**INTRODUCTION**

The internet of things, or IoT, is a system of interrelated computing devices, mechanical and digital machines, objects, animals or people that are provided with unique identifiers ([UIDs](https://internetofthingsagenda.techtarget.com/definition/unique-identifier-UID)) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction.

Flame Sensors, Smoke Sensors, Fire Alarms etc. are part of a safety equipment that help us in keeping our homes, offices and stores safe from fire accidents. Almost all modern houses, apartments, malls, cinema halls, theatres, office buildings and shops are equipped with such safety equipment and it is mandatory in some regions to fire safety devices.

A flame detector is a sensor designed to detect and respond to the presence of a flame 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.

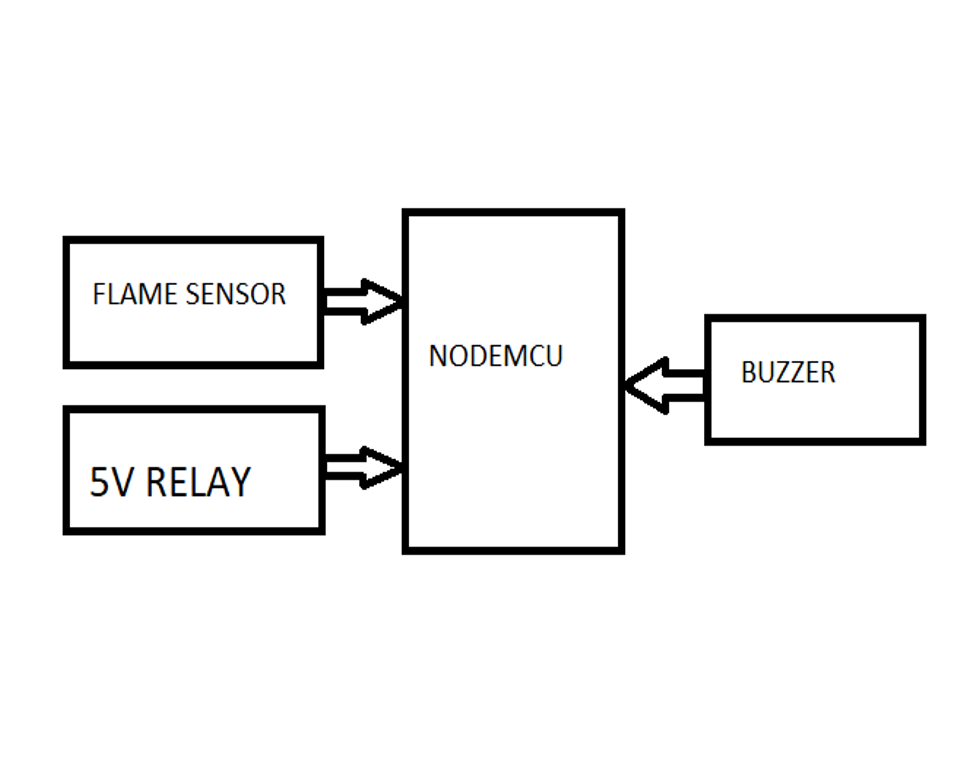
There are different types of flame detection methods. Some of them are: Ultraviolet detector, near IR array detector, infrared (IR) detector, Infrared thermal cameras, UV/IR detector etc.

When fire burns it emits a small amount of Infra-red light, this light will be received by the Photodiode (IR receiver) on the sensor module. Then we use an Op-Amp to check for change in voltage across the IR Receiver, so that if a fire is detected the output pin (DO) will give 0V(LOW) and if the is no fire the output pin will be 5V(HIGH).

In this project we are using an IR based flame sensor. It is based on the YG1006 sensor which is a high speed and high sensitive NPN silicon phototransistor. It can detect infrared light with a wavelength ranging from 700nm to 1000nm and its detection angle is about 60°. Flame sensor module consists of a photodiode (IR receiver), resistor, capacitor, potentiometer, and LM393 comparator in an integrated circuit. The sensitivity can be adjusted by varying the on board potentiometer. Working voltage is between 3.3v and 5v DC, with a digital output. Logic high on the output indicates presence of flame or fire. Logic low on output indicates absence of flame or fire.

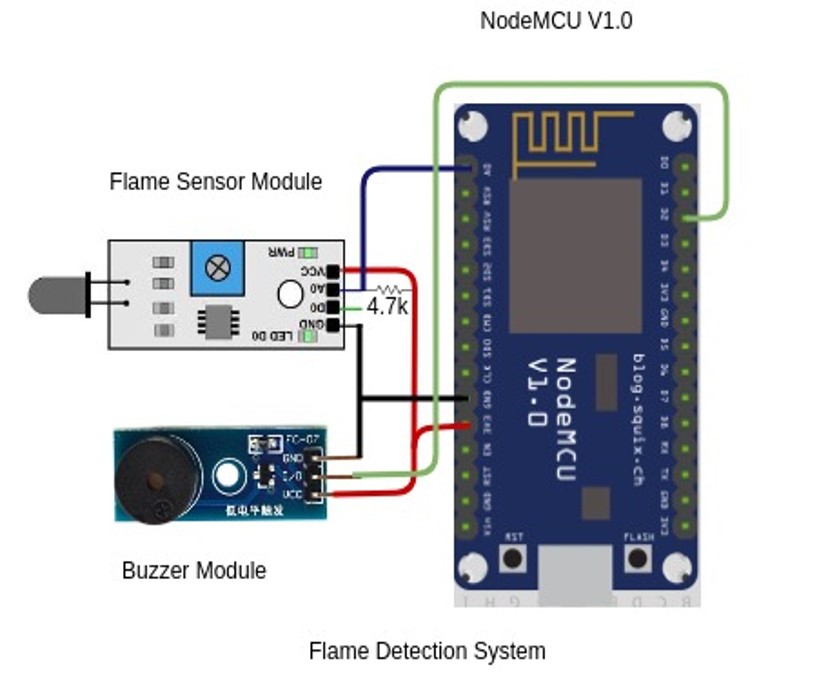
**DESIGN**

**FLOWCHART:**

****

1. We are going to define pins for Flame sensor, and buzzer which are connected to Nodemcu.
2. Based on analog value we will detect the presence of flame. Based on the value stored in “threshold”, we have to turn on the buzzer.
3. If the value of threshold is high then under normal conditions, the output from the Flame Sensor is HIGH.
4. When the sensor detects any fire, its output becomes LOW.
5. Arduino detects this LOW signal on its input pin and activates the Buzzer. compare the value with 0 or 1. If its equal to 1, it indicates that flame has been detected.
6. We have to turn on buzzer and then send value to the web page.
7. If its equal to 0, then it indicates that no flame has been detected. This process is repeated.

CIRCUIT DIAGRAM:



**CONNECTIONS:**

|  |  |  |
| --- | --- | --- |
| **NODEMCU** | **FLAME SENSOR** | **BUZZER** |
| GND | GND | **-** |
| D0 | A0 | **-** |
| VCC | VCC |  |
| GND | **-** | NEGATIVE TERMINAL |
| D1 | **-** | POSITIVE TERMINAL |

**IMPLEMENTATION**

#include <ESP8266WiFi.h>

const char\* ssid = "sarala";

const char\* password = "chandana96";

const int BuzzerPin = 16; // Buzzer output pin

const int threshold = 40; // Flame level threshold (You can vary the value depends on your need)

const int analogPin = A0;

int WiFiStrength = 0;

WiFiServer server(80);

void setup() {

Serial.begin(115200);

delay(10);

pinMode(BuzzerPin, OUTPUT);

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// Connect to Wi-Fi network with SSID and password

Serial.print("Connecting to ");

Serial.println(ssid);

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED) {

delay(500);

Serial.print(".");

}

// Print local IP address and start web server

Serial.println("");

Serial.println("WiFi connected.");

Serial.println("IP address: ");

Serial.println(WiFi.localIP());

server.begin();

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// Connect to WiFi network

//Serial.println();

//Serial.println();

//Serial.print("Connecting to ");

//Serial.println(ssid);

//WiFi.begin(ssid, password);

// Set the ip address of the webserver

// WiFi.config(WebServerIP, Gatway, Subnet)

// or comment out the line below and DHCP will be used to obtain an IP address

// which will be displayed via the serial console

//WiFi.config(IPAddress(192, 168, 1, 221), IPAddress(192, 168, 1, 1), IPAddress(255, 255, 255, 0));

// connect to WiFi router

//while (WiFi.status() != WL\_CONNECTED) {

//delay(500);

//Serial.print(".");

//}

//Serial.println("");

//Serial.println("WiFi connected");

// Start the server

//server.begin();

//Serial.println("Server started");

// Print the IP address

//Serial.print("Use this URL to connect: ");

//Serial.print("http://");

//Serial.print(WiFi.localIP());

//Serial.println("/");

}

double analogValue = 0.0;

void loop() {

WiFiStrength = WiFi.RSSI();

int analogValue = analogRead(analogPin);

Serial.println(analogValue);//serial print the FLAME sensor value

// convert the analog signal to voltage

// the ESP2866 A0 reads between 0 and ~3 volts, producing a corresponding value

// between 0 and 1024. The equation below will convert the value to a voltage value.

if (analogValue < threshold) {

digitalWrite(BuzzerPin, HIGH);

Serial.print("High FLAME");

delay(500);

}

else {

digitalWrite(BuzzerPin, LOW);

Serial.print("No flame");

delay(100);

}

// Serial data

Serial.print("Analog raw: ");

Serial.println(analogValue);

Serial.print("Analog V: ");

Serial.print("millis(): ");

Serial.println(millis());

Serial.print("WiFi Strength: ");

Serial.print(WiFiStrength); Serial.println("dBm");

Serial.println(" ");

delay(1000); // slows amount of data sent via serial

// check to for any web server requests. ie - browser requesting a page from the webserver

WiFiClient client = server.available();

if (!client) {

return; }

// Wait until the client sends some data

Serial.println("new client");

// Read the first line of the request

String request = client.readStringUntil('\r');

Serial.println(request);

client.flush();

// Return the response

client.println("HTTP/1.1 200 OK");

client.println("Content-Type: text/html");

client.println(""); // do not forget this one

client.println("<!DOCTYPE HTML>");

client.println("<html>");

client.println(" <head>");

client.println("<meta http-equiv=\"refresh\" content=\"60\">");

client.println(" </script>");

client.println(" </head>");

client.println(" <body style=\"background-color:powderblue;\">");

client.println("<center>");

client.print("<h1 style=\"size:20px;\">FIRE ALARM SYSTEM</h1>");

client.println("<img src=\"C:\\Users\\Divya P Hathwar\\Documents\\Arduino\\flame\\ff\\fire.png\" width=\"10px\" height=\"10px\">");

// show some data on the webpage and the guage

client.println("<table><tr><td>");

client.print("<p style=\"size:15px;\">WiFi Signal Strength:</p>");

client.println(WiFiStrength);

client.println("dBm<br>");

client.print("<p style=\"size=15px;\">Analog Raw:</p>");

client.println(analogValue);

client.println("<br>");

if(analogValue<threshold)

client.println("<strong>HIGH FLAME</strong>");

else

client.println("<strong>NO FLAME</strong>");

client.println("</td></tr></table>");

client.println("</center>");

client.println("<script>setInterval(function(){window.location.reload();},3000);</script>");

client.println("<body >");

client.println("</html>");

delay(1);

Serial.println("Client disonnected");

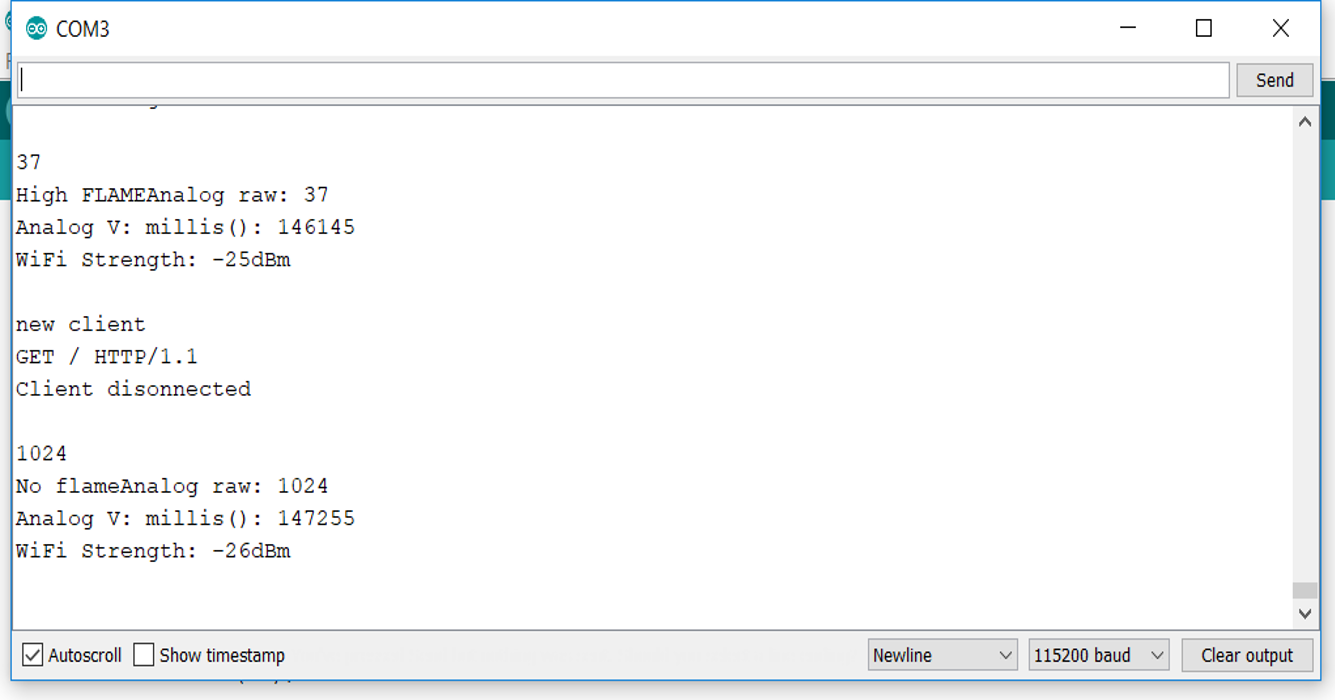
Serial.println("");

}}

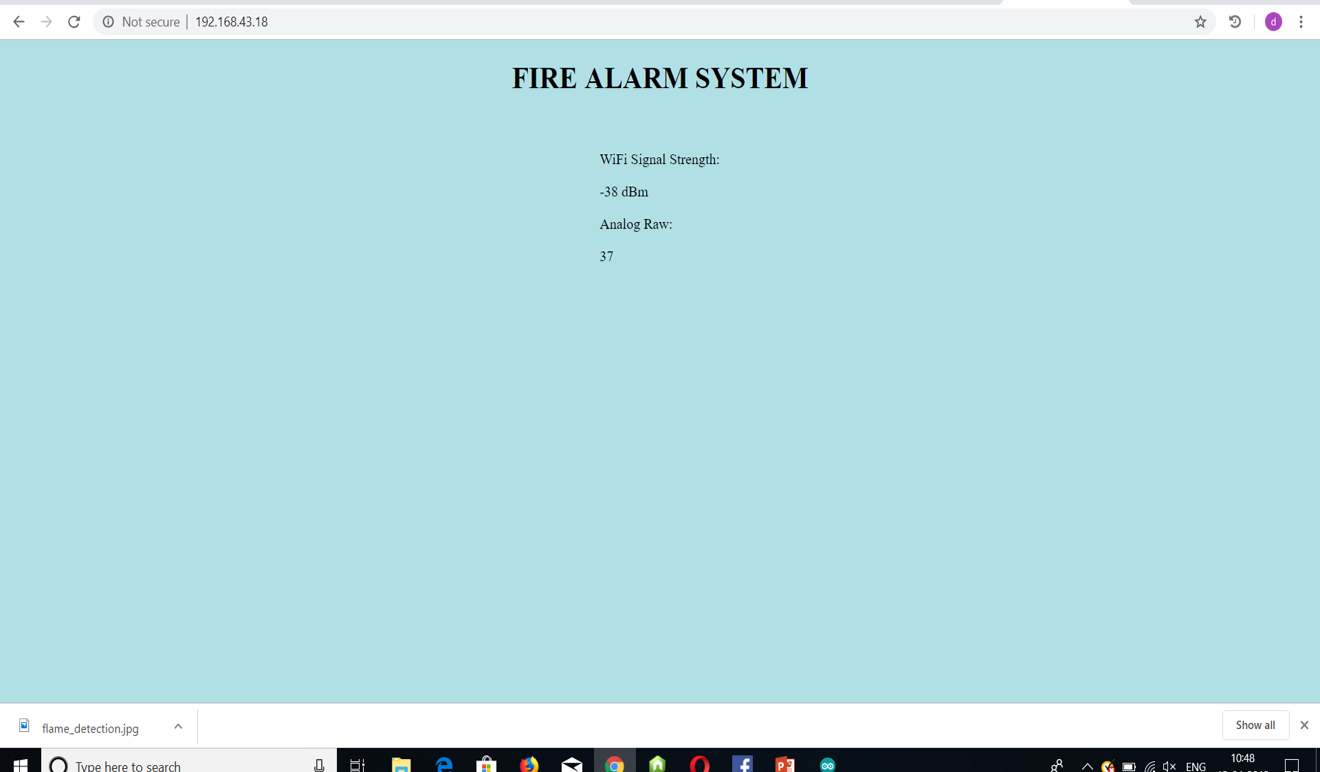
}

**OUTPUT**

RESULTS SNAPSHOT:



WEB INTERFACE:



**CONCLUSION**

# Fire alarm system using flame sensor is one way of making sure the house is safe from fire as it sends a message and the buzzer beeps when there is fire. It’s effective and easy to use. It operates to alert people to evacuate the location is a fire is detected with the help of a buzzer. Our model depicts a Home where our fire alarm system is grounded. The fire alarm constructed by this project is reliable and accurate.

# FUTURE ENHANCEMENTS

* In the future updates there will be notifications sent to the user on phone in case of any fire
* A water sprinkler can be added to decrease or put out the fire.

**REFERENCES**

[1] S. Wadhwani,Uday Singh,”Smart home automation and security system using Arduino and IOT”,IMS engineering college, vol.05, no.2,2018.

[2] Reshma Shinde,Ritika Pardeshi,”Need for Wireless fire detection system using IOT”,Dept.of computer enginnering DYPIET college, vol.04, 2017.

[3] Abinav Sharma,”IOT enabled forest fire detection and monitoring system”, GLBITM, Gr.Noida UP, vol.03,pp. 50- 54, 2017. [4] R. Nirajana,”Dr.T.Hemalatha”, An Autonomous IoT Infrastructure for Forest Fire Detection and Alerting System”, Dept.of computer science and engineering, PSNACET,vol. 119, no. 12, pp.2-10295-10304,2018

[4] <https://www.electronicshub.org/arduino-flame-sensor-interface/>

[5] <https://circuitdigest.com/microcontroller-projects/arduino-flame-sensor-interfacing>