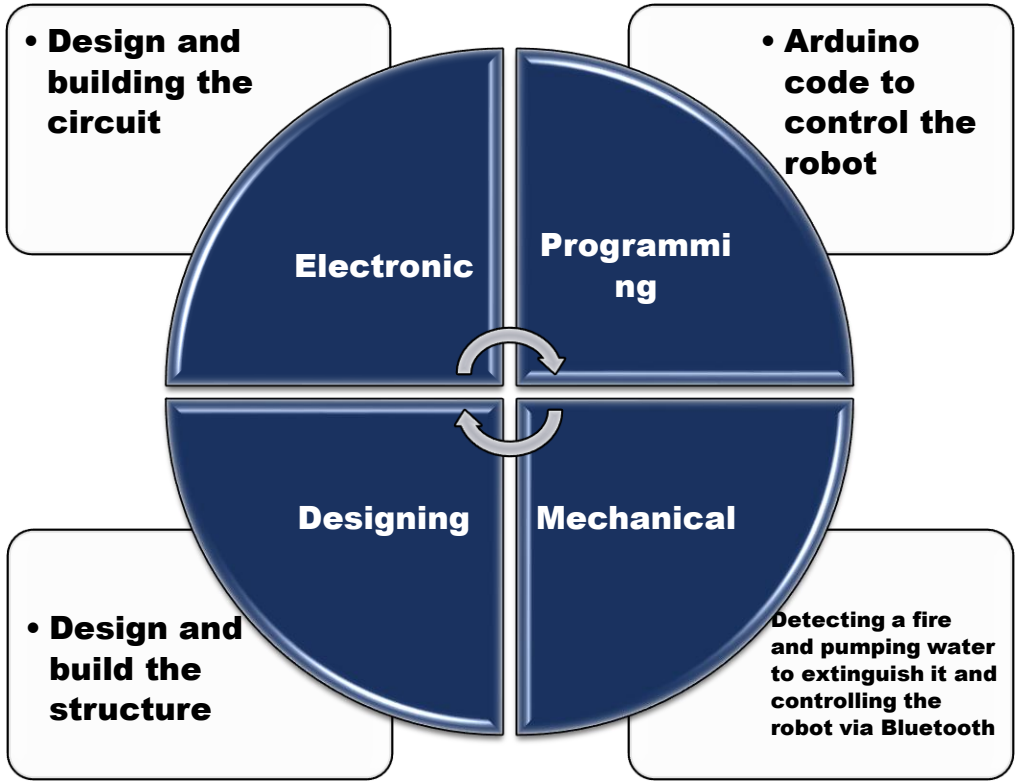
FIRE FIGHTING ROBOT

GROUP 1

GROUP MEMBERS

- W.S.C RODRIGO (19224)
- H.A.T NADEERANGA (19188)
- B.M.S.M.B BASNAYAKE (19186)
- G.S.R LAKSHAN (19191)
- M.D.O.C.V.B MAYADUNNE (19373)

TIMELINE


• Design and building the circuit
• Arduino code to control the robot
Electronic
Programmi ng
Designing
Mechanical
• Design and build the structure
Detecting a fire and pumping water to extinguish it and controlling the robot via Bluetooth

Time Period

Task	Assigned Members	September 1st-3rd week	September 25th	November (Whole Month)	January 1st Week	January 2nd week	February 1st week	February 2nd week	February 3rd week	March 1st week	March 2nd week		
Planning and schematic design	All												
Project Proposal	Oshan												
Researching and designing the Circuit\structure\mechanism	Sadul,Saliya,Tharindu,Ravindu												
Circuit making	Sadul,Tharindu,Ravindu												
Coding	Saliya,Oshan												
Progress Report	Oshan												
Making of the structure	Tharindu,Ravindu,Sadul												
Troubleshooting	Saliya,Oshan												
Preparing The final report	Oshan												
Finalizing the product	All												
Demonstration	All												

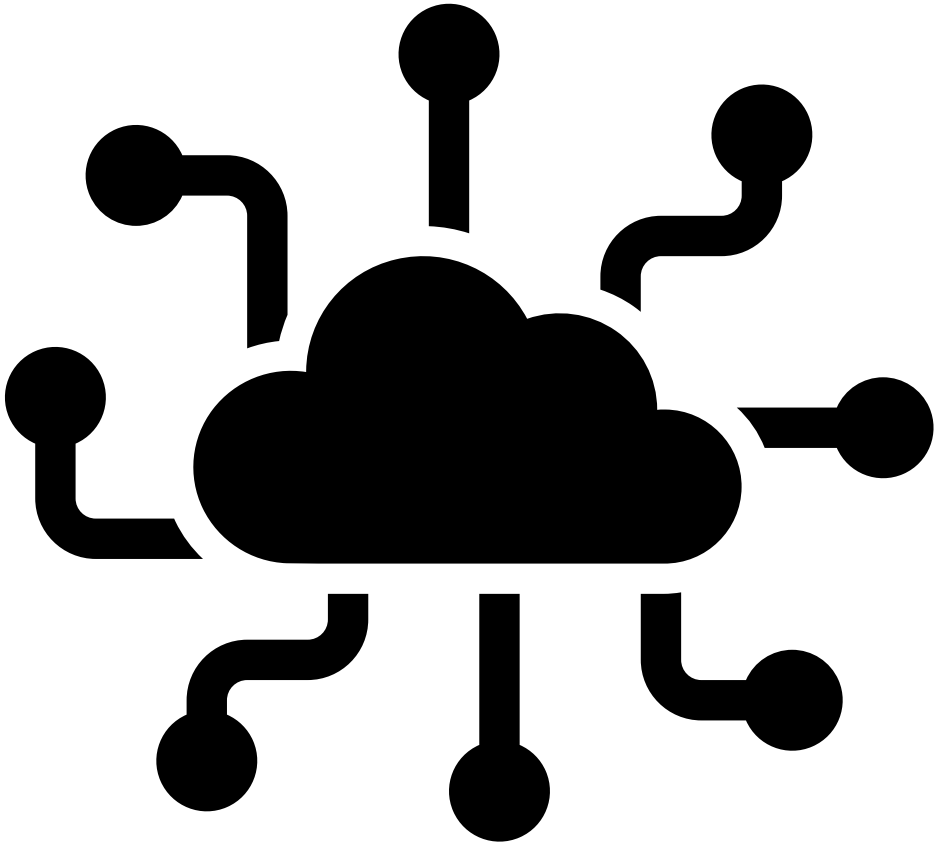
OBJECTIVES AND AIMS

In this project , our group will be able to develop a firefighting robot that have a dual advantage which it can sense the fire and can put it off before the fire gets bigger and also it can put off the fire before it gets out of control. This robot is designed to have a thermal sensor to sense the fire before it became out of control, mini pump will be used where the water flow. This is built not only for household protection, but it can also be used in a store and a company and this robot will be put nearby a flammable material.

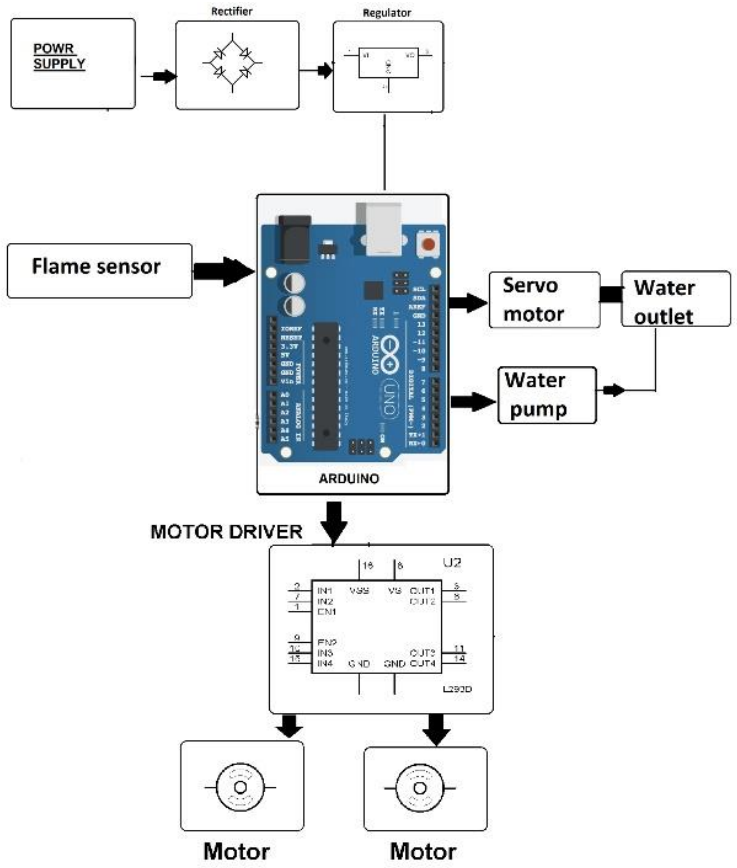
EQUIPMENT

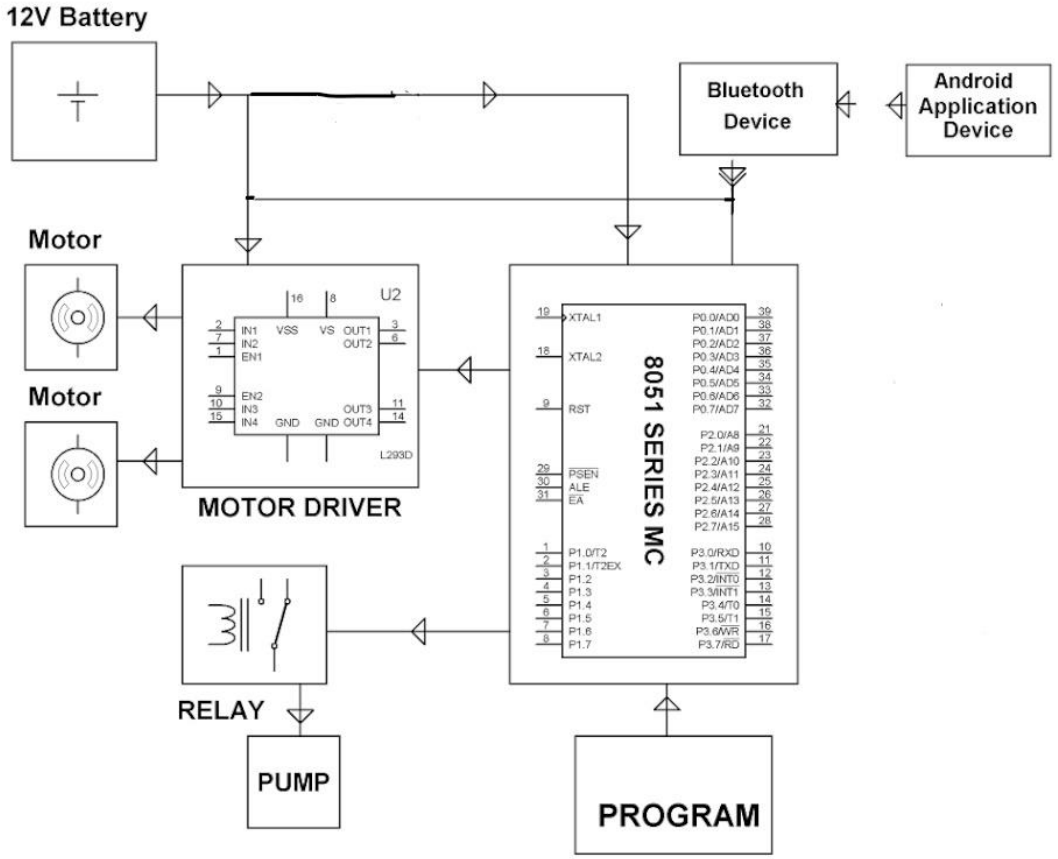
- Arduino Uno
- L298 Motor Driver
- DC motors × 4
- Wheels × 4
- Switch
- Highly adhesive glue
- Aluminum plates ×2
- Relay
- Drill
- Grinder
- Saw
- Pump
- 12v Battery
- Jumper wires
- Flame Sensor × 3
- Water Container
- Servo Motor
- Water Hose
- Bluetooth module

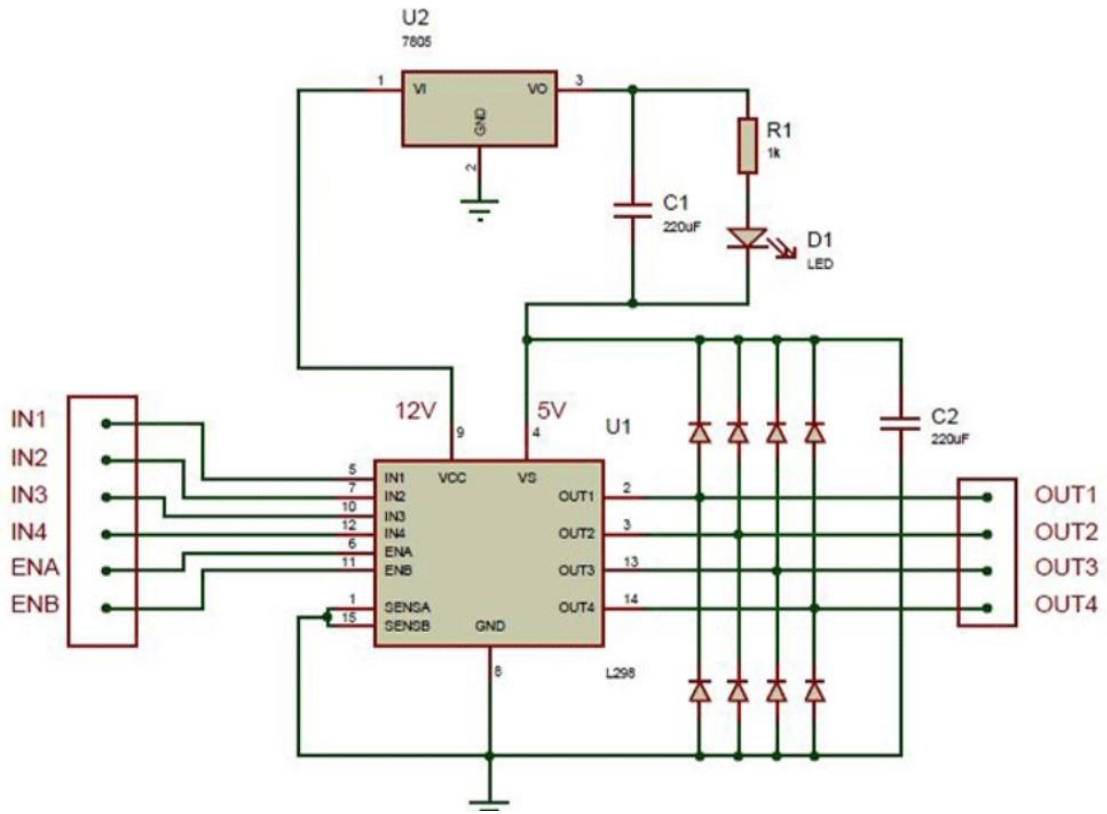
ELECTRONIC PART



SCHEMATIC DESIGN

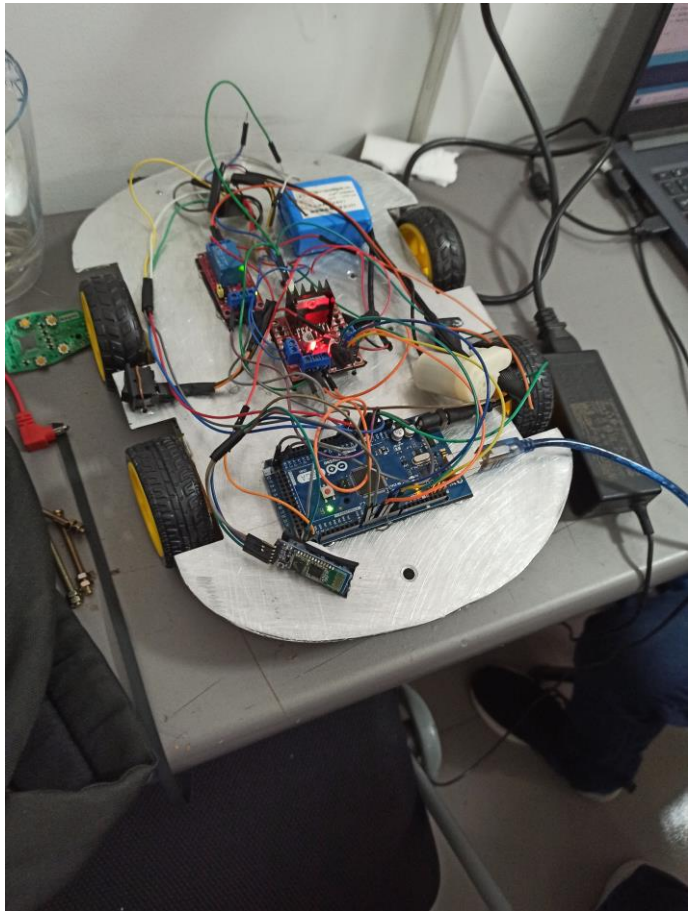


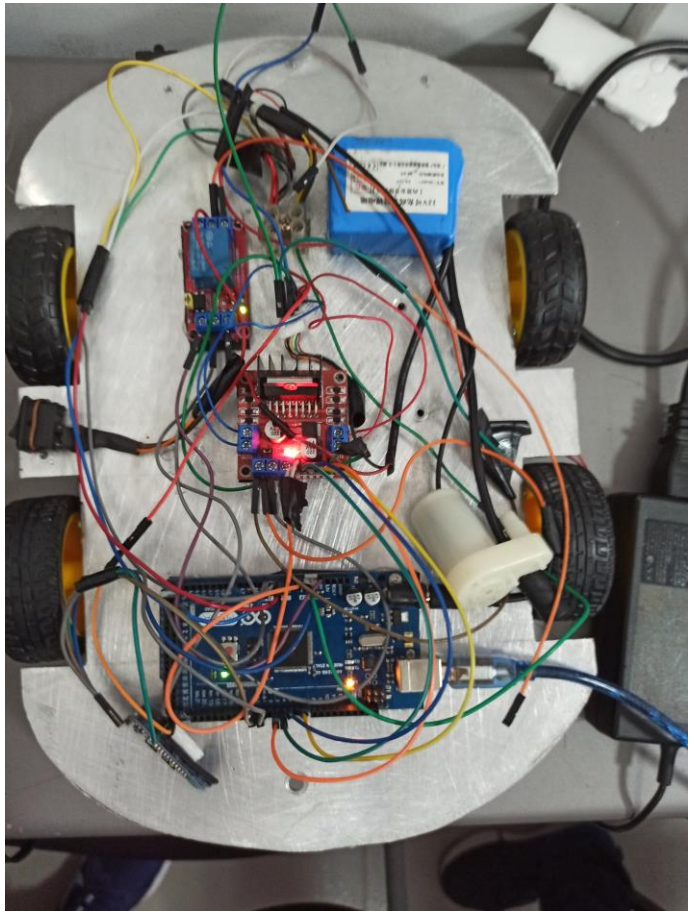


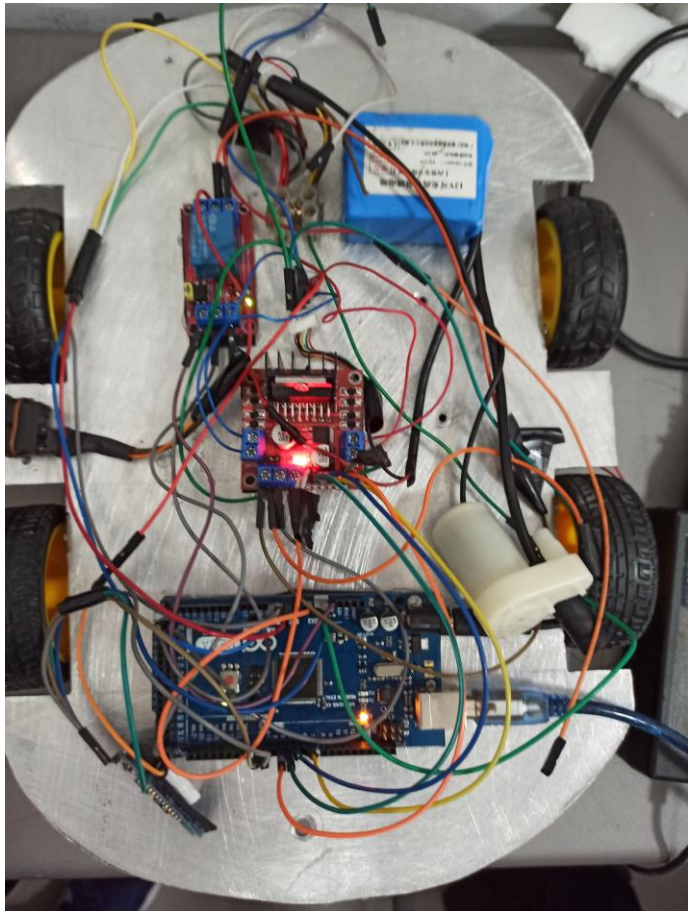

U2
7805
VI
VO
GND
R1
1k
C1
220uF
D1
LED
12V
5V
U1
C2
220uF
IN1
IN2
IN3
IN4
ENA
ENB
VCC
VS
SENSA
SENSB
GND
OUT1
OUT2
OUT3
OUT4
L298

CALCULATIONS USED WHEN SELECTING THE POWER SUPPLY AND CONNECTING CIRCUITRY

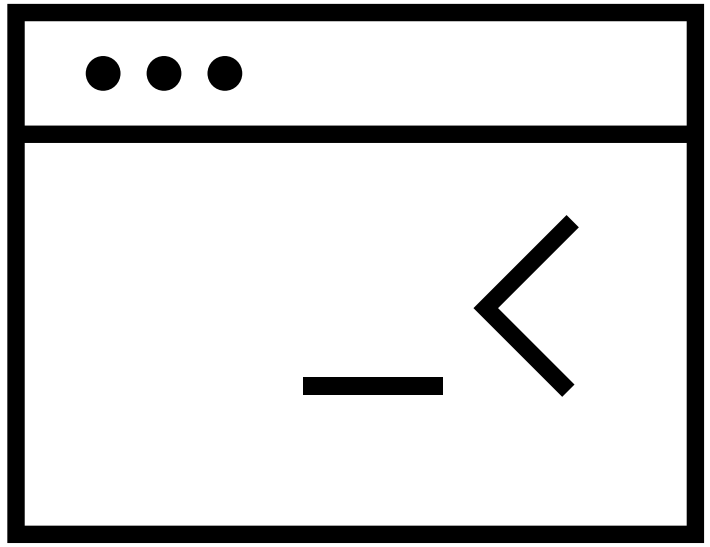
Arduino Uno Board	**1.8*100Ω**	**12v**	**0.066A**
Motor driver	4kΩ	12v	0.003A
Flame Sensor	200Ω	5v	0.025A
Pump	22Ω	12v	0.54A
Relay	1.8kΩ	12v	0.006A
DC motor	450Ω	12v	0.026A
Servo Motor	6.5Ω	5v	0.7A
			1.36A







PROGRAMMING PART



USED

Downloads

Arduino IDE 1.8.13

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the **Getting Started** page for Installation instructions.

SOURCE CODE

Active development of the Arduino software is **hosted by GitHub**. See the instructions for **building the code**. Latest release source code archives are available **here**. The archives are PGP-signed so they can be verified using **this** gpg key.

DOWNLOAD OPTIONS

Windows Win 7 and newer
Windows ZIP file

Windows app Win 8.1 or 10 Get

Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits

Mac OS X 10.10 or newer

Release Notes Checksums (sha512)

project_final | Arduino 1.8.13

File Edit Sketch Tools Help

project_final §

```
#include <Servo.h>  //include servo.h library
Servo myservo;

#define pump1 11
char input;

int pos = 0;

const int flamepin1=A1; //Defining flamesensors
const int flamepin2=A2;
const int flamepin3=A3;

int flamesensvalue1=0;
int flamesensvalue2=0;
int flamesensvalue3=0;


void setup() {
  pinMode(6,OUTPUT); //defining wheels
  pinMode(5,OUTPUT);
  pinMode(4,OUTPUT);
  pinMode(3,OUTPUT);

  pinMode(8,OUTPUT); //defining led  to LED
  pinMode(9,OUTPUT);


pinMode(flamepin1, INPUT);//sensors
pinMode(flamepin2, INPUT);
pinMode(flamepin3, INPUT);
```

Error compiling for board ESP32 Wrover Module.

Copy error messages

```
WARNING: library Servo claims to run on avr, megaavr, sam, samd, nrf52, stm32f4, mbed architecture(s) and may be incompatible with your current board which runs on esp32 architecture(s).
In file included from C:\Users\HP\Downloads\project_final\project_final\project_final.ino:3:0:
C:\Program Files (x86)\Arduino\libraries\Servo\src/Servo.h:77:2: error: #error "This library only supports boards with an AVR, SAM, SAMD, NRF52 or STM32F4 processor."
```

160 - 157

ESP32 Wrover Module, Huge APP (3MB No OTA/1MB SPIFFS), QIO, 40MHz, 115200, None on COM3

Type here to search

9:52 AM
4/1/2021

project_final | Arduino 1.8.13

File Edit Sketch Tools Help

project_final §

```
Serial.begin(9600);
Serial1.begin(115200);

  myservo.attach(11); //servo motor pin =11
  myservo.write(90);


}

void loop() {

int flamesensvalue1=analogRead(flamepin1);
int flamesensvalue2=analogRead(flamepin2);
int flamesensvalue3=analogRead(flamepin3);




Serial.println(flamesensvalue1);
Serial.println(flamesensvalue2);
Serial.println(flamesensvalue3);



boolean buzz1= buzz(flamesensvalue1,flamesensvalue2,flamesensvalue3); // this funtion will return 1 when fire ditected with some distance

if(buzz1){
    digitalWrite(9 ,LOW);
     digitalWrite(8 ,HIGH);
        }
 else{
    digitalWrite(9,HIGH);
       digitalWrite(8 ,LOW);
```

Error compiling for board ESP32 Wrover Module.

Copy error messages

```
WARNING: library Servo claims to run on avr, megaavr, sam, samd, nrf52, stm32f4, mbed architecture(s) and may be incompatible with your current board which runs on esp32 architecture(s).
In file included from C:\Users\HP\Downloads\project_final\project_final\project_final.ino:3:0:
C:\Program Files (x86)\Arduino\libraries\Servo\src/Servo.h:77:2: error: #error "This library only supports boards with an AVR, SAM, SAMD, NRF52 or STM32F4 processor."
```

160 - 157

ESP32 Wrover Module, Huge APP (3MB No OTA/1MB SPIFFS), QIO, 40MHz, 115200, None on COM3

Type here to search

9:52 AM
4/1/2021

project_final | Arduino 1.8.13

File Edit Sketch Tools Help

project_final §

```
if(buzz1){
    digitalWrite(9 ,LOW);
     digitalWrite(8 ,HIGH);
        }
 else{
    digitalWrite(9,HIGH);
        digitalWrite(8 ,LOW);
    }

 boolean pump= pumpcontrol(flamesensvalue1,flamesensvalue2,flamesensvalue3);// when nearly there is a fire this funtion return true

 if(pump){//fire having
  while(pump==0 || input=='6'){
    over(); //callling over funtion to stop motion when nearly fire detected

   digitalWrite(pump1, HIGH);

    for (pos = 50; pos <= 130; pos += 1) {
    myservo.write(pos);
    delay(10);
  }
  for (pos = 130; pos >= 50; pos -= 1) {
    myservo.write(pos);
    delay(10);
  }

 digitalWrite(pump1,LOW);
 myservo.write(90);

    }

 }else{
    digitalWrite(pump1,LOW);
   }//fire havent
```

Error compiling for board ESP32 Wrover Module.

Copy error messages

```
WARNING: library Servo claims to run on avr, megaavr, sam, samd, nrf52, stm32f4, mbed architecture(s) and may be incompatible with your current board which runs on esp32 architecture(s).
In file included from C:\Users\HP\Downloads\project_final\project_final\project_final.ino:3:0:
C:\Program Files (x86)\Arduino\libraries\Servo\src/Servo.h:77:2: error: #error "This library only supports boards with an AVR, SAM, SAMD, NRF52 or STM32F4 processor."
```

160 - 157

ESP32 Wrover Module, Huge APP (3MB No OTA/1MB SPIFFS), QIO, 40MHz, 115200, None on COM3

Type here to search

9:52 AM
4/1/2021

project_final | Arduino 1.8.13

File Edit Sketch Tools Help

project_final §

```
void front()
  {
  analogWrite(6,170);
  analogWrite(5,0);
  analogWrite(4,170);
  analogWrite(3,0);
  }
 void back()
  {
  analogWrite(6,0);
  analogWrite(5,170);
  analogWrite(4,0);
  analogWrite(3,170);
  }

  void right()
  {
  analogWrite(6,0);
  analogWrite(5,0);
  analogWrite(4,170);
  analogWrite(3,0);
   }

   void left(){
  analogWrite(6,170);
  analogWrite(5,0);
  analogWrite(4,0);
  analogWrite(3,0);
   }

    void over()
   {
  analogWrite(6,0);
  analogWrite(5,0);
  analogWrite(4,0);
```

Error compiling for board ESP32 Wrover Module.

Copy error messages

```
WARNING: library Servo claims to run on avr, megaavr, sam, samd, nrf52, stm32f4, mbed architecture(s) and may be incompatible with your current board which runs on esp32 architecture(s).
In file included from C:\Users\HP\Downloads\project_final\project_final\project_final.ino:3:0:
C:\Program Files (x86)\Arduino\libraries\Servo\src/Servo.h:77:2: error: #error "This library only supports boards with an AVR, SAM, SAMD, NRF52 or STM32F4 processor."
```

160 - 157

ESP32 Wrover Module, Huge APP (3MB No OTA/1MB SPIFFS), QIO, 40MHz, 115200, None on COM3

Type here to search

9:53 AM
4/1/2021

project_final | Arduino 1.8.13

File Edit Sketch Tools Help

project_final §

```
    void over()
   {
  analogWrite(6,0);
  analogWrite(5,0);
  analogWrite(4,0);
  analogWrite(3,0);
   }


boolean buzz(int first,int second,int third){//this is function to detect fire with some distanse

  if(first<700 || second<700|| third<700){
    return true;

  }
  else{
    return false;
    }
  }

 boolean pumpcontrol(int value1,int value2,int value3){//this funtion to nearly fire
       if(value1<100 || value2<100|| value3<100){
         over();
        return true;


  }
  else{
    return false;
    }

  }
```

Error compiling for board ESP32 Wrover Module.

Copy error messages

```
WARNING: library Servo claims to run on avr, megaavr, sam, samd, nrf52, stm32f4, mbed architecture(s) and may be incompatible with your current board which runs on esp32 architecture(s).
In file included from C:\Users\HP\Downloads\project_final\project_final\project_final.ino:3:0:
C:\Program Files (x86)\Arduino\libraries\Servo\src/Servo.h:77:2: error: #error "This library only supports boards with an AVR, SAM, SAMD, NRF52 or STM32F4 processor."
```

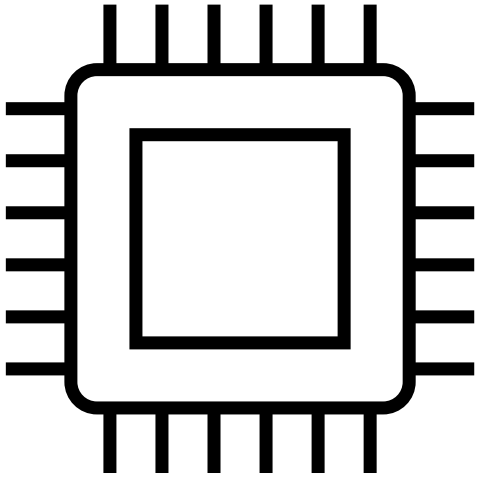
160 - 157

ESP32 Wrover Module, Huge APP (3MB No OTA/1MB SPIFFS), QIO, 40MHz, 115200, None on COM3

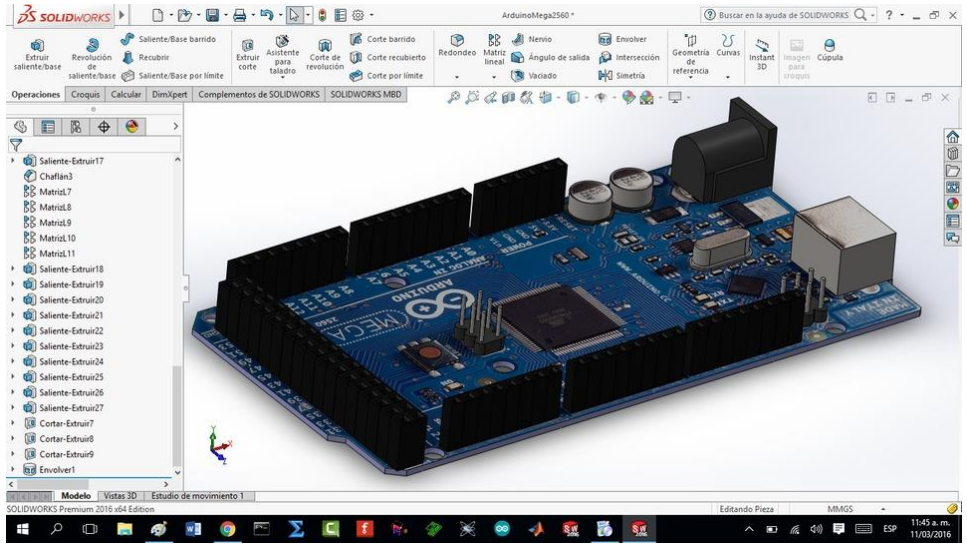
Type here to search

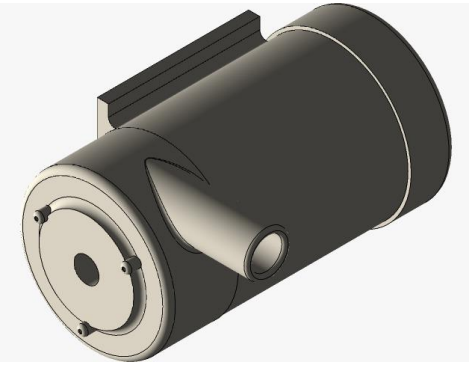
9:53 AM
4/1/2021

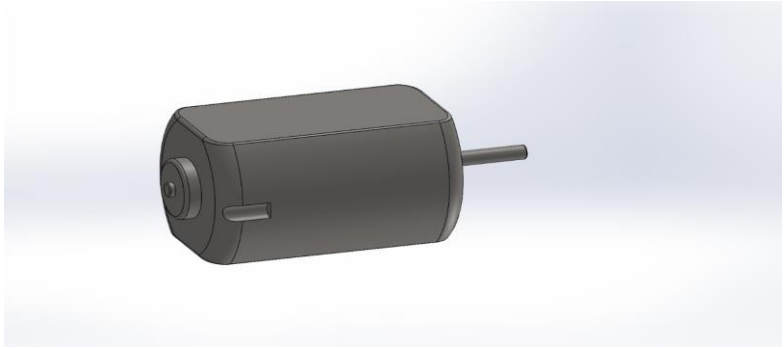
DESIGNING PART

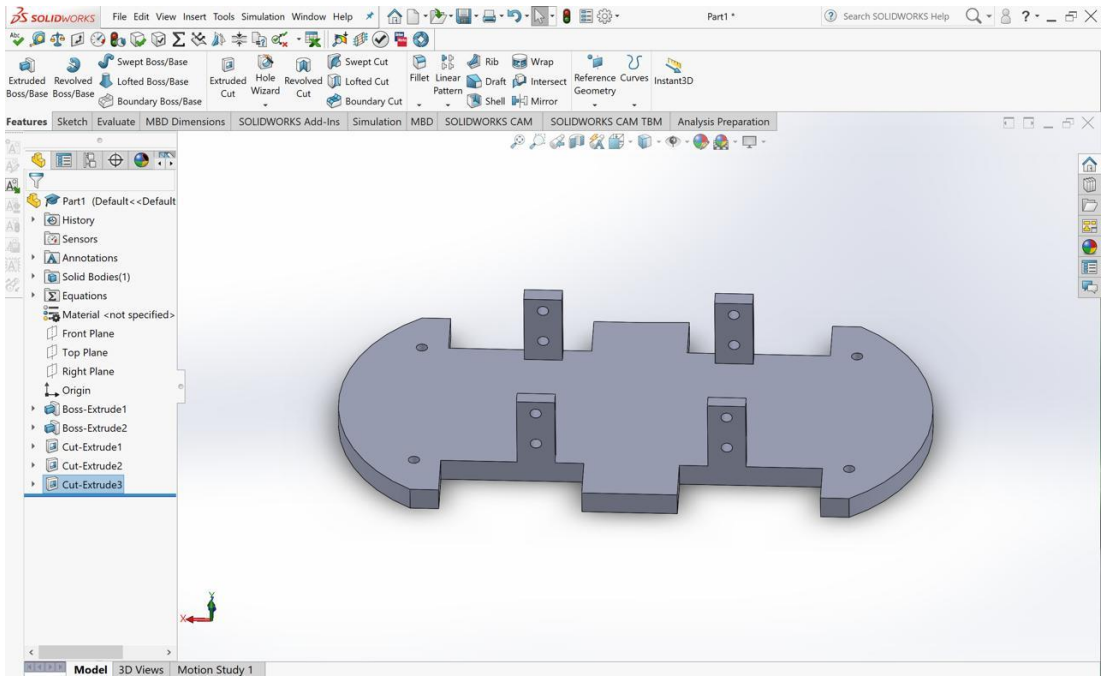


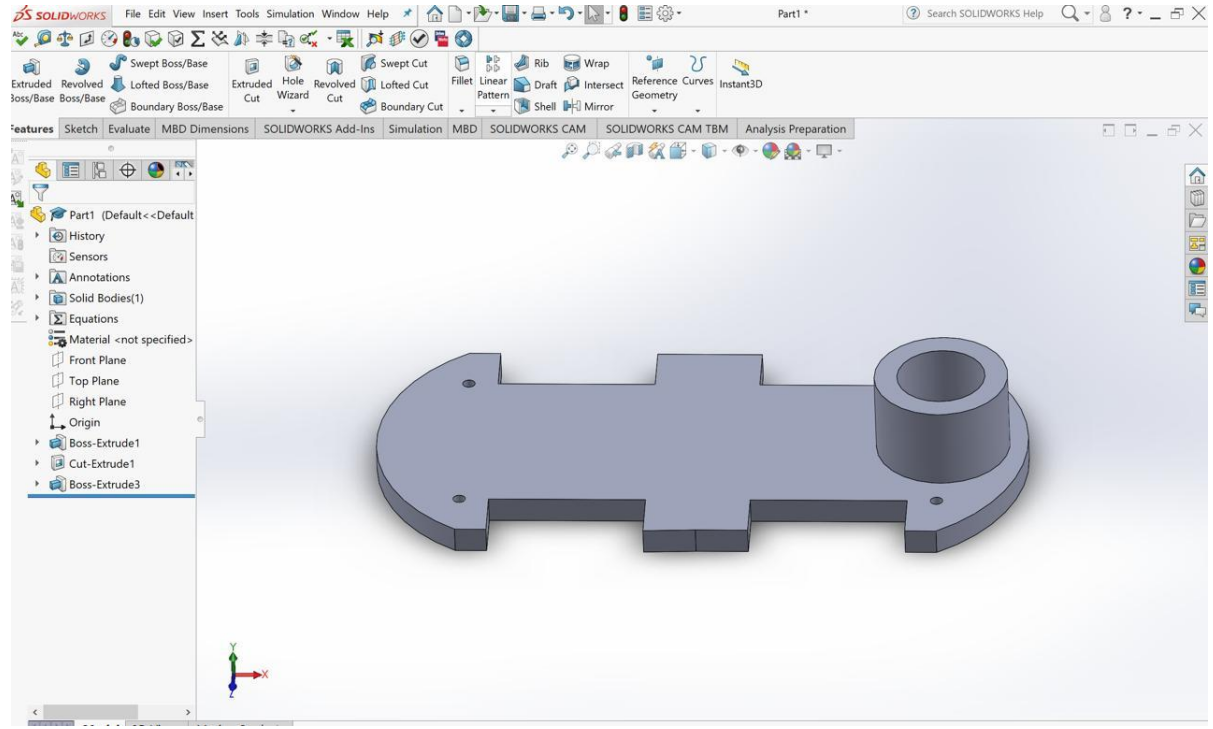
DESIGN STRUCTURE



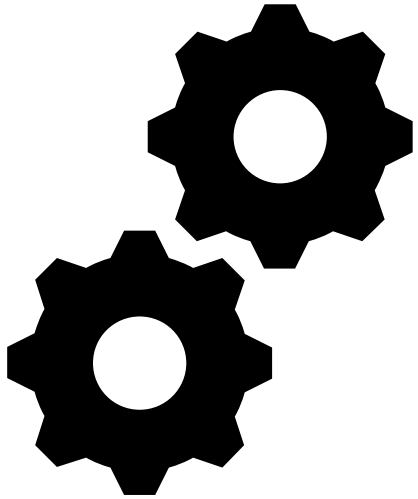




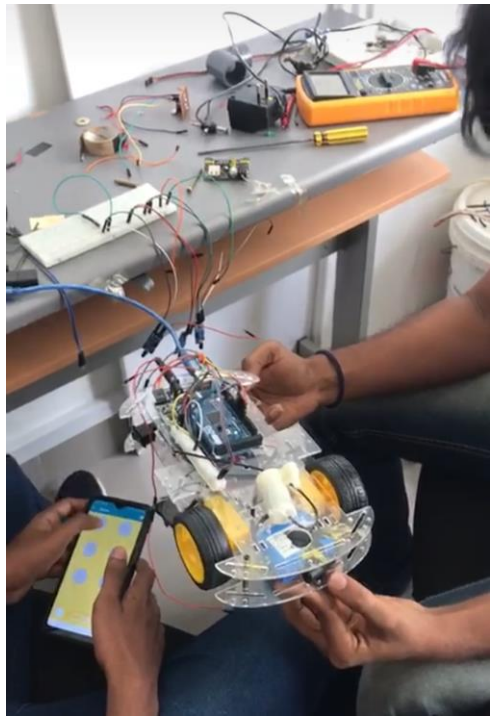

SOLIDWORKS
File Edit View Insert Tools Simulation Window Help
Part1 *
Search SOLIDWORKS Help
Extruded Boss/Base
Revolved Boss/Base
Swept Boss/Base
Lofted Boss/Base
Boundary Boss/Base
Extruded Cut
Hole Wizard
Revolved Cut
Swept Cut
Lofted Cut
Boundary Cut
Fillet
Linear Pattern
Rib
Draft
Shell
Wrap
Intersect
Mirror
Reference Geometry
Curves
Instant3D
Features
Sketch
Evaluate
MBD Dimensions
SOLIDWORKS Add-Ins
Simulation
MBD
SOLIDWORKS CAM
SOLIDWORKS CAM TBM
Analysis Preparation
Part1 (Default<<Default
History
Sensors
Annotations
Solid Bodies(1)
Equations
Material <not specified>
Front Plane
Top Plane
Right Plane
Origin
Boss-Extrude1
Boss-Extrude2
Cut-Extrude1
Cut-Extrude2
Cut-Extrude3
Model
3D Views
Motion Study 1

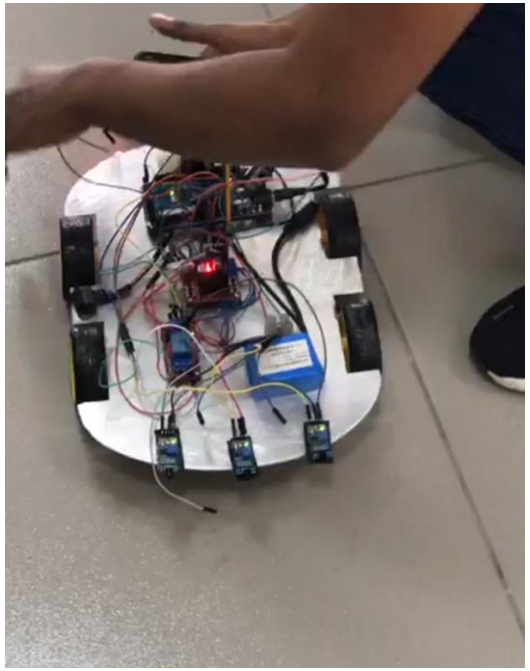

SOLIDWORKS
File Edit View Insert Tools Simulation Window Help
Part1 *
Search SOLIDWORKS Help
Extruded Boss/Base
Revolved Boss/Base
Swept Boss/Base
Lofted Boss/Base
Boundary Boss/Base
Extruded Cut
Hole Wizard
Revolved Cut
Swept Cut
Lofted Cut
Boundary Cut
Fillet
Linear Pattern
Rib
Draft
Shell
Wrap
Intersect
Mirror
Reference Geometry
Curves
Instant3D
Features
Sketch
Evaluate
MBD Dimensions
SOLIDWORKS Add-Ins
Simulation
MBD
SOLIDWORKS CAM
SOLIDWORKS CAM TBM
Analysis Preparation
Part1 (Default<<Default
History
Sensors
Annotations
Solid Bodies(1)
Equations
Material <not specified>
Front Plane
Top Plane
Right Plane
Origin
Boss-Extrude1
Cut-Extrude1
Boss-Extrude3

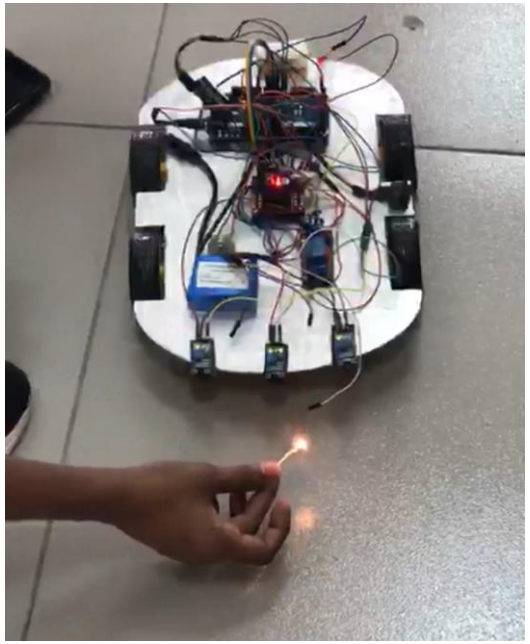
MECHANICAL PART

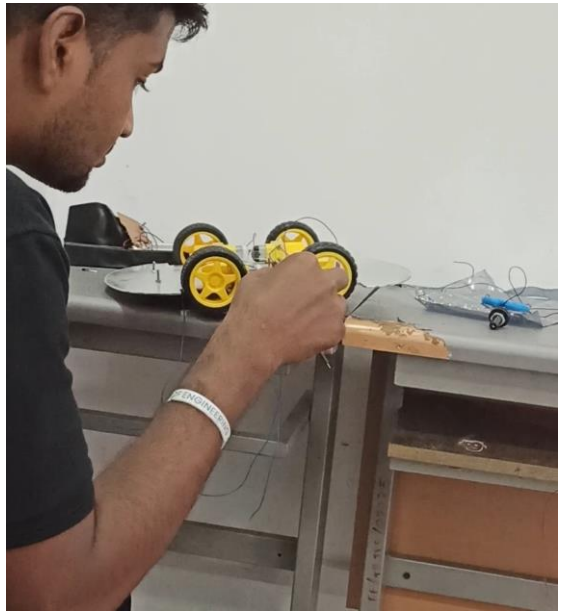


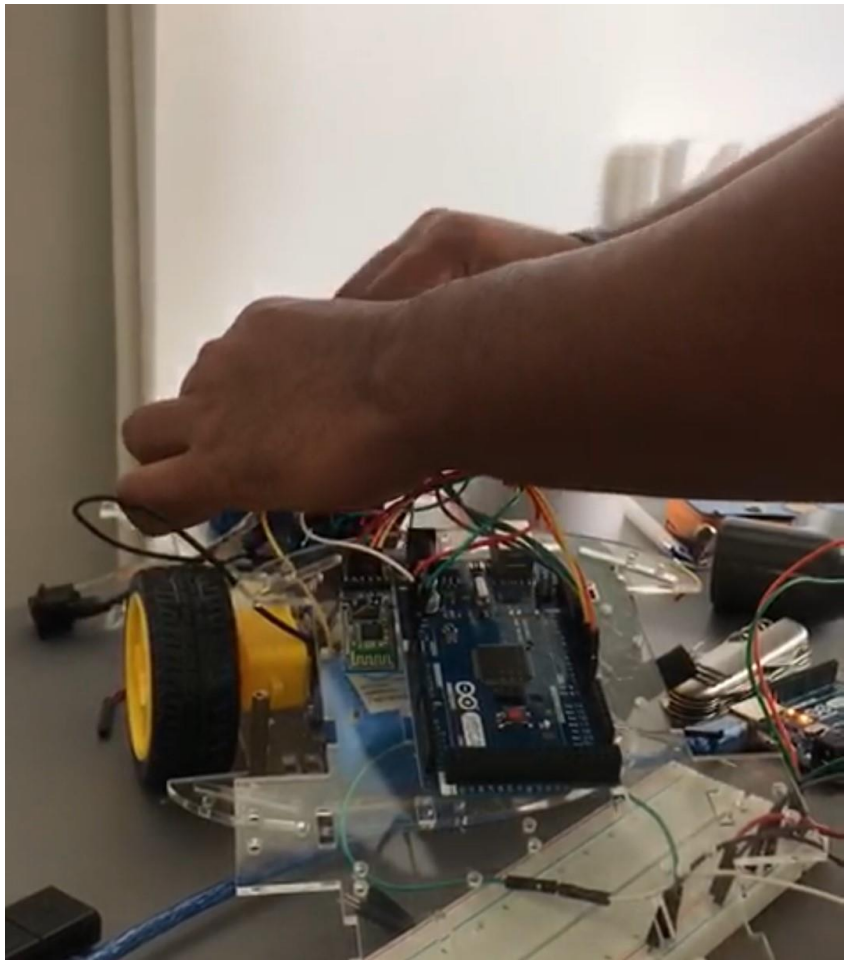
- Wheels are controlled with DC motors which uses a motor driver to control the motors so we can control the whole robot via Bluetooth by a mobile application.
- When a fire is detected by thermal sensors the pump starts pumping water from the container.
- A servo motor rotates the hose so a maximum distance can be obtained.

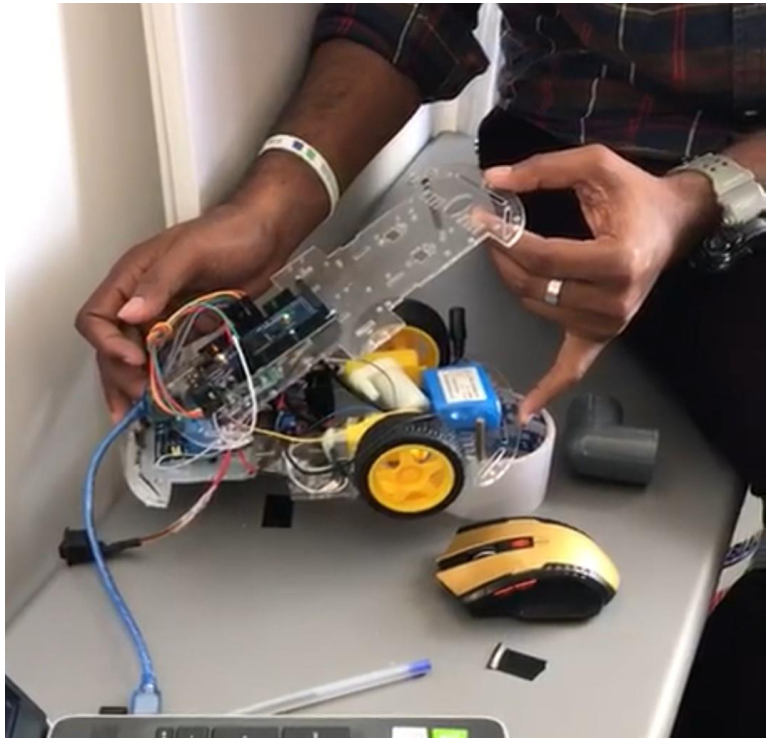


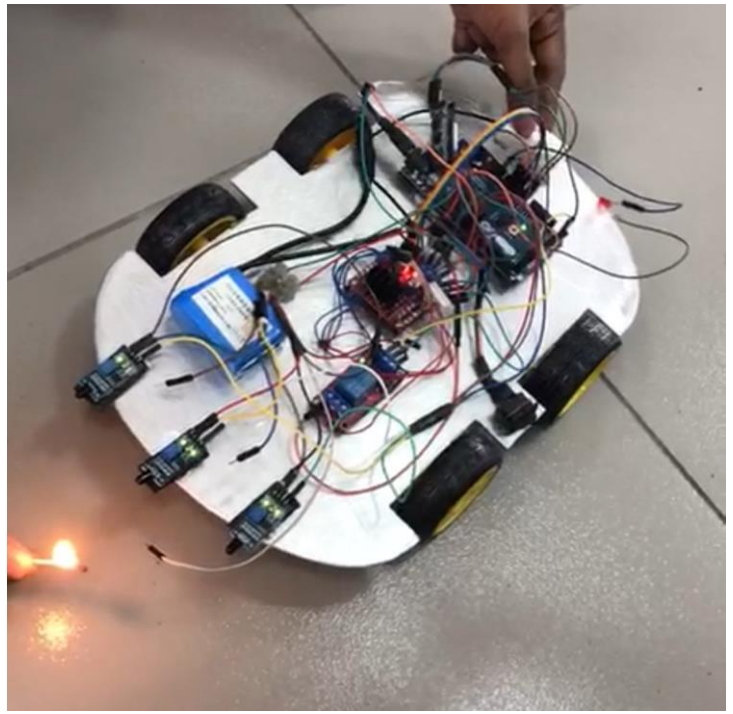












THANK YOU