Styrofoam, it all depends on the robots intended use. As long as you are able to fit all the internal or working parts of the robot and it is made out of a material that fits its purpose then you have a robot chassis that works.

There are some other details that are important depending on the robot you are creating such as its ability to protect components and the weight and dimensions of the chassis. This is the most customizable part of the robot and can be used to give it character as well as functionality.

## 3.5.8 Motors and Wheels

There are many ways to make a ground-based robot move, and one of the more reliable and easier to use methods is using wheels and motors. There are many variations of wheels and motors that you can use depending on the task that you want to accomplish. Wheels can vary in size and count, and the motors can vary in size and the amount of power that they generate.

If you have already chosen what driver module you want to use then you just need to pick a motor that is compatible with it and choose a number and size of wheels that you want to use. Remember that with the motor driver the number of wheels you can control may change. For example, the L293 motor that we chose to use can control one stepper motor or 2 DC motors. An issue arises if you want to have a four wheeled robot, in this case you would need 4 DC motors and 2 motor drivers to control them, or 4 motor drivers with 4 stepper motors.

### 3.5.9 Connection Wires

Figure 21: Connecting Wires

For the prototype we used a less permanent connection method than regular wires so that we could easily make modifications without great effort. To do this we used a type of wire called a jump wire, which are these are short metal wires with a rubber coating and on either end of the wire is a small plastic box and can come in the following configurations:

- Two male connections
- One male connection and one female connection
- Two female connections

Jump wires are designed for use with a breadboard and they are especially useful when testing connections and figuring out how your circuits are supposed to work effectively. Once our prototype was finalized, we switched to a more permanent connection method by having some of the necessary connections soldered onto the PCB.

# 3.6 Part Research and Selection

This section will be a relatively extensive exploration of all of the research that went into the necessary parts for the project. Not only will this section go into the depths of the underlying research, but it will also go through the selection process and the criteria we considered in selecting each part. The previous section explored some of the hardware components, and as such, this section will provide further justification as to why some of those parts were selected. Given that this project is primarily a robotics project, there are a great number of parts that had to be considered, studied, researched, selected, and compared with the other selected parts to make sure that every individual part could work cohesively with the overarching system. At the end of each given subsection, there will be a table that compares the different kinds of metrics for each of the competing components, and this will be the main factor that helped determine which component was used in the project, keeping in mind the main purpose of the robot and what the given part was to be used for.

There are five main components that are what could be considered as the main parts of the robot. These hardware parts that will be shortly explained further are the following five:

<u>Sensors:</u> Specifically, the sensors chosen are those that can best detect fire. All kinds of sensors were studied and considered, and the means through which they detect fire will all differ greatly in order to have a greater range of options. These methods range from using ultraviolet light as the means of detection to using infrared, a combination of both ultraviolet light and infrared light, and other means of detection such as plainly detecting visible light, and more advanced forms of infrared light detection. In total, five different kinds of fire detectors were considered in the following section and then also further explained.

<u>Motor Drivers:</u> In order to effectively operate the motors house by the robot chassis, there needs to be a motor driver that operates the motors since the microcontroller unit is incapable of performing this task without any added aid. For this section, only two kinds of motor drivers are compared, as these two are the most commonly used motor drivers, and their similarities and differences were weighed in order to decide on the ideal motor driver.

<u>Microcontroller Units:</u> The Microcontroller Unit is arguably the most important part of the robot, since it controls the entire robot as the central point of communication between all parts and it runs the software written for it. There are many kinds of microcontroller units and some have been previously used by the group members. There are three different kinds of microcontroller units that were explored, each with a different method of programming and a different level of embedded system coding, ranging from the more involved register level programming to the more accessible library level programming, which we ended up using.

Pumps: As previously stated, the submersible water pump was found to be the most appropriate application of a water pump for this type of firefighting project. As such, this section will focus primarily on two different kinds of submersible water pumps that could be used for the project. Mainly, one of the bigger selection criteria will also be how these water pumps are connected to the robot and what level of power they provide, as these pumps are generally not as strong as jet stream pumps that are used in other kinds of firefighting robots.

Power Supply Selection: Another extremely important part of the design and operation of the robot is the method of power supply. There are countless ways of powering a machine that all have its benefits and drawbacks, such as the capability of wireless power dependent on a given time frame, or an infinite amount of powered time that is taken back by the connection to a wall. Different kinds of wired and wireless power methods were explored through wall adapters, USB connections, battery packs, and voltage generators.

## 3.6.1 Fire Sensors

Sensors are an important part of a lot of different applications, devices, and their functionalities. Being able to sense the change of different events is what allows a lot of applications and devices to operate. The accuracy of the sensor is, a lot of the time, what determines the quality of the product since it directly relates to how well and how precisely the product operates.

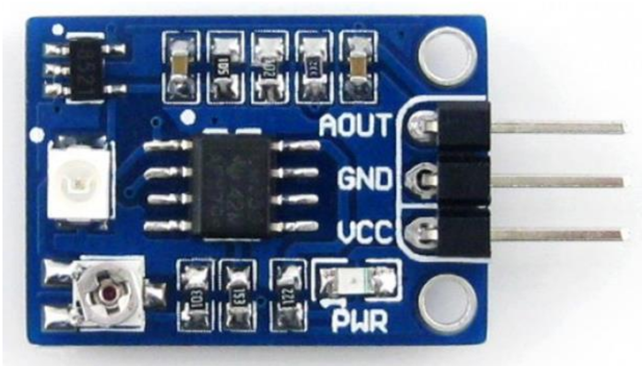
Our project uses sensors to sense the environment for the appearance of any flames that need to be put out. With that in mind, it was important that we chose the appropriate sensors that fit our requirements and our needs. There are a variety of sensors that are capable of sensing the presence of a flame in the environment. Each sensor has a different approach to sensing the flames and because of that they all have different advantages and disadvantages, all of which will be explored in this section.

## Ultraviolet Sensor (UV)

One of the sensors that are capable of sensing flames is the ultraviolet (UV) sensors. These types of sensors are based on ultraviolet ray detectors that can respond to the radiation of the rays. These sensors are usually employed inside places where they cannot be influenced by different sources of radiation. The reason for this is that outside sources of radiation like electrical discharges and sparks can end up triggering the sensor and resulting in a false detection of a flame. False alarms are a problem since they directly affect the reliability and accuracy of the project.

Due to the possibility of triggering a false alarm, this type of sensor is usually used for close environments that help to increase the reliability of the sensor. For the ultraviolet sensor, thick smoke is another variable that can reduce the accuracy of the sensor. The smoke hinders the ability of the sensor to detect the radiation that is necessary for it to detect the presence of a flame. The sensors that use ultraviolet radiation to detect flames have a range of around 15 meters. This range meets the specifications for the project, but it was found that this benefit was negated by the drawback of its decreased visibility through a potential cloud of smoke that many fires generate.

Figure 22: UV Sensor




# Infrared Sensor (IR)

Another type of sensor that can detect the presence of flames is the infrared (IR) based sensor. This sensor uses the infrared radiation light that is emitted from a flame in comparison to a base level of infrared light emitted or reflected by surfaces in order to detect an anomaly of high temperatures such as a fire. All flames, even the small ones, emit the infrared light that is used by the sensor. The sensor works by sensing the infrared spectral band that comes from the hot gases of the flame. Because of that, it can distinguish the infrared signal that comes from a flame from the signals that come from other sources such as lightning, electric discharges and welding.

This sensor is also not affected by sunlight since the infrared radiation that comes from the sun and could lead the sensor to false alarms is lost in the atmosphere. However, even though it is not affected by the radiation of the sun, the presence of water can interfere with its accuracy. The reason for that is because water vapor can absorb a great amount of infrared radiation and that would interfere with the reliability of the sensor. This is a very important point to consider since water is major player in the robot, as its final purpose is to put out fires through the use of a water pump. In addition, other sources of infrared radiation may also interfere with the measurements of the sensor. Just like the ultraviolet sensor, the IR sensor has a sensing range of around 15 meters, which meets the desired specifications for our project. All of these advantages and disadvantages are weighed and considered at the end of this section alongside the other types of sensors that will be talked about.

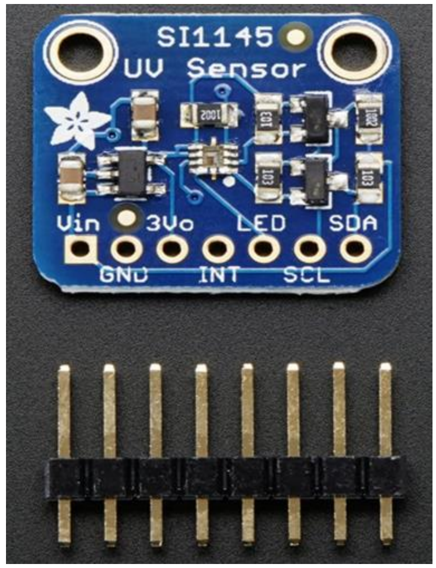
Figure 23: IR Fire Sensor

# Ultraviolet and Infrared Sensor (UV/IR)

A sensor that combines the technology of ultraviolet (UV) and infrared (IR) is great for increasing the accuracy and reliability of the sensor. This sensor uses ultraviolet and infrared radiation as a tool to detect flames. The combination of both technologies allows the sensor to have different perspectives on events. Due to the different perspectives, this type of sensor can minimize false alarm alerts. It uses both the information from the ultraviolet radiation and infrared radiation to detect the flame based on the conclusions from both of the inputs. It is important to note the way that this sensor functions, and that is, if a flame is detected by both inputs of the sensor, as opposed to having at least one detect a flame, a correct alarm is set.

However, since this sensor is a combination of the ultraviolet sensor and the infrared sensor it also has the drawbacks that each sensor possesses. Due to needing the response from both inputs to operate, the impairment of one of the sensors hinders the operation of the other. The operation of the ultraviolet sensor may be affected by heavy and thick smoke. And the operation of the infrared sensor may be compromised in places with a lot of water vapor and humidity. This feature of this sensor may be a heavy drawback that does counterbalance the benefits of having both technologies present in the sensor. Since the main purpose of having a specific type of sensor is to increase the overall accuracy of true positives and decrease the occurrence of false negatives, it matters relatively little to have increased kinds of technologies if, for the sake of the application, the group finds that the increased power of the technology used is, after all, a greater hindrance to the overall project as opposed to having chosen one over the other. This sensor, like the other two, has a range of around 15 meters, which is enough to meet the specifications of the project.

Figure 24: UV/IR Sensor




# Visual Flame Sensor

Another way of detecting flames is through using a visual flame imaging detector. This type of detector does not sense radiation as the ones previously mentioned. The visual flame imaging detector uses a charged couple device (CCD) image sensor to detect the flames. It also uses a software detection algorithm to analyze the image from the sensor and arrive and a conclusion if a flame has or has not been detected. By analyzing the shape and development of the images the software is able to detect the flame.

Due to the nature of this sensor, it is not affected by different radiation types and their intensity. This type of sensor is usually necessary when you need to distinguish different types of flames or fires and not just if it's present or not. The disadvantages of this type of sensor include not being able to sense flames that are not visible through the image sensor like hydrogen flames. Heavy smoke is another factor that can affect the ability of the sensor to detect flames. The range for this sensor depends on the resolution of the CCD image sensor that is used. Considering that this sensor not only has the same heavy drawback of smoke impairing the vision of the sensor as well as its own dedicated drawbacks from its use of visual light as the means of sensing, this sensor was not considered one of the forerunners, as it does have the greatest number of drawbacks and not a great many number of advantages that could potentially outmatch the advantages of the other kinds of sensors in the list.

Figure 25: Visual Flame Sensor


# Advanced Infrared Technology Sensor (IR/IR and IR3)

There are other sensors that also use infrared technology. These are the IR/IR flame sensors and the IR3 flame detectors. The IR/IR sensors use two infrared sensors to analyze and compare two different frequencies. This helps to better enhance the reliability and accuracy of the sensor by reducing the false alarm occurrence rate. The IR3 flame detectors use 3 different infrared sensors to analyze and compare 3 different radiation frequencies for different infrared spectral bands. By comparing the radiation signals of the 3 sensors, the IR3 flame detector can distinguish between the radiation that is generated by flames and the radiation that comes from non-flame sources.

Because of the use of multiple infrared sensors, this sensor is the most accurate among the ones previously discussed since it has a very low rate of false alarms. The IR3 flame detector also has a very long range of around 60 meters. This is enough range to cover the specifications of the project. Due to being very accurate, reliable and having a wide range this type of sensor is also very expensive. Considering price is also one of the biggest constraints the group faced, as will be further explained in a subsequent section. With all of the advantages this kind of technology obtains, it was not worth the price that was four of five times the price of the other sensors.

Figure 26: IR3 Flame Sensor


After comparing and contrasting the benefits and drawbacks of every sensor listed in the previous collection of sensors, the group decided to settle on the infrared sensor as the one that is used in the project. The infrared sensor is very reliable and accurate enough to be used for the project. It is best used inside buildings, and it can distinguish between a variety of other phenomena that also emit infrared radiation. It also has an appropriate range for the operation of the project and is capable of detecting the required small to medium sized flames. On top of that, it has a high-speed response when detecting a flame and is affordable. Even if it may not be quite as accurate as the IR3 Sensor that was just briefly mentioned, the small difference in accuracy cannot justify the wide difference in overall price.

There are different IR sensors that could've been used to fulfill the requirements of the project. For this project a three pin IR sensor would be ideal. Among the IR sensors available, the three sensors that are shown below in the table are potential candidates to be used in the project. The three that will be decided between are the Walfront IR Flame Sensor, the Osepp FLAME-01 Sensor, and the HiLetgo IR Infrared Sensor Module which are all compared in Table 2.

Table 2: Product Details for Different Infrared Sensors

| | Walfront IR Flame Sensor Module Detector | Osepp FLAME-01 Sensor | HiLetgo IR Infrared Sensor Module |
|---|---|---|---|
| Operating Voltage | 3.3 V to 5 V | 4.75 V to 5 V | 3.3 V to 5 V |
| Detection Angle | Around 60 degrees | Around 40 degrees | Around 35 degrees |
| Wavelength | 760 nm to 1100 nm | 760 nm to 1100 nm | 760 nm to 1100 nm |
| Sensing Range | Effective sensing range between 0 cm and 80 cm | Effective sensing range between 0 cm and 100 cm | Effective sensing range between 2 cm and 30 cm |
| Compatible with Arduino Uno | Yes | Yes | Yes |
| Price | $1.138 | $6.55 | $0.879 |

Table 3: Table for the Pinout and their Description for the Sensors Above

| Pinout of the Sensors from the table above. | Description. |
|---|---|
| Digital Output (DO) | Output that comes out from the sensor and is analyzed by the microcontroller. |
| GND | Ground. |
| Vcc | Power supply of the sensor. Between 3.3 V and 5 V. |

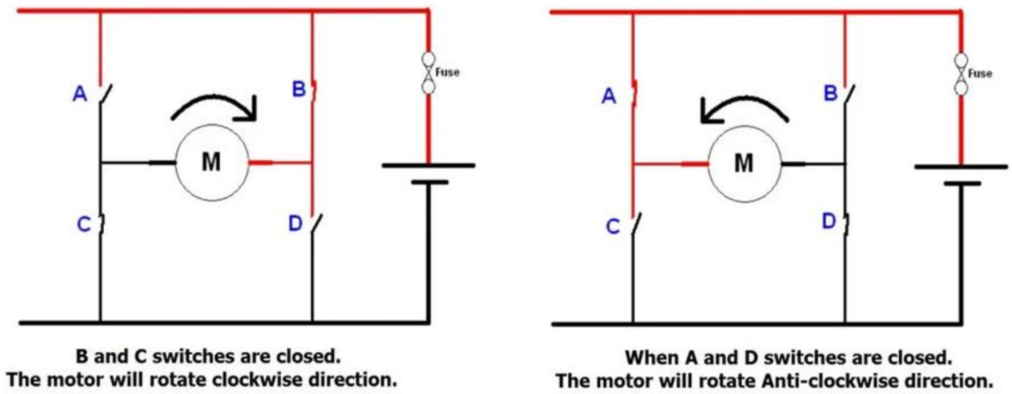
By looking at the tables above, it’s clear that the sensors analyzed have a lot of similar characteristics. The most important characteristics for the project would be the detection angle, operating voltage and sensing range. Looking at these characteristics we can see that a good choice for the infrared sensor used in the project was the Walfront IR Flame Sensor Module Detector. The reason for choosing this sensor is that it has very similar characteristics compared to the HiLetgo sensor, however the choice between them was made based on the fact that the Walfront sensor has a higher detection angle and sensing range. Because of that, when comparing both of these sensors the Walfront sensor has the edge by delivering a better performance when compared to the HiLetgo sensor. When comparing the Walfront sensor with the Osepp sensor we can see that some characteristics are different. The main differences between them are the operating voltage, the sensing range and the detection angle. The Osepp sensor can sense up to 1 meter, however it requires a higher operational voltage. Since multiple sensors would be used to achieve a higher detection angle having them all operating in the voltage range of the Osepp sensor would not be ideal considering that it only has a higher sensing range by 20 cm when compared to the sensing range of the Walfront sensor. Another difference is the fact that the Walfront sensor is much cheaper than the Osepp Sensor. Because of that, after considering the tradeoffs offered by the different sensors, the one selected for the project is the Walfront IR Flame Sensor Module Detector.

## 3.6.2 Motor Drivers

In our project, the motor is an important part of its functionality. It is responsible for the motion of the robot and because of that it is heavily related to how efficient the robot is. The motor is controlled by the Arduino Uno microcontroller. The Arduino can be considered the brain of the robot since it is responsible for monitoring the outputs from sensors and giving instructions for the motors to work accordingly. However, the microcontroller is not able to directly communicate with the motors. The reason for that is that the motors require higher voltage to operate than the output voltage that is provided by the connection from the output from the microcontroller unit. In order to solve this issue, we decided to connect a motor driver to the microcontroller that is responsible for controlling the motors for the robot to move. This motor driver then becomes the defining connective tissue between the “brain” of the robot, as we mentioned earlier of the microcontroller unit, and the “legs” of the robot, or the two DC motors that dictate the two-dimensional motion the robot is responsible for accessing and performing.

Two of the most used motor drivers in projects similar to this one are the L293D Motor Driver and the L298N Motor Driver. These motor drivers are both designed to control two DC motors at the same time, which is exactly the design of the robot, and both of them use the H-bridge setup. The H-bridge is a circuit configuration that is in both motor drivers and enables the voltage to be applied across the load through different directions. The reason this type of circuitry is needed in this project is because this circuit gives the microcontroller full control over the motor in order to determine the specific kind of motion it wishes to participate in. The flexibility that the H-bridge provides is such that it allows the robot to go in different directions such as forward, backwards and braking when needed. This kind of technology makes the design of the project much easier, and even though the motor driver is an extra part that needs to be considered, the added benefits of it as part of the robot makes it very much desirable as part of the overall design since it comes with an easy way of giving the robot the option of not only forward motion, but also on-demand braking that does not have to rely on outside forces such as friction or air resistance for a full stop.

Figure 27: Demonstration of the Operation of the H-bridge

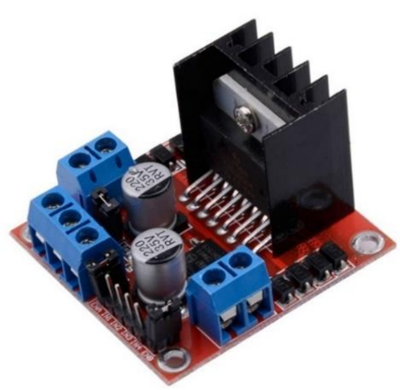



# L298N Motor Driver

One of the commonly used motor drivers is the L298N. The L298N is a Dual H-bridge motor driver. Because of that it can use both H-bridges to control a bi-polar stepper motor, or it can use one of the H-bridges to control the direction of the DC motors as specified by the microcontroller. This driver can work with high supplied voltage in the range of 4.5 V to 46 V. It also can generate up to 2 A of current that can be drawn from both channels of the driver. Due to it being able to generate an output current of up to 2 A the L298N is used with motors that require higher output current of around 2 A.

The microcontroller only needs to provide the logic signals to the driver, and it operates the motor according to what was received from the controller. The driver also has protection against the increase in temperature and a 78M05 voltage regulator. The over-temperature protector is responsible for not letting the driver get too hot due to its high current output. The 78M05 voltage regulator is used when the supply is less than 12 V, then the driver circuitry will end up being powered by the voltage regulator.

Figure 28: L298N Motor Driver



# L293D Motor Driver

The other commonly used motor driver is the L293D. Unlike the L298N driver, the L293D driver has four half H-bridges. These bridges can be operated as 2 full H-bridges as well. This gives the option of using half H-bridges or using them together if full H-bridges are required. By using the half H-bridge configuration the motors can be operated in one direction. By using the full H-bridge configuration, the motors can be operated in more than one direction by changing the current flow that goes through the motor. When compared to the L298N driver, the L293D driver has smaller input and output ranges. The supply voltage range for the L293D driver is between 4.5 V and 36 V. Another difference is in the current supplied by the driver. While the L298N can supply up to 2 A of continuous current per channel, the L293D can supply up to 600 mA of continuous current per channel. This driver also has a higher noise immunity when compared to the L298N driver.

The L293D motor driver also operates based on the logic signals that are received from the microcontroller. Once something is sensed by the microcontroller, it is processed and sent to the motor driver if the motors are required to start operating. When comparing both motor drivers, the L298N is used when a motor that requires a higher voltage to operate is necessary since it has a higher supply voltage and a higher output current than the L293D. On the other hand, the L293D is a better choice if the motors used do not require voltages that surpass its range. The reason for that is that it is a flexible option that can use the half H-bridge and full H-bridge configuration if necessary. It also has a higher noise immunity and does not dissipate as much heat when compared to the L298N motor driver. Since the fire extinguisher robot does not using any motor that requires high voltage to work, the best option to be used was the L293D motor driver since it met all the requirements for the project and it is an affordable driver. It is compatible with the microcontroller that is used for the project which is the Arduino Uno.

Figure 29: L293D Motor Driver


Just like with the other components for the project, L293D motors have different characteristics depending on the manufacturer of the component. Due to the difference in the parameters for the different parts, some of them are more applicable to the requirements of the project than others. For this project the L293D motor driver from 3 different manufacturers were considered to be included in the design. The following manufacturers had their L293D motor drivers analyzed to be included in the project: ACEIRMC, BOJACK and Texas Instruments.

Table 4: Product Details for Different L293D Motor Drivers

| | ACEIRMC L293D Motor Drivers Controllers | BOJACK L293D Motor Drivers Controllers | Texas Instruments L293D Motor Drivers Controllers | Adafruit Dual H-Bridge L293D Motor Driver |
|---|---|---|---|---|
| Supply Voltage Range | Between 7 V and 36 V | Between 4.5 V and 36 V | Between 4.5 V and 36 V | Between 4.5 V and 36 V |
| Channel Output Current | Around 600 mA | Around 600 mA | Around 600 mA | Around 600 mA |
| Peak Channel Output Current | Around 1.2 A | Around 1.2 A | Around 1.2 A | Around 1.2 A |
| Logic Level | DTL or TTL | DTL or TTL | TTL | TTL |
| Type | Dual H-Bridge | Quad Half H-Bridge | Quad Half H-Bridge | Dual H-Bridge |
| Compatible with Arduino | Yes | Yes | Yes | Yes |
| Price | $0.899 | $0.899 | $8.95 | $8.95 |

By looking at the table that has the product details for the motor drivers for the different manufactures analyzed. When comparing the ACEIRMC and the BOJACK motor drivers differ only on their type and their supply voltage range. The BOJACK motor driver offers a higher supply voltage range and the option of being a Quad Half H-Bridge that allows for different configurations to be used. When comparing the BOJACK motor driver and the Texas Instruments motor driver we can see that they also only differ in some different categories. The BOJACK motor driver has the advantage of being able to have two different logic levels when compared to the Texas instrument motor driver. The other big difference is the price of the motor drivers, the BOJACK motor driver is cheaper than the Texas Instruments component. Because of all that, the BOJACK motor driver offers the best performance and the lowest price for the project, and as such the group has decided that this is what it will go forward with.

Update: After doing testing using the BOJACK motor driver during senior design 2 we realized that we would need a different part from a different manufacturer for the motor driver. The reason for that is that the BOJACK motor drivers were not operating as expected. After some research we decided to use the Adafruit L293D motor driver. Adafruit is a very reliable manufacturer and the product (highlighted in red) has very similar specifications to the one we have been using in our designs. The only downside of using the Adafruit L293D motor driver is that it is more expensive when compared to the motor driver we were previously planning on using. However, we think that it is a worth tradeoff considering that the new motor driver is more reliable and can properly operate according to the project needs.

## 3.6.3 Microcontrollers

Microcontrollers are a very important part of many automated devices and controls that are used nowadays. Microcontrollers are small independent computers that are used to control automated devices and machines. These controllers have at least one CPU that is used as the processor of the controller, input, and output ports to receive and send out information and lastly, they also have memory to store necessary data. The microcontroller will have three different roles in the project. The first one is to sense the output the comes out from the infrared sensors and decide if a flame has been detected or not. It compares the sensor output to the values that are established in the coding of the controller to determine if the IR light sensed is coming from a flame source. The second role of the microcontroller is to control the motors of the robot. Once a flame is detected, the microcontroller controls the motor to drive the robot closer to the flame so that it gets in the range of the water pump so that it can put out the fire. The last role of the microcontroller is to control the water pump so that it is directed to the flame. Using the information from the sensors, the microcontroller will be able to correctly position the water pump and trigger it as soon as the flame is within its range. The microcontroller is a crucial part of the project since it is the one that will be responsible for putting the information from the sensors to use and making sure the other parts of the robot are working properly.

## Raspberry Pi 3 - RP 2040 Chip

One of the most used microcontrollers for projects like this one is the family of the Raspberry Pi microcontrollers. After looking through many of the different Raspberry Pi microcontrollers and their different generations, the group came to the consensus that the Raspberry Pi 3 model is the one that would be most appropriate for the project. This model is designed to be of great use to advanced applications and to have a very fast response. It is a good option to read the output from different sensors and control the different applications that the robot will be responsible for doing. The Raspberry Pi 3 can also be changed and customized to better fit the application required by the user.

This microcontroller also has a low power consumption and because of that, it also does not generate as much heat, which is perfect for a project that attempts to reduce heat as much as possible. In addition, if highly heated elements were to come in contact with the water in the pump, it could cause a very unwanted electrical reaction. It is good that it has low power consumption since it works with the power generated by a battery, and since a fire extinguishing robot has to be available at any given moment instead of turned on preemptively, which would defeat the whole purpose of the project. This is not a familiar controller, so extra research will be needed in order to use it correctly. This microcontroller uses the Broadcom BCM2837 64bit Quad Core Processor microprocessor. This microcontroller has a clock speed of 1.2 GHz and 1 GB of internal RAM. Because of that, it can work with more advanced applications than other microcontrollers. On top of this, the Raspberry Pi 3 also has the highest bit count when compared to the other MCUs on this list, at 64 bits.

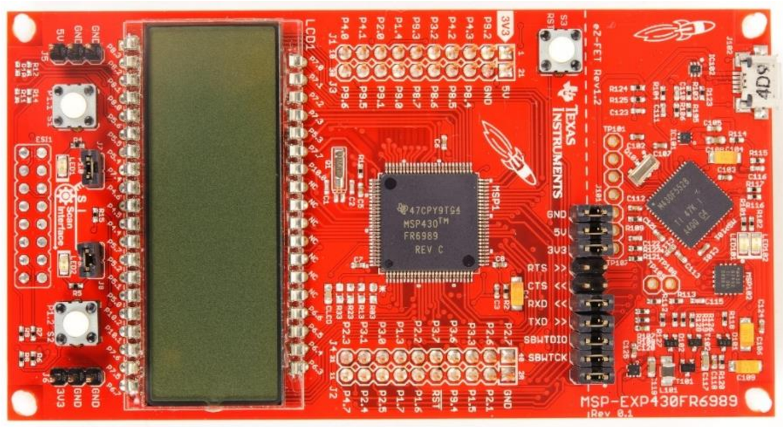
Figure 30: Raspberry Pi 3 Microcontroller


# MSP430FR6989 - MSP430

The MSP430 family of microcontrollers are another set of options for finding a microcontroller that could be used for the project. This microcontroller can be used for a variety of functions in the project. The main roles for our project like analyzing the analog output after converting them to digital values, controlling the motors and the water pump are included in the list of things that it can do. The MSP430 also has the option to use a low power mode. Staying in low power mode while no flames are detected helps to reduce the battery consumption. The transition from low power mode to active mode is around 6 milliseconds, which is negligible and won’t increase considerably the response time of the system.

A huge benefit of using this microcontroller is that it is extremely familiar since it has been used previously (specifically, the MSP430F6989 model) in different classes for various functions by all members of the group. However, it takes considerably more coding to use its functions correctly when compared to other microcontrollers. Since the MSP430 microcontroller unit is based on register level programming as opposed to the library level that is used by other microcontrollers such as the Arduino microcontroller, it gives the programmer a lot more control over the specific behaviors that are needed or desired for the project. However, this comes along with the drawback of having a much more complex method for programming. This would have meant that the programming phase of the project would have taken much more time, not only because of the amount of programming done, but also the amount of time one would need to learn about what each register does, and how to access each register. This would have involved a long time invested into the reading of the user’s manual and familiarizing oneself with the registers implemented in the design of the microcontroller. Although it is familiar to the group, this is a huge drawback that was easily replaced by a less flexible microcontroller that could still do the same quality job as the MSP430, but without the drawback of a very low level approach to the programming of the robot.

Figure 31: MSP430F6989 Microcontroller




# Arduino Uno - ATMEGA328

Another option of microcontroller that can be used for this project is the Arduino Uno. The Arduino Uno microcontroller can be used in the same ways as the previous microcontrollers mentioned. It can be powered with a battery as required for this project and has low power consumption. When compared to the previous microcontrollers, the Arduino Uno is less efficient and less powerful than the others. However, it is much simpler and user friendly to setup and program when compared to the other microcontrollers. This microcontroller uses the Atmel ATmega328 microcontroller chip. With this chip, it can go up to 20 MHz of clock frequency, which is even higher than the frequency of the MSP430 board. Programming the different roles required for the project like getting the output from the sensor and quickly deciding the actions to take are much simpler to be done using the Arduino Uno. It also has low power consumption since it only requires an operating voltage of 5 V. The main disadvantage of choosing the Arduino Uno as the microcontroller is the not so familiar programming language of the controller. The programmer has worked with this microcontroller before, but not as much as the C programming language that could be used for the other microcontrollers.

By analyzing the characteristics of the different microcontrollers, it was decided which one would be used for the project. The Arduino Uno microcontroller was chosen mainly for its ease of programming when compared to the other microcontrollers. After discussing about the pros and cons of the controllers, a conclusion was made that even without as many experiences on the Arduino programming language, it was better to research it and use it instead of using the other more complicated and harder to setup microcontrollers researched. And in terms of performance, the Arduino Uno was able to function just as well as the other microcontrollers would since the project does not require the extra processing power that would be gained by selecting the Raspberry Pi 3 or the MSP430 microcontroller. The ATMEGA328 is also a great option for our project because it is really flexible. It can be easily setup with some capacitors, resistors and crystal to function like an Arduino board in our final PCB. This chip can also be programmed on the PCB which allows for adjustments after the entire PCB is assembled.

Figure 32: Arduino Uno Microcontroller

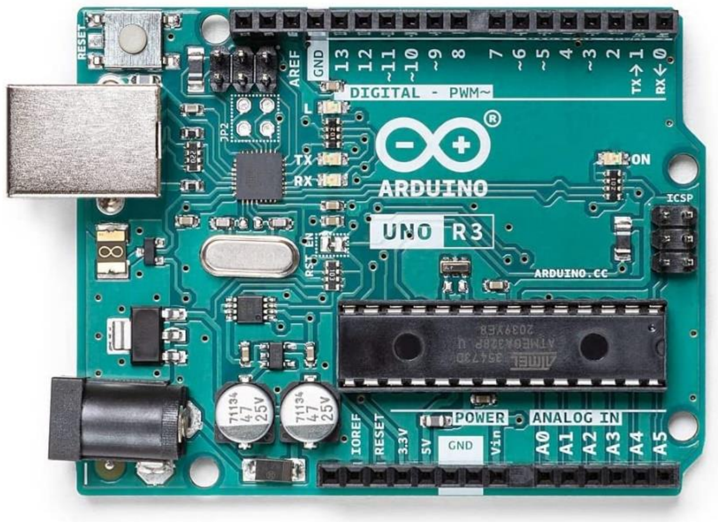



Table 5: Comparison of the Microcontrollers

| **Characteristics** | **Raspberry Pi 3 - RP 2040 Chip** | **MSP430 - MSP430** | **Arduino Uno - ATMEGA328** |
|---|---|---|---|
| Operating Voltage | 5 V | 1.8 V to 3.6 V | 5 V |
| Maximum Total Current Drawn | 54mA | 6 mA | 20 mA |
| Internal RAM | 1GB DDR2 | 66 KB | 2 KB SRAM |
| Clock Frequency | 1.2GHz | 16 MHz | 20 MHz |
| Operating Temperature | -40ºC to +85ºC | -55°C to +95°C | 40°C to 85°C |
| GPIO Pin Count | 40-pins | 24-pins | 28-pins |
| Power Consumption/Low Power | 3.7 W<br>Low Power MCU | 330 µA at 1MHz<br>Low Power MCU | 0.3 W<br>Low Power MCU |
| Bit Count | 64-bit | 16-bit | 8-bit |
| Price | $35 | $24 | $19.75 |

# 3.6.4 Types of Pumps

Earlier in the document, there were many different kinds of water extinguishers that were discussed and thoroughly examined. For the sake of being thorough in our exploration of fire extinguishing, all of the six kinds of extinguishers and their technologies were considered as part of the selection process of the hardware components and background research on the theme of extinguishing fires. In order to effectively select what technology would be used in this project, the different technologies were weighed in terms of how frequent different kinds of fires would be extinguished with the robot. Also, considering the first prototype of the robot, the conclusion came out to be that the best first step in choosing a type of extinguishing technology would be to first try to deal with water extinguishing technology as opposed to any of the other more advanced types of fire extinguisher technologies such as the powder or carbon based sprays.

With that being chosen, the group then turned to selecting how the water based fire extinguisher would be mounted on the robot, and after some deliberation, it was determined that using a full fire extinguisher was less ideal for the prototype of the robot, and so the attention was turned away from fire extinguishers and towards more easily accessible and easily manipulatable water pumps. In the following section there will be a high coverage on the background information of water pumps and the different kinds that the group considered before making a final choice on which kind of water pump would be selected for the project.

## Water Pumps

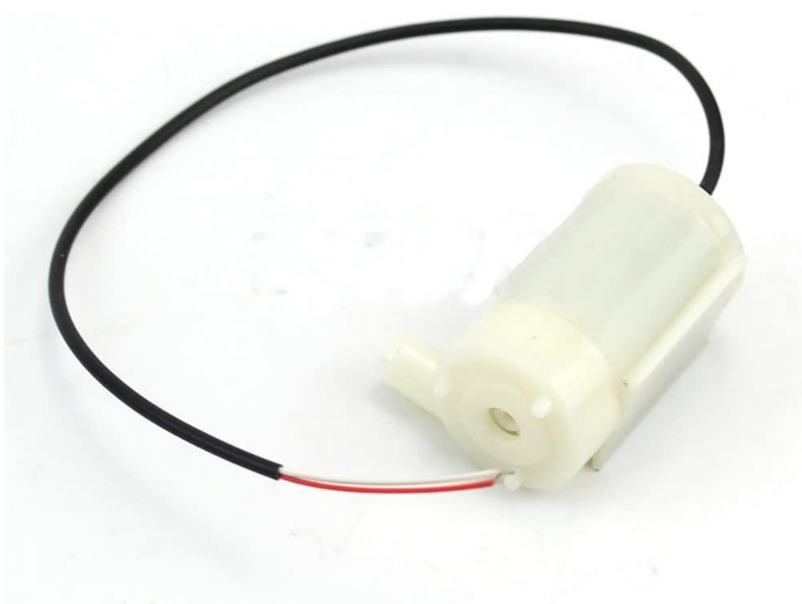
The water pump is an important part of the project and design of the robot. For our project, the pump needs to be able to pump a certain amount of water to be able to put out the fire and work in adverse conditions. The biggest factors in this capacity would be not only quantity of water, but also accurate aim in pumping the water, but also, to an extent, the velocity and pressure that the pump is able to produce. There are different types of water pumps that can be used depending on the specifications and requirements of the project. The two main types are based on the power supply of the pumps. One of the types is DC powered pumps and the other type is AC powered pumps. There are differences between these types of pumps and what they can offer. When compared to the AC water pumps, the DC pumps are smaller and more portable. They also have similar control and operation speeds. The DC pumps have a higher energy efficiency and can be connected to a battery to work as a power supply for the pump. On the other hand, the AC pumps have longer lifespans due to not using a battery and have higher power when compared to the DC powered pumps. For the project the DC pump is the best suited option between these two power supplies, since the focus of the robot in its early stages will be that of getting a functioning robot, and in the future it could be potentially altered to something more powerful. Also, it’s important to note that the firefighting robot uses a battery that can also be used to power the water pump.

Among the DC powered pumps, there are some different types that are better suited to different applications. The submersible pump and the peristaltic pump are examples of the different categories of pumps available. The best suited one based on the requirements of the project is the submersible pump. One of the reasons for choosing the submersible pump is that they are resistant against pump cavitation. This problem is caused by high pressure environments that can lead to damage to the pump. Due to the robot working close to fire it is preferable to have a pump that is more resistant against high pressure environments. As was stated earlier, the resistance to changing pressures is considered of utmost importance for this project since fires tend to affect the overall pressure of the environment, and just as the sensors were chosen under this condition, the pump will follow the same criteria. This type of pump also best fits the design of the robot since it is able to be located inside the water container reducing the overall size of the robot.

# Dicrey Micro Water Pump Mini Submersible Pump

The Dicrey micro water pump is one of the pumps that satisfy the requirements of the design for the project. The Dicrey pump works with a 5 V power supply and 100 mA of current. These values can be achieved in accordance with the other parts of the project. It is a small pump and weighs around 30 grams. Because of that it does not compromise the robot by being too big or too heavy to be carried on the chassis of the robot. Other important aspects of the pump to consider are the max height of the water and the max flow rate of the pump. The max height for the water with this pump is around 55 cm, this value is good enough for its application in firefighting robot. And the max flow rate of the pump is around 100 L/H. This is also good value for the flow rate of the pump. However, one of the down sides of this pump is that it has a continuous work life is around 500 hours, which is not appropriate for its application.

Figure 33: Dicrey Micro Water Pump Mini Submersible Pump



# PULACO DC 5V USB Mini Submersible Water Pump

The other pump that could be used for the project is the PULACO mini submersible water pump. This pump is a good upgrade from the one previously mentioned. It also works with a 5 V DC power supply and has a better performance than the previously mentioned pump. This pump has a flowrate of around 190 L/H which is almost double the flow rate from the Dicrey Micro pump. The PULACO pump has a lift height for the water to be around 61 cm, which is a slight increase over the previous pump. The continuous work life for this pump is also majorly increased when compared to the continuous work life of the Dicrey pump. The continuous work life for the PULACO pump is around 20000 hours, which is more than enough for the application of the robot. The only tradeoff when compared to the Dicrey pump is that the PULACO pump weighs around 127 grams, which is a big increase. However, it is still an appropriate weight and it is worth using the PULACO pump instead of the Dicrey pump due to all the higher performance it has on a lot of its features.

Figure 34: PULACO DC 5V USB Mini Submersible Water Pump

Just like for the previous components, similar components with different manufacturers have different characteristics and parameters. The various different submersible water pumps available in the market can fit better depending on the application based on their advantages and disadvantages. The best fit for the project was a submersible water pump that is immune to pump cavitation and can work of a DC power supply. The 3 submersible water pump manufacturers that were considered for the project were PULACO, Heneng and MONSUP.

Table 6: Product Details for Different Submersible Water Pumps

| | PULACO Mini Submersible Water Pump | Heneng Mini Submersible Water Pump | MONSUP Mini Submersible Water Pump | ALAMSCN 5V Micro Submersible Mini Water Pump |
|---|---|---|---|---|
| Operating Voltage | DC 5 V | DC 5 V | DC 5 V | DC 5V |
| Maximum Flow Rate | 189 L/H | 97 L/H | 200 L/H | 100 L/H |
| Maximum Lifting Height | 61 cm | 91.44 cm | Between 40 cm and 150 cm | Between 40 cm and 110 cm |
| Power | 3 W | 1 W | Between 1 W and 3W | 1 W |
| Connector | USB | USB | USB | Positive and negative wires |
| Price | $11.99 | $12.99 | $10.99 | $2.85 |

The table above contains the comparisons of different submersible water pumps from different manufacturers and their characteristics. When comparing all the water pumps in the table it is visible that they are very similar in some respects and vary in others. The PULACO and the Heneng water pumps are only similar in their operating voltage and their type of connector. The PULACO water pump has a better flow rate, has smaller maximum lifting distance and is a little cheaper than the Heneng water pump. Considering these differences, the PULACO water pump is the best choice for the project since it is worth sacrificing some lifting distance and consuming more power in order to achieve almost double the flow rate of the Heneng water pump. When comparing the PULACO and the MONSUP water pumps we can see that they are very similar in their operating voltage, flow rate and type of connector. The MONSUP water pump has similar values for the power that are related to the lifting height of the pump. This pump is also cheaper than the PULACO water pump. However, the MONSUP water pump is out of stock and is less consistent than the PULACO water pump. To conclude, the PULACO submersible water pump is the best choice for the design of the project considering the options analyzed.

Update: After conducting different tests during Senior Design 2, we realized we would need a pump with a different connector in order to properly connect it to the PCB without having to add a USB port to the PCB. Because of that we decided to go with the ALAMSCN 5V Micro Submersible Mini Water Pump. This pump has similar specifications when compared to the options that were previously considered during Senior Design 1 and it has the correct connector wires for us to attach it to our PCB. It's also less expensive than the previously considered pump options, which helps to reduce the overall final cost of the design.

## 3.6.5 Power Supply Selection

The power supplies for a project are one of the most important parts of the design. The power supply is what is responsible for giving power to the entire system and making sure every part has its needed amount of voltage to operate properly as intended. The power supply is a part that needs to be accurately designed in order not to compromise the system. If a power supply does not deliver enough power, then some of the parts in the system are not able to operate as intended with full capacity. With that said, it is clear that the power supply is one of the most important components that needs to be considered in terms of making an ideal choice, and not just one that works under certain conditions, but rather all conditions.

As said before, the battery needs to be powerful enough in order to power the entire robot and meet all of its voltage requirements, but on the other hand, if it delivers much more power than what is accepted by the different parts it will end up damaging the parts through burn out, causing the system not to work properly anymore and causing temporary, or even permanent, damage to the entire system, leading to a need to rebuild. There are a lot of different ways to supply power to the components of the project. Some of the different ways to supply power for the design is going to be discussed in this section.

## USB Connection

One of the various ways to power a project is to use a direct USB connection to another device that could supply the required voltage. Using this method, the Arduino Uno microcontroller is usually connected to a laptop or computer. The device that is connected to the Arduino Uno is the one that is responsible for supplying power to the microcontroller. This is a good strategy to use when the Arduino Uno microcontroller is being tested. For the testing of the components that is a good strategy to use. The reason for that is because during the test of components a computer is usually available and used so that is a good option for a power supply for the Arduino Uno microcontroller. However, as convenient as this is, this option does not apply to the design of the project. One of the reasons for that is the fact that it is used only by the Arduino Uno microcontroller, and it doesn't work for other components of the design, such as the L293D motor driver, that also require a power supply, and so the USB connection would be useless. If not all components can share the same power source, it would be unwise and inefficient to use it, since having multiple power sources is irresponsible and more costly in terms of economic and other kinds of resources that are currently limited by our different constraints. Another reason is that it is not practical to depend on a computer or a laptop to power up the microcontroller considering their size and weight, and especially since the requirements for the project outlined earlier have already specified that the robot needs to be under a certain level of weight to maintain ease of transportation. This essentially brings the USB connection to a low level of priority when having to determine what kind of power source will be used going further with the project.

Figure 35: USB Micro-B Cable

# Variable DC Benchtop Power Supply

The Variable DC Benchtop Power Supply is another option for powering the system of the project. For this method, a multi-range DC power supply is required for providing the power to the system. These power supplies usually have multiple channels that can be used to provide a voltage supply for more than one component. It could be used to supply the voltage for the Arduino Uno microcontroller and for the L293D motor driver. However, just like the last option it is not practical to use due to its size and weight. Another problem with this option is the fact that it depends on a plug to a wall power output for it to work.

Figure 36: Power Supply-Multi-Range DC

# AC to DC Wall Adapters

The AC to DC wall adapters is another option for supplying power to a system. They are usually plugged in the wall to supply power to the system. These adapters are able to supply power to the Arduino Uno microcontroller. However, just like the USB connection alternative, the AC to DC adapters can only supply power to the microcontroller and not the L293D motor driver. Another aspect of that goes against the use of the AC to DC adapters is the fact that it has to be connected to the wall at all times for the system to be supplied with power. Due to the fact that the robot is expected to move around on its own with no restrictions, the AC to DC adapters is not ideal since they restrict the movement of the robot and that goes against the requirements for the project.

One of the main considerations of the project in terms of motion is a free range of two dimensional motion, and even if there was the possibility of acquiring a sufficiently long cable for the robot to move throughout a given space, one would still run into the problem of the robot becoming tangled in its own power wire due to how unpredictable the movement of the robot could be, leading it to potentially tangle itself by its constant change of motion throughout the given space. With that said, it becomes clear that providing wired power to the robot is a less than ideal solution, and so it becomes necessary to turn to a different kind of power production. Namely, this kind of power would have to be wireless, and as such we turn to exploring different kinds of wireless power production, such as different kinds of batteries.

Figure 37: Wall Adapter Power Supply

# Batteries

As said earlier, batteries are another good option to supply power to the entire system required by the project’s design. With the right battery and voltage, it is possible to supply voltage to the entire system with a single battery. Batteries are reliable for supplying power for certain applications and they are used in a variety of occasions. The batteries allow the system to have increased mobility due to its operating on its own and not requiring any connections to outside places. By using a voltage regulator, it is possible to regulate the voltage that comes out of the battery. With that, it is possible to supply power to the entire system with the correct voltage for the chosen battery. Another positive aspect of using the battery with the correct voltage value for the system is that it is possible to reduce the power dissipation of the system but correctly applying the right value to the motor. Booster converters and switching regulators can also be used if there is a need to get a specific voltage requirement or a little more current for the entire system.

Figure 38: Rechargeable Battery


After considering all of the different options for a power supply for the system it was decided that a battery is the best option for the project. It meets the requirements for the of the project and because of that is the best option for supplying power for the system. However, there are a lot of different battery types because of the different materials that are used during the confection of the product. Some of the most common battery types used will be described in be commented on below to help choose the best one for the project. The batteries that are going to be discussed are the Lithium-Ion batteries (Li-Ion), the Lithium-Ion Polymer batteries (Li-Po), the Nickel-Metal-Hydride batteries (NiMH) and the Nickel-Cadmium batteries (NiCd).

# Lithium-Ion battery (Li-Ion)

The lithium-Ion battery is a product that has been in research discussion for a long time. This type of battery dates back to the 1960s and was first commercialized by Sony in the early 1990s. The Lithium-Ion battery is a rechargeable battery that works based on its lithium ions. The ions move between negative and positive electrodes of the battery through a liquid electrolyte material. Some of the characteristics of this battery is that it has low self-energy discharge, high energy density and no memory effect. Due to it having no memory effect, the problem of the battery having to be discharged to not compromise the battery life of the product is no longer a concern.

This type of battery is commonly used in diverse electronics applications such as portable electronics such as cellphones. Due to being used in a lot of portable electronics such as cellphones and laptops, these batteries are small and could be used in the project without any space or weight problem. At first glance, this becomes an extremely convenient option for the project because not only do they provide a lightweight option for a battery, but they have also been previously used to power lightweight devices and electronics, which is something that the group is in need of when deciding on a type of battery to power the lightweight robot. These batteries also could have high current ratings of around 30 amperes depending on the manufacturer of the product. However, because of these reasons Lithium based batteries are more expensive than the other batteries.

Figure 39: Lithium-Ion Battery (Li-Ion)


# Lithium-Ion Polymer battery (Li-Po)

The Lithium-Ion Polymer battery is a rechargeable battery that is also based on lithium ions. The main difference between the normal lithium-ion battery and the lithium-ion polymer battery is the material that is used in between the positive and negative electrodes of the battery. The lithium-ion polymer battery uses a polymer electrolyte instead of the liquid electrolyte used in the regular lithium-ion battery. The polymer electrolytes are composed of high conductivity semisolid polymers. Because of that these batteries also have higher energy density when compared to the other types of lithium-ion technology batteries.

These batteries are also commonly used in various portable electronics due to their size and capacity. Another common application for them is in different devices and applications where weight is a critical factor in the design constraints. Some examples are radio-controlled aircraft and electric vehicles. This type of battery was created after lithium-ion batteries started being commercialized by Sony in the early 1990s. After that, other package forms were developed, including the Lithium-Ion Polymer battery that has a flat pouch format.

Just like the Lithium-Ion battery, the Lithium-Ion Polymer battery has a lot of positive parts such as no having a memory effect and being lighter than other batteries. However, it also suffers from the same downside of being more expensive when compared to other batteries on the market.

Figure 40: Lithium-Ion Polymer Battery (Li-Po)

# Nickel-Cadmium Batteries (NiCd)

The Nickel-Cadmium battery is a rechargeable battery that uses 2 different types of electrodes to generate energy. The materials that are used as electrodes are nickel oxide hydroxide and metallic cadmium. These batteries were first developed in 1899 at Sweden. The lead-acid battery was the main competitor of the Nickel-Cadmium battery at the time. And over time the Nickel-Cadmium batteries started to be more used than their main competitor. The Nickel-Cadmium batteries are cheaper when compared to the lithium-ion batteries previously discussed. These batteries provide small voltage and current and are a good option to be used for small electronics.

Due to it being a small voltage battery, multiple battery cells would have to be connected in parallel to provide enough power for the system to fully function. By using multiple cells in a parallel configuration, it causes a space and constraint problem that could affect the main robot design. These batteries also contain the memory effect. Due to this effect, the battery life of the cell can be compromised if not fully discharged before recharging. Because of these reasons the Nickel-Cadmium battery is not the ideal battery to be used in the project.

Figure 41: Nickel-Cadmium Batteries (NiCd)


# Nickel-Metal-Hydride Batteries (NiMH)

The Nickel-Metal-Hydride battery is a rechargeable battery with an operation similar to the Nickel-Cadmium battery. On the positive electrode, the chemical reaction is still using nickel oxide hydroxide. However, on the negative electrode, instead of using the cadmium element it implements a hydrogen absorbing alloy. With this change, the Nickel-Metal-

Hydride batteries can have as much as three times the total capacity of the Nickel-Cadmium batteries. It also contains a significantly higher energy density than the previously discussed Nickel battery. On the other hand, when compared to the Lithium-Ion batteries, the NiHM battery still has a smaller energy density. These batteries are usually applied to substitute similar sized alkaline batteries. The reason for that is that they generally offer more safety by being much less susceptible to leakage and explosion.

The market and search for Nickel-Metal-Hydride batteries has been declining since lithium-ion batteries started to become more prevalent in the industry. These batteries are commonly used in small electronics applications such as remote-controlled devices. Just like the previous nickel-based battery, this one can only supply the required power levels when multiple cells are connected in series. Because of that, it is troublesome to fit all the required battery cells in the design due to their size and weight being combined. Even though the Nickel-Metal-Hydride batteries are a better option than the Nickel-Cadmium batteries they are still an inferior option when compared to the lithium-ion batteries.

Figure 42: Nickel-Metal-Hydride Batteries (NiMH)


Table 7: Components and Their Voltage and Current Requirements

| Component | Number of Components used | Voltage (V) | Current (mA) |
|---|---|---|---|
| L293D Motor Driver | 1 | 12 | 600 |
| Motor | 2 | 12 | Around 10000 |
| Sensor | 3 | 5 | 30 |
| Arduino Uno Microcontroller | 1 | 5 | 1000 |
| Submersible Water Pump | 1 | 5 | 220 |

Table 8: Properties and Characteristics for Different Batteries

| | Tenergy 12 V Rechargeable Battery Pack | TalentCell Rechargeable 12V Battery | Turnigy 12 V Battery | Anker PowerCore 5 V Power Bank |
|---|---|---|---|---|
| Voltage | 12 V | 12 V | 12 V | 5 V |
| Capacity | 2000 mAh | 3000 mAh | 5000 mAh | 10000mAh |
| Discharge | 1 C | 3 C | 20 C | 15 C |
| Dimensions | 119933.63 $mm^3$ | 160317 $mm^3$ | 167739 $mm^3$ | 10.59 x 5.21 x 2.49 cm |
| Weight | 255 g | 190 g | 360 g | 192 g |
| Material | NiMH | Lithium-Ion | Lithium-Ion Polymer | Lithium-Ion |
| Compatible with project | Yes | Yes | Yes | Yes |
| Price | $21.98 | $28.79 | $24.99 | $41.99 |

When looking at the different options of battery researched it is noticeable that one of the batteries has a better performance when compared to the others by looking at their comparison table. When comparing the characteristics for the Tenergy Rechargeable Battery and the Turnigy Battery we can see that the first option has a smaller size, weight and is cheaper which is preferable. However, the Turnigy Battery has a much higher discharge and capacity ratio. Because of that the tradeoff of having to deal with a heavier and bigger part in order to get a satisfactory power supply is preferable. TheTurnigy Battery is also made using the lithium-Ion Polymer material which is preferable for the project requirements. When comparing the Turnigy Battery and the TalentCell Rechargeable Battery, just like the previous comparison, it is worth to tradeoff of dealing with the extra weight and size of the Turnigy Battery if we consider that it is supplies the necessary power for the system. The Turnigy Battery is also cheaper than the TalentCell Rechargeable

Battery. Because of the reasons listed above, the Turnigy Battery can be considered the best option for the design of the project.

Update: After further testing that was done during Senior Design 2, we realized that there was no need to utilize a 12 V battery for our project. We tested all the components and built a prototype using a 5 V power supply and that was enough for the whole project to work properly. Because of that we looked for a reliable 5 V power supply for our project and the Anker PowerCore 5 V Power Bank is the best option we found. It is very compact and weighs less than the previously considered options and it has a big capacity. It also is rechargeable and has a big battery life for our project. The only downside of this option is that it is more expensive than the previously considered options, but we agreed that it is worth the extra money to have a more reliable and fitting power supply for our project.

# 4. Design Standards and Realistic Design Constraints

This chapter will deal mainly with standards and constraints for the project, and as such will only contain two subheadings, with multiple sub-subheadings for each. The standards section will have 6 sub-subsections while the constraints section has more sub-subheadings at nine different constraints.

Standards related to the design of the robot and constraints related to the project are important factors that need to be covered. The standards are related to each individual's technology and software. Each standard details the implementation of the referred technology and how it is supposed to be used in the project. Following the standard specifications and procedures is what facilitates the integration between different designs from different businesses. Different aspects of the design will have their own standard that helps with its integration and implementation. The Arduino uno that is used for the project has its own standards that can be referenced to help and facilitate the design of how to best utilize this device.

The different kinds of standards considered will be:

1. Safety for AMR
2. PCB Design
3. Battery
4. USB
5. Fire Safety
6. C Language

The constraints are also important factors that directly affect the design and decision process of the project. Constraints are various factors that for different reasons limit the development of the design. These constraints could vary from economic factors to social and political factors.

The different kinds of constraints considered will be:

1. Economic
2. Time
3. Environmental
4. Health
5. Sustainability
6. Covid-19
7. Manufacturability
8. Safety
9. Social, Political, and Ethical

## 4.1 Standards

As mentioned in the previous section, standards are one of the most important parts to be considered when designing a project. There are an immense number of different standards that have already been created and should be followed when doing a design. All of the standards are evaluated and analyzed by different organizations that will later on qualify it as a valid standard or an invalid standard. One important organization that deals with standards is the American National Standards Institute (ANSI). This non-profit organization does not develop the standards that are used in the industry. Other groups and companies are the ones that are responsible for the development of different standards related to their products. The ANSI is responsible for developing and overseeing the requirements that are requested of any group or company that wants to develop their standards. The American National Standards Institute is responsible to oversee and give credit to the various companies and organizations that follow the requirements established for developing standards by the ANSI.

Another important organization when that needs to be mentioned when talking about standards is the Institute of Electrical and Electronics Engineers (IEEE). The IEEE is a renowned organization in the electronics world. They are a non-profit organization that is responsible for having developed and published over 900 different standards that are related to electronics and their applications. Because of that, some of the standards that have been published by IEEE will be responsible for some of the standards that are related to the design of the project. The main objective of most of the standards is to help, facilitate and promote the national market to better integrate itself with the different markets around the world that follow similar standards and therefore can easily integrate different products. There are two other standard organizations that develop standards that are related to the design of the project. The first one is the National Fire Protection Association (NFPA). This organization develops standards related to fire, electrical applications and user safety. Considering that the project deals with fire, electrical components and safety of the user the standards of the National Fire Protection Association can be considered for the project. The other organization that produces related standards to the project is the Association Connecting Electronics Industries (IPC). This organization is responsible for standards related to the assembly of electronic components. Its standards cover topics such as design, printed circuit boards (PCBs) Manufacturing and electronic components assembly. Considering the nature of the project, some IPC standards will also apply to the design of the project.

The American National Standards Institute (ANSI) is one of the members of the International Organization for Standards (ISO). The ISO is an independent organization that is responsible for being the biggest developer of international standards in the entire world. The main objective of this organization is to facilitate trading and commerce among its member countries by setting international standards that are common among its member nations. Because of that, the standards that were followed and agreed upon during the design of this project are all related to the standards that have been developed by the members of the International Organization for Standards.

For the design of the project, a good number of standards from the different organizations mentioned above will be used in order to standardize the design. The electronic related standards were used in the design to achieve a final project design that is in accordance with all of the related standards possible. The remaining standards that are not related to electronic components will also be followed during the design to make sure the electronic section can be well integrated with the safety and fire dealing standards.

### 4.1.1 Safety Standard for AMR (Autonomous Mobile Robots)

The R15.08 defines the safety standards for Industrial Mobile Robots (IMR). With the new advancements in technology and the versatility of these robots, new safety requirements have arisen. It was crucial to have these standards set in place as there are risks to these vehicles and their systems.

The Robotic Industrial Association (RIA) issued R15.08-20 part I, which discusses the safety standards for industrial mobile robots and their attachments (i.e. wheels, manipulators). Part I is for the manufacturer to examine and understand how to design and manufacture a safe IMR. With each new robot, a relevant Risk Assessment analysis must be met.

Part II of the document is still in the works. This section will specify the safety requirements for mobile robots installed in particular conditions. To go more in depth, this part will include the appropriate speed of the robot depending on the presence of safety sensors and escape clearance for operators. "With Part 1 you will have a safe robot, with Part 2 you will have a safe robot professionally installed in a given working environment." Part 3 will eventually describe the user responsibilities for safe operation of the mobile robots and their systems.

Due to the cost of the official document, we were only able to read a summary of the main points discussed. Listed below are the most relevant elements of the R15.08.

1. Application of the IMR

   R15.08 is applicable only to robots in the workplace and the people who work around the robots. The application must be industrial, meaning factories, warehouses, laboratories, etc. R15.08 does not cover AMRs intended for consumer or household use, military use, medical use or rehabilitative use. This standard is not applicable in environments with untrained public as the actions of the robot are not predictable.

2. Mobility

   Mobility refers to ground-based systems that are able to navigate autonomously. Systems that are not included in the R15.08 scope include airborne, waterborne, and rail systems.

3. Navigation

The navigation of the robot could be autonomous, automated, or both. The industrial robots can navigate to meet a specific destination by any method stated previously. An AMR navigates autonomously by either following a guided path or deciding its own path. The sensors on the AMR will detect any obstacles in its path and maneuver its way around them.

4. Inclusion of a Manipulator

An Industrial Mobile Robot may include a manipulator but it is not mandatory.

5. Presence of Addition of Other Attachments

An Industrial Mobile Robot may include an attachment but it is not mandatory.

The R15.08 identifies three different types of IMRs: Type A, Type B, and Type C. Type A and Type B have an AMR as a mobile platform, while Type C might use an AMR mobile platform or an AGV (Automated Guided Vehicle) instead. A diagram is presented in the R15.08 document which outlines the different types of IMRs.

An IMR Type A is the most basic AMR without any attachment. Type B is an AMR Type A plus an active or passive attachment. Attachments include conveyors, roller tables, lifting devices, etc. Type C is an AMR or AGV base with a robotic manipulator. Robots that are out of the scope of the R15.08 include robots that are able to transport riders, automated/autonomous forklifts, and robots that are not intended for industrial applications.

Another important section included in this document is a hazard identification and risk assessment chapter. It lists hazardous materials that may arise in an IMR and considers various circumstances that could occur. Some of the key points highlighted in this chapter are listed below.

1. Perform a hazard analysis to identify potential hazards
2. Perform a risk assessment to eliminate or reduce those hazards

One of the more important sections in the document includes the design requirements and risk reduction measures. Depending on the hazard identification and risk assessment result, different measures need to be implemented to have a safe IMR.

Requirements considered when designing an IMR:

- Conformity with other standards
- Modes of operation – Automatic, Semi-automatic, manual mode, maintenance mode
- Portable control unit
- Control functions – Restart functions, emergency stop function, protective stop function

- Navigation and control – Collision avoidance, velocity limiting
- Presence-sensing devices – Non-contact presence-sensing devices, contact presence-sensing devices, suspended or variable presence-sensing, safety distances

A complete list of the various design requirements are listed in the R15.08 document.

Another chapter included in the document is about verification and validation of risk reduction measures. The manufacture and manufacturer must comply with the specs in order to minimize risks. Practical tests, measurements, review of schematics, circuit diagrams and other measures are taken in order to figure out the exact verifications associated with the IMR.

Lastly, the information for use and markings section of the R15.08 document contains the instructions necessary to ensure safe and correct use of the IMR. It provides information and warnings to the user about any risks associated. All the documentation to ensure the safety standards are met are outlined in this chapter as well as how the operating environment is set up.

There is a lot of information in the R15.08 document that is necessary to comply with. This set of standards is strictly mandatory when designing a robot in order to ensure a safe vehicle is designed.

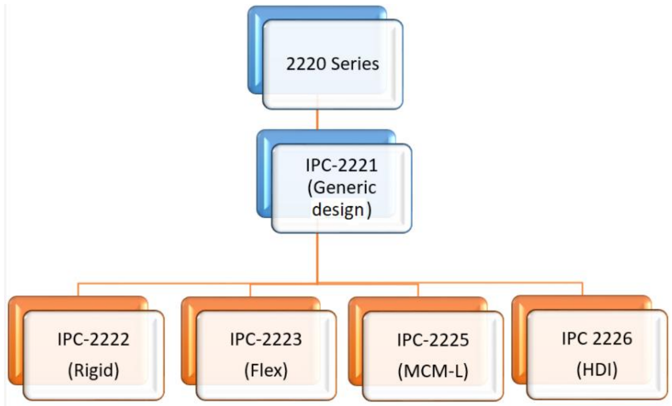
### 4.1.2 PCB Design Standard

IPC-2221 is a generic standard for any given circuit board design and lays down the requirements for a PCB design. Also, it provides a list of various forms of component mounting/interconnection structures. The IPC-2221 states the basic design standards and principles for the circuit boards established and also any recommendations that can be given for the team building circuitry.

When IPC was founded, it stood for Institute of Printed Circuits. Its name was then subsequently changed to Institute for Interconnecting and Packaging Electronic Circuits, although the name is still declared IPC to this day. IPC standards are established for the electronic manufacturing industry and are issued by the trade association. IPC sets the standards for the foundation of the design, assembly, packaging, interconnection, material, performance, and inspection specifications for the electronic industry and are followed throughout the world.

The IPC-2221 standard belongs to a family of documents under the IPC-2220. Established in these documents are a conglomeration of different components of a PCB design. These include not only the generic printed circuit board design, but also the different components, the mounting, and the interconnections present in the circuitry. A document set is identified with a four-digit number that ends with the number zero. In order to clarify the structure of the series, the hierarchy of the 2220 series is shown below in the following figure.

Figure 43: Hierarchy of 2220 Series




Based on the figure above, it can be determined that for each type of board there are specific standards. To design rigid boards, for example, the IPC-2221 should be referred to as well as the IPC-2222. All generic standards established by the IPC-2221 should be used in conjunction with the specific standards and requirements for the type of circuit board.

The IPC-2221 focuses on a variety of aspects including the board's reliability, manufacturing difficulties, and costs. The PCB designer should balance the required electrical, mechanical, and thermal performances.

Due to the costs of the official document, a sample version was obtained in order to examine the key points discussed in the IPC-2221.

1. Clearance in Board Design

The distance between two conductors or nodes, measured in air, is known as the clearance distance. The IPC-2221 refers to many different clearances related to different aspects of the circuit board. Below are a few of the important ones mentioned in this standard.

Figure 44: IPC-2221 Standards

| Surface | finishes | for | HDI | PCBs | | |
|---|---|---|---|---|---|---|
| HASL | (vertical | and | horizontal) | | | |
| Lead-free | HASL | | | | | |
| OSP | (Shikoku | F2 | and | Entek) | | |
| ENIG | (electroless | nickel | immersion | gold) | | |
| ENEPIG | (electroless | nickel | electroless | palladium | immersion | gold) |
| Immersion | silver | | | | | |
| Tin | nickel | | | | | |
| Electrolytic | soft | gold | | | | |
| Electrolytic | hard | gold | | | | |

2. Creepage in a PCB

Creepage refers to the distance between conductors or nodes along the surface of an insulator. The space between the conductors should always be maximized and optimized to the extent. Conductor spacing is allotted so that there is enough space for the etch compensation of the other physical features. The etch compensation should be twice the etched copper thickness. Another consideration that should be made is the conduction imperfections and copper wicking between PTHs and adjacent plane layers.

3. Thickness of the Traces

To carry a specific amount of current, a PCB trace needs to have an appropriate thickness. If it is less than what is required, the trace will burn while the current passes through it. The thickness and width of the traces will depend on the signal characteristics, current carrying capacity, and the maximum allowable temperature. The thickness will also change based off the construction requirements of the board.

The trace width for allowable current can be calculated by:

Width (mils) = Area (mils^2) / (Thickness (oz) * 1.378 (mils/oz))

The cross-sectional area, A is calculated by:

A = ( I / (k * (deltaT^.44))^(1/.725)

I is the maximum current in Amps, k is a constant, deltaT is the temperature rise above ambient in Celsius, and A is the cross-sectional area of the trace in mils^2.

For internal layers, k = .024 and the conductor thickness equals the copper foil thickness of the base laminate. If blind and buried vias are implemented then the conductor thickness equals the copper foil thickness that includes the copper plating. For external layers, k = .048 and the conductor thickness equals the thickness of the base foil and the plated copper of PTH without including the thickness of the solder coating, tin-lead plating, or secondary plating's.

Insulation is a basic requirement in a PCB for protection against short circuits between conductors. This happens by accidental contact, overheating caused by electric conduction, and corrosion or other environmental damages. A lot of materials can be used to insulate the board, depending on the board application. The IPC-2221 suggests some insulation resistance tests and individual test coupons listed below.

Insulation tests:

1. Insulation resistance test

   "In this test, a voltage is applied across the PCB which induces current flow. This current is measured to calculate the quantifiable value for the whole of the product's insulation" (Sierra Circuits).

2. HiPot testing

   "This test is used to check whether the insulation provided is enough to protect the circuit board or not. For this test, a high voltage is applied to the PCB and the resulting current flow through the insulation is measured. This current is called the leakage current and it is measured using the HiPot tester. If the high voltage does not break down the insulation, then the insulation will be good enough to protect the board. This test is also known as the dielectric withstanding voltage (DWV) test and is usually carried out after conducting the dielectric breakdown test" (Sierra Circuits).

Insulation resistance test coupons:

1. Moisture and insulation resistance coupons

   This coupon evaluates the insulation resistance and the bulk resistance of the circuit board. This evaluation is done once the board has been exposed to different humidity and temperature environments with specific voltages applied to it.

2. E-coupon

   This coupon evaluates the moisture and insulation resistance of the laminated base materials. A maximum of 10 layers can be tested using this coupon.

3. Legacy E coupon

   This coupon has a Y pattern and it is helpful in evaluating the cleanliness and insulation resistance properties.

Surface insulation resistance coupons:

1. H coupon

   This coupon measures the effects of the process or residues on the surface insulation resistance.

2. Legacy H coupon

   The legacy H coupon is used for higher level insulation testing.

The last topic discussed in the preview of this document is high voltage circuits. When it comes to high voltage circuit boards, the values should be rechecked. There is a higher chance of a flashover when a high voltage is applied across the conductors. Different standards should be considered when working with a high voltage design. There are high voltage design tools that can be used to evaluate values for specific voltages.

Following the standards set by IPC-2221 allowed for a proper and safe design. Considering that the main priority of the project is in fact the safety of the user, since it is used to combat the dangers of unsupervised and unplanned fires, it is imperative that all forms of safety protocols are followed in order to make sure that the robot does not provide any extra threats.

### 4.1.3 Battery Standard

IEC 60086 is an international standard that encourages manufacturers to ensure that their batteries are interchangeable according to the standard form, fit, and function. This standard is put into place to help users and manufacturers of devices that use batteries.

Due to the costs of the official document, only a preview was able to examined. Below are the key takeaways.

Benefits to demonstrating that a battery complies with the IEC battery standard:

- Enhance consumer trust in the brand
- Increase the likelihood that device manufacturers will include or recommend the batteries in their products
- Demonstrate safety
- Batteries that do not coincide with the IEC 60086 mark may be prohibited from certain markets

A complete list of the different types of battery standards can be found in the document. Stated below are the general battery standards.

Figure 45: General Battery Standards

| Standard Number | Title |
|---|---|
| IEC 60050 | International electro technical vocabulary. Chapter 486: Secondary cells and batteries. |
| IEC 60086-1, BS 387 | Primary Batteries - General |
| IEC 60086-2, BS | Batteries - General |
| ANSI C18.1M | Portable Primary Cells and Batteries with Aqueous Electrolyte - General and Specifications |
| ANSI C18.2M | Portable Rechargeable Cells and Batteries - General and Specifications |
| ANSI C18.3M | Portable Lithium Primary Cells and Batteries - General and Specifications |
| UL 2054 | Safety of Commercial and Household Battery Packs - Testing |
| IEEE 1625 | Standard for Rechargeable Batteries for Mobile Computers |
| USNEC Article 480 | Storage Batteries |

There is a highly efficient testing and certification process for a wide range of battery types to ensure that all safety standards are met and any design project that involves the use of a battery can be completed efficiently. As previously stated, it can be dangerous to not follow the safety standards that are provided, and so the group was highly aware of the need to not only follow these standards, but also go beyond them to ensure the safety of all who intend to use the project in the future.

# 4.1.4 USB Standard

The USB standards have shifted from USB 1 to USB 2, USB 3 and now USB 4, each with different performance specifications and capabilities. For a product to be successful, it has to comply with standards and keep up with the latest technology advancements. This is exactly what the Universal Serial Bus (USB) has done with each updated version of their standards. The revisions included refining the performance and other improvements to the system.

With the use of the USB being so widespread, backwards compatibility is very important along with a future upgrade path. A preview of the official document was examined and the key ideas are futured below.

USB Implementers Forum:

The USB standard is developed and maintained by the USB Implementers Forum, USB-IF, which is a non-profit corporation that has been founded by the companies that developed the USB standard and want to use and develop it. The corporation ensures that the USB is an industry standard. The USB-IF develops and maintains the USB standards, including Wireless USBs, and runs a compliance program to maintain that the quality of the products and compatibility between devices.

Wired USB Standards:

Looking at each of the different versions of the USB and understanding the various notations and performances is important in understanding how each of the standards came about.

USB 1.1- This was the original version of the USB which was released September of 1998 after discovering a few problems with the USB 1.0. specification, released January 1996. In this document was a Master/Slave interface and a tiered star topology that was capable of supporting up to 127 devices and a maximum of six tiers or hubs. The master or “Host” device was normally a PC with the slaves or “Devices” linked via the cable.

One of the aims of the USB standard was to minimize the complexity within the device by enabling the Host to perform the processing, meaning the devices would be cheap and accessible.

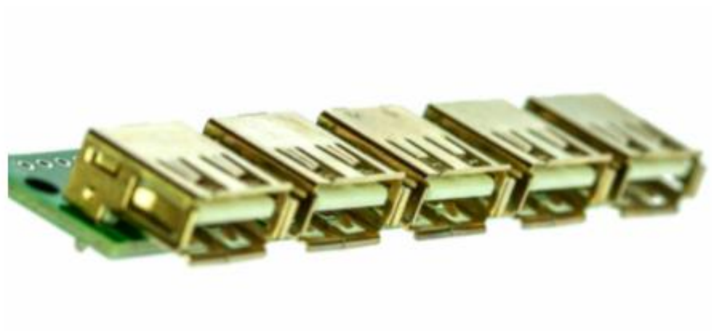
The data transfer rates of USB 1.1 are:

- Low speed – 1.5 Mbps
- Full speed – 12 Mbps

The data encoding method for the USB is Unicode. Other parameters include a cable length limited to 5 meters and a power consumption up to 500 mA, although it is limited to 100 mA during start-up.

USB 2.0- The main difference between USB 2.0 and USB 1.1 was the data transfer speed, which was increased up to a high speed rate of 480 Mbps. The data encoding for this version of the USB is also Unicode. Another improvement in version 2.0 was an increase in the power up to 1.8 A. This allowed the USB to provide charge for smartphones and external drives.

Figure 46: USB 2 Type A Sockets on a PCB



USB 3.0- The USB 3.0 included a feature called SuperSpeed bus, which provided a fourth transfer mode giving data transfer rates 4.8 Gbit/s. The USB 3.0 moved from Unicode to 8b/10b encoding.

Figure 47: USB 3 Type A Connector

USB 3.1- This version of the USB is known as SuperSpeed+. The speed of data transfer was double the speed compared to USB 3.0. It provided raw data transfer of 10 Gbit/s and also reduced the line encoding overhead to 3 percent, which was done by changing the encoding scheme to 128b/132b. The charging capability was increased to 20 V, 5 A, with the capability to reduce it to 5 V as appropriate. This allowed users to charge larger devices such as computers. Version 3.1 significantly increased the speed and functionality of the USB.

USB 3.2- USB 3.2 introduced two new SuperSpeed+ transfer modes over the USB-C connector with data rates of 10 and 20 Gbit/s. The increase in bandwidth was a result of multi-lane operation over existing wires that were intended for flip-flop capabilities of the USB-C connector. Another update was that the USB-IF introduced a new naming scheme for the different variants. The USB-IF decided on a 2 by 2 notation for the highest speed version.

Figure 48: USB C Connector

USB 4- USB-IF introduced version 4 in August of 2019. This standard provided more flexibility and functionality and was based around the Thunderbolt 3 protocol specification. USB 4.0 supports data through up to 40 Gbps. It uses the USB-C type connector because of its ability to accommodate the speed and mode of the USB 4.0.

Wireless USB- This version provides a wireless connection over which data can be transferred. This version uses frequencies in the band 3.1-10.6 GHz and provides a data bandwidth of 53-480 Mbps with a distance up to 3 to 10 meters.

The levels of functionality for a USB today are large with the evolution of technology. The increasing requirements for faster and larger levels of data transfer as well as increased levels of convenience and capability have led to a number of new standards being introduced.

## 4.1.5 Fire Safety Standard

The requirements for fire protection are subject to numerous standards. The leading resource in the United States for fire safety standards is the NFPA (National Fire Protection Association). While some codes apply to residential and commercial settings, hazardous-area fire and gas detection equipment must be certified for use in that hazardous location, which are outlined by the OSHA (Occupational Safety and Health Administration).

The attributes that make a process or location “hazardous” are discussed below.

The National Fire Protection Association has a goal of minimizing the risks and effects of fires by breaking down hazardous areas into three classes:

- Class I – Areas containing flammable gases, flammable liquid-produced vapors, or combustible liquid-produced vapors are or may be present in the air in quantities sufficient to produce explosive or ignitable mixtures
- Class II – Areas containing combustible dust

- Class III – Locations in which easily ignitable fibers or flying's (rayon, cotton, jute, hemp, and cocoa fiber) are present

Each class has divisions based on the concentrations of flammable materials. The breakdowns are presented in the official document which was not able to be accessed due to costs.

It is important to understand how to control fire risk. To be able to do this, the components for how a fire starts needs to be looked at. The essential elements are known as the "fire triangle."

1. Oxygen – Primary source is the ambient air
2. Fuel – Present in structures and used in processes of all kinds
3. Ignition Source (i.e. heat) – Depends on fuel and conditions; Can be spark of electric switch, hot surface, etc.

It is necessary to control sources of ignition. For example, if electrical equipment must be located in a hazardous area, they must be designed to limit or isolate potential sources of ignition, per NFPA. NFPA 70 Section 500.7 "Protection Techniques" lists several techniques for how to control ignition risk.

The three acceptable protection methods for Class I Division 1 are:

- Explosion Proof (XP) – Sparks or explosions are contained within housing
- Purged and Pressurized – Combustible gases and vapors are denied entry into the enclosure
- Intrinsically Safe (IS) – The entire power of the system is limited

"A fire protection system is comprised of several subsystems that can include, but are not limited to: flame, smoke and gas detection; notification and/or suppression activation; and a controller that receives the inputs form the detection devices, makes decisions and initiates appropriate action or actions" (IML Group PLC).

There are many options for fire detection that depends on the materials and fuels present, the processes involved, the environment, and other control measures present. NFPA 72 references the several codes and standards for fire detection and handling which can be found in the official document.

Other NFPA Standards that reference NFPA 72 include:

- NFPA 15 Standard for Water Spray Fixed Systems for Fire Protection
- NFPA 30 Flammable and Combustible Liquids Code
- NFPA 59A Standard for the Production, Storage, and Handling of Liquefied Natural Gas
- NFPA 70 National Electrical Code
- NFPA 409 Standard on Aircraft Hangars

NFPA 72 describes a flame detector as “a radiant energy-sensing fire detector that detects the radiant energy emitted by a flame” (IML Group PLC). Several sensing technologies can be implemented into these devices. Factors that come into play when selecting the flame detector are listed below:

- Size of the fire that is to be detected
- Fuel involved
- Sensitivity of the sensor
- Distance between fire and detector
- Purpose of detection system
- Response time required

NFPA 72 addresses these selection factors as well as several others in the document. While the standards of this document are very detailed and wide ranging, there are other organizations that include a variety of other standards that need to be considered when dealing with fire safety.

### 4.1.6 C Language Standard

The current standard for the programming language C is ISO/IEC 9899:2018, or C18. C18 is a revision of C11. It does not contain any major changes from the previous edition except for some technical corrections and clarifications.

The C language is one of the most popular programming languages used worldwide for operating systems, network drivers, databases, language interpreters, utilities, and a variety of other areas. The C language originated in the 70s and was expanded upon in 1989, also known as the standard ANSI X3.159-1989. It has since been adopted internationally and developed by ISO/IEC.

ISO/IEC 9899:2018 addresses many scopes, specifically the ones stated below:

- The representation of C in programs
- The syntax and constraints of the C language
- The semantic rules for interpreting C programs
- The representation of input data to be processed by C programs
- The representation of output data processed by C programs
- The restrictions and limits imposed by a conforming implementation of C

The document is divided into four major subdivisions: (1) preliminary elements, (2) the characteristics of environments that translate and execute C programs, (3) the language syntax, constraints, and semantics, (4) and finally the library facilities. The user should note that the document undergoes periodic revisions and new features could be added or withdrawn from the document. The ISO/IEC 9899:2018 does not address the size or complexity of a program and its data that will exceed the capacity of a processing system nor does it cover minimal requirements of a data-processing system that is capable of supporting a conforming implementation.

Due to the fact that version C18 could not be purchased because of costs, version C11 was reviewed and the main topics were noted below. There were only minor changes between these two versions, as stated previously.

C standard leaves some behavior of C constructs as undefined and some as unspecified to allow for some flexibility and simplification in the implementation. Different compilers can produce different results so it is important to understand what the standards say about different topics of C programs.

What the standard says about prototype of main:

```
The function called at program startup is named main. The
implementation declares no prototype for this function. It shall be
defined with a return

type of int and with no parameters:

       int main(void) { /* ... */ }

or with two parameters (referred to here as argc and argv, though any
names may be used, as they are local to the function in which they
are declared):

       int main(int argc, char *argv[]) { /* ... */ }

or equivalent;10) or in some other implementation-defined manner.
```

An example of different compilers producing different results:

```
#include<stdio.h>
int main()
{
    int i = 1;
    printf("%d %d %d\n", ++i, i++, i);
    return 0;
}
```

```
2 1 3 - using g++ 4.2.1 on Linux.i686
1 2 3 - using SunStudio C++ 5.9 on Linux.i686
2 1 3 - using g++ 4.2.1 on SunOS.x86pc
1 2 3 - using SunStudio C++ 5.9 on SunOS.x86pc
1 2 3 - using g++ 4.2.1 on SunOS.sun4u
1 2 3 - using SunStudio C++ 5.9 on SunOS.sun4u
```

The statement in the C standards that is related to the circumstance above is as follows:

```
The order in which the function designator, arguments, and
subexpressions within the arguments are evaluated in a function
call (6.5.2.2).
```

As stated in the C standards, it is not a good idea to use programming constructs whose behavior is undefined or unspecified, as the output of such programs may change with the compiler.

The standards listed above were found in a preview of the ISO/IEC 9899:2011 (C11) document. There are several other standards that should be studied before writing a program in order to ensure that the outputs are what is expected.

# 4.2 Constraints

There are a variety of constraints that need to be considered when dealing with building a robot. This subheading will deal with nine different kinds of constraints that will range from economics constraints, to social, political, and ethical constraints. Each will be discussed at length and each was applied to the project.

The first of these is economic constraints that deal with the group's limitations on budget, especially dealing with all of the different occasions that the project's parts may need to be purchased more than once. Part of this constrained will be explained in more detail in a later section.

The next constraint discussed will be that of time. In this project, this was probably the most important as it is the one that is most pressing for our purposes. Since the project is under the context of a senior design project, the time constraint is one that absolutely cannot be stretched further.

The environmental constraint comes next, and this will discuss the difficulties of not having appropriate equipment at our disposal at all times, and how sometimes the group was held back from working due to the environments we are all working from, which is primarily at home.

The next constraint is that of health. The health constraint is mainly due to the Covid-19 pandemic, but also to other potential diseases and illnesses. However, the Covid-19 constraint will also be expanded on in a subsequent sub-subheading.

Following the health constraint is the sustainability constraint which deals with what the robot has to be able to handle. This deals with everything from extreme temperatures to being waterproof.

Next, the Covid-19 constraint is basically a subset of the health constraint, but dealing specifically with the Covid-19 pandemic and how the university has a different set of rules ever since the pandemic hit.

The manufacturability constraint follows the Covid-19 constraint and deals mainly with how it can be designed and built, and how the determined complexity of the robot could have been a potential drawback.

The penultimate constraint will be the safety constraint which, for this project, has a lot of relevance, since we're dealing with a robot that puts out fires. This section also outlines how the robot has to be safe from certain conditions like humidity and moisture from the water it uses.

Lastly, the final constraint is a conglomerate of social, political, and ethical constraints, and these have to do with the probability of this project being approved by political and social powers, and how the design has to meet certain ethical qualifications and standards.

## 4.2.1 Economic Constraints

Due to the ongoing situation involving a shortage of supplies, it was hard to get certain components needed for our project. As a result, we had to make adjustments to our robot design and make due with what components we can find. We had to be flexible and creative in coming up with solutions to the problems we faced.

Something else that was taken into consideration is inflation. With the current state of the economy, prices in all aspects of life have greatly increased. Due to the increase of prices, certain components were more expensive than usual. The prices are constantly fluctuating so we had to keep an eye on the different components and when they were at a good price. In order to preemptively combat this, we had to plan accordingly and made sure to budget.

Another constraint for our group was the cost of the robot's design. Since all components of our project were paid out of pocket, we had to keep a reasonable budget in mind. This constraint also limited us to what parts we will want to buy. As outlined previously in Table 4, the total cost of the robot was not overwhelmingly expensive, but other constraints needed to be kept in mind when it came to testing the parts that were purchased.

For example, as will be expounded on in section 4.5, Sustainability Constraints, the robot needs to be waterproof. With this in mind, it is possible that, in testing, some electronic components might have been damaged, since there were multiple testing instances where water was involved. If these parts had been damaged, most of the budget will have to be doubled. Since electronic equipment takes up the majority of the overall budget (around 70%), it was possible that the overall budget of the project would have had to be increased a considerable amount.

## 4.2.2 Time Constraints

The most crucial constraint in this project was time. We had two semesters to design and implement our project as well as write the documentation for how it was all constructed. With this in mind, we had to be diligent with our time to stay ahead of task. Being proactive each week to get a little bit of the project done was key so we did not have to cram work at the last second. Each team member has a very different schedule so we had to communicate effectively in order to know what needed to be done and when. Even though each individual was very busy, working together and getting bits done each week helped us make the best progress. The tentative schedule was to have each team member writing a specified number of pages of documentation per week, depending on how many pages needed to be done, given a certain time.

Since this team graduates in the fall and not in the summer, the team was afforded the summer as extra time in which the design could have been altered if necessary, or the building phase of the project could have been started early. However, the summer was constrained by the plans of the individual members. Given the extra time of summer, and the fact that some of the team members had summer plans, this slowed down what could have been accomplished during this time. It was important that the summer was considered for advancing the project, but because of the ambiguous nature of that time, and the team members' other engagements, it was necessary that the summer be considered not so much as guaranteed extended time, but as a potential economy of time that could have been used under specific circumstances.

## 4.2.3 Environmental Constraints

In order to properly test our robot and its components, we needed the right equipment. This was a constraint at times if we were not able to get to the lab during one of the week. We did not have the devices at home to check if everything was working as expected so we had to plan out each week accordingly.

Another issue we could have faced was the laboratory being full when we wanted to do testing on our project. With many students trying to design and test their project, the environment was not always suitable or available for us to complete what we needed to get done. In order to prevent this, our group worked on the robot design piece by piece and testing the different components as we went along, as opposed to trying to get testing done all at one time, which could have been overwhelming.

As mentioned previously, a shortage of supplies has been an ongoing issue. This was another environmental constraint we had to keep in mind as it was out of our control and we had to have a backup plan. To prevent this from happening, we were diligent in buying parts and did not wait till the last minute.

### 4.2.4 Health Constraints

This worldwide pandemic has put a halt to many things, and as of now, there are still some limits that the pandemic has placed that were detrimental to the building of the project. This could have potentially caused issues as one or more of the team members could have become sick and not made it to a meeting. Especially in highly contagious seasons such as the flu season, it was important for every member of the group to what we could to keep ourselves from spreading any kind of illness and disease. If even one group member was limited in our ability to help in the project, the entire process could have been slowed down up to a factor of twenty five percent.

As a result, this could have pushed our progress back. To prevent this from happening, we had to take necessary precautions and made sure we were all doing our part to try and stay healthy. Not only is the wearing of masks still a viable option to keep the spread of the virus at a minimum, but there is also the availability of vaccinations, and the prudent choice of staying home and isolated whenever one of us started to feel sick. In terms of the availability of laboratory space, it was a wise option to consider reserving spaces in order to avoid being sent away due to a person limit.

### 4.2.5 Sustainability Constraints

Another constraint or obstacle we may have faced was testing our robot. A lot of the equipment used in this project is not waterproof so we had to find a way to protect the components. The main objective in this project is for the robot to pump out water and put the flame out, so it was vital for us to not damage any of the parts that are not waterproof. As shown in previous sections, this was one of the most important constraints since it could have sent us back many months in the case that new parts had to be purchased again if there was permanent water damage in any of the parts.

Our robot was also able to withstand high temperatures as it will be detecting and extinguishing flames/fires. The parts picked for the design of our robot were durable and sturdy. Our robot also had to be able to maneuver its way through obstacles to reach the fire so we needed to consider the functionality of the various components. On top of that, since fires tend to change atmospheric pressure to a decently high degree, it was important for the robot to be able to withstand atmospheric pressure changes in terms of its ability to detect fires and to extinguish them.

### 4.2.6 Covid-19 Constraint

In Spring of 2020, the world was stopped by the advent of the COVID-19 virus that caused the shutdown of many of the facilities and institutions that were necessary for the acquiring and testing of many necessary parts. Due to the pandemic and its risks, it has been complicated for the group members together. The pandemic has affected various types of business in both positive and negative ways. As a result, the manufacturing business is one that has been negatively affected. Because of that, it was difficult at times to find specific parts that were used in the design of the project. The limited to lack of availability of different parts was a constraint that was faced due to the effects of the Covid-19 pandemic.

Not only that, but there were many different limits that could have slowed down the progress of the robot, such as a limit on the number of people allowed in a lab at once. If the robot needed testing for a specific part and the labs were unavailable for one of these reasons, it could have highly affected the speed at which the project was built.

## 4.2.7 Manufacturability Constraint

The manufacturability of the project was related to how easily it could have been manufactured once it had been designed. How easily the design of the project can be reproduced and constructed for mass production is what defines how manufacturable the design is. Third party parts and items were selected in order to keep the manufacturability of the design as high as possible. One of the goals of the project was for it to be simple and effective which also ties into the manufacturability of the design.

The main design uses a simple platform that is used to sustain the water container with the pump and the servo on top. The electronic parts of the circuit were acquired from third parties manufactures that provided these parts in higher quantity if needed, like the sensors. All the parts were simply assembled around the main bottom platform. However, using the third-party companies as the main suppliers for the parties also affected the manufacturing of the project in a negative way. Delays due to shipment problems of the parts sometimes caused a reduction of the manufacturing of the project. With that said, the choice of using parts that are provided by third-party companies was a good tradeoff that helped to keep the manufacturing of the design simple without requiring these parts to be built from scratch along with the robot.

## 4.2.8 Safety Constraint

The safety constraint is related to the safety of the user and the economic constraint as well. Due to the reasons mentioned in the economic constraints, the supplies and parts used for the final design were not always the desired ones because of the expected budget. As a consequence, the robot only works for a specific range that is the sensing range from the infrared sensor that was chosen. And due to it not having the best range, that also compromises the working range of the robot. Also related to this topic, the sensor don’t have the best resolution and sensitivity available. Because of that it does not always detect small flames that don’t produce enough infrared light to be detected by the sensor.

Another constraint is the fact that the design involves water that can damage the electronic parts of the project. Making some of the electronic components waterproof by covering them with the material of the shell helped to reduce the possibility of shock and malfunction of the robot due to the presence of water. To conclude, this fact can put at risk the safety of the user that fully relies on the decision making of the robot. If an undetectable flame could quickly spread and reach levels where the robot is no longer able to put it out it could cause major damage to its environment.

### 4.2.9 Social, Political and Ethical Constraints

These constraints are related to the other products available on the market and the approval of their usage by a government agency. There are no products that are completely similar to this project. Because of that it would already be very different and unique when compared to devices that have similar functions. Its unique sensing and extinguishing ability would be enough to help the users to feel safe when the designed robot is dealing with its target flame size. The difference and uniqueness of the project design when compared to possible market competitors is enough to cover the social and ethical constraints. Regarding the political constraint, the project design would have to be approved by a government agency that licenses and takes care of similar types of projects. The political constraint can be resolved when the fully functioning design can be demonstrated and showcased.

# 5. Hardware Design

This section is going to cover the hardware design of the project. The hardware part of the expression refers to the physical parts and components that are used in the project. The word design refers to the act on how the hardware physical components are assembled together in order to create a functioning product. So, this section is going to cover the assembly of the different components that are needed to have fully completed project and have it working as intended.

The first part of the hardware design section is going to cover the microcontroller that is used in the project, the Arduino Uno and its connections to other important components in the design. One of the connections that is going to be covered in this section is the connections to the Walfront IR Flame Sensor Module Detector that is the IR sensor selected for the project. Since the detection angle of each of the sensor chosen is around 60 degrees, three sensors are used together and positioned in a way that maximizes the detection angle of the robot to a total of 240 degrees combined among all three sensors. By increasing the number of sensors in the robot and positioning them correctly the overall detection of the entire system can increase by a factor of 3 and its performance can also increase since it can detect a flame in a wider angle covering a greater area. The correct pins of the Arduino Uno microcontroller and of the sensor that need to be connected in order for the robot to fully function as expected are covered in this hardware design section.

Another aspect that is going to be covered in this section is the L293D motor driver integrated circuit and its connections to the Arduino Uno microcontroller. This component is the one that is responsible for receiving the logic signals from the Arduino Uno microcontroller and using the signals to operate the two DC motors of the robot. There are two techniques that are used to control the speed and direction of the motors. These techniques are the Pulse Width Modulation (PWM) and the use of H-Bridge circuits that are already in the L293D motor driver. Both of these different techniques are going to be covered in the part of this section that covers the connections for the L293D motor.

The power supply of the robot along with its connections is also a topic that is going to be discussed in this hardware design section. Due to the fact that the robot is required to work autonomously it cannot be required to be plugged into a power supply that could limit its movement ability. Because of that a battery is used to power the different components on the robot. Picking the correct battery is an important task since it is a key aspect of the system. It is responsible for powering up the various required components in the robot and making sure it works properly. The battery also has size constraints since it needs to be of moderate size and cannot compromise the functionalities of the other parts of the robot due to its size, weight or heat generated. The two main components that are powered directly by the battery are the Arduino Uno microcontroller and the L293D motor driver. There are other components that need power to function such as the Walfront IR Flame Sensor Module Detector and the PULACO Mini Submersible Water Pump, but those components are powered by their connection with the Arduino Uno microcontroller.

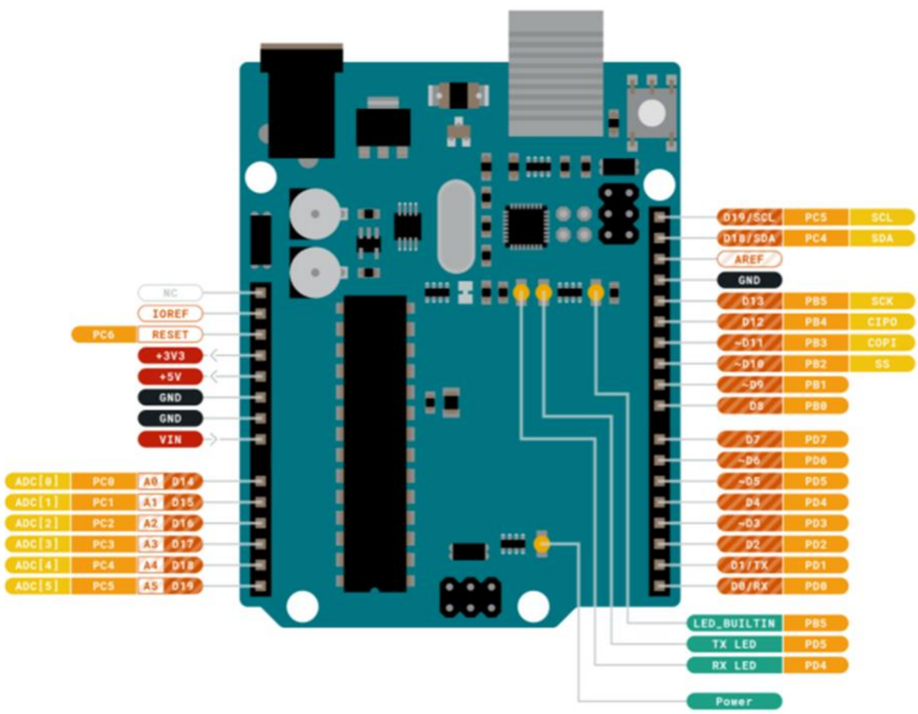
Overall, this section is going to cover all of the connections, operations and physical components that are used in the design of the project. The pin connections, their purpose and how the components interact among themselves will be discussed along in this section.

# 5.1. Arduino Uno

The microcontroller used in our firefighting robot is the Arduino Uno. This device is the primary component as it is used to run and control all the other parts in our design. The Arduino Uno has a 5-volt input source which is used to supply power. Other components connected to our microcontroller are the three flame sensors, the motor driver, and a water pump.

Using the pinout sheet of the Arduino Uno, we connected the various parts in order to get our robot to function properly.

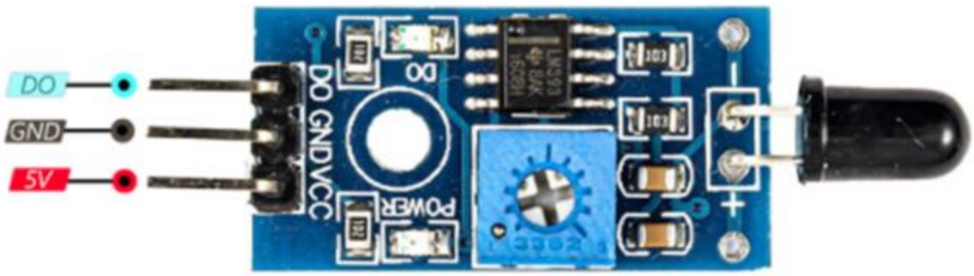
Figure 49: Pinout of the Arduino Uno




Arduino Uno connection to the three flame sensors:

The flame sensors used in our design have three pins which are VCC, GND, and D0. The VCC is used for the 5-volt power supply, GND is used for ground of the power supply, and D0 is used for the digital output. The figure below shows where each pin lies in the design of the flame sensor.

Figure 50: Pins of the IR Flame Sensor




IR Flame Sensor 1-

- The D0 pin on the IR flame sensor is connected to the D10 pin on the Arduino Uno microcontroller.
- The GND pin of flame sensor is connected to the GND pin on the Arduino Uno microcontroller.
- The VCC pin of flame sensor is connected to +5 V on the Arduino Uno microcontroller.

IR Flame Sensor 2-

- The D0 pin on the IR flame sensor is connected to the D9 pin on the Arduino Uno microcontroller.
- The GND pin of flame sensor is connected to the GND pin on the Arduino Uno microcontroller.
- The VCC pin of flame sensor is connected to +5 V on the Arduino Uno microcontroller.

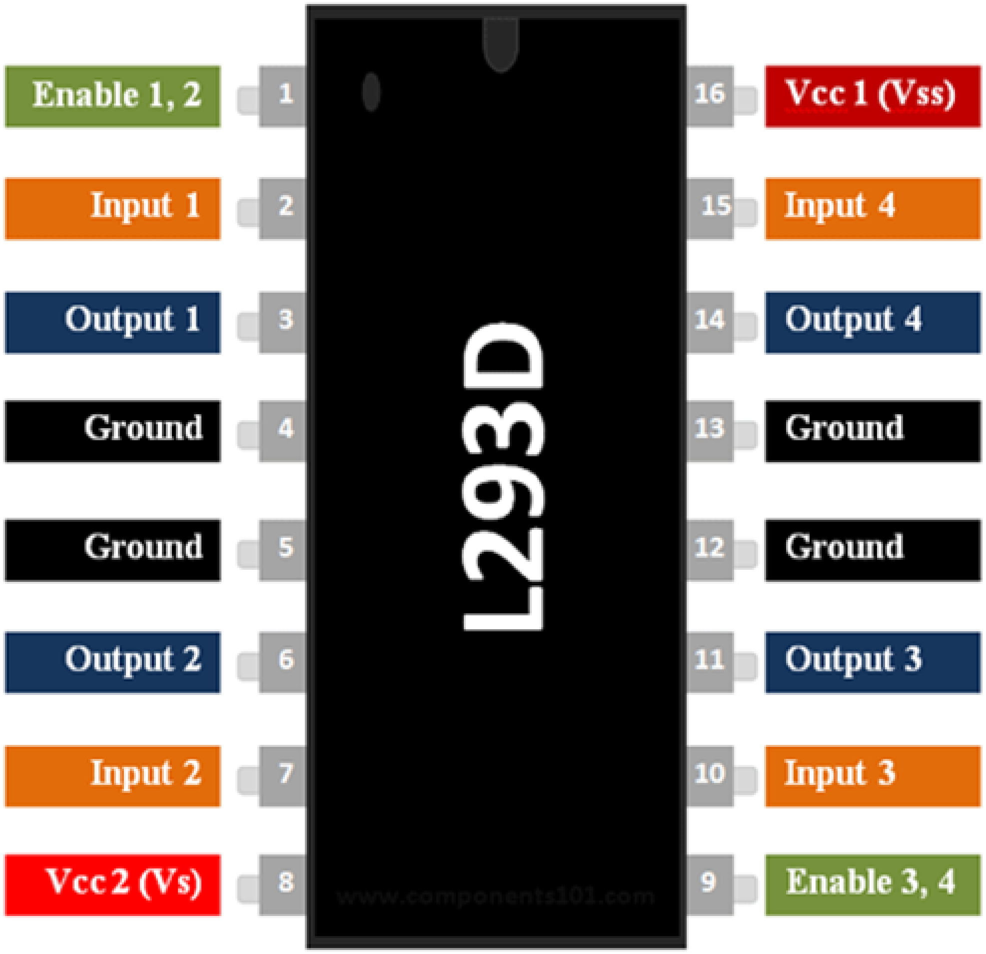
IR Flame Sensor 3-

- The D0 pin on the IR flame sensor is connected to the D8 pin on the Arduino Uno microcontroller.
- The GND pin of flame sensor is connected to the GND pin on the Arduino Uno microcontroller.
- The VCC pin of flame sensor is connected to +5 V on the Arduino Uno microcontroller.

Arduino Uno connection to the motor driver:

The Arduino Uno is connected to the L293D motor driver which is responsible for the movement of the DC motors, since, as stated before, the Arduino Uno microcontroller unit cannot provide the necessary power to run the two DC motors in order to move the robot.

Figure 51: Pin Connections of L293D Motor Driver




- IN1 (pin 2) on the motor driver is connected to the D5 pin on the Arduino Uno.
- IN2 (pin 7) on the motor driver is connected to the D4 pin on the Arduino Uno.
- IN3 (pin 10) on the motor driver is connected to the D3 pin on the Arduino Uno.
- IN4 (pin 15) on the motor driver is connected to the D2 pin on the Arduino Uno.
- VSS (pin 16) on the motor driver is connected to the +5-volt port on the Arduino Uno.
- GND (pins 4, 5, 12, 13) on the motor driver is connected to GND on the Arduino Uno.

Arduino Uno connection to the water pump:

The 5 V mini submersible water pump is used to pump water in order to extinguish the fire. This allows the robot to work on an automatic watering system.

- The +5-volt port on the water pump is connected to the D6 pin on the Arduino Uno microcontroller.
- The GND pin on the water pump is connected to the GND pin on the Arduino Uno microcontroller.

Update: We used a relay module in order to supply power to our water pump. The relay module was connected to 5 V on the Arduino, GND on the Arduino, and pin 6 on the Arduino. It was also connected to the positive terminal of our pump via the NO. The COM on the relay module was connected to 5 V.

We also supplied an additional 12 V power supply for the L293D Motor Driver IC from the battery. The connections for the battery to both the motor driver and the Arduino Uno microcontroller are outlined below.

- The positive terminal of the battery is connected to VS (pin 8) of the L293D motor driver.
- The negative terminal of the battery is connected to the GND on the Arduino Uno microcontroller.

Update: No additional 12 V power supply was needed for our design. The VS pin on the motor driver was supplied with 5 V.

The many different aspects previously discussed were all critical in the functioning of our fire extinguishing robot. Before connecting and wiring each of the components, we made sure to examine and understand the datasheets of each of the given components. Using the devices in the wrong way or supplying too much or too little power could temporarily, or even permanently, damage the part, which is why we took note of a few of the features outlined below.

Important features of the Arduino Uno:

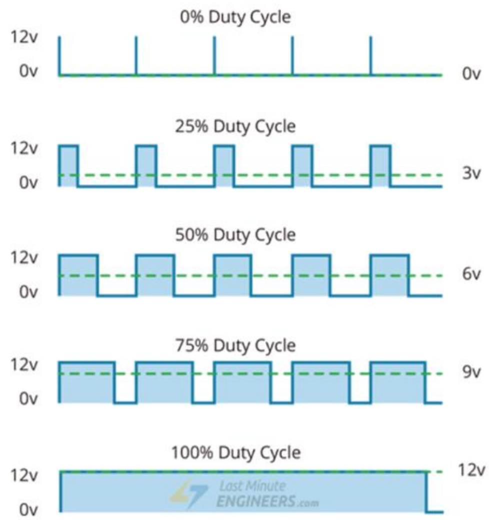
- 14 digital I/O pins (6 of which can be used as PWM outputs)
- 6 analog inputs
- USB connection
- 5 V operating voltage
- DC current for each I/O is 40 mA

# 5.2 L293D Motor Driver IC

The L293D Motor Driver is a key component in the assembly of our firefighting robot. This part will be used to run the two DC motors, controlling both speed and direction. There are two different techniques that provide a method to control both speed and direction.

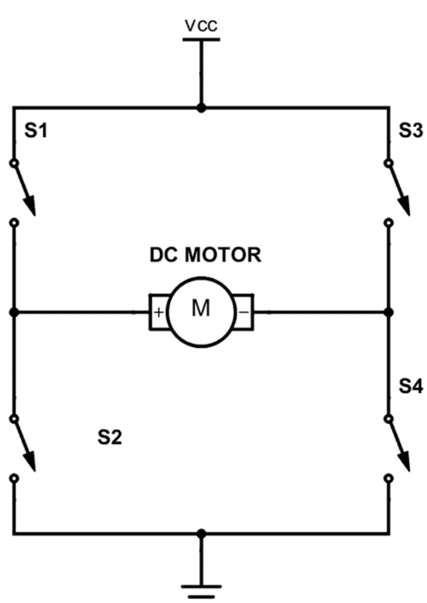
In order to control the speed, we use the technique of PWM (Pulse Width Modulation). PWM is done by taking the average value of the input voltage and adjusting it by sending a series of ON-OFF pulses. The average voltage is proportional to the width of the pulses, which is known as the duty cycle. A higher duty cycle results in a greater average voltage and therefore a higher speed being applied to the DC motor, which may sometimes be necessary for the project, especially for further improvements.

Figure 52: PWM Technique




The second technique controls the rotation direction, which is done by using an H-Bridge. The DC motor's spinning direction can be controlled by changing the polarity of its input voltage. An H-Bridge circuit contains four switches with the motor at the center. When two switches close at the same time, the polarity of the voltage applied to the motor reverses, causing a change in the spinning direction.

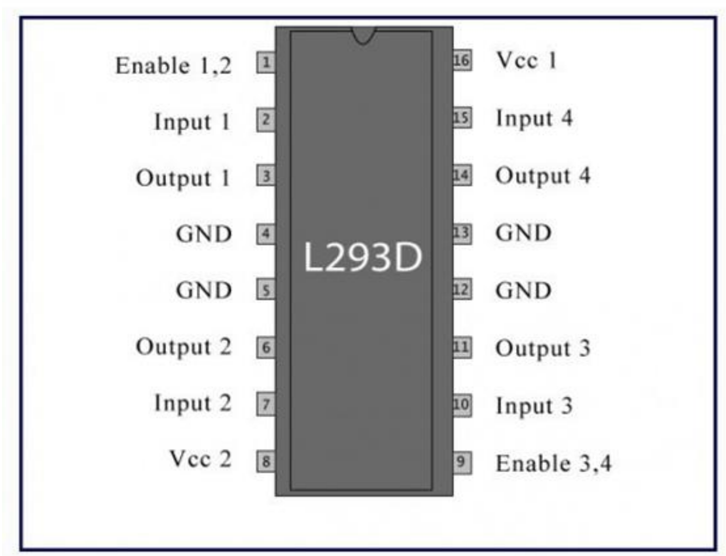
Figure 53: H-Bridge



After a fair amount of deliberating in order to figure out the ideal choices for the robot, and after realizing that a motor driver was necessary, since the Arduino Uno microcontroller unit cannot drive the two DC motors that were chosen due to a lack of sufficient voltage outputted by the Arduino Uno, the group decided that the project was to use the L293D Motor Driver IC through using the dual-channel H-Bridge that was briefly described above. Also, there is an external voltage source connected to the circuit in order to supply power to the L293D Motor Driver.

In order to have a functioning robot, each device and feature must be connected properly, and in order to figure out the proper connection paradigms, the documentation of each component must be consulted. In an attempt. The figure below shows the pinout of the L293D Motor Driver IC.

Figure 54: Pinout of L293D Motor Driver IC




One of the most important aspects of a design when dealing specifically with hardware is being certain of what each pin in the component does and how it can be leveraged for a specific use. These pins are not the same for every piece of hardware, and so they need to be studied and understood through the hardware manual. After doing this, one can be sure of the design decisions one makes since the design is based on the manual's pin configurations. In the case of the L293D Motor Driver, there are sixteen different pins, each of which has a different job and each of which is named something differently. Some pins control input, some pins control output, some pins are simply strictly connected to ground, and some pins have a potentially different connection to another part of a separate component altogether. These sixteen pins have been identified and studied through the motor driver's documentation and they have been outlined below. This table is extremely important because it shows exactly what each pin does, and this is a nonnegotiable part of the design process. As such, the table below identifies each pin and its particular use for us in our design.

Table 9: Pinout Specifications of L293D

| Pin Number | Pin Name | Description |
|---|---|---|
| 1 | Enable 1, 2 | Enables the input pin Input 1 (pin 2) and Input 2 (pin 7) |
| 2 | Input 1 | Directly controls Output 1 pin; controlled by digital circuits; controls spinning direction of motor 1 |
| 3 | Output 1 | Connected to one end of motor 1 |
| 4 | GND | Connected to ground of circuit |
| 5 | GND | Connected to ground of circuit |
| 6 | Output 2 | Connected to another end of motor 1 |
| 7 | Input 2 | Directly controls Output 2 pin; controlled by digital circuits; controls spinning direction of motor 1 |
| 8 | Vcc 2 | Connected to voltage pin for running motors (4.5V to 36V); connected to 5 V |
| 9 | Enable 3, 4 | Enables the input pin Input 3 (pin 10) and Input 4 (pin 15) |
| 10 | Input 3 | Directly controls Output 3 pin; controlled by digital circuits; controls spinning direction of motor 2 |
| 11 | Output 3 | Connected to one end of motor 2 |
| 12 | GND | Connected to ground of circuit |
| 13 | GND | Connected to ground of circuit |
| 14 | Output 4 | Connected to another end of motor 2 |
| 15 | Input 4 | Directly controls Output 4 pin; controlled by digital circuits; controls spinning direction of motor 2 |
| 16 | Vcc 1 | Connected to +5V to enable IC function |

The L293D contains two types of control pins in which the speed and spinning direction of the DC motors are controlled. Input 1 and Input 2 control the spinning direction of motor 1 while Input 3 and Input 4 control the spinning direction of motor 2. This is done by applying a logic HIGH (5 V) or a logic LOW (ground) to the pins, illustrated by the table below.

Table 10: Example of How to Control Spinning Direction of Motor

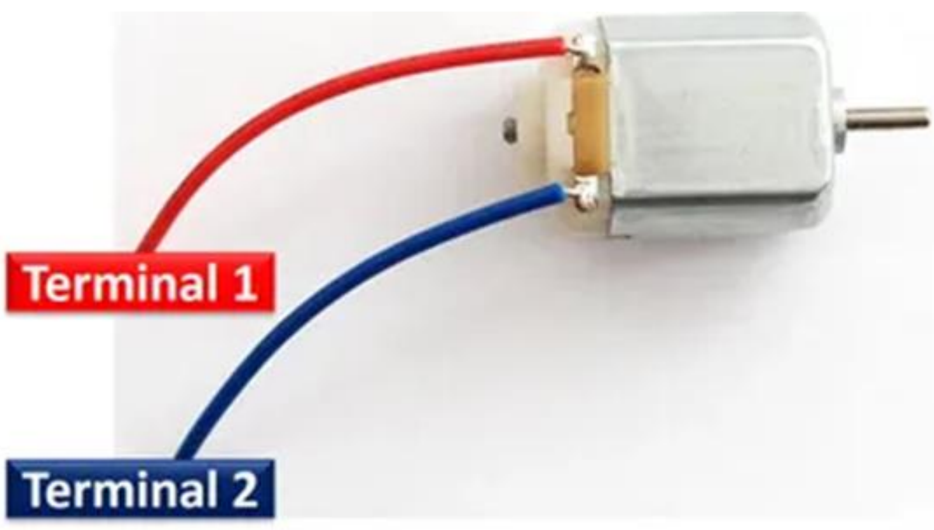
| Input 1 | Input 2 | Spinning Direction |
|---|---|---|
| Low (0) | Low (0) | Motor OFF |
| High (1) | Low (0) | Forward |
| Low (0) | High (1) | Backward |
| High (1) | High (1) | Motor OFF |

The speed control pins are Enable 1,2 and Enable 3,4. Pulling the pins to HIGH will make the motors spin, while pulling them LOW will make them stop. By using the method of PWM, we can control the speed of the motors.

L293D motor driver connection to the two DC motors:

The motors are important for the robot to be able to navigate its way to a detected fire. The wiring of the motors to the motor driver is stated below.

Figure 55: DC Motor Terminals




Left motor-

- OUT1 (pin 3) is connected to one side of the motor
- OUT2 (pin 6) is connected to another side of the motor

Right motor-

- OUT3 (pin 11) is connected to one side of the motor
- OUT4 (pin 14) is connected to another side of the motor

The connections involving the motor driver to the Arduino Uno are stated in a previous section. Any connections that were not stated previously and are important for the design of our robot are shown below.

Other connections-

- VS (pin 8) on the L293D motor driver is connected to the 12 V battery

Update: Pin 8 is connected to 5 V

Understanding the various aspects that enable the component to work properly is vital. We have to understand how the part works, how to wire it correctly with the other components in our project, and what voltages, currents, etc. are required in order for the part to reach its full capacity. Without this understanding, the motor can get damaged as well as other parts in the design. To prevent any of these situations from happening, we took note of the special features of the L293D Motor Driver IC.

Important features of the L293D:

- Motor voltage range of 4.5V to 36V
- Can be used to run two DC motors with the same IC
- Maximum peak motor current of 1.2 A
- Maximum continuous motor current of 600 mA
- Supply voltage of 4.5V to 7V

# 6. Software Design

The software portion of the firefighting robot is written in C code and runs on that Arduino Uno unit. The Arduino Uno unit takes inputs from the IR sensors and produces outputs that it sends to the motor drivers, water pump, and servos. This allows the robot to know where the fire is and orient itself in the direction of a fire and point the cannon at it to extinguish it. The functionality of the code will be explained in the sub sections of 6.1.

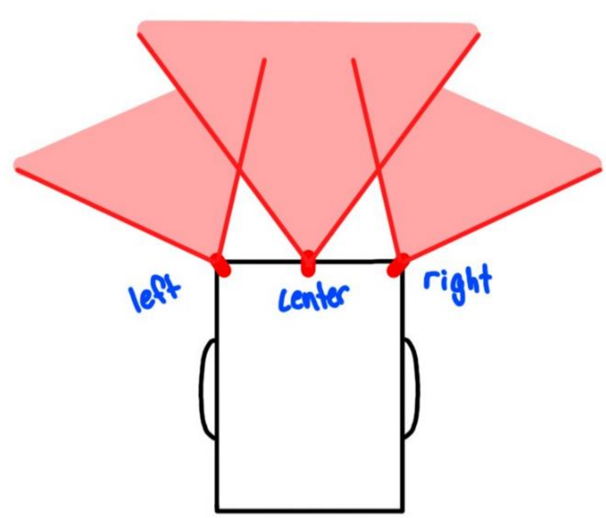
## 6.1 Software Functionality

The software portion of the firefighting robot is broken down into two main tasks: finding the fire and responding accordingly. To do this the robot uses its sensors to find a fire in its proximity and then responds using the motors to turn or move towards a fire and engage. These tasks are programmed in the Arduino Uno through the Arduino Integrated Development Environment IDE.

### 6.1.1 Sensor Input

The purpose of the sensors is to find the location of a fire in proximity to the robot. There are three small flame sensors mounted to the robot chassis. The first sensor is the center sensor, located at 90 degrees, or the very front of the robot. The second and third sensors are the left and right ones, they are both located between thirty and forty-five degrees from the center sensor or between 45 and 60 degrees and 120 and 135 degrees respectively. This gives the flame sensors approximately a 150-degree field of vision if positioned correctly.

Figure 56: Example Field of View of the Robot




Now that we know the location and field of view of the sensors we can determine where the flames are and what information we receive from them, as well as how to respond to receiving the information. First things first, you need to connect the sensor using the #define macro. In the setup stage the sensors need to be defined as an input so that the code knows what to do with the information it receives from them. From this point you need to know how a flame sensor determines if there is a flame in front of it in terms of input and output. If there is no flame in front of the sensor it will output a low or a 0, but if there is a flame in front of the sensor it returns a high or a 1.

The flame sensor is now defined, we know the area of coverage, and what the input and outputs returned by the sensor are. There are 4 main statements used to control the robot based on the information the sensors return. If none of the flame sensors can see a flame, or they are all returning 0's, then our firefighting robot remains dormant and stay in place. However, if one of the sensors sees a flame then it returns a 1, or high, and this changes the if statement to one that moves the bot into position and triggers the function to put the fire out. There are 4 main if statements that run based on the input from the sensors, which allows the robot to determine where the flame is in relation to the robot itself and moves the robot accordingly.

In order for the robot to activate the extinguish function we created a Boolean and set it equal to false, and whenever one of the sensors finds a fire, it moves into position and then sets the Boolean equal to true. Then we nest a call for the extinguish function inside a while statement for the fire Boolean being equal to true. This means that it constantly attacks the fire with water as long as one of the sensors is seeing a flame. If the front sensor no longer sees a flame due to it being put out, but one of the side sensors does, the robot repositions itself to better attack the flame from the front of the chassis where it has the greatest reach and water coverage.

### 6.1.2 Output

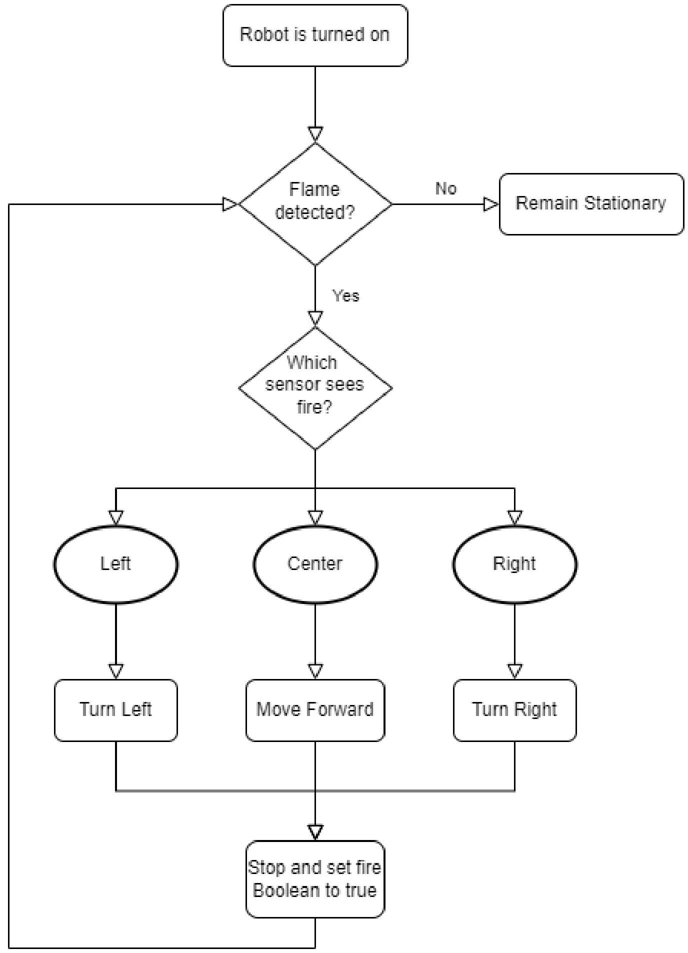
In the first section we covered the input from the flame sensors on the firefighting robot, now we will cover the output. The output is divided among the 3 moving parts of the robot: the wheel motors, turret servo, and the water reservoir pump. These three sections have their own code that is activated through the input sensors and causes the robot to react to the fire in a way that best allows it to put out the flames.

### 6.1.3 Motor Output

The motor has a rather simple algorithm, but it is also the only portion of the code that has a different reaction based on which sensor sees the flames. As we know, electric motors standard stand-by is when they are set to digital HIGH, this means in order to make the robot move in a certain direction, the motor for a wheel must be set to digital LOW.

Taking this into account, we set the motors to be digital high when the flame sensors see nothing, which leaves the robot on and functional, but remaining stationary. However, when the flame sensors see a fire, the robot positions itself so that it can best fight the fire. Due to the lack of range of the water turret this means positioning the flame front and center to the chassis where the turret gets the most coverage.

Figure 57: Flow Chart of the Movement of the Firefighting Robot




Since we chose a motor driver that can use two motors, we must program the second motor even though we are not using it. Since we do not initialize the motors to be in a shut-off state we need to set the motors to their desired values based on the input sensors. This leads to some lines of code that seem useless since they will never change. However, these lines are necessary if you decide to add another pair of wheels to the robot to give it more functionality so we coded them anyway. These lines look something like the following:

- digitalwrite (Left motor 2, HIGH);
- digitalwrite (Right motor 2, HIGH);

The name of the motor will change depending on what they are initialized as, but the format will always be the same. You are writing the value of HIGH to both motors so that they remain off, even though there are no motors. This is because we chose a motor controller that controls two different DC motors each. Since we are using two of the motor controllers, one on each side, all four motors must be accounted for in the code.

## 6.1.4 Servo Output

The servo output is rather simple but required a bit of testing before we had it all set and finalized. Due to the nature of the pump and servo interaction there can be a large variation in the code for the servo. The servo is what aims the water pump, and this is done through small delays and coordinate updates. The water pump forces water through a tube which is attached to the servo which performs a sweeping motion as is standard for the use of something like a fire extinguisher. This helps the robot not only cover a wider area but also allows the robot to be a little more relaxed in its positioning relevant to the flames.

The servo begins in a middle position facing directly forward, upon the fire detection Boolean being set to true the extinguish function activates and starts the movement of the turret via the servo. This is done through two for loops that read the current position of the servo and write the current position and add one. Since the field of view of our sensors is about 150 degrees it is important to keep the range of motion below that field of view so that we aren't spraying at things that are not in the vision of the robot. The first for loop changes the servo position from the initial 90 degrees to 30 degrees, or the bottom of the range of the servo arm. It then slowly increments the servo up to 150 degrees where it stops and the current for loop expires. In this position the new for loop's requirements have now been fulfilled where it then takes the new position of 150 degrees and lowers one degree at a time down to 30 degrees again. These two for loops then continue to be set off if the flame sensors detect a fire and the robot is in a position to fight it.

## 6.1.5 Pump Output

The pump output is extremely simple and only a couple of lines. Since the pump doesn't have any codable moving parts all we do is turn it on and turn it off. The difficult part of coding the pump is finding where in the code to place the enable and disable, or the LOW and HIGH for the pump. Water pumps are naturally off when they are set to digital LOW, this is the opposite of the wheel motors so it can be a bit confusing at times.

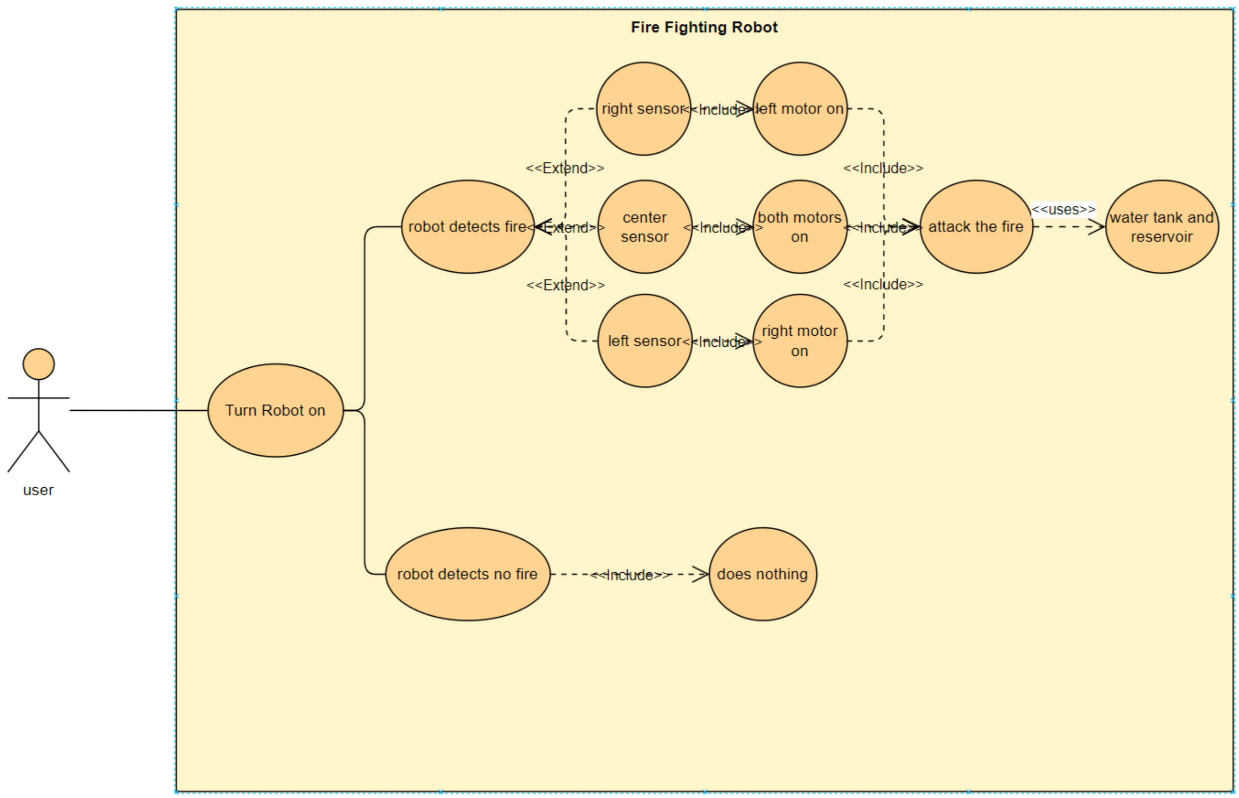
We decided to place the digital High right before the first for loop. This means that while the servo is aiming the water directly down the center the water pump turns on and continues pumping water throughout the for-loop's duration. When both loops have ended, and the servo has done one complete rotation left to right the pump turns off, the servo returns to the center position, and the Boolean is set to false. However, if one of the flame sensors continues to see a fire then it immediately changes the Boolean back to true which reactivates the for loops and starts up the pump again.

All of this happens fast enough that it appears the pump never shuts off. This more or less gives a continuous stream of water to fight the fire that turns off after each full sweep of the servo. Since the code is written this way it allows the firefighting robot to save water in the revisor but also fight more than just the fire that is currently seen. If the fire is put out halfway through the sweep the pump does not turn off and it continues to do its full sweep of the danger zone. There was a consideration to have the pump only activated while the Boolean is set to true, but if there is a flame just outside of reach of the sensors it would turn the pump off even if it could still be hit with the water. It would also turn off the pump each time the robot repositions itself whereas this way we can keep the robot fighting the fire for longer for a higher success rate.

# 6.2 Software Algorithms

The algorithms for the software are the bread and butter of the robot. They are how the robot determines where things are and how to respond to them. In section 6.1 we discussed the coding for the robot and why we made the decisions we made; in section 6.2 we will show pseudo code for the robot.

Figure 58: Use Case Diagram for Robot




# 6.2.1 Sensor Algorithm

The sensors are the only form of input the robot has, so it is important that the code be simple but also effective. The simpler the code is the less likely it was for us to run into something that could throw the robot off or break over time. The robot in its current configuration has about a 150-degree field of view, this allows it to detect just about any flames in front of it out to 30 centimeters away. The first thing you need to do when coding the sensors is define them as an input using the #define macro that comes standard with the C language. When you define your sensors, you need to also set it to the pins they are dedicated to and set your pinMode to either input or output, in this case it is input for the sensor. On the left is the assignment of each pin to a sensor and on the right is defining that sensor to the category of input

Table 11: Sensor Defines and Input Declarations

| | |
|---|---|
| #define CenterSensor 8<br>#define LeftSensor 9<br>#define RightSensor 10 | pinMode(CenterSensor, INPUT);<br>pinMode(LeftSensor, INPUT);<br>pinMode(RightSensor, INPUT); |

Below is a table breakdown of the sensor input that shows what the motors do when each of the sensors detects a flame. As discussed, flame sensors have a default of LOW or 0 when there is no flame in view and a HIGH or 1 when there is.

Table 12: Sensor Inputs and the Robot's Reaction to them

| **IF** | **THEN** |
|---|---|
| All sensors == 0 | Left motor == high<br>Right motor == high<br>Robot moves does not move |
| Center sensor == 1 | Left motor == low<br>Right motor == low<br>Move forward and trigger extinguish function |
| Left sensor == 1 | Left motor == high<br>Right motor == low<br>Move to left |
| Right sensor == 1 | Left motor == low<br>Right motor == high<br>Move to right |

## 6.2.2 Motor Algorithm

The motors are the first and direct reaction to the input from the flame sensors, thus we already gave the pseudo code for the wheel motors in that section. When coding the motors, you must also use the #define macro the declare them as part of the output. The motors have 4 pins even though we only use two of the four motors.

Table 13: Pin Assignments for Motors

| | |
|---|---|
| Pin 2 | Left motor 1 |
| Pin 3 | Left motor 2 |
| Pin 4 | Right motor 1 |
| Pin 5 | Right motor 2 |

Since we have two unused motors, they remain off the whole time, but we code them each time so that it is easy to change them to active motors if another set of wheels are added to the robot.

Table 14: Code for Robot Movement

| | |
|---|---|
| Robot remains stationary | DigitalWrite (LeftMotor1, HIGH);<br>DigitalWrite (LeftMotor2, HIGH);<br>DigitalWrite (RightMotor1, HIGH);<br>DigitalWrite (RightMotor2, HIGH); |
| Robot moves forward | DigitalWrite (LeftMotor1, LOW);<br>DigitalWrite (LeftMotor2, HIGH);<br>DigitalWrite (RightMotor1, LOW);<br>DigitalWrite (RightMotor2, HIGH); |
| Robot turns left | DigitalWrite (LeftMotor1, HIGH);<br>DigitalWrite (LeftMotor2, HIGH);<br>DigitalWrite (RightMotor1, LOW);<br>DigitalWrite (RightMotor2, HIGH); |
| Robot turns right | DigitalWrite (LeftMotor1, LOW);<br>DigitalWrite (LeftMotor2, HIGH);<br>DigitalWrite (RightMotor1, HIGH);<br>DigitalWrite (RightMotor2, HIGH); |

As you can see two of the motors, LeftMotor2 and RightMotor2, never get set to LOW or on. These are the two unused motors. Whenever the robot wants to turn right, the left motor turns on moving that side of the robot forward and the right wheel then acts as a pivot point for the robot allowing it to turn on a dime. If the robot wants to move left the same thing happens but on the other side. The left motor stays off allowing that wheel to act as pivot while the right wheel drives the right side of the robot forward turning it left.

We chose to code it this way rather than having the turn aim at the flame depending on which sensor sees it because this gives the field of attack a much larger area. If one sensor is seeing a flame it could just be an offshoot of the fire where it would spray at that small portion and be done. By repositioning to fight the fire head on it gives us the maximum field of view and range of motion of the water turrets arm.

## 6.2.3 Pump Algorithm

The code for the pump is quite simple, but because of its interaction with the servo it is a little more complex since the two are in tandem, so they are considered the same algorithm. The only time the pump algorithm is used is when the extinguish function is called which causes the pump and the servo to work to put out the flames in front of the robot.

The robot once in position calls the extinguish function which begins the firefighting process. The servo which is facing forward oscillates left and right while the pump is activated and pumping water. This gives a sweeping motion that is commonly used when fighting fires. To do this we need two for loops. First the robot turns the pump on and wait a moment before starting the loops. The first loop sets the position to 90 which it should already be at, then it incrementally moves the position one degree at a time to around the

15-degree mark. This is the first part of the sweep and moves the aim to the right side of the robot. The second for loop sets the position to the right most position, 15 degrees, where the robot is aiming and then move servo one degree at a time all the way to the 165-degree position, or a full sweep from right to left. It then turns the pump off and set the servo back to the 90-degree position.

This is where it would be easy to make the code more complicated and have it continue sweeping while it sees fire, but instead we took a simpler approach. Since the code moves so quickly turning the pump off does not affect the firefighting abilities since it immediately reactivates the pump upon seeing fire. This means that it runs through those loops again. In practice while the robot observes a flame, it appears to be actively fighting the fire the whole time even though it is turning the pump on and off and constantly adjusting the servo to the start and end positions.

## 6.3 User Interface

The user interface is a very important aspect of any project and ours is no exception. Our user interface is simply a power supply being plugged into the micro controller. The robot needs to be kept it simple to minimize the number of points it can fail at so problems can be solved as they happen. This means that we had to remove any sort of user interface for the testing/prototyping phase and having the robot function while being plugged into a power source via USB cable. However, with that being said, it is very easy to upgrade the user interface after the fact and the user interface can be made more complex in future iterations of this same project besides just having a microcontroller as the main aspect of the user interface.

The robot needs to be programmed via USB cable wired directly into the microcontroller, as this not only codes the robot but also powers the microcontroller which powers the rest of the robot. To make the robot more mobile, all we need to do is remove the type A side of the USB cable and plug it into a power pack's output port. This needs to be done after loading the code on the robot so it has the memory of the code and can operate without being plugged into the computer. There are a variety of ways that this can be accomplished. One of these solutions is simply to attach the power pack to the robot. Whether this is done through taping or attaching with zip ties, the moment that the power pack is attached to the robot chassis, you have a fully mobile firefighting robot without any distance restrictions.

There were plans to upgrade the user interface to a switch or a button by the end of the prototyping phase. It was discussed to add multiple functions so one can choose what the robot does it via buttons, switches, or potentially an LCD touch screen display, but ultimately these were not added to the final product. All of these kinds of future upgrades were discussed but time constraints kept them from taking place. Although they may not be necessary in the strictest sense of the word, they still provide an extra level of user friendliness that is essential in building a consumer product such as this firefighting robot. As a way to summarize all of these potential future upgrades, there is a table provided below that outlines all of the different upgrade ideas that the team has come up with.

Table 15: Future Planned Modes for Robot Integrated via User Interface

| | |
|---|---|
| Sentry Mode | Remain in place and monitor the surroundings for fire. Can remain totally still with the sensors on or rotate in place to get more coverage. |
| Low Power Mode | The sensors turn off and all functions are suspended while the robot returns to a designated home base |
| Perimeter Monitoring | Follow a wall and scout the perimeter for a fire to fight |
| Bounce Mode | Travel in a straight line until it runs into a wall then it will turn a random amount and continue in a straight line |
| Consistent coverage | Starts in a corner and runs in straight lines covering an entire space to ensure there are no flames within a room |

# 7. Project Testing and Prototype Construction

The action of prototype testing is very important because the project involves a lot of different parts that could not be working properly. Manufacturing defects are common when using different and diverse components from different manufactures. A lot of different factors could cause a component to not work as intended and because of that, tests were conducted on every component required for the design in order to make sure they were all working as intended. Another fact that could increase the defects on the components and parts is that they are being handled by students with little experience on the topic. The testing of all of the components will be discussed in this section.

Some of the most important parts that were tested were the Arduino Uno microcontroller that is responsible for controlling the actions of the robot based on the programming code it has. The acquired Walfront IR Flame Sensor Module Detector detects the presence of flames through IR light. The BOJACK L293D Motor Drivers Controllers control the direction and speed of the motors based on the input received from the microcontroller. The PULACO Mini Submersible Water Pump pumps water from the container to put out the fire. And finally, the voltage regulator design is responsible for regulating the voltage and keeping it stable.

Each component was tested in different ways that varied greatly depending on the component and on what was tested. One example is the Arduino Uno microcontroller. To test the microcontroller a circuit on the breadboard was used to simulate a section of the final project and analyze the output generated by the microcontroller to check if it corresponds to what was expected from it. Another test that was conducted was a simple test where a voltage source is used to simulate the battery of the final project and a simple programming code is implemented to see the response from the microcontroller to determine if it was accurate.

This section is also going to contain information on some prototype designs and information for some of the parts of the project. Prototypes are models that resemble the functionalities and characteristics of some parts of the project that represent initial designs for some parts. The prototype models were used to test its functionalities and to make sure it worked according to what was expected. One of the prototypes commented on in this section is the PCB design for the voltage regulator that serves to regulate the voltage for the microcontroller and could have been adapted to regulate voltage for other components as well.

To conclude, in this section it is going to be covered the tests and analysis of different components that are used in for the project. Extra information on how the tests is conducted along with some analysis tables. These tables are going to contain data on the tests that are realized and their output results. Engineering knowledge and evaluation should be used when testing the different components to make sure they work as intended and can be further used in the design of the project.

# 7.1 Hardware Testing

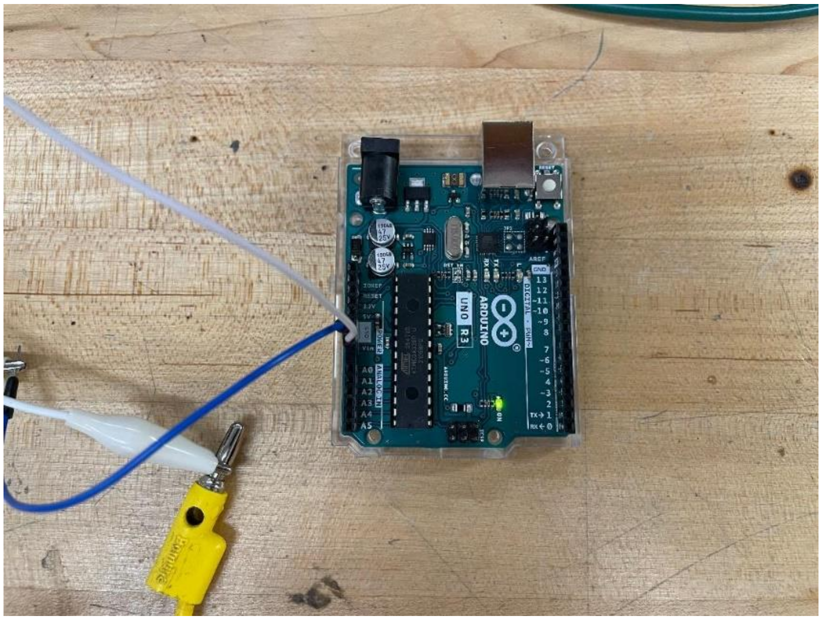
Testing is a crucial part to any design project. This ensures that the components of a design will work as expected. The testing of our parts was done in the Senior Design lab.

# 7.1.1 Arduino Uno R3

We began by testing our microcontroller, which is one of the main components of our fire extinguishing robot. The Arduino Uno R3 has a Vin input pin which we used throughout the duration of our project to connect the other components to it. We began by turning on the DC Power Supply and setting Channel 1 to 5 Volts, since this is the voltage required for many of our components. We connected this 5 Volts to the Vin pin on the Arduino using wires. We also made sure to connect the Ground pin on the Arduino. This is an important step to take as not connecting a ground on components can damage them. As shown in the images below, the Arduino powered on and was reading the pins correctly, as indicated by the green LED.

Figure 59: DC Power Supply

Figure 60: Arduino Uno R3 Powered On

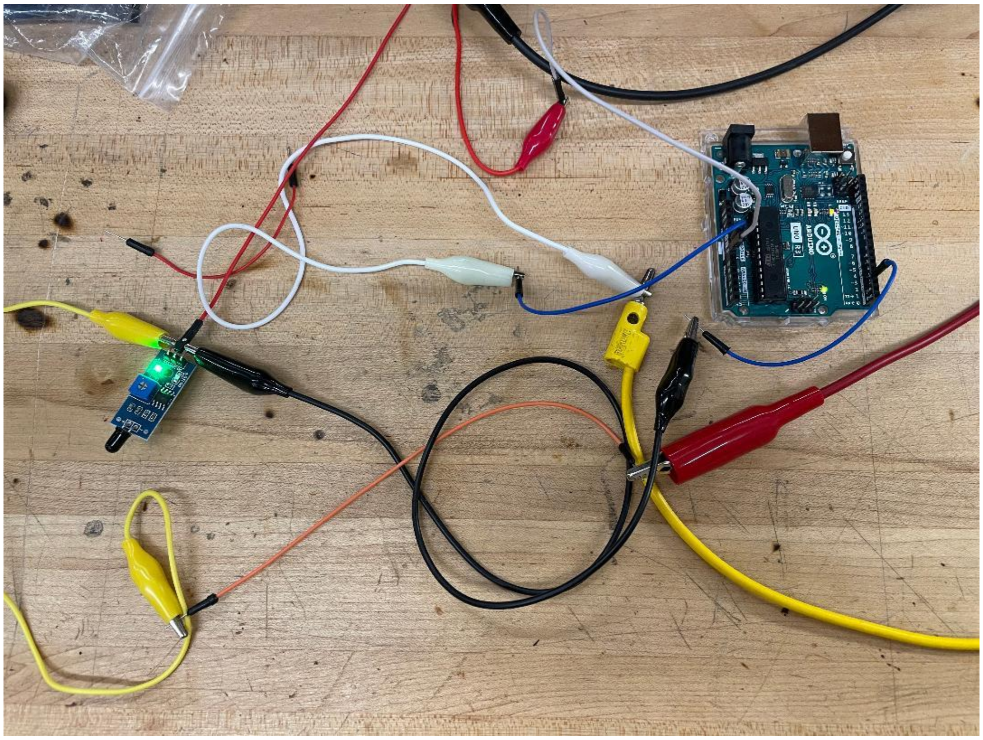



## 7.1.2 IR Flame Sensor

For the testing of our IR Flame Sensor, we wanted to make sure the pins from the Arduino to the sensor were reading correctly. We turned the DC Power Supply on and set Channel 1 and Channel 2 to both be 5 Volts, as the Arduino Uno and IR Flame Sensor will be using that voltage. The IR Flame Sensor has 3 pins, the Digital Output (DO), Ground (GND), and a voltage pin (VCC). We connected VCC to Channel 2 of the DC Power Supply, which was 5 Volts. We then connected the DO pin on the flame sensor to one of the PWM pins on the Arduino. Lastly, we made sure to ground the component. As indicated by the images below, the pins between the two components were reading correctly and each of the LEDs were on.

Figure 61: DC Power Supply

Figure 62: Arduino Uno R3 and IR Flame Sensor Connections



## 7.1.3 L293D Motor Driver

Due to supply issues, we did not receive this component in time to test during the main testing portion of the project. However, when we received this part, testing was done to ensure it was working properly. This was done by connecting the driver to the microcontroller and the motors and making sure that the wheels were working appropriately.

# 7.2 Software Testing

A very important step to any project is the testing stage, here is where we tested out the parts to make sure they function as intended and the code that we used worked properly. The code for the firefighting robot was written in the Arduino coding IDE for easier implementation directly into the microcontroller. So, for the testing stage we coded each of the parts of the robot directly: the flame sensors, motors, servo, and pump. Each of these were done separately to ensure that they function properly and code from one of the other parts does not interact poorly with a specific part. Once all of them had been properly tested they were brought together and tested together before being permanently installed on the chassis of the robot.

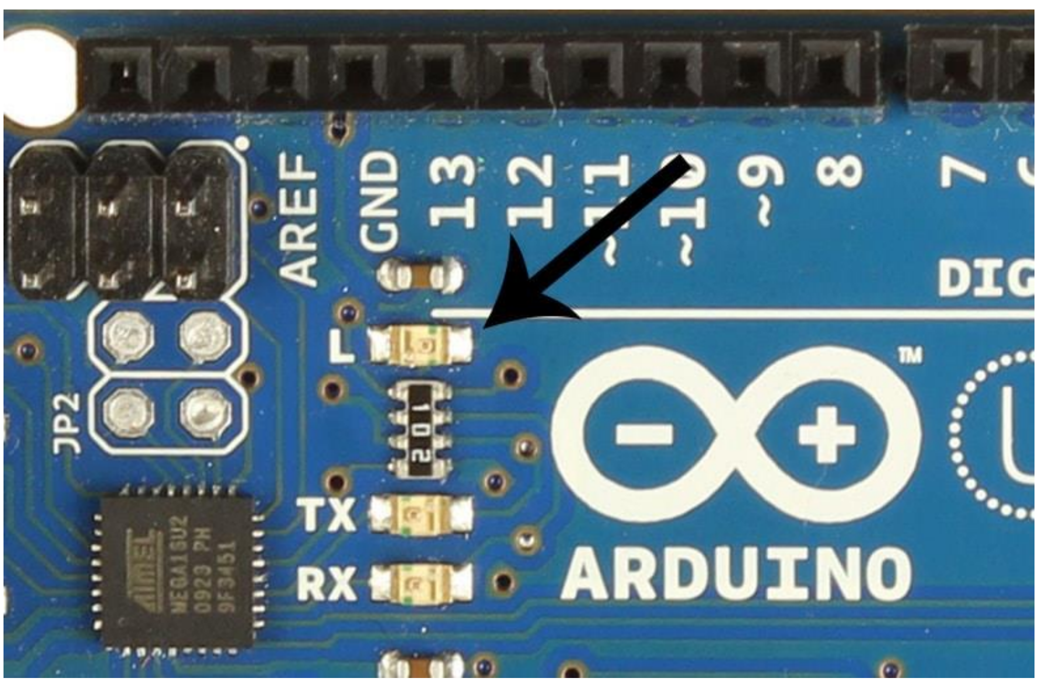
For all the testing, the initial setup had to be done in order for the parts to properly interact with the microcontroller and code. After the setup and defining the parts, we were able to do whatever we wanted to the part with code. For the purposes of testing, we wanted to do something similar to what the fully put together and functional code is designed to do. This ensured that our robot was capable of accomplishing the end goal even when in the testing phase, it just needed to be put together with the rest of the parts.

After opening the Arduino IDE, you should see some code for setup and loop, these are useful for organizing your code. First thing you want to do is go to the tools tab and then down to the board selection, make sure that the "Arduino/Genuino Uno" option is selected. Then you need to make sure you have the right COM port selected; this is the next option under the board type. There is no right answer here, it depends on where you plugged it into your PC. You can find the COM pin in your device manager under the pin's section.

## 7.2.1 Sensor Testing

The sensor testing had to be done a little differently than the rest of the testing, this is because there are no moving parts or indicators on the sensors. To compensate for this, we had to program an indicator into the microcontroller since those are the only two parts we are using for the initial testing. This was easily accomplished thanks to Arduinos built in LED lights, which gave us a way to tell if the flame sensors are seeing a fire and program a light to turn on. Each of the sensors were tested individually on the given pin that they will be assigned to.

Figure 63: Arduino Uno On Board LED




Starting with an example code makes this a little easier, so we did not have to look for the LED on the board and write the code out for it. Under the file tab you can open example code that makes the LED blink. Open this one and it loads in the code. Outside of both the setup and loop groups you need to #define your sensor and the pin that it will be associated with, then inside the setup you will pinMode the sensor as the input and the LED as the output. The LED should already be defined as the output thanks to the example code that we loaded in. Now it is as simple as creating an if-function inside the loop and then putting all the current contents of the loop inside the if-function. The conditions for the if function should be that it only activates if the sensor is HIGH or equal to 1.

This would mean nothing happens until the flame sensor sees a flame in which case it sends a high to the microcontroller and the microcontroller activates the code if loop and starts the blinking of the LED. If the sensor stops seeing the flame, then the LED stops blinking.

Now that the code was written we sent it to the microcontroller and held a flame in front of the sensor, which was done by using a lighter in front of it. If the LED blinked, then our code and part were both working, and we could sub out our pseudo code that was written for testing and sub in our real code for the robot. We made sure that all 3 sensors were tested in each of the positions so there were no conflicts, and everything was checked to be working properly.

### 7.2.2 Motor Testing

Motor testing also has a setup phase to it, but it does not have to have a work around since all you need to do to test the motors is power them on. Testing outputs is much easier than testing inputs. To set up the motors you need to use the #define macro on each of the motors for the pins they are associated with, that means even the 2 motors that aren't actually used. Next, in the setup group you need to pinMode each of the motors to outputs, so the microcontroller knows what they are. Now that they are set up it's as simple as writing code that turns them on to check if they run properly.

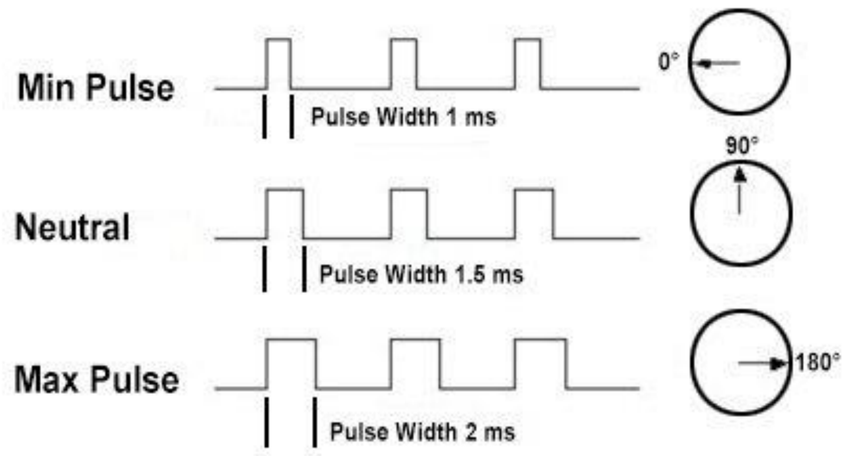
To make sure that both motors can run in tandem, and they have the same timing as each other we wrote some code with a delay between turning it on and off. Inside the loop we set both motors to LOW, this turns them on since they are off when set to HIGH. The two motors that we do not use since the robot only runs on two wheels is set to HIGH the entire time and does not change to LOW at all through the testing or prototyping phase. We wrote in a delay of 1000 before setting both motors back to HIGH. This makes the motor run for 1 second and then turn off again. After we have turned them off, we wrote in another 1000 delay to leave them off for another second and then we close the loop. This will alternate between turning both motors on for a second, then turning both motors off for a second. This lets us see if both motors work and if either of them has a timing problem or delay.

Load the code onto the robot and press play and the wheels should start spinning intermittently, if they do not then there is a problem with the code or the hardware. Double check your code and if the problem persists try another motor. If the two motors are not spinning with the same timing, then you can also try another motor or try modifying the delay of one of the motors to get them to have the same timing. However, if everything works as intended then you are done testing your motors and they should be good to go. Remove the testing code and replace it with the code for the firefighting robot.

### 7.2.3 Servo Testing

The setup for the servo testing was a little different since it is not considered an input or an output. To begin you need to use the #include macro to include servo functions into your code. Then you need to set the initial position of the servo, the actual position does not really matter you just need to declare a position integer.

Figure 64: Servo Rotation Based on Electronic Pule Duration




The testing phase for servos is very important, this is because all servos are made differently and smaller/cheaper made ones like the ones we use for the firefighting robot are more often than not made with poor standards. A few varying specifications that servos can run into from model to model and even each individual servo of the same design are as follows:

- Angular speed
- Rotation range
- Stall Torque
- Size
- Gear material/quality

Now inside the setup group you use the myservo.attach function that was included in the macro for servos, this allows you to declare the pin that it is connected to on the microcontroller. Once you have that you can write the initial position you want the servo to start at. A good starting position is 0 since we want to test the functionality of the servo, we want to test the full range of motion. Next you want to make two for loops, one loop moves the servo in a full 360-degree rotation to the left and the other does a full 360-degree rotation to the right. To do this you need to set a starting position for the first loop where it is equal to 0. The next parameter should be if for the position being less than or equal to 360, and the final parameter should be the position integer followed by two plus signs. This translated means it will start at position 0 and will add one point to the current position, or one degree, and then it will start the loop again. The servo will do a full rotation starting at position 0 and ending at position 360 or back to the position.

The second for loop should be the same thing but the first two parameters are switched, and the last parameter should be the position integer followed by two minus signs. This would activate as soon as the first loop ends since it will end at the 360 position and will start the second loop's initial requirements. Then it will return to the 0-position going the opposite direction of the first loop. This will allow us to test the full range of motion of our robot's servo which will also be the water cannon it uses for fighting fires. It is also a good idea to adjust the positions in the code so that we can test the actual position that our robot will be using. We want to ensure that our calculations for the robot’s reach are correct and testing its range of motion before fastening it to the robot for this portion is helpful. You also need to program delays into the code, or the servo will do a full rotation in less than and snap back to its original position in less than a second which is not great for observing in the testing phase.

# 7.2.4 Pump Testing

The pump was hard to test without making a mess and since we did not have a fully functioning robot and we were still plugged into a PC with an exposed microcontroller it was not safe to test the functionality of the pump with water. Instead, we tested it by plugging the part into a wall socket and testing with water in a safer environment, and then we wired it to our microcontroller board and test it with code to make sure it worked properly with our system.

Figure 65: Water Pump Showing Input and Output Ports

The first thing to do is use the #define macro to define your pump and set it to the proper pin. Next in the setup group you need to define your pump as an output, to do this use the same pinMode function that we used for the last few tests. Pumps run digital LOW when they are off so in order to turn it on, we need to do a digitialWrite for the pump output and set it to HIGH. I then would code in a delay of some sort around 2000 or 2 seconds since all we need to do is test if it comes on and test the suction on it. After writing the code send it to the microcontroller and run it. The pump should turn on and begin sucking through the input port and blowing out of the output port. To test this place your hand directly on the input port, and you should be able to feel it pulling on your hand. To test the output port just place your hand about an inch away from it and let it blow on your hand. If both ports are working properly, you can remove the testing code and reprogram the board with the code for the working firefighting robot since we are now done with all the testing phases of the software.

## 7.3 Prototype PCB Designs and Testing

A printed circuit board is the core of electronic devices and it is a very important aspect in many design projects. Prototyping is a very efficient way to test various types of hardware and software functionalities and it allows for a visual of a design to be examined.

### 7.3.1 Voltage Regulator

Maintaining the voltage of the system within certain limits is essential to keep the components working as intended and not compromising the functionalities of the system. For these reasons, the voltage regulators are used to control the voltage from a power supply and make sure it is maintained within certain operating values for different parts of the circuit design. There are different types of voltage regulators that are used depending on the required application. Some of the most used types of voltage regulators are the linear voltage regulator and the switching voltage regulator.

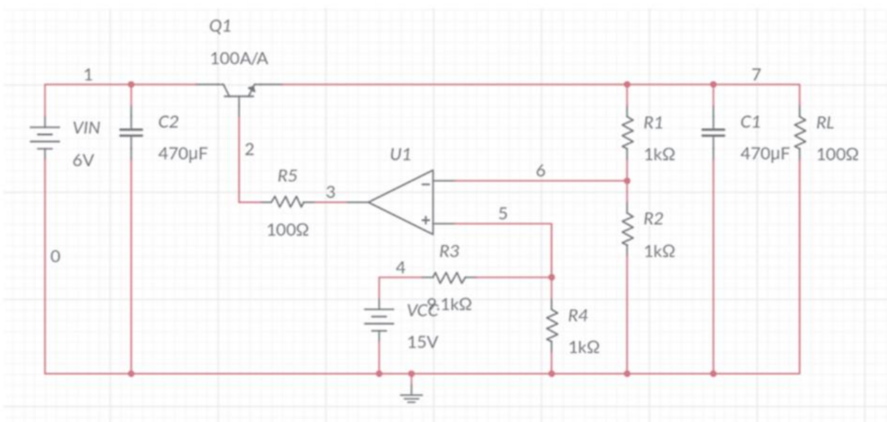
#### Linear Voltage Regulators

The linear voltage regulators are very common for their simplicity and ease to use. They are generally good options to use when powering low power circuits or circuits where the power supply that is going to be regulated is not that different of voltage from the regulated value. Due to it having low complexity and being easy to use, one of the tradeoffs of using a linear voltage regulator is that it is inefficient when compared to the other types of voltage regulators. The reason that they are inefficient when compared to other regulators is that the difference between the input voltage and output voltage is dissipated exclusively as heat. The main advantages of using a linear regulator are that it generates low current, it regulates the output very consistently and it requires a small number of external components to be implemented. The aspects of the linear regulator are its low efficiency, its power and heat dissipation from the wasted input voltage and its power capacity.

# Testing Linear Voltage Regulator

Once we researched the linear regulator configuration we also went to the lab and tested the design. First, we simulated the schematic of the circuit using the Multisim software to see if the design would work. The design that we simulated using Multisim and later on built it on the breadboard can be seen in the figure below.

Figure 66: Schematic of the Linear Voltage Regulator




Once we simulated the circuit and confirmed its functionality, we went on to build the circuit on the breadboard to analyze its response. Due to the availability of components at the time, the regulator built had a regulated voltage of 3.5 V. The circuit on the breadboard and the signal captured from the circuit can be seen in the following figures below.

Figure 67: Linear Voltage Regulator on the Breadboard

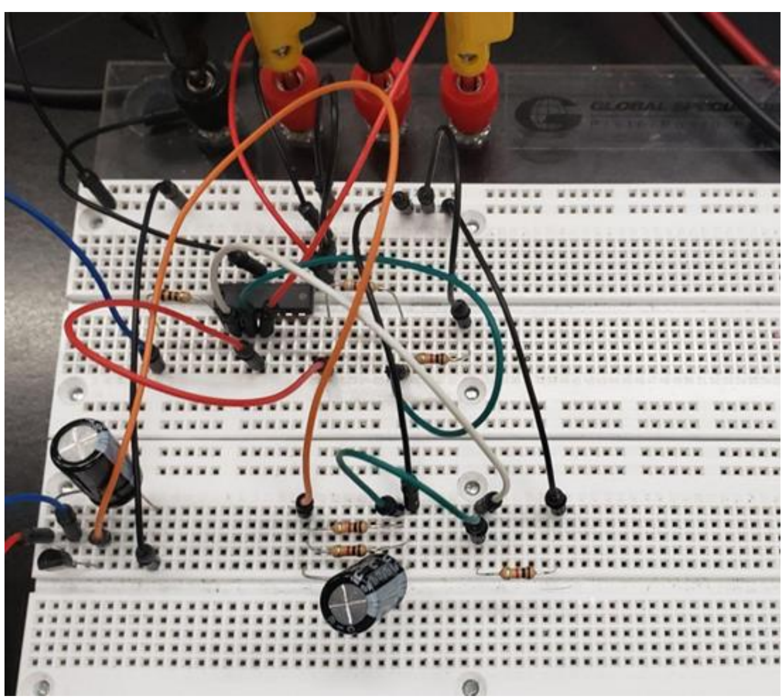
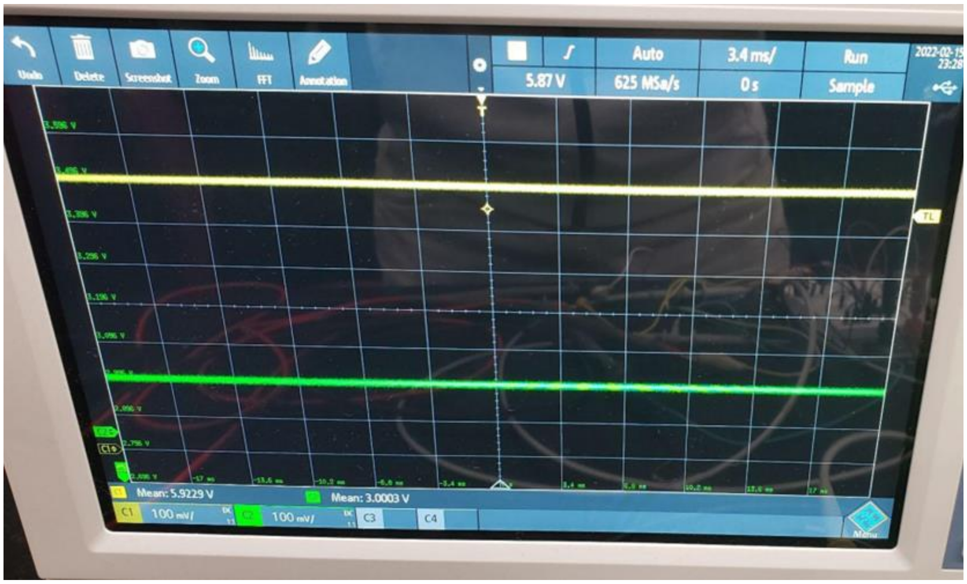


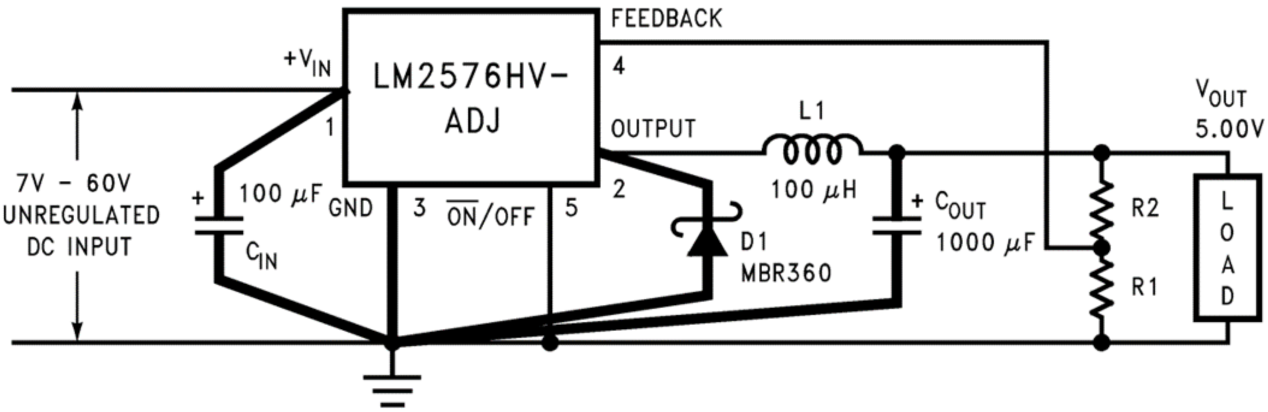
Figure 68: Simulation of the Linear Voltage Regulator




# Switching Regulators

The switching regulators are different than the linear regulators in the way that they are used to regulate the voltage from the input power supply. As the name suggests the switching regulators use the switch technology to regulate the voltage of the input voltage to the desired voltage level. The FETs are used as switches for this operation. The FETs are responsible for storing the voltage of the input and later releasing it to generating only the required amount needed to power the selected function from the design. This switching technology is what allows the switching regulator to have a higher efficiency when compared to the linear regulator. Due to the switching technology, the regulators also have close to zero power dissipation. The reason for that is that the power that is not used is not wasted and instead it is stored and can be used later to provide power to the system. Another difference between the linear regulator and switching regulator is that the switching regulator can generate output voltage signals that are even higher or in opposite polarity from the input signal. That can happen due to the design flexibility that can be used in different configurations such as the boost, buck, buck-boost, flyback, isolated inverted and isolated non-inverting configurations. Being able to use this different configuration is what increases the flexibly of the switching regulators. The main downside of this type of regulator is the fact that it has higher noise when compared to the linear regulator and it has a higher complexity. However, these downsides were worthy tradeoffs when considering the positive gains from using the switching regulators. The main positive aspects of the switching regulators include its efficiency along with its low power dissipation and its ability to be used for high power applications.

Figure 69: Voltage Regulator Schematic Obtained from the National Semiconductor LM 2576-ADJ Datasheet




To design the 5 V switching voltage regulator the figure above was used as a basis. The design has been done using the AutoDesk Eagle software. The switching voltage regulator has an input voltage of around 12 V and regulaates it down to 5 V. The 5 V is used for different aspects of the design including powering up the Arduino Uno microcontroller. The components used and their values can be seen in the list below:

- Resistor (R1) with a resistance value of 1 kΩ.
- Resistor (R2) with a resistance value of 3.1 kΩ.
- Capacitor ($C_{in}$) with a capacitance value of 100 μF.
- Capacitor ($C_{out}$) with a capacitance value of 1000 μF.
- Inductor (L1) with an inductance value of 100 μH.
- MBR360 Diode (D1).
- LM2576HV-ADJ Simple Switcher Step-Down Voltage Regulator.

Using the components above the following schematic and PCB design was created using the Autodesk Eagle software.

Figure 70: Schematic of the Voltage Regulator Designed Using the Eagle Software

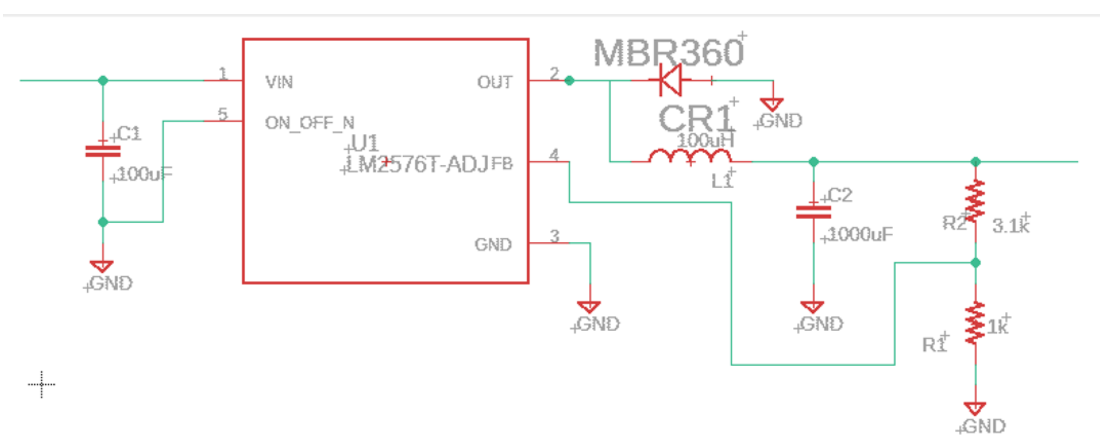
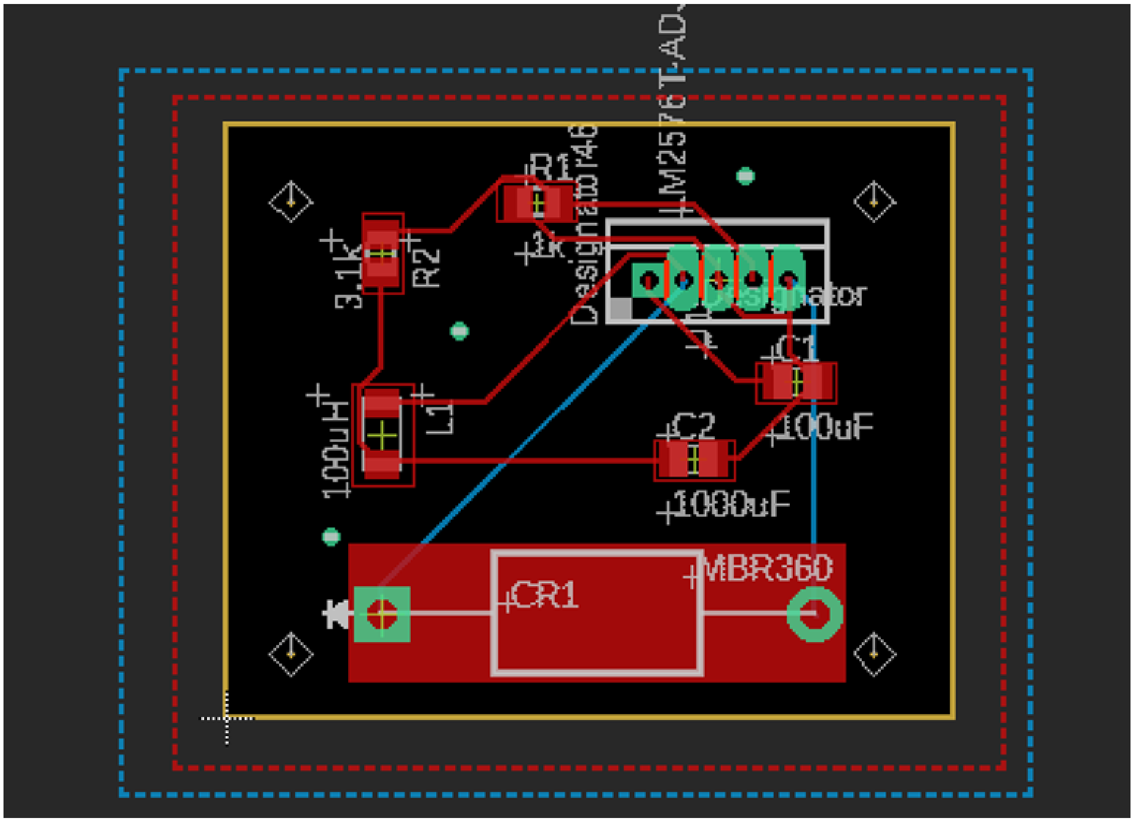


Figure 71: PCB Design of the Switching Voltage Regulator




# Testing Switching Voltage Regulator

After going over the characteristics of the switching voltage regulator we went on to simulate the circuit that is shown in the schematic for this part of the design. After simulating the design and confirming its functionalities we went on to build the circuit on the breadboard to confirm its functionality using the components available in the lab. The circuit built on the breadboard and the 5 V regulated output signal from the circuit can be seen in the following figures.

Figure 72: Switching Voltage Regulator on the Breadboard

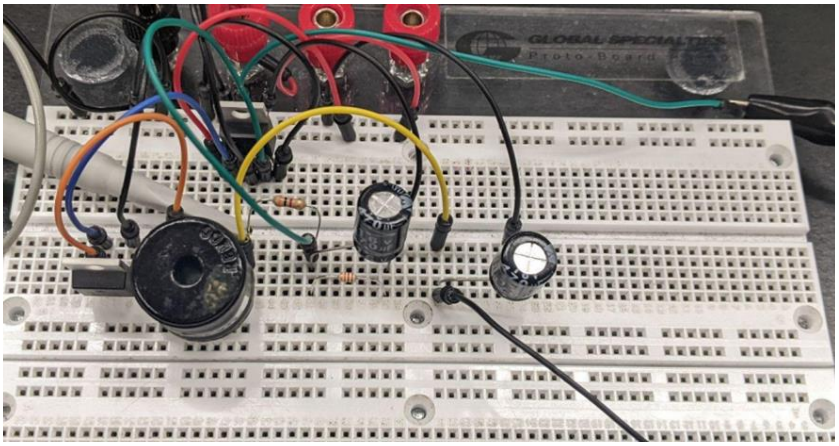
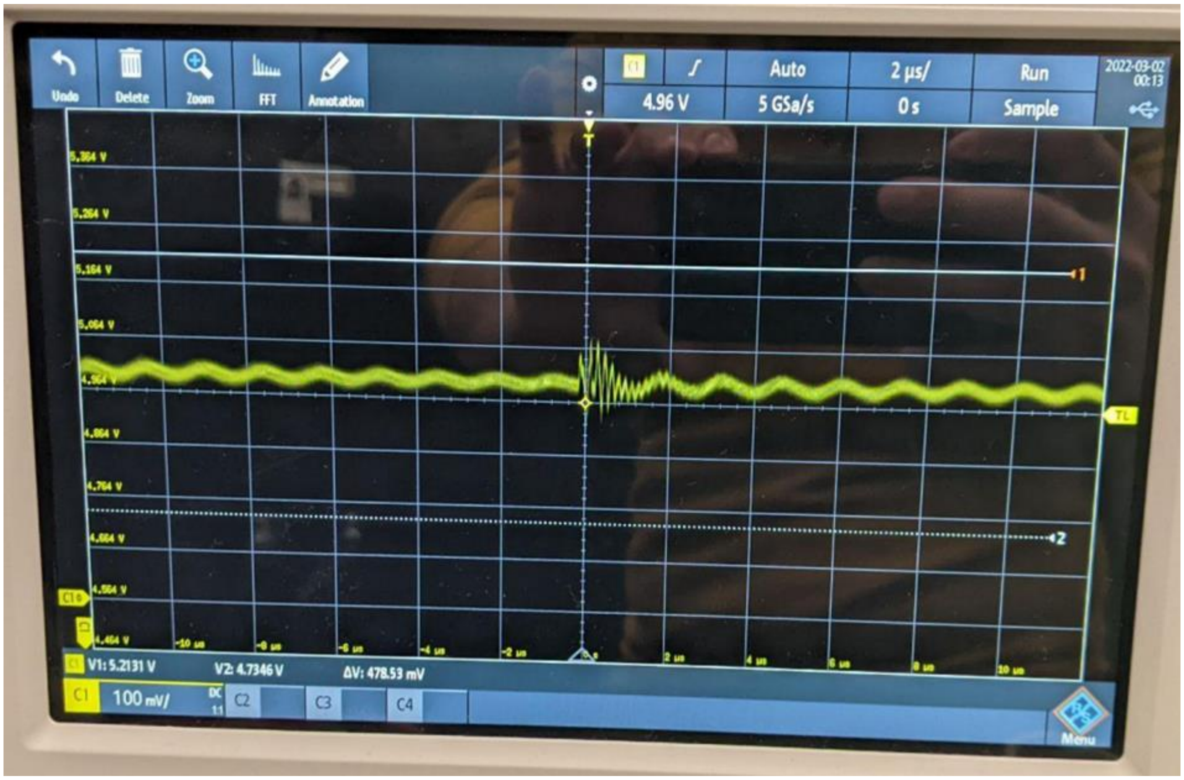


Figure 73: Simulation of the Switching Voltage Regulator




Update: After making some changes to the power design of our project in Senior Design 2, we decided that there was no need to use a voltage regulator in our system. We tested the project prototype with a 5 V battery that we used and that was sufficient to power the entire project. Because of that we decided on not using the 12 V power supply and use a 5 V power supply instead. Since everything uses 5 V and we are using a 5 V power supply we decided to remove the regulator from the design since it wasn’t needed anymore. In addition, the Anker PowerCore 5 V Power Bank that we are using as our power supply has a 5 V voltage regulator included in it so that helped us to decide on the removal of the regulator from our design since it was already included in the power supply.

## 7.3.2 Overall Project Design

Shown below are three different images of our overall project design for the sake of providing a thorough visual design. The first image depicts a breadboard view of our fire extinguishing robot and provides a visual means for the various components in our project as well as a rough outline of their connections. The second image is the overall schematic. This schematic shows all the important connections that need to be made between all of the hardware parts in order to have a fully functioning robot. It connects all the parts and allows them to function together. The last image and diagram is the PCB of the fire extinguishing robot that also shows the interconnected hardware pieces through a third visual means.

All of this was done in the Fritzing software due to issues with Eagle after multiple attempts to make it work. Everything shown below was discussed in detail in other sections of this document, including but not limited to, the various components and their functionality, how to make the connections between the parts, discussion of the data sheets, etc. These sections can be examined and scrutinized closely in comparison to the given figures to see consistency of build and design.

Figure 74: Breadboard Representation of Project Design

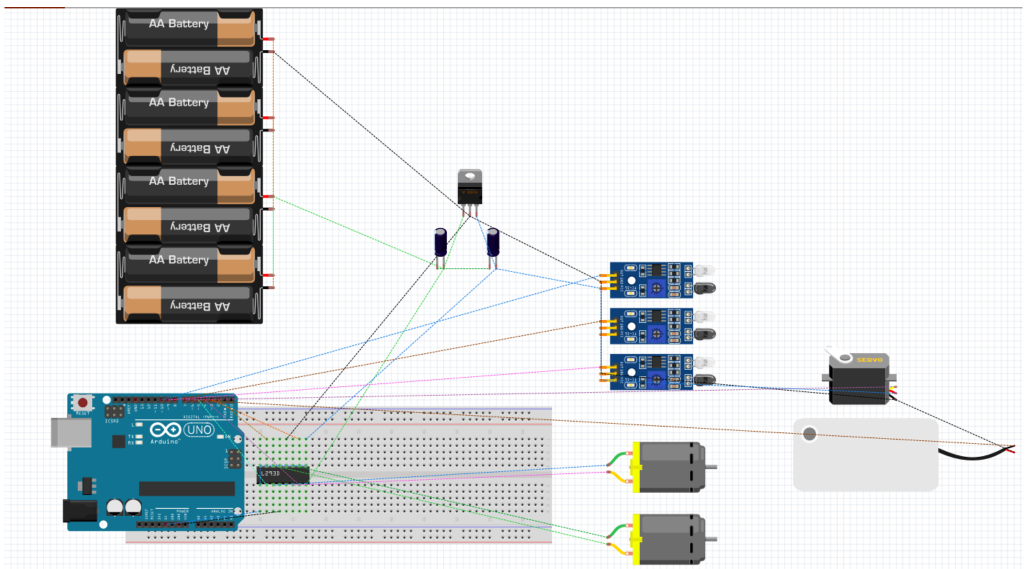



Figure 75: Schematic of Project Design

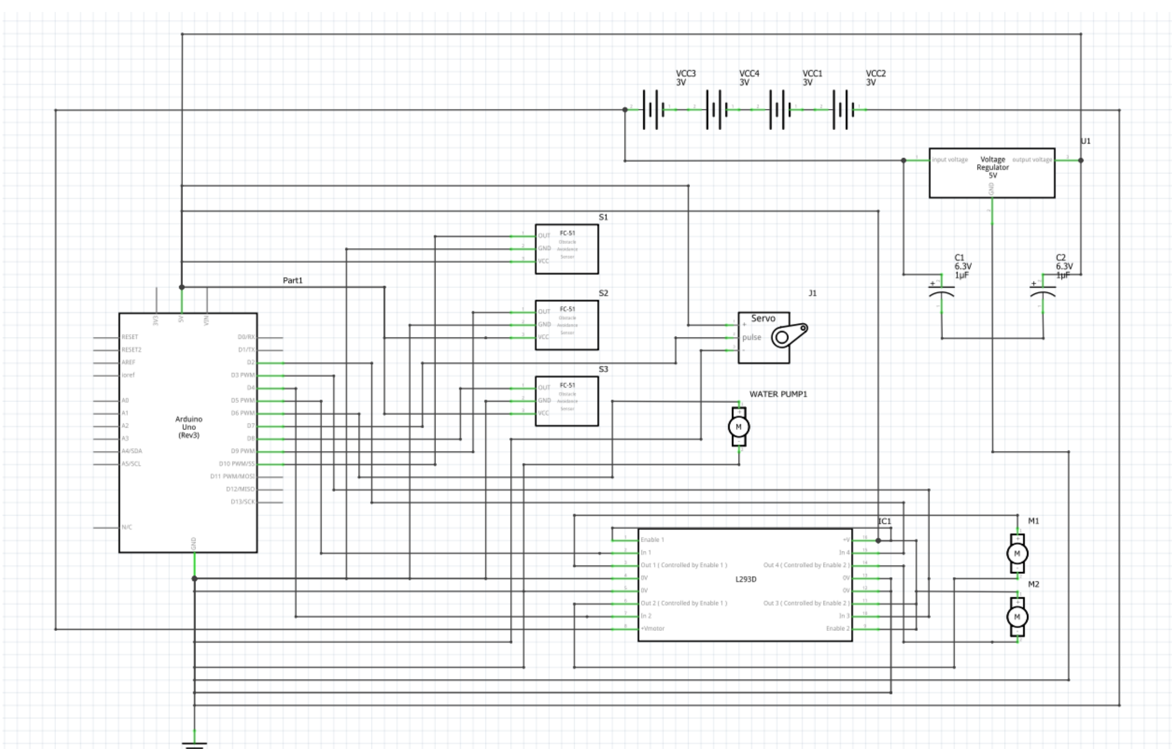


Figure 76: PCB of Fire Extinguishing Robot

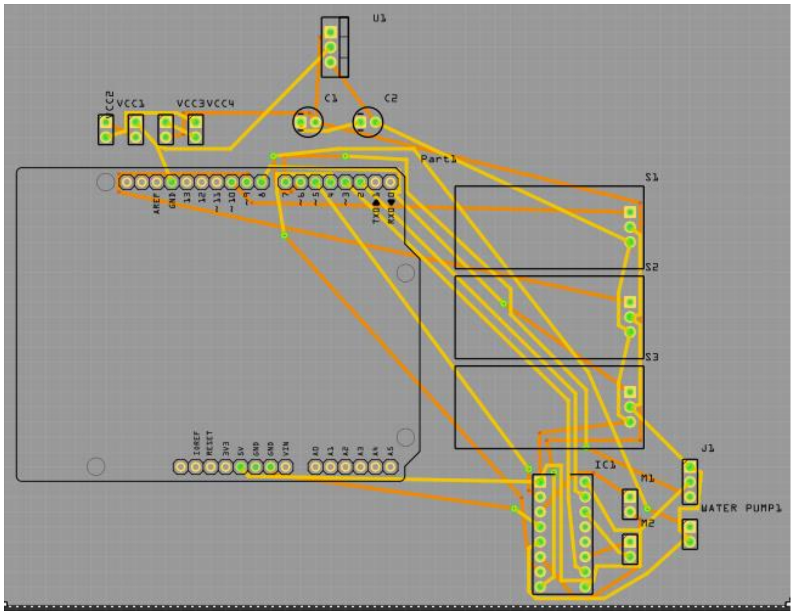



Figure 77: Final Schematic of Project Design

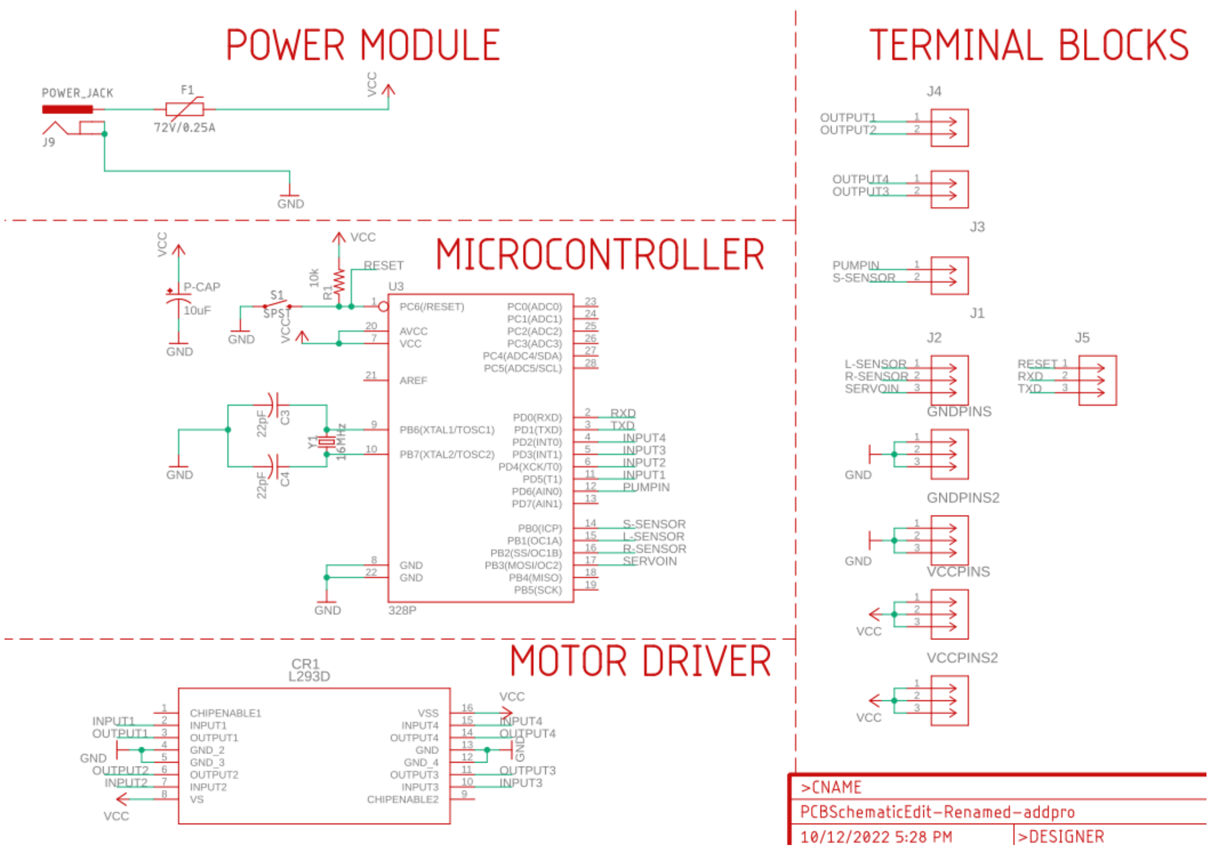
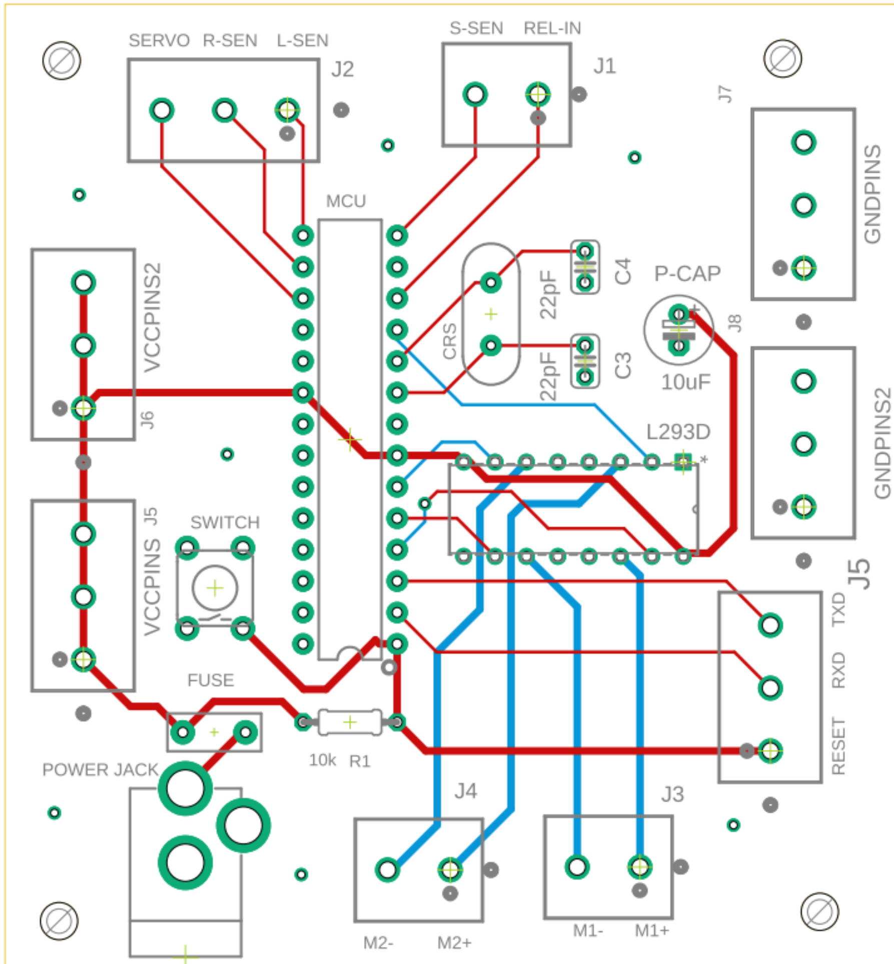


Figure 78: Final PCB of Fire Extinguishing Robot



# 8. Administrative Content

This section outlines and highlights key milestones that need to be met in order to have a fully functioning robot by the given deadline. In greater detail, this section will cover the key administrative content for our project including but not limited to: milestones, budget/financial, project roles, design problems, and what we plan to do moving forward. The milestone section will discuss each of the tasks were completed and those that we wanted to complete but did not get to before the end of senior design 2. The budget and finances section will discuss how we broke up the cost of the project and the breakdown of the costs associated with the project itself. The budget section is a necessary topic of discussion for every project design as it determines whether or not the project will be able to be completed with each individual's financial situation. The project roles are a vital part in the project as each person had to be able to contribute in order to get the task done. The project roles section will cover how we planned to divide up the workload for senior design one and senior design two so that we all had a course load that we were able to comfortably handle, which took into account things like individual class schedules and our knowledge/skills with certain aspects of the project.

While problems certainly did come about throughout the process of this, like any other, design project, it is an important stepping stone to overcome and help each individual grow as an engineer. The next section discusses the problems that we were facing when designing the firefighting robot in all areas including hardware, software, and even economical. It is always good to have an end goal in mind. With that being said, there are always improvements or changes to be made. The final section looks at what we planned to do moving forward past senior design one. When we were at the end of senior design one and had a good idea of what our project would look like when completed, what were some things that we want to implement moving forward? Upgrades and the like will be discussed in the looking forward section. The process of completing a design project is always a learning experience and a lot can be taken away from the end results.

## 8.1 Milestone Discussion

This section breaks down the tasks that needed to be completed in Senior Design I and Senior Design 2. These tasks were not only important to complete in order to have a working project, but they were also necessary to graduate from The University of Central Florida with a degree in engineering. Milestones are an especially important topic to discuss as this course in its nature is time sensitive. Our group chose to take senior design one in the spring and follow up with senior design two in the fall. This gave us an advantage to strategize over the summer semester so that we could be ready to take on the project as soon as we got into senior design two in the fall. The tables below highlight the important due dates/major tasks that needed to be accomplished. The ordering of the parts and the testing of our project will also be discussed in a later section. We highlight the assembly of our robot and any redesigns that may come about.

## Senior Design I

Senior design one mostly had to do with researching our topic, creating a game plan, and preparing ourselves for senior design two. For this reason, most of the milestones will be due dates for the various divide and conquer submissions that we turned in. However, there are some other important milestones that we accomplished throughout the semester. Some of which include team meetings, project breakthroughs, and testing dates for the components of the project. The table below highlights some of these milestones in chronological order.

Table 16: Tasks for Senior Design I

| **Task** | **Date** |
|---|---|
| First team meeting | January 21, 2022 |
| Senior Design Bootcamp | January 20, 2022 |
| Second team meeting | January 28, 2022 |
| Project topic decided (firefighting robot) | January 28, 2022 |
| Senior Design Bootcamp assignment/quiz due | January 28, 2022 |
| Divide and Conquer Version 1 | February 4, 2022 |
| Meeting with Dr. Wei | February 7, 2022 |
| Third team meeting | February 11, 2022 |
| Divide and Conquer Version 2 | February 18, 2022 |
| Fourth team meeting | February 25, 2022 |
| Fifth team meeting | March 4, 2022 |
| Assignment on Standards | March 11, 2022 |
| Sixth team meeting | March 18, 2022 |
| Quizzes due | March 18, 2022 |
| Senior Design I Documentation (60 pages) | March 25, 2022 |
| Meeting with Dr. Wei | March 28, 2022 |
| Seventh team meeting | April 1, 2022 |
| Hardware Testing Complete | April 3, 2022 |
| Software Testing Code Complete | April 6, 2022 |
| Updated Report (100 pages) | April 8, 2022 |
| Eighth team meeting | April 15, 2022 |
| Final team meeting | April 22, 2022 |
| Final Document (120 pages) | April 26, 2022 |

## Senior Design II

Senior design two is inherently different than senior design one because it shifts our focus away from research and begins the stage of creating and testing, as well as fixing any problems and potentially upgrading the project further into senior design two if there is time and the team has the skills required to do so. This is our final timeline that we updated as we went along. Below is the task blueprint with our dates as to when in the Fall semester we had our tasks completed.

Table 17: Tasks for Senior Design II

| Task | Date (exact dates TBA) |
|---|---|
| Get Remaining Parts | August 29, 2022 |
| First Meeting | August 29, 2022 |
| Testing Hardware | August 30, 2022 |
| Testing Software | August 30, 2022 |
| Fully Code Firefighting Robot | August 30, 2022 |
| Have all Hardware Functional | September 12, 2022 |
| Design PCB | September 20, 2022 |
| Finalize Prototype | October 7, 2022 |
| Finalize Build for Midterm Demo | October 27, 2022 |
| Midterm Demo | October 31, 2022 |
| Test PCB | November 1, 2022 |
| Finalize Robot build | November 1, 2022 |
| Finish Testing and Modifying | November 8-18, 2022 |
| Final Demo/Presentation | November 23, 2022 |

# 8.2 Budget and Finance Discussion

Examining and planning out a budget for a design project is always one of the initial steps that needs to be taken in order for the project to be successful.

This project was financed by our project members. We split any of the costs evenly between the four of us. Some members already had parts we needed for the project so that was not included in the costs. The prices and components are subject to change in the future and adjustments can be made accordingly.

Our project was without a sponsor, that means that for the financial support of the project the group members are the sole benefactors. This severely limited our potential for parts to lower end parts and a smaller robot. In the case that we found it possible to upgrade the size of the robot, which provided a benefit that was substantial enough to order all new parts, then we were willing to attempt a larger, more financially burdening, robot.

As discussed in some previous sections, we had some stretch goals and potential upgrades that would have increased the prices of many, if not all, the parts. The price sheet we have created is based on the small prototype robot that we initially created as well as the final robot design and our printed circuit board. The following table shows the total costs for all the aspects to our project design. It includes components we bought in Senior Design 1 and Senior Design 2 in order to complete our fire extinguishing robot. Overall, our budget was a little more than we initially planned but it was still manageable for our group.

Table 18: Overall Price for Project Design

| Part Description | Quantity | Price |
|---|---|---|
| Arduino Uno REV3 | 1 | $22.79 |
| Arduino Software | 1 | Free |
| Eagle CAD Software | 1 | Free Trial |
| Flame Sensor | 15 | $11.68 |
| Servo Motor | 5 | $10.99 |
| L293D Motor Driver | 10 | $13.87 |
| Mini DC Submersible Water Pump | 4 | $12.13 |
| 12 V DC Motor | 2 | $6.89 |
| Driver + 2 DC Motor Set | 1 | $18.79 |
| DC Motor and Wheel Set | 1 | $12.50 |
| Low Voltage Pump | 2 | $7.90 |
| Relay | 2 | $7.27 |
| Robot Chassis | 1 | $13.49 |
| Ceramic Capacitor | 230 | $12.99 |
| 16 MHz Crystal | 10 | $6.99 |
| ATmega328P-PU | 2 | $29.99 |
| USB 2.0 Male Plug 2pin Wire | 2 | $8.99 |
| Push Button | 50 | $7.36 |
| USB to DC Power Cord | 1 | $9.19 |
| DC Barrel Power Jack | 5 | $8.90 |
| 3-Pin and 2-Pin Connector | 13 | $25.00 |
| Reset Fuse | 5 | $7.73 |
| Anker PowerCore Power Bank | 1 | $41.99 |
| Total | | $297.43 |

# 8.3 Project Roles

The project roles are a very important topic to discuss as we wanted to choose things that made each person comfortable and played towards their strengths and weaknesses. Each role was picked by the person who is fulfilling that role. Everyone didn't get to do exactly what they wanted but they chose to do the roles they were assigned, nonetheless. It was important that we did it this way to minimize tension and have an expected accuracy for the final product that could be achieved in a minimum amount of time. This reduced the amount of time each person works on the project because they know their role and what they are expected to produce by each of the deadlines. Our group got very lucky, and we ended up with an even split of Electrical engineering students and Computer engineering students, this way we were able to divide up the hardware and software work of the project to best suit each person. Most of this information is outlined in the block diagrams that were created in section 2.4.3.

To do this we needed to outline what steps we had to take to create the final product of the firefighting robot. A very generic outline is shown in the second part of the milestone's discussion. This goes over each of the steps we wanted to take in the stages of senior design two. Here's a quick outline of what we need to accomplish to finish the initial prototype.

- Build the robot from hardware parts following circuit diagram
    - Power supply
    - Motors
    - Sensors
    - Pump
    - Servos
- Test the hardware
- Interface the hardware with the Arduino Uno
- Model and create a chassis
- Test the interfacing of the software and hardware
- Code the robot
    - Sensors/input
    - Motors, pump, and servo/output

To start with the two Electrical Engineering students, Sydney and Gabriel handled the hardware of the robot and testing. Sydney focused on the power supply and its connection to the two drivers which will control the motors. Gabriel focused on setting up the sensors with an optimal field of view using the provided field of view of each sensor. He also connected them to the power supply and the pump and its connection to the power supply. The two of them used circuit diagrams for the connections. Each of their portions will be connected via the breadboard for the protype.

After the two electrical engineering students connected and tested each of their portions of the robot, it was passed off to Juan who focused on the interfacing of the Arduino Uno with the hardware. This means he set up the connections to the microcontroller from the breadboard. To do this he followed the pin assignments that were created in a previous section. After the microcontroller had been connected to each of the hardware components, the physical robot was done, and passed off to Noah.

Noah then took the robot that had been fully constructed and tested all the parts individually using the testing code that was written in the research phase of the project. After that he programed it using the code discussed in the Software section of this report. Once the robot was coded and tested, he began designing a chassis cover for the robot using OnShape. The chassis cover sits on top of any electric parts to hopefully keep them from potentially getting wet from the water spraying and to keep any wires from getting snagged in the process of moving.

# 8.4 Project Design Problems

During the research phase we encountered very few problems with the design as expected. The issues tended to arise when we got to the prototyping stage of the project. However, there were a few issues in the research phase of the project. One issue we ran into was the licensing for one of the softwares that we needed to use. A common software to build and test circuits is Autodesk's Eagle PCB Design and Electrical Schematic Software. We attempted to use it for our circuit design, but we ran into an issue with licensing. Eagle is supposed to have a free subscription for students, but we kept running into an issue that wouldn't allow us to get the free subscription. This may have been due to all of us taking Junior Design and using the software for our project schematics. The issue persisted and we had to find another way to create and test our circuits without using the software that we planned to use because of familiarity with it.

Another issue we ran into rather frequently was part availability. When we were ordering our parts to try to do some hardware and software testing, we kept running into issues with the parts being back ordered because of the supply chain problems. There was a lot that happened this semester with the supply chain, and it caused manufacturers to have problems producing and even shipping and importing/exporting products. When we attempted to order our motor driver for testing there were issues with the manufacturer that backlogged our order all the way to May 2022. This meant that there was no way for us to get our part in time to have our testing done and written about in our final paper by the end of Senior Design I. Luckily, we have examples of testing motor drivers and were prepared to have the information logged and the product tested as soon as we got our hands on it in May.

# 8.5 Looking Forward

A very important aspect of any project is the future of that project, you never want to stay locked in the present or only do the bare minimum of what is required. This project is a huge resume boost and probably the most experience we will be getting in the engineering field outside of an internship or co-op. So, the team wanted to do something that not only got us a passing grade in Senior Design I and Senior Design II but also something that challenged us and provides something beneficial to our portfolios.

Looking forward there were quite a few things we planned to do with our project. The team had discussed various upgrades for the robot. The main goal we had in the final months of Senior Design I and the first few months of Senior Design II was finishing testing and getting the prototype functioning, this needed to be completed before we start considering implementing any potential upgrades that we had thought of. However, after our robot was running properly, we needed to start looking at upgrading our basic robot. The team had big plans for upgrading the robot so it was good for us to look at it early so that we could plan for what we would like to do in the future. In this section we will discuss some of the potential upgrades that we had in mind for the robot.

The initial prototype had no on/off switch. We were planning to remedy this issue with a switch or button of some sort, but because of the way the Arduino Uno functions, the robot has full functionality while plugged into the computer that it is being coded from. When programming the robot, you send the code to the microcontroller, and it will automatically start running it while it is plugged into a power source. The computer that it is plugged into is a power source so that is the first stage power source. Eventually when the external power source was added after programming the robot, the firefighter had free range to move anywhere it likes. However, this means that as soon as the power source is plugged in it starts running the robot's code. The solution can be added in one of two ways, we could have add a switch that cuts off the power source somewhere within the circuit, or we could have gotten a power source that comes with an on off switch built into it that will cut the voltage supply off when activated. Another upgrade that we talked about was programming the robot to have several modes. This heavily depends on the amount of bug fixes and testing that is required in the software testing phase and other tasks that must be carried out by the programmer. The programming is almost entirely done by one team member, so adding different modes is somewhere in between a long shot and a doable feature that we can add. Adding different modes to the robot would have required quite a bit of modification not only to the software but also to the hardware of the robot as well.

Table 19: Future Planned Modes for Robot Integrated via User Interface

| | |
|---|---|
| Sentry Mode | Remain in place and monitor the surroundings for fire. Can remain totally still with the sensors on or rotate in place to get more coverage. |
| Low Power Mode | The sensors turn off and all functions are suspended while the robot returns to a designated home base |
| Perimeter Monitoring | Follow a wall and scout the perimeter for a fire to fight |
| Bounce Mode | Travel in a straight line until it runs into a wall then it turns to a random amount and continue in a straight line |
| Consistent coverage | Starts in a corner and runs in straight lines covering an entire space to ensure there are no flames within a room |

Some of these require an extensive amount of coding and hardware upgrades. Going through the list we can evaluate how much of a change we need for each mode. Any modes that we add would have required a mode selection in the hardware whether that be a switch, an LCD display, or even a knob of some sort. The sentry mode would have been the easiest mode to add to the robot, by adding a small loop to the code we could have had the wheels rotate left and right increasing the field of view of the robot by having it turn repeatedly while it doesn't sense a flame. The low power mode would have had to monitor the battery level and have a GPS device somewhere on the robot, so it knows where it is and where it needs to move to in order to return to the home base. Perimeter monitoring would have needed an ultrasonic sensor so that the robot could evaluate its distance from the wall and maintain that distance while moving. A bounce mode would have also needed an ultrasonic sensor but would have most likely required more than one for it to function efficiently. A consistent coverage mode could have been hard programmed if you had years to map out the room in that way but would have most likely required a mix of GPS and ultrasonic sensors to dodge obstacles and properly follow the lines in the room. This is because motors tend to lose accuracy when turning so hard programming the 180 degrees turn would lose accuracy as the battery drains.

The final upgrade the team discussed was a ratchetting system for us to use fire extinguishers for the robot instead of water. As we have discussed throughout the paper, there are various kinds of fires and water is not good at putting out most of them. To make the robot more flexible with the fires it fights we came up with the idea of replacing the water reservoir with small fire extinguishers. To do this upgrade our robot would have to be remarkably bigger than we are currently planning on having it but would allow you to use a different extinguisher depending on the place you plan to have the robot. If, for example, you plan on having the robot in your kitchen you can use a class B extinguisher which would efficiently be able to put out a grease fire. There was also talk about having the robot hold multiple classes of extinguishers so it could select the type of extinguisher to use that would most effectively put out the flames. We would have to have a way for the robot to determine which class of fire it is looking at so it would be able to determine the type of extinguisher to use.

# 9. Conclusion

The problem of unwanted fires has always been a problem for the human person, and over millennia, this problem has not gone away despite all the different ways the human race has tried to eradicate it. It likely will never be a problem that could be permanently dispensed with, but this reality, instead of destroying the hope of solving it, should actually spur people into imaginative solutions that could very well dimmish the problem to a negligible minimum. Our group has attempted to combat this problem through our iteration of a fully functional, self-driven, wireless firefighting robot. In an attempt to highlight our approach, this documentation went through a series of eight chapters that thoroughly explored the progression of the project from its infancy to the beginnings of the first prototype stage. It compiled the set of data that was used from the beginning to set up a foundational framework for the project based on a series of decisions made to focus the project on a given problem statement.

After determining the specific focus of the problem statement, the document took on the challenge of determining a plan of action in figuring out very detailed specifications. Certain decisions had to be made without any reasonable direction for the sake of giving the group a path to focus on. These specifications were made with the idea of a plan in mind, and following these specifications, the plans could be laid out looking forward.

Following this, a large amount of data was collected through the use of articles and research tools in order to develop an organized idea of what the final project could realistically look like. Not only was there a need to dive deep into the research of the types of fires, but also how the universal problem of unwanted fires has been traditionally and recently addressed through the use of modern technology. This led us to question what current technologies could be used not only as inspiration but also as foundations upon which to build, improve, or from which change the direction of the focus. With this in mind, the parts were then carefully chosen after having research a very extensive list of motors, drivers, microcontrollers, sensors, pumps, etc. This provided the group with the direction we needed to begin designing the actual hardware and software, but not before taking necessary security measures.

This prompted the group to look into the requirements that are described by the set of American Standards, which then provided the group with a set of boundaries upon which the rest of the hardware and software could be built upon. The hardware and the software were then outlined thoroughly in chapters five and six, and with these two aspects of the project fully resolved, the rest of the documentation looked at the way in which both could be married to one another to work as a cohesive whole and the testing of all the parts. With all of this ready, the end of the documentation placed the group in a position ready to begin the prototypal stage of the project. Looking forward from that point, there was much that could potentially change in the final product, but this set up the team with a great foundation to begin assembling, confident of the research that has gone into the robot, with high expectations of the functionality and effectiveness of the idea, that before this documentation process, seemed a faraway hope of addressing this ever-present danger of unwanted fires.

# 10. Appendices

This section shows all the references used throughout this paper. In depth research was conducted in order to get a proper documentation of our design project. A section for permissions also follows to show permission for material used that was not created by our group members.

## 10.1 Bibliography

Staples, P. (2019, March 2). *Robots used in Everyday Life*. Sciencing. Retrieved April 15, 2022, from https://sciencing.com/robots-used-in-everyday-life-12084150.html

*House fire statistics and facts (data from 2022) - the zebra*. (n.d.). Retrieved April 15, 2022, from https://www.thezebra.com/resources/research/house-fire-statistics/

*Fire extinguishing robot using Arduino - irjet.net*. (n.d.). Retrieved April 15, 2022, from https://www.irjet.net/archives/V8/i6/IRJET-V8I6583.pdf

*General monitors flame detector selection guide*. (n.d.). Retrieved April 15, 2022, from https://www.aft.net/wp-content/uploads/2018/10/General-Monitors-Flame-Detector-Selection-Guide.pdf

*All about flame sensors (flame rods, optical sensors, and Visual Flame Imaging)*. (Flame Rods, Optical Sensors, and Visual Flame Imaging). (n.d.). Retrieved April 15, 2022, from https://www.thomasnet.com/articles/instruments-controls/all-about-flame-sensors/

Chakraborty, S. (2022, March 28). *Flame sensor: Principle: 4 important types: Application*. Lambda Geeks. Retrieved April 15, 2022, from https://lambdageeks.com/flame-sensor/

*Total fire protection for business & industry | associated ...* (n.d.). Retrieved April 15, 2022, from http://www.associatedfire.net/source/documents/Products/Detection-Devices/FlameDetectionSystems.pdf

Babu, P. S., & Akram, M. (2018, October 6). *Flame detectors working principle*. Inst Tools. Retrieved April 15, 2022, from https://instrumentationtools.com/flame-detectors-working-principle/

*The basics of Flame Sensors*. BDC. (2016, March 29). Retrieved April 15, 2022, from https://www.gobdc.com/basics-flame-sensors/

What is Arduino Uno? A getting started guide. (n.d.). Retrieved April 15, 2022, from https://www.rs-online.com/designspark/what-is-arduino-uno-a-getting-started-guide#:~:text=Arduino%20UNO%20is%20a%20low,and%20motors%20as%20an%20output.

*Motor driver IC*. Technology Robotix Society, IIT Kharagpur. (n.d.). Retrieved April 15, 2022, from https://2020.robotix.in/tutorial/auto/motor_driver/

*The use of robotics in Firefighting*. EKU Online. (2020, July 22). Retrieved April 15, 2022, from https://safetymanagement.eku.edu/blog/the-use-of-robotics-in-firefighting/#:~:text=Firefighting%20Technology,of%20fire%20control%20and%20suppression.

Atsolution. (2021, January 27). *The four different types of fire detectors: Fire protection*. A Total Solution. Retrieved April 15, 2022, from https://www.atotalsolution.com/blog/fire-protection-the-four-types-of-fire-detectors/#:~:text=Ionization%2FPhotoelectric%20%E2%80%93%20A%20favorite%20of,detected%20as%20soon%20as%20possible.

*Tech in 2020: How the fire service leveled up its technology game*. FireRescue1. (2020, December 26). Retrieved April 15, 2022, from https://www.firerescue1.com/fire-products/technology/articles/tech-in-2020-how-the-fire-service-leveled-up-its-technology-game-emYpkAml4WrUybD8/

Macfos. (2021, February 3). *What is the difference between l293d and L298n motor driver?* Robu.in | Indian Online Store | RC Hobby | Robotics. Retrieved April 15, 2022, from https://robu.in/what-is-the-difference-between-l293d-and-l298n-motor-driver/

ago, yida 3 years, Yida, posts, yidaS. author's, & posts, S. author's. (2020, October 30). *L298, a dual H-Bridge Motor Driver Module*. Latest Open Tech From Seeed. Retrieved April 15, 2022, from https://www.seeedstudio.com/blog/2019/10/08/l298-all-about-l298-motor-driver/#:~:text=Characteristics%20of%20L298,-The%20main%20features&text=High%2DVoltage%20and%20Current%20full,like%20DC%20and%20Stepper%20Motors.&text=Has%20a%20high%2Dcapacity%20filter,current%20of%20an%20inductive%20load.

*L298N motor driver module*. Components101. (n.d.). Retrieved April 15, 2022, from https://components101.com/modules/l293n-motor-driver-module

*L293d dual H-Bridge Motor Driver*. ProtoSupplies. (2022, March 12). Retrieved April 15, 2022, from https://protosupplies.com/product/dual-h-bridge-motor-driver-l293d/#:~:text=KEY%20FEATURES%20OF%20L293D%20DUAL,4.5%20to%2036V%20motor%20voltage

*L293D motor driver IC*. Components101. (n.d.). Retrieved April 15, 2022, from https://components101.com/ics/l293d-pinout-features-datasheet

Inc, E. P. (2022, April 15). *All you need to know about L293D: Custom*. Maker Pro. Retrieved April 15, 2022, from https://maker.pro/custom/projects/all-you-need-to-know-about-l293d

*Raspberry pi 3*. Components101. (n.d.). Retrieved April 15, 2022, from https://components101.com/microcontrollers/raspberry-pi-3-pinout-features-datasheet

*Power consumption benchmarks*. Power Consumption Benchmarks | Raspberry Pi Dramble. (n.d.). Retrieved April 15, 2022, from https://www.pidramble.com/wiki/benchmarks/power-consumption

Wikimedia Foundation. (2022, March 27). *Ti MSP430*. Wikipedia. Retrieved April 15, 2022, from https://en.wikipedia.org/wiki/TI_MSP430#:~:text=The%20MSP430%20does%20not%20have,large%20buffers%20or%20data%20tables.

Basics of MSP430. (n.d.). Retrieved April 15, 2022, from https://www.rs-online.com/designspark/basics-of-msp430

Admin. (2021, October 14). *Arduino Uno Board: Features and real time applications*. WatElectronics.com. Retrieved April 15, 2022, from https://www.watelectronics.com/arduino-uno-board-tutorial-and-its-applications/

Last Minute Engineers. (2020, December 18). *In-depth: Control DC motors with L293D Motor Driver IC & Arduino*. Last Minute Engineers. Retrieved April 15, 2022, from https://lastminuteengineers.com/l293d-dc-motor-arduino-tutorial/

*L293D motor driver IC*. Components101. (n.d.). Retrieved April 15, 2022, from https://components101.com/ics/l293d-pinout-features-datasheet

*DC powered pumps information*. DC Powered Pumps Selection Guide: Types, Features, Applications | Engineering360. (n.d.). Retrieved April 15, 2022, from https://www.globalspec.com/learnmore/flow_control_flow_transfer/pumps/dc_pumps

*Amazon.com: Dicrey Micro Water Pump Mini submersible pump ...* (n.d.). Retrieved April 15, 2022, from https://www.amazon.com/Dikley-Micro-Submersible-Fountain-Aquarium/dp/B0791FSR2S

*Pulaco DC 5V USB mini submersible water pump for pet fountain, aquarium fish tank, pond, fountain, hydroponics*. Sparrow Pets. (n.d.). Retrieved April 15, 2022, from https://sparrowpets.com/product/pulaco-dc-5v-usb-mini-submersible-water-pump-for-pet-fountain-aquarium-fish-tank-pond-fountain-hydroponics/

Isaac. (2020, August 7). *Water pump for Arduino: Everything you need to know*. Hardware libre. Retrieved April 15, 2022, from https://www.hwlibre.com/en/bomba-de-agua-arduino/

*Best 5V arduino water pumps*. ThePlantBot.com. (2020, February 16). Retrieved April 15, 2022, from https://theplantbot.com/arduino-water-pump/

Team, T. A. (n.d.). *Uno R3: Arduino documentation*. Arduino Documentation | Arduino Documentation. Retrieved April 15, 2022, from https://docs.arduino.cc/hardware/uno-rev3

Administrator. (2021, September 4). *Arduino Flame Sensor Interface - working, circuit diagram, code*. Electronics Hub. Retrieved April 15, 2022, from https://www.electronicshub.org/arduino-flame-sensor-interface/

Admin. (2020, November 18). *IR Infrared Flame Sensor Module: How fire sensor works " electroduino*. ElectroDuino. Retrieved April 15, 2022, from https://www.electroduino.com/ir-infrared-flame-sensor-module/

Security, K. F. &. (2022, March 1). *What are the different types of fire extinguishers & their uses*. Blog. Retrieved April 15, 2022, from https://blog.koorsen.com/what-are-the-different-types-of-fire-extinguishers-their-uses

*Safe fire direct*. Safe Fire Direct. (n.d.). Retrieved April 15, 2022, from https://www.safefiredirect.co.uk/Page/32/carbon-dioxide-(c02)-fire-extinguisher.aspx

Ansul. (n.d.). Retrieved April 15, 2022, from https://www.ansul.com/en/us/pages/ProductDetail.aspx?productdetail=CLEANGUARD%2BClean%2BAgent%2BExtinguishers

*How to build a house of quality with technical and competitive benchmarking*. ASQ. (n.d.). Retrieved April 15, 2022, from https://asq.org/quality-resources/house-of-quality

Kelechava, B. (2021, January 22). *The current C programming language standard - ISO/IEC 9899:2018 (C18)*. The ANSI Blog. Retrieved April 15, 2022, from https://blog.ansi.org/2018/11/c-language-standard-iso-iec-9899-2018-c18/#gref

GeeksforGeeds. (2021, May 6). *C programming language standard*. GeeksforGeeks. Retrieved April 25, 2022, from https://www.geeksforgeeks.org/c-programming-language-standard/

electronic-notes.com. (n.d.). *USB standards: USB 1, USB 2, USB 3, USB 4 - capabilities & comparisons*. Electronics Notes. Retrieved April 25, 2022, from https://www.electronics-notes.com/articles/connectivity/usb-universal-serial-bus/standards.php

*IEC 60086 Battery Standard Testing and certification - TUV SUD*. TUVSUD. (n.d.). Retrieved April 25, 2022, from https://www.tuvsud.com/en-us/services/product-certification/iec-60086-battery-standard

Pate, S., Hosch, M., & Indgjer, M. (n.d.). *Meeting NFPA standards for Flame, smoke and ... - det-tronics*. hazardex. Retrieved April 25, 2022, from https://www.det-tronics.com/content/documents/Meeting_NFPA_Standards-Fire_Safety_AR-1123.pdf

Sagar. (2021, April 14). *Applying IPC-2221 standards in circuit board design*. Sierra Circuits. Retrieved April 25, 2022, from https://www.protoexpress.com/blog/ipc-2221-circuit-board-design/

*AGV and Amr Robot Home*. AGV network. (n.d.). Retrieved April 25, 2022, from https://www.agvnetwork.com/r15-08-safety-amr

Araujo, K. (n.d.). *Battery standards*. Epec Engineered Technologies. Retrieved April 25, 2022, from https://www.epectec.com/batteries/battery-standards.html

Kumar, S. (2018, August 2). *Interfacing flame sensor with Arduino to build a fire alarm system*. Circuit Digest. Retrieved April 25, 2022, from https://circuitdigest.com/microcontroller-projects/arduino-flame-sensor-interfacing

*Amazon.com. spend less. smile more.* Amazon. (n.d.). Retrieved April 25, 2022, from https://www.amazon.com/

*Buy Electronics Components Online in India*. QuartzComponents. (n.d.). Retrieved April 25, 2022, from https://quartzcomponents.com/

Huxley, A. (n.d.). adafruit. Retrieved April 25, 2022, from https://www.adafruit.com/

*Save money. live better*. Walmart.com. (n.d.). Retrieved April 25, 2022, from https://www.walmart.com/

*AGV and Amr Robot Home*. AGV network. (n.d.). Retrieved April 25, 2022, from https://www.agvnetwork.com/r15-08-safety-amr

*Amazon.com. spend less. smile more.* Amazon. (n.d.). Retrieved April 25, 2022, from https://www.amazon.com/

Araujo, K. (n.d.). *Battery standards*. Epec Engineered Technologies. Retrieved April 25, 2022, from https://www.epectec.com/batteries/battery-standards.html

*Buy Electronics Components Online in India*. QuartzComponents. (n.d.). Retrieved April 25, 2022, from https://quartzcomponents.com/

electronic-notes.com. (n.d.). *USB standards: USB 1, USB 2, USB 3, USB 4 - capabilities & comparisons*. Electronics Notes. Retrieved April 25, 2022, from https://www.electronics-notes.com/articles/connectivity/usb-universal-serial-bus/standards.php

GeeksforGeeds. (2021, May 6). *C programming language standard*. GeeksforGeeks. Retrieved April 25, 2022, from https://www.geeksforgeeks.org/c-programming-language-standard/

Huxley, A. (n.d.). adafruit. Retrieved April 25, 2022, from https://www.adafruit.com/

*IEC 60086 Battery Standard Testing and certification - TUV SUD*. TUVSUD. (n.d.). Retrieved April 25, 2022, from https://www.tuvsud.com/en-us/services/product-certification/iec-60086-battery-standard

Kumar, S. (2018, August 2). *Interfacing flame sensor with Arduino to build a fire alarm system*. Circuit Digest. Retrieved April 25, 2022, from https://circuitdigest.com/microcontroller-projects/arduino-flame-sensor-interfacing

Pate, S., Hosch, M., & Indgjer, M. (n.d.). *Meeting NFPA standards for Flame, smoke and ... - det-tronics*. hazardex. Retrieved April 25, 2022, from https://www.det-tronics.com/content/documents/Meeting_NFPA_Standards-Fire_Safety_AR-1123.pdf

Sagar. (2021, April 14). *Applying IPC-2221 standards in circuit board design*. Sierra Circuits. Retrieved April 25, 2022, from https://www.protoexpress.com/blog/ipc-2221-circuit-board-design/

*Save money. live better*. Walmart.com. (n.d.). Retrieved April 25, 2022, from https://www.walmart.com/

American National Standards Institute. (n.d.). *IBR standards hosted by ANSI*. List of Standards Organizations Providing Standards Incorporated by Reference. Retrieved April 22, 2022, from https://ibr.ansi.org/Standards/Default.aspx

Bartleson, K. (2013, May 14). *10 standards organizations that affect you (whether you ...* Electronic Design. Retrieved April 22, 2022, from https://www.electronicdesign.com/technologies/communications/article/21796419/10-standards-organizations-that-affect-you-whether-you-know-it-or-not

*IPC Standards*. IPC International, Inc. (2021, October 5). Retrieved April 22, 2022, from https://www.ipc.org/ipc-standards

Wikimedia Foundation. (2022, April 13). *International Organization for Standardization*. Wikipedia. Retrieved April 22, 2022, from https://en.wikipedia.org/wiki/International_Organization_for_Standardization

Hussaini, U. (2020, April 4). *Four ways to power up the arduino uno*. Technobyte. Retrieved April 22, 2022, from https://technobyte.org/2016/07/power-up-the-arduino-uno/

Dee, J. (n.d.). *How to Power a Project*. Spark Fun. Retrieved April 22, 2022, from https://learn.sparkfun.com/tutorials/how-to-power-a-project/all

Fauzan. (2017, June 17). *Power supply to L293D Motor Driver*. Arduino Stack Exchange. Retrieved April 22, 2022, from https://arduino.stackexchange.com/questions/39617/power-supply-to-l293d-motor-driver

MYTECTUTOR. (2020, July 25). *L293D motor driver control of DC and Stepper*. YouTube. Retrieved April 22, 2022, from https://www.youtube.com/watch?v=PnnLw9KX3SQ

Staff. (2021, September 29). *Which rechargeable battery is best for your arduino project?* DigiKey. Retrieved April 22, 2022, from https://www.digikey.com/en/maker/blogs/2021/which-rechargeable-battery-is-best-for-your-arduino-project#:~:text=and%20cost%2Deffective.-,Summary,have%20their%20benefits%20and%20drawbacks.

Wikimedia Foundation. (2020, April 3). *Lithium-Ion Battery*. Wikipedia. Retrieved April 22, 2022, from https://en.wikipedia.org/wiki/Lithium-ion_battery

Wikimedia Foundation. (2020, September 21). *Lithium polymer battery*. Wikipedia. Retrieved April 22, 2022, from https://en.wikipedia.org/wiki/Lithium_polymer_battery

Wikimedia Foundation. (2022, March 23). *Nickel–cadmium battery*. Wikipedia. Retrieved April 22, 2022, from https://en.wikipedia.org/wiki/Nickel%E2%80%93cadmium_battery

Wikimedia Foundation. (2022, March 29). *Nickel–metal hydride battery*. Wikipedia. Retrieved April 22, 2022, from https://en.wikipedia.org/wiki/Nickel%E2%80%93metal_hydride_battery

*Arduino infrared collision avoidance* . Rhydolabz. (n.d.). Retrieved April 22, 2022, from https://www.rhydolabz.com/documents/26/IR_line_obstacle_detection.pdf

*Flame Sensor Module*. OSEPP. (2022). Retrieved April 22, 2022, from https://www.osepp.com/electronic-modules/sensor-modules/60-flame-sensor-module

*Turnigy 5000mah 3s 20c Lipo Pack W/XT-60*. Hobbyking. (n.d.). Retrieved April 22, 2022, from https://hobbyking.com/en_us/turnigy-5000mah-3s-20c-lipo-pack-xt-90.html

*Jameco*. (n.d.). Retrieved April 22, 2022, from https://www.jameco.com/Jameco/Products/ProdDS/836086.pdf

Ultra Librarian. (n.d.). *Texas Instruments LM2576T-adj*. Texas Instruments LM2576T-ADJ: Symbol, Footprint, 3D STEP Model. Retrieved April 22, 2022, from https://app.ultralibrarian.com/details/15e2c26c-103f-11e9-ab3a-0a3560a4cccc/Texas-Instruments/LM2576T-ADJ

Ultra Librarian. (n.d.). *Vishay MBR360*. Vishay MBR360: Symbol, Footprint, 3D STEP Model. Retrieved April 22, 2022, from https://app.ultralibrarian.com/details/57565ccd-109c-11e9-ab3a-0a3560a4cccc/Vishay/MBR360

Renesas Electronics. (n.d.). *Linear vs. Switching Regulators*. Renesas. Retrieved April 22, 2022, from https://www.renesas.com/us/en/products/power-power-management/linear-vs-switching-regulators

# 10.2 Permissions

SB Sydney Brown
Mon 4/18/2022 7:43 PM
To: james.careless@gmail.com

Hello Mr. Careless,

I am an electrical engineering student at the University of Central Florida. I am writing to request your permission to use figures from your article (Tech in 2020: How the fire service leveled up its technology game (firerescue1.com)) in a design paper that will not be published.

Thank you

Sydney Brown
sydneymbrown@knights.ucf.edu

Tech in 2020: How the fire service leveled up its technology game - FireRescue1

The Los Angeles Fire Department introduced its new firefighting robot at a press conference Tuesday. RS3 can blast 2,500 gallons of water or foam per minute and is designed to battle fires in ...

www.firerescue1.com

## Permission to use figures

JC James Careless <james.careless@gmail.com>
Tue 4/19/2022 8:50 AM
To: Sydney Brown

Hi Sydney,

Please go ahead.

Yours,

jc

...

James Careless
613-778-8681
***Please note my new email address that I am moving to: james@jamescareless.com***
http://www.jamescareless.com/

Reply | Forward

## Permission to use figures

ElectronicsHub.org <admin@electronicshub.org>
Mon 4/18/2022 10:51 PM
To: Sydney Brown

Yes, of course you can.

...

Reply | Forward

Sydney Brown
Mon 4/18/2022 7:52 PM
To: admin@electronicshub.org

Hello,

I am an electrical engineering student at the University of Central Florida. I am writing to request your permission to use figures from your article (Arduino Flame Sensor Interface - Working, Circuit Diagram, Code (electronicshub.org)) in a paper for Senior Design that will not be published.

Thank you

Sydney Brown
sydneymbrown@knights.ucf.edu

## Contact Form

Fields marked with an * are required

**Name ***

Sydney Brown

**Email ***

sydneymbrown@knights.ucf.edu

**Message ***

Hello,

I am an electrical engineering student at the University of Central Florida. I am writing to request your permission to use figures from your article (https://lastminuteengineers.com/l293d-dc-motor-arduino-tutorial/) in a paper for Senior Design that will not be published.

Thank you

**What is thirteen minus 6? ***

7