Chapter 1

Introduction

1.1 Fire Fighting Vehicle:

With the development in the field of Vehicles, human intrusion has become less and Vehicles are being widely used for safety purpose. In our day-to-day life, fire accidents have become common and sometimes may lead to hazards that make it hard for the firemen to protect human life. In such cases, a fire fighting Vehicle is used to guard human lives, wealth, and surroundings from the fire accidents. This firefighting Vehicle project is an advanced project for engineering students, who are interested in Vehicles. This project incorporates RF technology for remote operation and also uses 8051 microcontroller. A firefighting Vehicle is capable of detecting fire if a house catches fire while someone in the house is either sleeping or not present in the house. By means of this firefighting Vehicle, people and properties can be saved from fire accidents.

A Vehicle is an automated device which performs functions usually attributed to humans or machines tasked with repetitive or flexible set of actions. Numerous studies have shown that Vehicle can be beneficial in medicine, rehabilitation, rescue operation and industry. Over the years, Vehicles has been introduced in various industries. The industrial Vehicles are multi-function manipulators designed for more specialized materials, divisions, gadgets or devices through various programmatic movements to perform various tasks. In line with the Fourth Industrial Revolution (4IR), there is demand for a one system that can control, communicate and integrate different Vehicles regardless of their types and specifications. Machine learning has also heated up interest in Vehicles, although only a portion of recent development in Vehicles can be associated with machine learning. Recent Vehicle development project has embedded machine learning algorithms to increase the intelligence in Vehicles. This will increase the productivity in industry while reducing the cost and electronic waste in a long run.

Studies on the use of humanoid Vehicles are actively carried out to minimize firefighters' injuries and deaths as well as increasing productivity, safety, efficiency and quality of the task given. Vehicle can be divided into several groups such as Tele-Vehicles, Telepresence Vehicles, Mobile Vehicles, Autonomous Vehicles and Androids Vehicles. Telepresence Vehicle are similar to a tele-Vehicle with the main difference of providing feedback from video, sound and other data. Hence, tele-presence Vehicles are widely used in many fields requiring monitoring capability, such as in child nursery and education, and on improving older adult's social and daily activities. Mobile Vehicle is designed to navigate and carry out tasks with the intervention of human beings. Meanwhile, autonomous Vehicles can perform the task independently and receive the power from the environment, as opposed to android Vehicles which are built to mimic humans.

In this paper, a firefighting Vehicle is proposed. The main function of this Vehicle is to become an unmanned support vehicle, developed to search and extinguish fire. There are several existing types of vehicles for firefighting at home and extinguish forest fires. Our proposed Vehicle is designed to be able to work on its own or be controlled remotely. By using such Vehicles, fire identification and rescue activities can be done with higher security without placing fire fighters at high risk and dangerous conditions. In other words, Vehicles can reduce the need for fire fighters to get into dangerous situations. Additionally, having a compact size and automatic control also allows the Vehicle to be used when fire occurs in small and narrow spaces with hazardous environments such as tunnels or nuclear power plants. Thermite and Fire Rob are two current available fire fighter Vehicles that have been used widely in industry. Thermite (produced by Howe and Howe Technologies Inc) is a firefighting Vehicle that uses a remote control and can operate as far as 400 m. It can deliver up to 1200 gpm of water or 150 psi of foam. The size of this Vehicle is 187.96 cm x 88.9 cm x 139.7 cm. This Vehicle powers up to 25 bhp (18.64 kW) using a diesel engine. The main component in the design of this Vehicle are multi-directional nozzle that is backed by a pump that can deliver 600 gpm (2271.25 l/min). This Vehicle is designed for use in extreme danger areas, such as planes fires, processing factories, chemical plants or nuclear reactors.

Fire Rob (Manufactured by Croatian manufacturer DOK- ING) is a fire-fighting vehicle controlled by a single operator via remote control. It extinguishes fire without intervention of fire fighters with a high pressure on a hydraulic arm that pumps water up to 55 m away. It also can carry 1800 liter of water and 600 liter of foam in its two on board tanks. The coating on Fire Rob allows it to withstand critical temperature of 250ºC and thermal radiation of 23 kW/m for a period of 30 minutes.

In this study, a compact and small firefighter Vehicle has been developed. This Vehicle is named QRob, which is short form of Rescue Vehicle. This Vehicle can evade obstacles, search and extinguish fire. Furthermore, this Vehicle can increase the productivity, safety, efficiency and quality of the task given. QRob is more compact and more flexible compared to Thermite and Fire Rob Vehicle. Another advantage of QRob is in its ability to enter location with small entrance or narrow space.

1.2 Objectives:

i) To construct firefighting Vehicle.

ii) To reduce human effort for firefighting.

iii) To ensure safe about human life.

1.3 Working of Fire Fighting Vehicle Project:

There are several possibilities of fire in any remote area or in an industry. For instance, in garments god owns, cotton mills, and fuel storage tanks, electric leakages may result in immense fire & harm. In the worst of cases & scenarios, fire causes heavy losses both financially and by taking lives. Vehicles is the best possible way to guard human lives, wealth and surroundings. A Firefighting Vehicle is designed and built with an embedded system. It is capable of navigating alone on a modeled floor while actively scanning the flames of fire. The Vehicle could be used as a path guide in a fireplace device or, in normal case, as an emergency device. This Vehicle is designed in such a way that it searches a fire, & douses it before the fire could spread out of range & control. This type of firefighting Vehicle will sooner or later work with firefighters, thus greatly reducing the danger of injury to victims. Apart from this, this Firefighting

Vehicle project will also help generate interest along with the innovations in the field of Vehicles while operating towards a sensible and obtainable solution to save lives and mitigate the danger to property.

The main intention of this project is to design a fire fighting Vehicle using Android application for remote operation. The firefighting Vehicle has a water tanker to pump water and spray it on fire; it is controlled through wireless communication. For the desired operation, 8051 microcontroller is used.

In the proposed system, an android application is used to send commands from the transmitter end to the receiver end for controlling the movement of the Vehicle in forward, backward, right or left directions. At the receiver side, two motors are interfaced to the 8051 microcontroller wherein two of them are used for the movement of the vehicle and the remaining one to place the arm of the Vehicle. Remote operation is done by android OS based Smartphone or tablet. The Android device transmitter acts as a remote control with the advantage of being having adequate range, while the receiver has a Bluetooth device fed to the microcontroller to drive DC motors through the motor driver IC for particular operation. Further, this project is developed by interfacing it with a wireless camera so that the person controlling it can view the operation of the Vehicle remotely on a display.

Chapter 2

LITERATURE REVIEW

2.1 Thermite RS1-T4 (1,250 GPM):

Thermite RS1-T4 is the most capable, durable and reliable firefighting Vehicle on the market. The RS1-T4 is remotely operated with a belly pack controller that provides high-definition and infrared video in real-time, allowing the machine to traverse hazardous terrain with maximum control. Manufactured by leaders in Vehicle innovation, RS1-T4 has the power to push vehicles from its path, the agility to climb stairs and the ability to output a maximum 1,250 gpm at 200 psi. Fabricated using durable construction-grade components, the RS1-T4 can navigate challenging terrains and withstand exposure to extreme elements. Thermite is designed as an advanced tool to help operators combat fires safely and efficiently.

Figure 2-1: Thermite RS1-T4 (1,250 GPM) in application field

SPECIFICATIONS:

- Kohler 24 hp water cooled diesel engine
- Nozzle options 600-1,250 gpm max
- Remote & tethered controls
- Durable industrial components

- Industrial grade tracks
- Live video feed via widescreen camera
- Zero degree turning radius
- Ascend slopes up to 50% grade
- Traverse side slopes up to 35% grade
- 5,000 lb winch
- Alternate color options available
- Multiple inlet options available
- High intensity lighting
- Standoff range of over 300-500 m
- Up to 20 hr of runtime without refueling
- Clearance: 8 in
- Vehicle and track sprayers provide 360-degree heat shielded protection

2.2 THERMITE RS3-T1:

The Thermite RS3-T1 is a super-high volume, low center of gravity, wide chassis, industrial firefighting Vehicle. The RS3-T1 is remotely operated with a bellypack controller that provides high-definition video feedback for ultimate maneuverability in difficult conditions. The RS3-T1's modular design and wider stance allows additional equipment to be incorporated, including a plow assembly and positive pressure ventilation (PPV) ventilator. Fabricated using industrial-grade steel reinforced rubber tracks, RS3-T1 can navigate rugged terrain and withstand exposure to the extreme elements.

Figure 2-2: THERMITE RS3-T1 in application field

SPECIFICATIONS:

- Yanmar 36.8 hp water cooled diesel engine
- Nozzle options 1,250-2,500 gpm max
- Integrated bellypack controller
- Durable industrial components
- Industrial grade tracks
- Live video feed via widescreen camera
- Zero degree turning radius
- Ascend slopes up to 50% grade
- Traverse side slopes up to 35% grade
- 8,000 lb winch
- Alternate color options available
-
- Multiple inlet options available
- High intensity lighting
- Standoff range of over 300-500 m
- Up to 20 hr of runtime without refueling
- Clearance: 10 in
- Vehicle and track sprayers provide 360-degree heat shielded protection

2.3 Vehicle firefighter Colossus:

There are firefighters of Paris who have acquired it and now are able to benefit the new opportunities thanks to this Vehicle which can do among others:

- Extinguish fires
- Go upstairs
- Be piloted up to 300 meters
- Take pictures
- Store GPS position
- Report a zoomed image on a control touch pad
- These last three points will be available in a more upmarket version.

It’s a little revolution for the fire soldiers that will be able to let the Vehicle to explore the dangerous zones before knowing if they can take action by themselves or not without putting their life in danger.
The Vehicle firefighter will also be able to shorten the time to assess the risk of a situation before intervention which can make problem when it’s very important.
Colossus is a 420kg Vehicle that still measures 1m60 in height. It’s also able to transport up to half a ton of payload and has a battery that gives it autonomy up to 4 hours. Motorization level is not outdone and has a double version that accumulates 8000 watts.

2.4. Fire Ox:

The Fire Ox is one of the few Vehicle firefighting vehicles that carries its own water tank. Designed as a first response unit, it suppresses fires, assists in search and rescue and can handle dangerous materials. Originally created by Lockheed Martin as a Squad Mission Support System (SMSS) for assisting soldiers with their gear in the field, it was retrofitted with a water tank and hose for distribution.

The Fire Ox is semi-autonomous and can be controlled from up to 200 miles away. Due to its mobility it has the ability to traverse situations unsafe for people, minimizing casualties and rescue time. It can also be used in a multitude of situations including wildfires and structure fires.
Firefighting Vehicles are still being developed and perfected for more common use in the field. Occupational safety professionals are working with other professionals and governmental agencies to make the futuristic technology a reality. Those who have a passion for defending first responders from overly dangerous situations can help shape the future of firefighting Vehicles.

The safety of firefighters and the victims caught in fires is paramount, urging the production of these Vehicles forward. They have the ability to withstand environments too hazardous for people and can prevent further damage that may happen left to traditional methods of suppression.

Chapter 3

Theory & Methodology

3.1 Introduction:

The methodology is the general research strategy that outlines the way in which research is to be undertaken and among other things, identifies the methods to be used in it. These methods, described in the methodology. Methodology does not define specific methods, even though much attention is given to the nature and kinds of processes to be followed in a particular procedure or to attain an objective.

When proper to a study of methodology, such processes constitute a constructive generic framework and may therefore be broken down into sub-processes, combined, or their sequence changed.

A paradigm is similar to a methodology in that it is also a constructive framework. In theoretical work, the development of paradigms satisfies most or all of the criteria for methodology. An algorithm, like a paradigm, is also a type of constructive framework, meaning that the construction is a logical, rather than a physical, array of connected elements.

Any description of a means of calculation of a specific result is always a description of a method and never a description of a methodology. It is thus important to avoid using methodology as a synonym for method or body of methods. Doing this shifts it away from its true epistemological meaning and reduces it to being the procedure itself, or the set of tools, or the instruments that should have been its outcome. A methodology is the design process for carrying out research or the development of a procedure and is not in itself an instrument, or method, or procedure for doing things.

Methodology and method are not interchangeable. In recent years, however, there has been a tendency to use methodology as a "pretentious substitute for the word method. Using methodology as a synonym for method or set of methods leads to confusion and misinterpretation and undermines the proper analysis that should go into designing research.

3.2 NodeMCU:

NodeMCU is an open source Lua based firmware for the ESP8266 WiFi SOC from Espressif and uses an on-module flash-based SPIFFS file system. NodeMCU is implemented in C and is layered on the Espressif NON-OS SDK.

The firmware was initially developed as is a companion project to the popular ESP8266-based NodeMCU development modules, but the project is now community-supported, and the firmware can now be run on any ESP module.

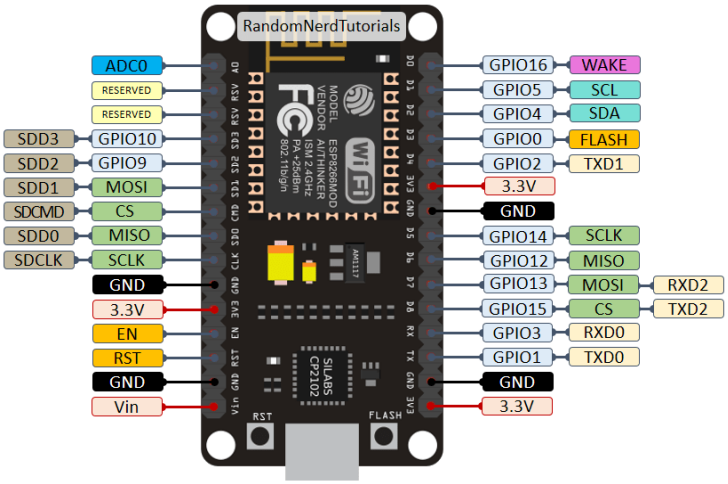

Figure 3-1: NodeMCU Pinout

The NodeMCU programming model is similar to that of Node.js, only in Lua. It is asynchronous and event-driven. Many functions, therefore, have parameters for callback functions. To give you an idea what a NodeMCU program looks like study the short snippets.

This project uses two main branches, master and dev. dev is actively worked on and it's also where PRs should be created against. Master thus can be considered "stable" even though there are no automated regression tests. The goal is to merge back to master roughly every 2 months. Depending on the current "heat" (issues, PRs) we accept changes to dev for 5-6 weeks and then hold back for 2-3 weeks before the next snap is completed.

A new tag is created every time dev is merged back to master. They are listed in the releases section on GitHub. Tag names follow the <SDK-version>-master_yyyymmdd pattern.

3.3 Fire Sensor:

A flame detector is a sensor designed to detect and respond to the presence of aflame or fire. Responses to a detected flame depend on the installation, but can include sounding an alarm, deactivating a fuel line (such as a propane or a natural gas line), and activating a fire suppression system. A fire detector works by detecting smoke and/or heat. Since a fire detector usually works by detecting smoke and heat, and not actual fire, these devices are not usually called "fire detectors". Instead, these devices are more appropriately called "smoke detectors" and "heat detectors".

Figure 3-2: Flame sensor

3.4 Bluetooth Module:

The HC-05 is a class 2 slave Bluetooth module designed for transparent wireless serial communication. Once it is paired to a master Bluetooth device such as PC, smart phones and tablet, its operation becomes transparent to the user. It uses the 2.45GHz frequency band. The connection can be point-to-point or multi-point where the maximum range is 10 meters. The transfer rate of the data is 1Mbps.HC-05 Bluetooth module provides switching mode between master and slave mode which means it able to use neither receiving nor transmitting data. The Bluetooth module is used to connect the mobile phone device with the module for controlling the fire. [4]

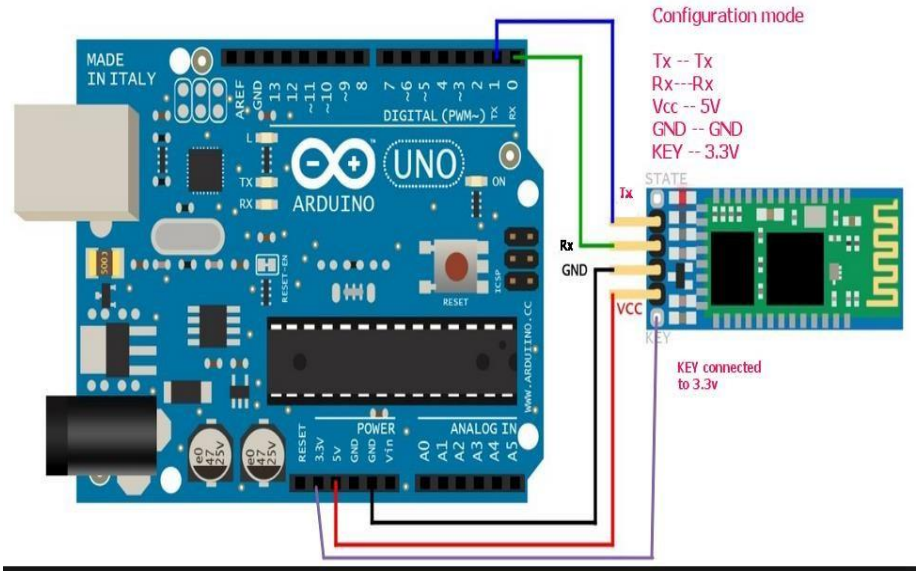

Figure 3-3: Bluetooth module HC-05

3.5 Motor Driver:

A motor driver is an integrated circuit chip which is usually used to control motors in autonomous Vehicles. Motor driver act as an interface between Arduino and the motors. The most commonly used motor driver IC's are from the L293 series such as L293D, L293NE, etc. Motor Driver circuits are current amplifiers. They act as a bridge between the controller and the motor in a motor drive. The input to the motor driver IC or motor driver circuit is a low current signal. The function of the circuit is to convert the low current signal to a high current signal. [4]

`Figure 3-4: Motor Driver

3.6 Pump:

Motor is an electrical machine which converts electrical energy into mechanical energy and pump is mechanical device which converts mechanical energy into work to be done. Pump is driven by motor only. A pump requires a driving mechanism such as a motor or an engine to operate. Flow rate 18.75ml/sec.

Figure 3-5: Mini pump

Speed	Angle in degree	Distance in feet
100%	0	0
100%	10	1.16
100%	20	2.33
100%	30	3.5
100%	40	4.66
100%	50	5.83
100%	60	7
100%	70	5.83

Table 1-1: Distance Measurement with Following Angle

3.7 Gear Motor:

A gear motor is an all-in-one combination of a motor and gearbox. The addition of a gear head to a motor reduces the speed while increasing the torque output. In order to select the most suitable gear motor for your application you must first compute the load, speed and torque requirements for your application.

Figure 3-6: Gear Motor

The electric motor used in most cranes is a type of gear motor that uses the basic principles of speed reduction to increase torque or force. The output speed of the rotor is reduced through a series of large ears until the rotating, RPM speed, of the final gear is very low. Power 8watt & 300rpm.

3.8 Port description of Easier Pro:

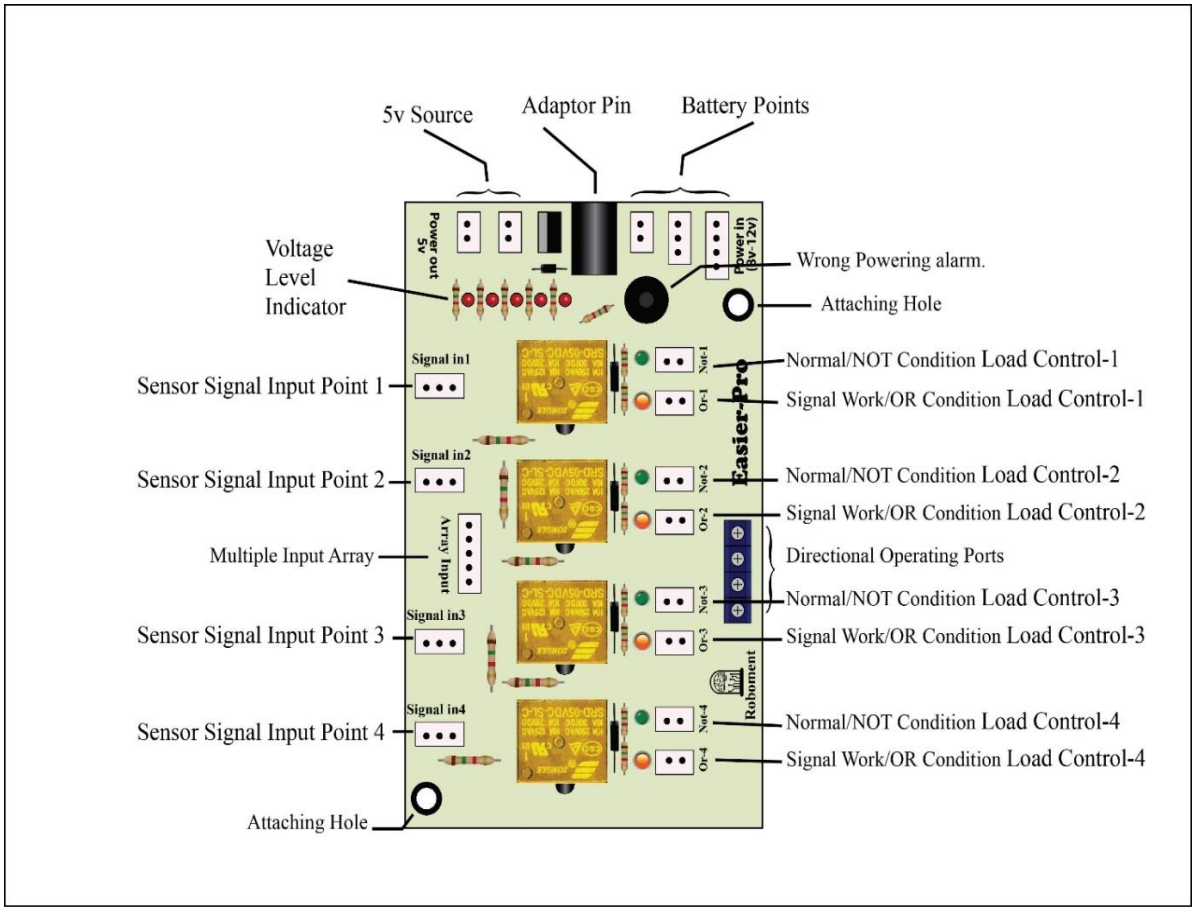

Figure 3-7: Easier Pro Pinout

3.9 Arduino:

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. Arduino consists of both a physical programmable circuit board (often referred to as a microcontroller) and a piece of software, or IDE (Integrated Development Environment) that runs on your computer, used to write and upload computer code to the physical board. In fact, you already are; the Arduino language is merely a set of C/C++ functions that can be called from your code. Your sketch undergoes minor changes (e.g. automatic generation of function prototypes) and then is passed directly to a C/C++ compiler

3.10 Arduino Nano:

One of the most popular Arduino boards out there is the Arduino Nano. While it was not actually the first board to be released, it remains to be the most actively used and most widely documented on the market. Because of its extreme popularity, the Arduino Nano has a ton of project tutorials and forums around the web that can help you get started or out of a jam. We're big fans of the nano because of its great features and ease of use.

Here are the components that make up an Arduino board and what each of their functions are.

Figure 3-8: Arduino Nano

Pinout of Arduino Nano:

- Reset Button – This will restart any code that is loaded to the Arduino board
- AREF – Stands for “Analog Reference” and is used to set an external reference voltage
- Ground Pin – There are a few ground pins on the Arduino and they all work the same
- Digital Input/Output – Pins 0-13 can be used for digital input or output
- PWM – The pins marked with the (~) symbol can simulate analog output
- USB Connection – Used for powering up your Arduino and uploading sketches
- TX/RX – Transmit and receive data indication LEDs
- ATmega Microcontroller – This is the brains and is where the programs are stored
- Power LED Indicator – This LED lights up anytime the board is plugged in a power source
- Voltage Regulator – This controls the amount of voltage going into the Arduino board
- DC Power Barrel Jack – This is used for powering your Arduino with a power supply
- 3.3V Pin – This pin supplies 3.3 volts of power to your projects
- 5V Pin – This pin supplies 5 volts of power to your projects
- Ground Pins – There are a few ground pins on the Arduino and they all work the same
- Analog Pins – These pins can read the signal from an analog sensor and convert it to digital.

3.11 Programming on Arduino:

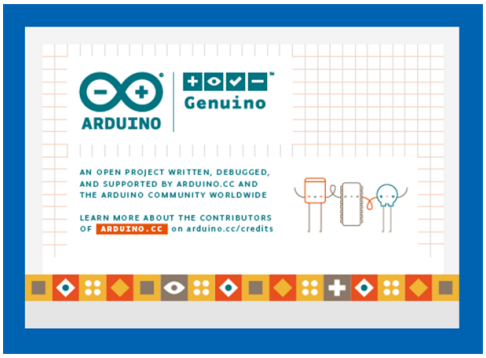

Figure 3-9: Opening Arduino IDE

Once the circuit has been created on the breadboard, you'll need to upload the program (known as a sketch) to the Arduino. The sketch is a set of instructions that tells the board what functions it needs to perform. An Arduino board can only hold and perform one sketch at a time. The software used to create Arduino sketches is called the IDE which stands for Integrated Development Environment. The software is free to download and can be found at https://www.arduino.cc/en/Main/Software

Once the software has been installed on your computer, go ahead and open it up. This is the Arduino IDE and is the place where all the programming will happen.

Take some time to look around and get comfortable with it. [2]

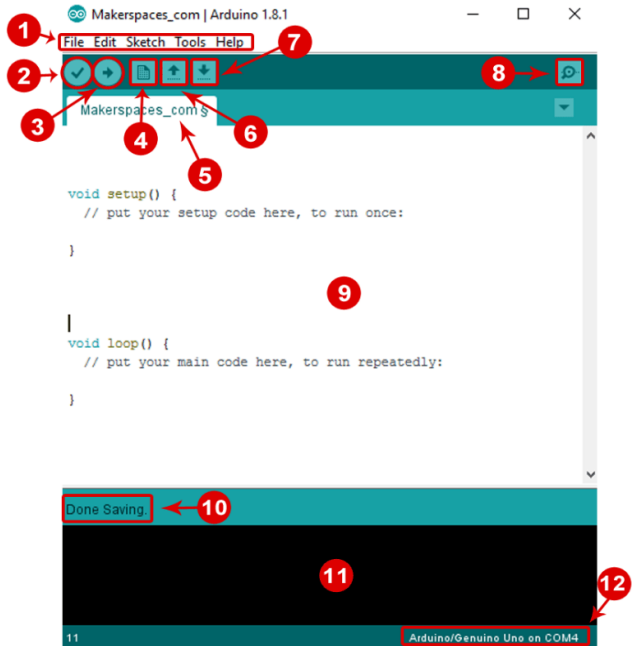

Figure 3-10: Sketch of Arduino IDE

- Menu Bar: Gives you access to the tools needed for creating and saving Arduino sketches.
- Verify Button: Compiles your code and checks for errors in spelling or syntax.
- Upload Button: Sends the code to the board that's connected such as Arduino Uno in this case. Lights on the board will blink rapidly when uploading.
- New Sketch: Opens up a new window containing a blank sketch.
- Sketch Name: When the sketch is saved, the name of the sketch is displayed here.

- Open Existing Sketch: Allows you to open a saved sketch or one from the stored examples.
- Save Sketch: This saves the sketch you currently have open.
- Serial Monitor: When the board is connected, this will display the serial information of your Arduino
- Code Area: This area is where you compose the code of the sketch that tells the board what to do.
- Message Area: This area tells you the status on saving, code compiling, errors and more.
- Text Console: Shows the details of an error messages, size of the program that was compiled and additional info.
- Board and Serial Port: Tells you what board is being used and what serial port it's connected to. [10]

At this point you are ready to connect your Arduino to your computer. Plug one end of the USB cable to the Arduino Uno and then the other end of the USB to your computer's USB port.

Once the board is connected, you will need to go to Tools then Board then finally select Arduino Uno.

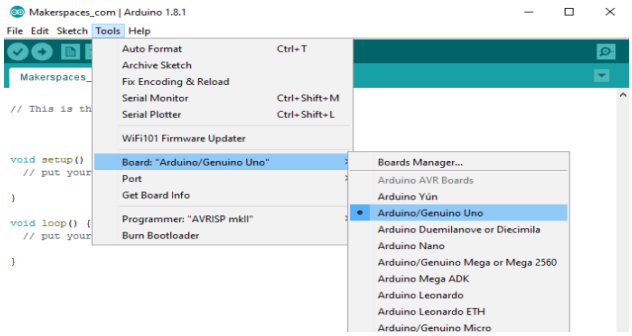

Figure 3-11: Arduino Board Selection

Next, you have to tell the Arduino which port you are using on your computer. To select the port, go to Tools then Port then select the port that says Arduino.

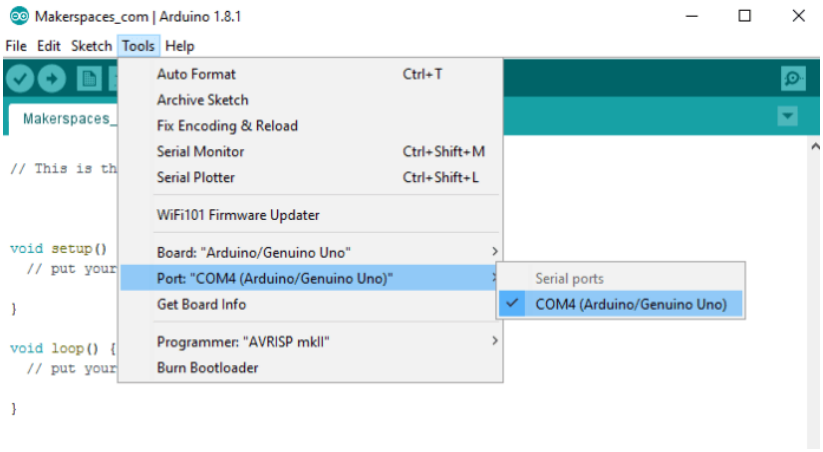

Figure 3-12: Communication port selection

3.12 Arduino Project 1: Blink an LED:

It's finally time to do your first Arduino project. In this example, we are going to make your Arduino board blink an LED.

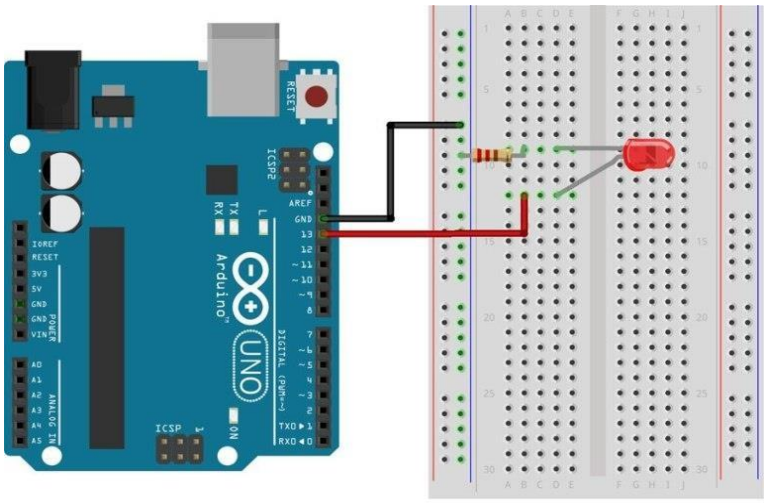

Figure 3-13: Experimental Setup

Required Parts:

- Arduino Uno Board
- Breadboard – half size
- Jumper Wires
- USB Cable
- LED (5mm)
- 220 Ohm Resistor

3.13 Connect the Parts:

You can build your Arduino circuit by looking at the breadboard image above or by using the written description below. In the written description, we will use a letter/number combo that refers to the location of the component. If we mention H19 for example, that refers to column H, row 19 on the breadboard.

Step 1 – Insert black jumper wire into the GND (Ground) pin on the Arduino and then in the GND rail of the breadboard row 15

Step 2 – Insert red jumper wire into pin 13 on the Arduino and then the other end into F7 on the breadboard

Step 3 – Place the LONG leg of the LED into H7

Step 4 – Place the SHORT leg of the LED into H4

Step 5 – Bend both legs of a 220 Ohm resistor and place one leg in the GND rail around row 4 and other leg in I4

Step 6 – Connect the Arduino Uno to your computer via USB cable

3.14 Upload the Blink Sketch:

Now it's time to upload the sketch (program) to the Arduino and tell it what to do. In the IDE, there are built-in example sketches that you can use which make it easy for beginners.

To open the blink sketch, you will need to go to File > Examples > Basics > Blink Now you should have a fully coded blink sketch that looks like the image below.

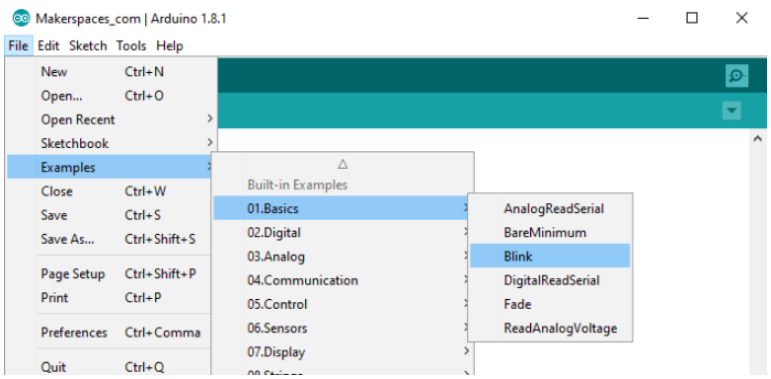

Figure 3-14: Scratch opening

Figure 3-15: Basic program of LED Blinking

Next, you need to click on the verify button (check mark) that's located in the top left of the IDE box. This will compile the sketch and look for errors. Once it says "Done Compiling" you are ready to upload it. Click the upload button (forward arrow) to send the program to the Arduino board.

The built-in LEDs on the Arduino board will flash rapidly for a few seconds and then the program will execute. If everything went correctly, the LED on the breadboard should turn on for a second and then off for a second and continue in a loop.

3.15. Arduino Nano is better than Arduino Uno:

Figure 3-16: Arduino Uno & Arduino Nano

Arduino UNO is one of the most famous board in Arduino family after Arduino Duemilanove. It is the latest design of the basic USB board. It comes with 6 analog inputs, 14 digitals output where 6 of them support PWM, and 16 MHz clock speed. Arduino UNO comes with 6 analog inputs and 14 digital I/O where 6 of them are PWM outputs. It is running on a ATmega328 processor with 32kB flash memory. The clock speed of this Arduino board is 16 MHz with the dimension of 68.6mm x 53.3mm. There are a lot of shields build to expend its functionality. Arduino Nano, as its name, Arduino Nano is a compact and breadboard-friendly version board based on Atmega328 processor. It is more or less same functionality as the Arduino UNO but in different package. Instead of using the standard USB to connect to the computer, it uses the mini usb but without the power plug for external power source that built on Arduino UNO. The dimension of Arduino Nano is only 43mm x 18mm, it comes with 6 PWM I/O from the total of 14 digitals I/O, 8 analog inputs, 16 MHz clock speed and 32kB of flash memory. [4]

Arduino Board	Family	MEMORY			Clock	UART	PWM	Digital	Analog	VCC	Vin Range	USB-Serial
		SRAM	FLASH	EEPROM								
Duemilanove (328)	ATmega328	2K	32k	1kB	16MHz	1	6	14	6	5V	7-12V	ATmega16U2
Uno	ATmega328	2k	32k	1kB	16MHz	1	6	14	6	5V	7-12V	ATmega16U2
Arduino Mega 2560	ATmega2560	8k	256k	1kB	16MHz	4	14	54	16	5V	7-18V	ATmega16U2
Arduino Mega ADK	ATmega2560	8k	256k	1kB	16MHz	4	14	50	16	5V	7-18V	ATmega16U2
Arduino Ethernet	ATmega328	2k	32k	1kB	16MHz	1	4	9	6	5V	6-18V	N/A
Arduino BT	ATmega328	2k	32k	1kB	16MHz	1	6	14	6	5.5V	1.2V-5.5V	Bluegiga WT11
Arduino Pro Mini 328 5V	ATmega328	2k	32k	1kB	16MHz	1	6	14	6	5V	5-12V	N/A
Arduino Nano 3.0	ATmega328	2k	32k	1kB	16MHz	1	6	14	8	5V	7-12V	FTDI FT232RL
Arduino Mini	ATmega328	2k	32k	1kB	16MHz	1	6	14	8	5V	7V-9V	N/A
Arduino Pro 3.3V	ATmega328P	2k	32k	1kB	8MHz	1	6	14	6	3.3V	3.35-12V	N/A
Arduino Pro 5V	ATmega328P	2k	32k	1kB	16MHz	1	6	14	6	5V	5-12V	N/A
Arduino Fio	ATmega328P	2k	32k	1kB	8MHz	1	6	14	8	3.3V	3.35-12V	N/A
LilyPad Simple Board	ATmega168P	1k	16k	512B	8MHz	1	5	9	4	2.7-5.5V		N/A
LilyPad 328 Main Board	ATmega328P	2k	32k	1kB	8MHz	1	6	14	6	2.7-5.5V		N/A

Table 1-2: Arduino Boards Comparison Chart

3.16 Thunkable:

Thunkable is the easiest way to build an app.

At Thunkable, our mission is simple – to enable everyone to build their own beautiful and powerful mobile apps.

Instead of writing code, simply add your favorite components and connect them together. Instead of choosing between iOS or Android, Thunkable app projects work on both iOS and Android devices.

Instead of starting your app from scratch, get inspired by and remix projects from our Public Gallery, featuring the work of our amazing community of app creators.

Common 5steps to get a customized android app:

- **Step 1(IDEA):** Start with a problem you want to solve. Or be inspired by problems that other people have solved with apps.
- **Step 2(Create):** Add your favorite technologies and write the story of how you'd like them to work together with blocks.
- **Step 3(Test Live):** Download the Thunkable Live companion and test your apps while you build them in real time. Even hard core developers don't have this.
- **Step 4(Publish):** For apps that you want to reach greater audiences, follow the steps to publish, promote and maintain them on the Play Store. $29 one-time fee required

- **Step 5(Ishare):** Share your app with your friends and your app source code with the larger Thunker community. The more sharing, the more problems that can be solved.

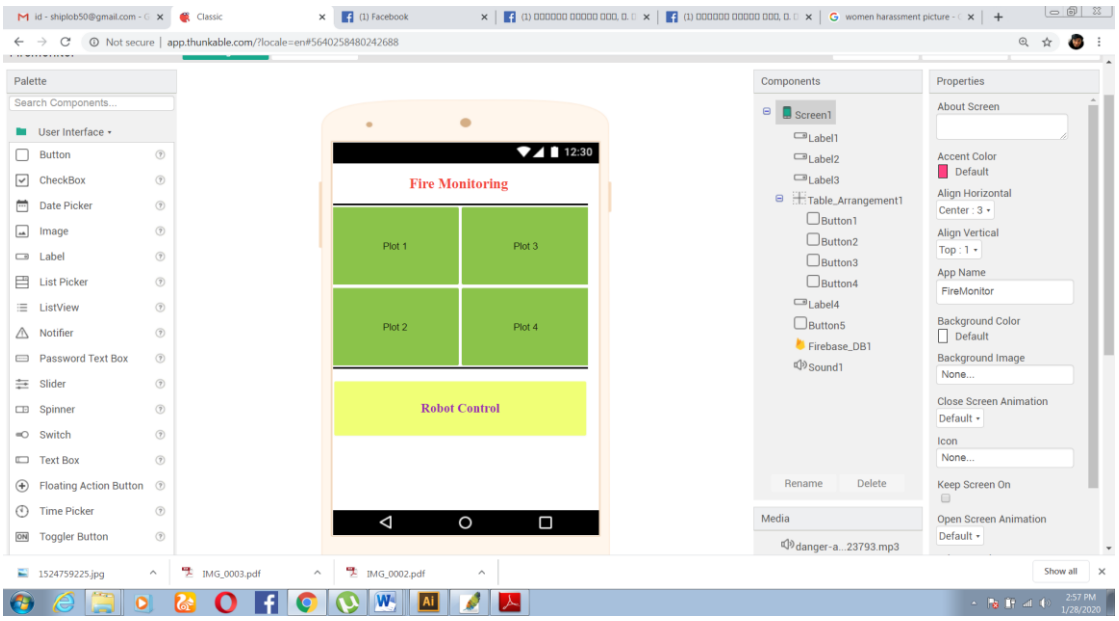

Figure 3-17: Main Dashboard of Thunkable app builder

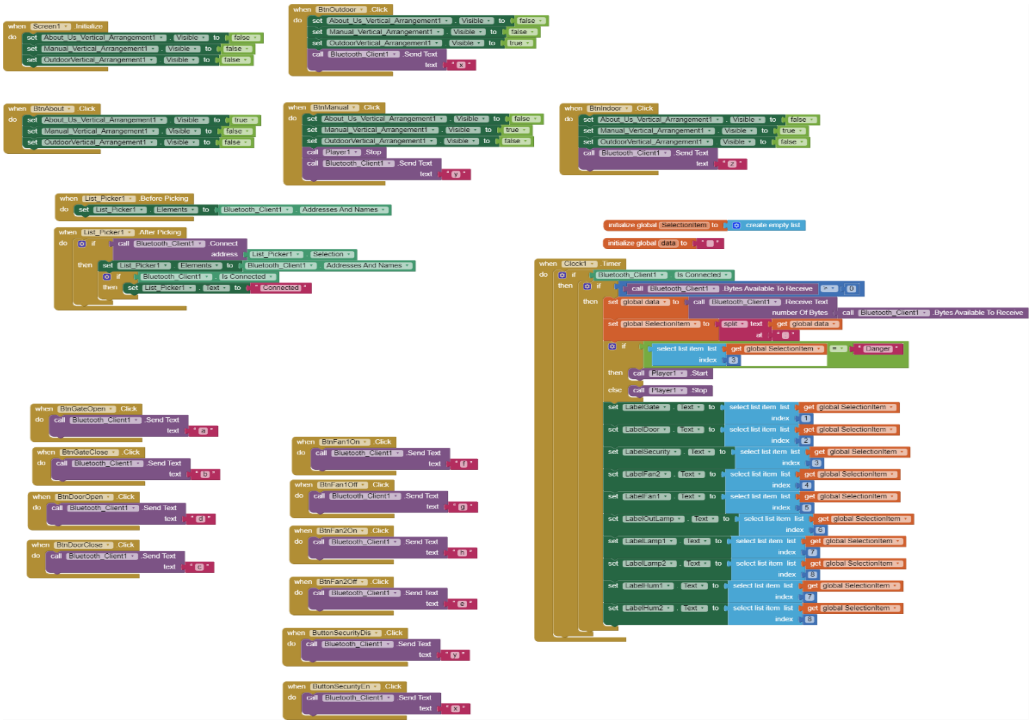

Figure 3-18: Block programming in Thunkable

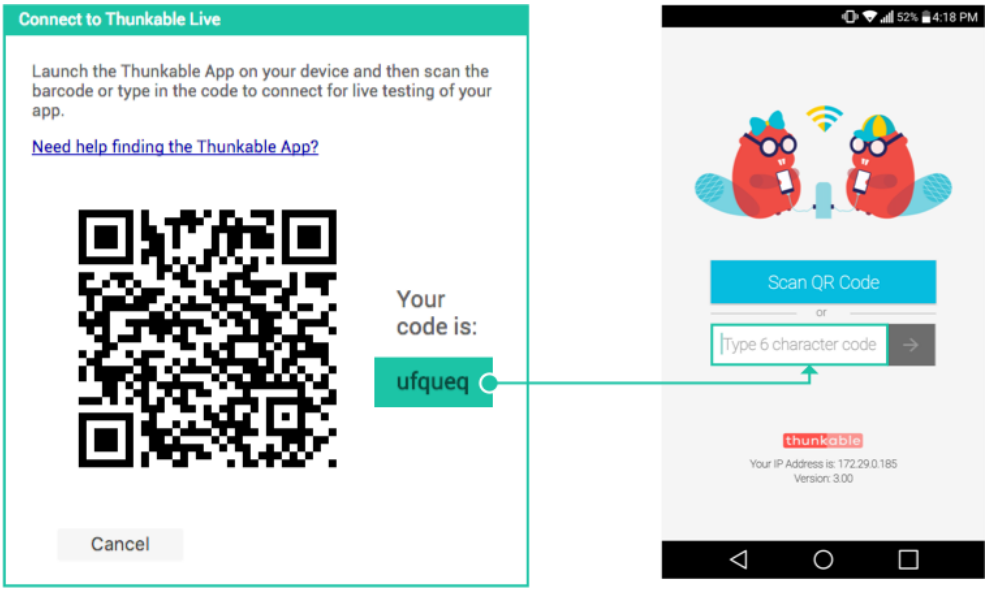

Figure 3-19: Live Test in Thunkable

3.17 Overview of Thunkable:

Thunkable is a cross-platform app builder that enables anyone to build their own native mobile apps. All apps built on Thunkable work for both Android and iOS devices.

Thunkable enables anyone to create beautiful and powerful mobile apps. On Thunkable, every app project you create works on both Android and iOS devices.

App Inventor is a blocks-based programming tool that uses the source code in the AIA file to create applications. The AIA file is also used to export and import projects to and from a computer to the web program. You can share your project in an executable form, which is the

.APK file, instead of source code form.

Surprisingly, the .aia file is just a regular .zip file. You can verify by saving a copy to your local disk drive, and then rename the file to have a .zip file extension instead of .aia. Then use Windows Explorer, StuffIt Expander or other utility to open and decompress the .zip file.

3.18 Block Diagram of Fire Plot Identification:

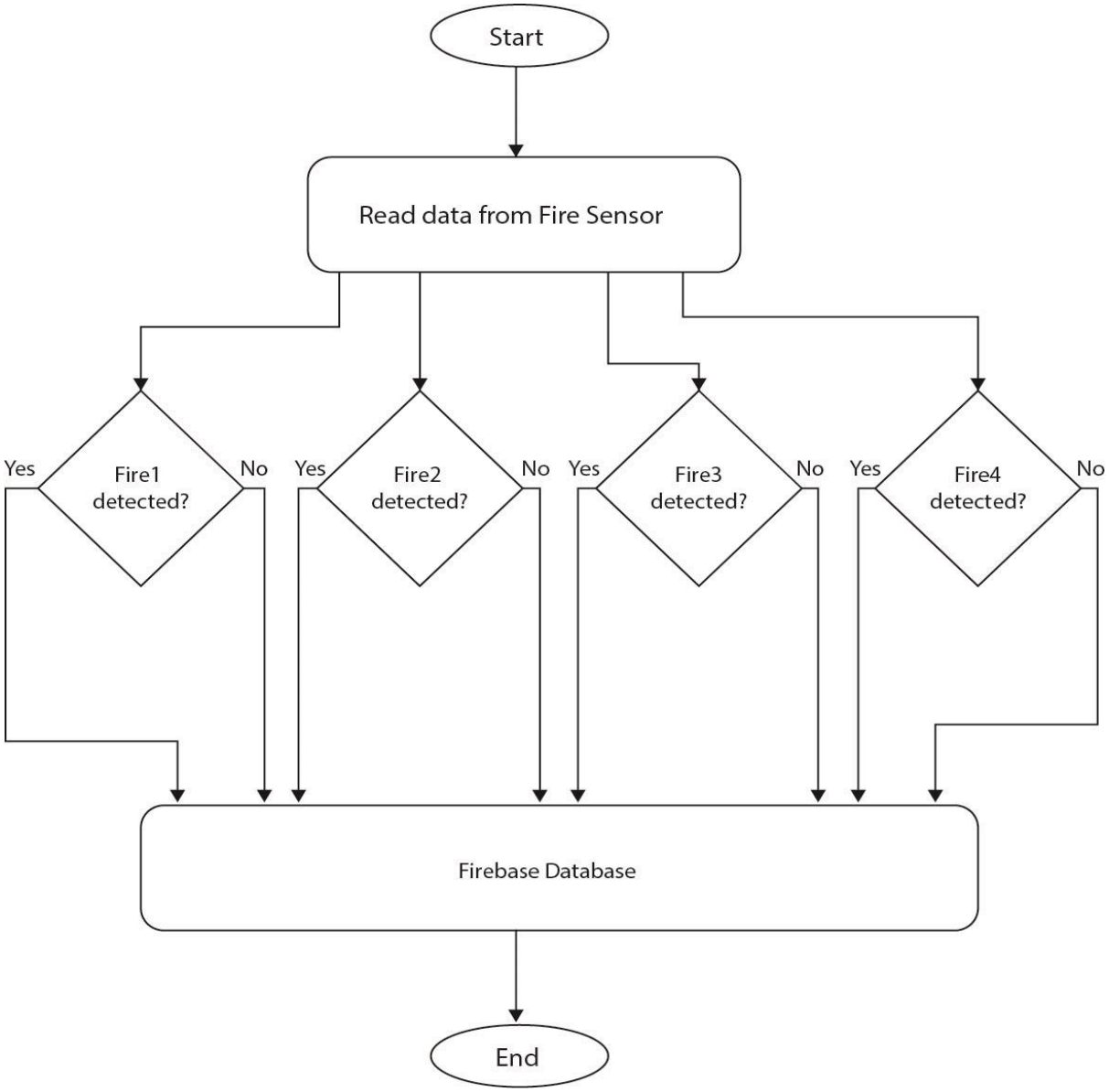

Figure 3-20: Block Diagram of Fire Plot Identification System

3.19 Principle:

Every Fire Sensor has 4ports. Vcc, Gnd, DO, AO. To get power, fire sensor required 5v in Vcc & Gnd. The DO pin supplies voltage if it seen any infrared ray. Fire has huge infrared ray. So, if any fire occurred then fire sensor sent signal to NodeMCU. After processed that data by nodeMCU it sent the data to database. Here android app able to access the data which real-time changing in Firebase Database.

3.20 Block Diagram of Fire Fighting Vehicle:

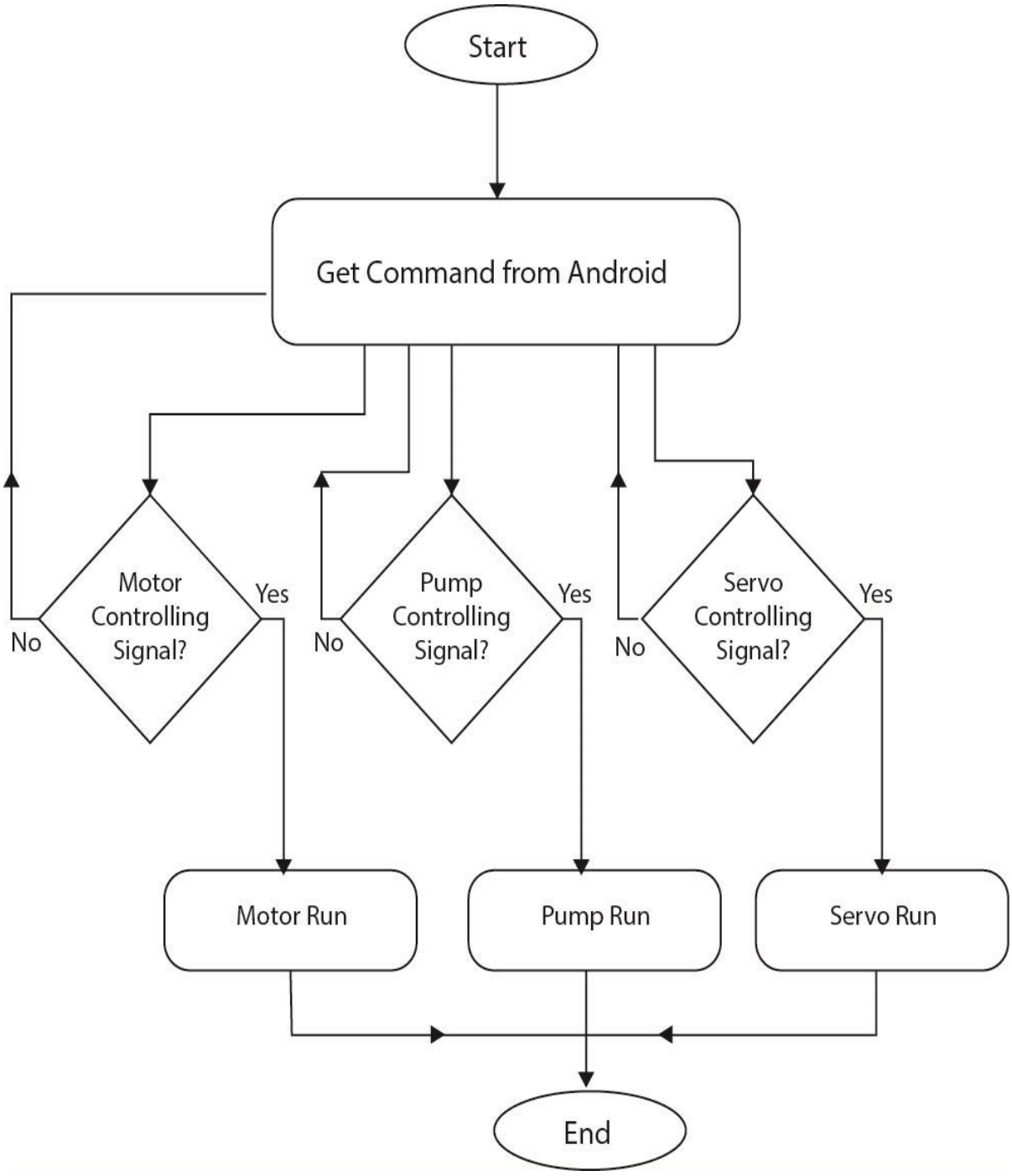

Figure 3-21: Block Diagram of Fire Fighting Vehicle

3.21 Principle:

Fire-fighting Vehicle consist battery, controller board, Arduino, Bluetooth module, four geared motor with wheels, one pump & servo motor. When signal sent from android application to drive geared motor, then Vehicle moves forward, backward, left turn & right turn. Servo moves zero to seventy degree by command of user via android application. Pump also changes its speed by command of user via android application.

3.22 Experimental Setup:

The process of circuit design can cover systems ranging from complex electronic systems all the way down to the individual transistors within an integrated circuit. For simple circuits the design process can often be done by one person without needing a planned or structured design process, but for more complex designs, teams of designers following a systematic approach with intelligently guided computer simulation are becoming increasingly common. In integrated circuit design automation, the term "circuit design" often refers to the step of the design cycle which outputs the schematics of the integrated circuit. Typically this is the step between logic design and physical design.

For the purpose of smart security design, a control system circuit has been developed to control and secure from fire. A circuit diagram of the control and security system has been shown in figure 3-22.

In this circuit there are several input and output devices. Input devices are Bluetooth module battery, controller board, Arduino, Bluetooth module, four geared motor with wheels, one pump & servo motor. A 12V power supply has been used for running this circuit.

3.23 Working procedure:

The working procedure of this project mainly depends upon the combination of Mechanical and electronic control system. To achieve the design & fabrication of a fire identification system and fire-fighting Vehicle here used fire sensor, NodeMCU, firebase database, android application, power system, arduino, geared motor, wheels, Easier pro, motor driver, Bluetooth module, servo motor, water pump etc.

To identify fire location, here used fire sensor which regular sent about fire condition in four different spots. NodeMCU is a special development board which mainly used to communicate with database via wifi communication. After get data from fire sensors, NodeMCU sent those data to firebase database. Android application able to access this database and shown those data by processing into color and sound. It mainly used for identification the spot of fire zone. In this app has another option to controlling fire-fighting Vehicle. In the time of opening the controlling screen in the app, mobile and Vehicle automatically connected with each other. There are four button to control Vehicle to forward, backward, left & right. There also have two drop down options to set pump speed and nozzle angle. Reconnect button used to connect Vehicle if it disconnected.

CHAPTER 4

RESULT

4.1 Result:

The project IoT based automatic fire detection and fire-fighting Vehicle completed successfully. To identify fire location, here used fire sensor which regular sent about fire condition in four different spots. NodeMCU is a special development board which mainly used to communicate with database via wifi communication. After get data from fire sensors, NodeMCU sent those data to firebase database. Android application able to access this database and shown those data by processing into color and sound. It mainly used for identification the spot of fire zone. In this app has another option to controlling fire-fighting Vehicle. In the time of opening the controlling screen in the app, mobile and Vehicle automatically connected with each other. There are four button to control Vehicle to forward, backward, left & right.

There also have two dropdown options to set pump speed and nozzle angle. Reconnect button used to connect Vehicle if it disconnected.

4.2 Photographic View:

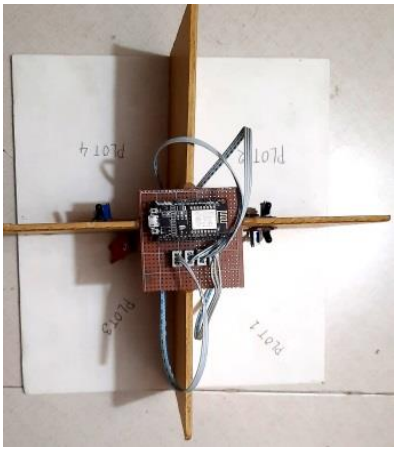

Figure 3-22: Mother board

Figure 3-23: Vehicle Top View

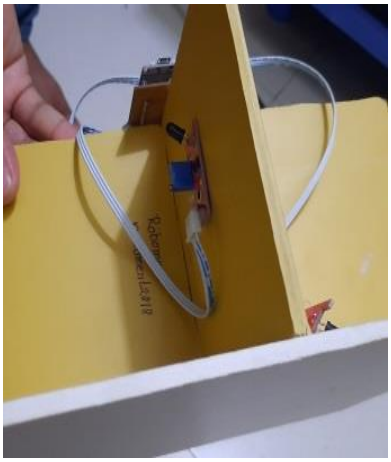

Figure 3-24: Fire Sensor

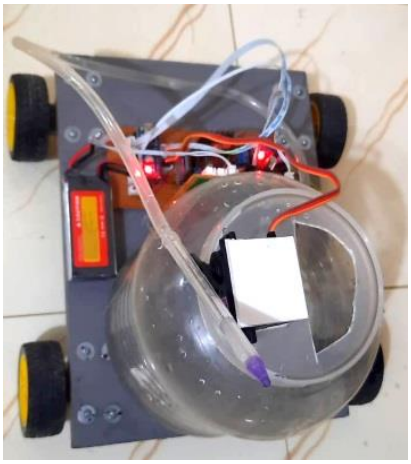

Figure 3-25: Tank Feeder

CHAPTER 5

CONCLUSION AND FUTUREWORK

Conclusion:

The objective of our project building automatic fire identification and Vehicle control system have fulfilled after completing of this project. This project gave an insight into designing of an automated fire just by using fire sensors. This helped in reducing the cost spent in construction of automated fires. Presence of every module has been reasoned out and placed carefully, thus contributing to the best working of the unit. Secondly, using highly advanced IC's with the help of growing technology, the project has been successfully implemented. Thus the project has been successfully designed and tested. This fire system can be used worldwide with many applications. This fire system is applicable from simple home to large banquet, hospital, airports & so on. The project provided us with an opportunity of working in different areas of engineering, Electronics & software. Due to this we are able to enhance knowledge along with electronics and software portion. To be more precise, the completion of the project has raised our confidence to next level. Where we feel more confident about our abilities as an engineer. We have made the project with best of our abilities. Hence, we conclude this project perceives a lot of experience and knowledge in the field of automation. Some updated features like controlling with android application has been used here.

Future work:

i) Here, Servo motor used for nozzle moving purpose. But practical application in future it have to be lead screw and geared motor.

ii) Components used only for project development purpose, but in practical application it have to be more higher quality.

iii) Thunkable used to make android application. It's looking so professional when android studio software use to make app.

Appendix

```
#include<Servo.h>
Servo servo;

void setup() {
  // put your setup code here, to run once:
  servo.attach(3);
  pinMode(A0,OUTPUT);
  pinMode(A1,OUTPUT);
  pinMode(A2,OUTPUT);
  pinMode(A0,OUTPUT);

  pinMode(6,OUTPUT);
  pinMode(7,OUTPUT);
  pinMode(8,OUTPUT);
  Serial.begin(9600);



}

void loop() {
  // put your main code here, to run repeatedly:
digitalWrite(8,1);
digitalWrite(7,0);
  if(Serial.available()>0)
   {
     char data= Serial.read(); // reading the data received from the bluetooth module
     if(data=='a') {digitalWrite(A0,0); digitalWrite(A1,1); digitalWrite(A2,1); digitalWrite(A3,0);}
     else if(data=='b'){digitalWrite(A0,1); digitalWrite(A1,0); digitalWrite(A2,0);
       digitalWrite(A3,1);}
     else if(data=='c'){digitalWrite(A0,0); digitalWrite(A1,1); digitalWrite(A2,0);
       digitalWrite(A3,1);}
     else if(data=='d'){digitalWrite(A0,1); digitalWrite(A1,0); digitalWrite(A2,1);
       digitalWrite(A3,0);}
     else if(data=='e'){digitalWrite(A0,0); digitalWrite(A1,0); digitalWrite(A2,0);
       digitalWrite(A3,0);}
     else if(data=='f')servo.write(20);
```

```
    else if(data=='g')servo.write(30);
    else if(data=='h')servo.write(40);
    else if(data=='i')servo.write(50);
    else if(data=='j')servo.write(60);
    else if(data=='k')servo.write(70);
    else if(data=='l')servo.write(80);
    else if(data=='m')servo.write(90);
    else if(data=='n')analogWrite(6,25);
    else if(data=='o')analogWrite(6,50);
    else if(data=='p')analogWrite(6,75);
    else if(data=='q')analogWrite(6,100);
    else if(data=='r')analogWrite(6,125);
    else if(data=='s')analogWrite(6,150);
    else if(data=='t')analogWrite(6,175);
    else if(data=='u')analogWrite(6,200);
    else if(data=='v')analogWrite(6,225);
    else if(data=='w')analogWrite(6,255);
    else if(data=='x')analogWrite(6,0);
  }
}
```

References

[1] Development of Fire Fighting Vehicle (QRob) by Mohd Aliff1, MI Yusof, Nor Samsiah Sani 2019
[2] Remote Controlled Fire Fighting Vehicle by Mangayarkarasi 2018
[3] Fire-Extinguishing Vehicle Design by Using Arduino by Abdülkadir ÇAKIR, Nyan Farooq Ezzulddın EZZULDDIN 2016
[4] A Fire-Fighting Vehicle and Its Impact on Educational Outcomes by D.J. Pack, A.M. Mankowski, and G.J. Freeman 2017
[5] Development of a Network-based Autonomous Firefighting Vehicle by Md. Hazrat Ali, Sultan Shamishev and Aidos Aitmaganbayev 2018
[6] Automotive Fire Fighting Vehicle by Sharavanan, Nithiya Devi, Venkat Raja, T. Venugopal 2017