

**UNIVERSITY OF CHITTAGONG**

**Department of Computer Science & Engineering**

**IoT Project Report: Fire Fighting Robot**

**Presented By:**

**Group: D1**

Sadia Afroj -21701031

Md. Shoaib-21701032

Misba Ul Hoque Arafat-21701033

Afroza Khanam Shati-21701034

Tonmoy Roy-21701035 (group leader)

Jahirul Islam-21701037

**Project Submitted to:**

Nasrin Sultana, Assistant Professor,

Computer Science and Engineering,

University of Chittagong, Chittagong-4331

1. **Project Name: Fire Fighting Robot**
2. **Components Required**

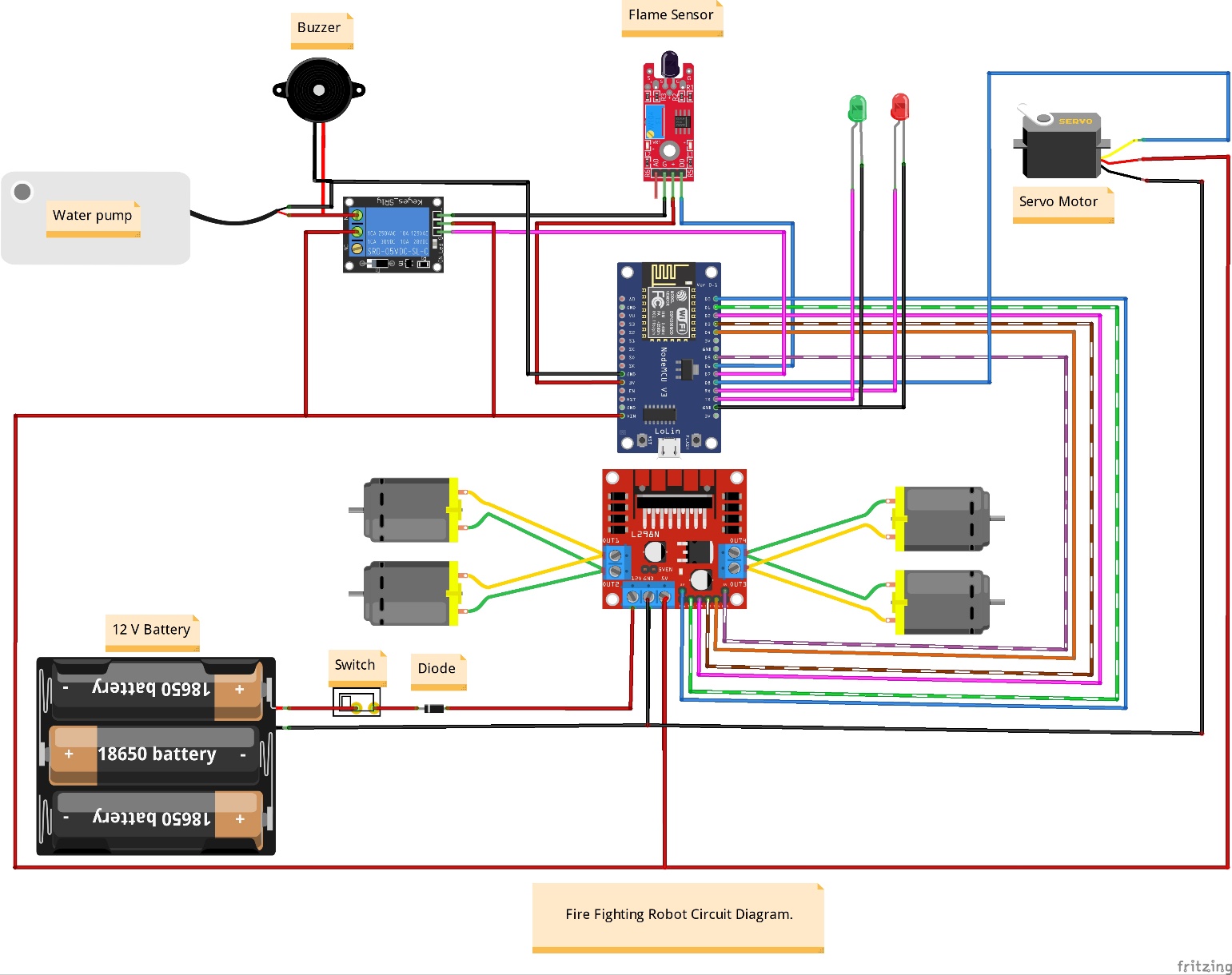
* NodeMCU esp8266
* USB – A to micro-USB cable
* Car chassis
* L298 motor driver module
* Flame sensor module
* Servo Motor
* Relay Module
* Buzzer
* Green LED
* Red LED
* Mini DC Submersible Pump
* 12V Battery
* On-Off- Switch
* DC Female Connector Jack
* Connecting wires
* Soldering iron
* Solder wire
* Hot Melt Glue Gun
* **Software Required**
* Arduino IDE

1. **Objective**

* Design and development of low-cost firefighting robot.
* Run automatic and manual firefighting robot.
* Fire Alarm system
* Send mail to the owner.
* Extinguish fire.

1. **Introduction**

Many fire disasters occur worldwide, resulting in high losses of buildings, factories, etc. Fire is often uncontrollable in buildings and many other places due to fewer sources. It’s hard to access the internal building rooms for firefighters due to high temperatures. There must also be many explosive materials that may result in large-scale complications. According to reports over 1.6 million fire incidents occur which results in 27,027 deaths. Looking at these problems faced by Bangladeshi firefighters, the robot is used to extinguish the fire by entering various buildings through a distance to reduce risk. The movements are controlled through a smartphone via Node MCU ESP8266. The 12V water pump is used as an extinguisher to extinguish the fire which is in direct phase with the battery. We’ve used a flame sensor to sense the fire by emission of visible UV and IR radiation. L298N driver IC is used for the movement of the motors commanding to move forward, backward, left, right, and stop. The robot was equipped with a 12V Battery, regulating 5V supply and producing 5V to onboard LED. In the fire examination, the robot can perform the task of fire extinguishing properly.

1. **Circuit Diagram**

Relay Module

1. **Code**

#define BLYNK\_PRINT Serial

#include <Servo.h>

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>

//Motor PINs

#define ENA D0

#define IN1 D1

#define IN2 D2

#define IN3 D3

#define IN4 D4

#define ENB D5

Servo motor1;

int F\_sensor=12;

int pump=13;//buzzer also

int servopin=15;

int green\_led=3;

int red\_led=1;

bool forward = 0;

bool backward = 0;

bool left = 0;

bool right = 0;

int Speed;

char auth[] = "\*\*\*\*\*\*\*"; //Enter your Blynk application auth token

char ssid[] = "\*\*\*\*\*\*\*"; //Enter your WIFI name

char pass[] = "\*\*\*\*\*\*\*"; //Enter your WIFI passowrd

void setup() {

//Initialize the serial monitor

Serial.begin(115200);

//Set the motor pins as the output pin

pinMode(ENA, OUTPUT);

pinMode(IN1, OUTPUT);

pinMode(IN2, OUTPUT);

pinMode(IN3, OUTPUT);

pinMode(IN4, OUTPUT);

pinMode(ENB, OUTPUT);

motor1.attach(servopin);

pinMode(pump, OUTPUT);

pinMode(green\_led, OUTPUT);

pinMode(red\_led, OUTPUT);

//Initialize the blynk communication

Blynk.begin(auth, ssid, pass);

}

void put\_off\_fire() {

delay (500);

Stop();

delay(500);

digitalWrite(pump, HIGH); delay(500);

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(pump,LOW);

myservo.write(90);

}

void cradle() {

//you begin your own personal code for servo here

int pos;

digitalWrite(pump,HIGH);

for (pos = 40; pos <= 140; pos ++) { // goes from 40 degrees to 140 degrees

// in steps of 1 degree

motor1.write(pos); // tell servo to go to position in variable 'pos'

delay(10); // waits 15ms for the servo to reach the position

}

for (pos = 140; pos >=40; pos --) { // goes from 140 degrees to 40 degrees

motor1.write(pos); // tell servo to go to position in variable 'pos'

delay(10); // waits 15ms for the servo to reach the position

}

digitalWrite(pump,LOW);

delay(10);

//your personal code for servo should end here

}

void automatic(){

while(digitalRead(F\_sensor)){

digitalWrite(red\_led,HIGH);

Blynk.logEvent("fire\_alarm","Fire Fire!!!");

put\_off\_fire();

}

digitalWrite(led,LOW);

}

//Get values from the widgets

BLYNK\_WRITE(V0) {

forward = param.asInt();

}

BLYNK\_WRITE(V1) {

backward = param.asInt();

}

BLYNK\_WRITE(V2) {

left = param.asInt();

}

BLYNK\_WRITE(V3) {

right = param.asInt();

}

BLYNK\_WRITE(V4) {

Speed = param.asInt();

}

//Extinguish fire

BLYNK\_WRITE(V5)

{

int pinValue = param.asInt();

if (pinValue == 1) { // if Button sends 1

Serial.println("servo and led is on");

digitalWrite(red\_led,HIGH);

digitalWrite(green\_led,LOW);

Blynk.logEvent("fire\_alarm","Fire Fire!!!");

cradle(); // start the function cradle

Blynk.run(); // Run rest of show in-between waiting for this loop to repeat or quit.

int pinValue = 0; // Set V5 status to 0 to quit, unless button is still pushed (as per below)

Blynk.syncVirtual(V5); // ...Then force BLYNK\_WRITE(V0) function check of button status to determine if repeating or done.

}

else{

digitalWrite(red\_led,LOW);

digitalWrite(green\_led,HIGH);

digitalWrite(pump,LOW);

Serial.println("servo and led is off");

}

}

//Autonomous mode

BLYNK\_WRITE(V6)

{

int pinValue = param.asInt();

if (pinValue == 1) { // if Button sends 1

if (digitalRead(F\_sensor) ==0) //If Fire not detected all sensors are zero

{

//Do not move the robot

digitalWrite(green\_led,HIGH);

digitalWrite(red\_led,LOW);

Stop();

}

else if (digitalRead(F\_sensor) ==1) //If Fire is straight ahead

{

//Move the robot forward

digitalWrite(green\_led,LOW);

digitalWrite(red\_led,HIGH);

Blynk.logEvent("fire\_alarm","Fire Fire!!!");

Forward();

}

delay(300); //Slow down the speed of robot

automatic(); // start the function put\_off\_fire

Blynk.run(); // Run rest of show in-between waiting for this loop to repeat or quit.

int pinValue = 0; // Set V6 status to 0 to quit, unless button is still pushed (as per below)

Blynk.syncVirtual(V6);

}

else{

Serial.println("auto servo and led is off");

}

}

//Check widget values using the IF condition

void smartcar() {

if (forward == 1) {

Forward();

Serial.println("Forward");

} else if (backward == 1) {

Backward();

Serial.println("Backward");

} else if (left == 1) {

Left();

Serial.println("Left");

} else if (right == 1) {

Right();

Serial.println("Right");

} else if (forward == 0 && backward == 0 && left == 0 && right == 0) {

Stop();

Serial.println("Stop");

}

}

void loop() {

//Run the blynk library

Blynk.run();

smartcar();

if(digitalRead(F\_sensor)==0){

digitalWrite(green\_led,HIGH)

}

//Motor control functions

void Forward() {

analogWrite(ENA, Speed);

analogWrite(ENB, Speed);

digitalWrite(IN1, LOW);

digitalWrite(IN2, HIGH);

digitalWrite(IN3, HIGH);

digitalWrite(IN4, LOW);

}

void Backward() {

analogWrite(ENA, Speed);

analogWrite(ENB, Speed);

digitalWrite(IN1, HIGH);

digitalWrite(IN2, LOW);

digitalWrite(IN3, LOW);

digitalWrite(IN4, HIGH);

}

void Left() {

analogWrite(ENA, Speed);

analogWrite(ENB, Speed);

digitalWrite(IN1, HIGH);

digitalWrite(IN2, LOW);

digitalWrite(IN3, HIGH);

digitalWrite(IN4, LOW);

}

void Right() {

analogWrite(ENA, Speed);

analogWrite(ENB, Speed);

digitalWrite(IN1, LOW);

digitalWrite(IN2, HIGH);

digitalWrite(IN3, LOW);

digitalWrite(IN4, HIGH);

}

void Stop() {

digitalWrite(IN1, LOW);

digitalWrite(IN2, LOW);

digitalWrite(IN3, LOW);

digitalWrite(IN4, LOW);

}

1. **Working**

So, in the working of Fire fighter robot, we are providing the algorithm first so you can have an idea about the basic working of the robot.

There we are using one IR flame sensor which is continuously seeking for fire or flame. The IR Flame sensor senses the warmth and heat of anybody. and we coded this sensor so that it could sense the flame around it. The sensor is always searching for fire, if the sensor finds it the robot will turn and start walking toward the fire.

How does it work? The flame sensor senses the fire and sends the information to the NodeMCU which is the brain of this robot. The brain will act according to the condition and information obtained from the sensor. NodeMCU will give commands to the Motors to start in the walk in the desired direction.

The robot will stop near the fire and start watering it till the fire is under control.

1. **Application**

* Fire fighter robot can be used in the Areas where a can’t go
* Fire fighter robot use in war
* Fire fighter robot can be used in big kitchen.
* Fire fighter robot can be used it Restaurent