IOT Based Fire Alarm System

Serial No. - 15

Registration No. – RA1711008010385

Name – Gourab Gupta

Batch – 2

Faculty Name – Dr. .Kayalvizhi Jayavel/IT

Abstract

IOT Based Fire Alerting System uses two Sensors, namely, Temperature and Smoke sensors. There is an ADC convertor, which converts the analog signals received at the sensor end to digital and then transmits them to the micro-controller, Arduino. The micro-controller is programmed to turn on the buzzer, when the temperature & the smoke reach a threshold value.

At the same time, Arduino sends the data to the Wi-Fi module ESP8266. ESP8266 is a chip which is used for connecting micro-controllers to Wi-Fi network. ESP8266 will then the following data to the IOT website, where, authorized people can take appropriate measure in order to curb the fire.

1. Temperature (in Degree Celsius)
2. Smoke Value (in Percentage)
3. Device ID
4. Date and Time Stamp

Device ID is the unique ID given to a device, which would help the personnel get information related to the location, where the fire is detected.

The Pre-requisite for this IoT based fire alarming system is that the Wi-Fi module should be connected to a Wi-Fi zone or a hotspot. This project is also implemented without the IOT module. In place of the IOT module, we have used GSM module, by which an SMS is triggered when the buzzer is turned ON.

Introduction

Fire Detectors play a very important role in Industries, Shops, Malls, Residential complexes, parking areas, etc. They help in detecting fire or smoke at an early stage and can help in saving lives. Commercial Fire detecting systems usually have an alarm signaling, with the help of a buzzer or Siren. We have designed an IOT based Fire Alerting System using Temperature and a smoke sensor which would not only signal the presence of fire in a particular premise but will also send related information through IOT.

Internet of Things (IoT) is basically the network of ‘things’ by which physical things can exchange data with the help of sensors, electronics, software, and connectivity. These systems do not require any human interaction. In this Arduino fire alarm system using temperature and smoke sensor using the IOT project, we can send LIVE information like Temperature, Smoke Value detected by a particular device to the Fire Department.

Requirements (Hardware and Software)

1. Arduino Uno

The Arduino Uno is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc. The board is equipped with sets of digital and analog input/output pins that may be interfaced to various expansion boards and other circuits.

1. Gas Sensor (MQ2)

Gas sensor as the name suggests, detects gas leaks or other emissions which can be used to deduct what is happening or going to happen and is very useful particularly in fire safety. The sensor which has the own board on the bottom has four pins to connect Raspberry Pi. These are power, ground, analogue output and digital output. MQ2 sensor has the sensitivity characteristics for each gas for detecting. According to these data curves, Resistance value of MQ-2 is different to various kinds and various concentration gases

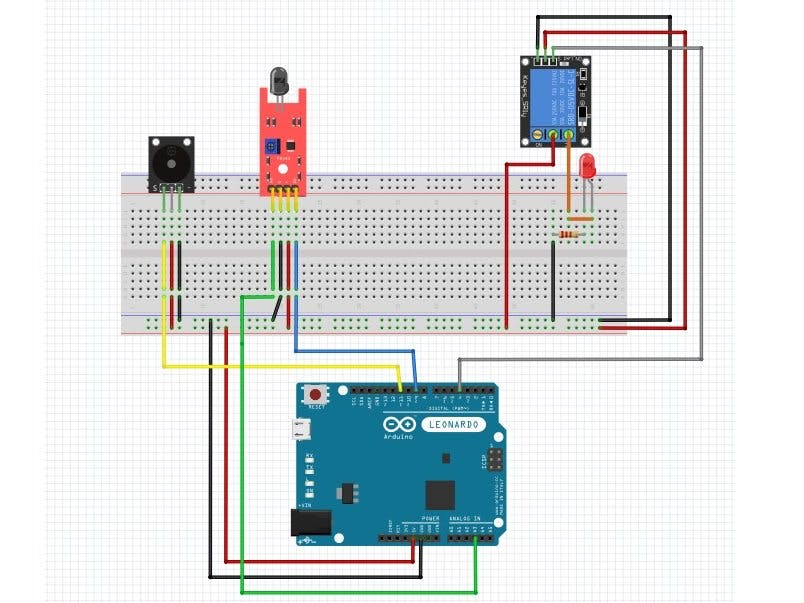
1. Google Firebase Cloud Messaging (FCM)

To send the notification to the mobile device, Google Firebase Cloud Messaging [8], hereafter FCM, is used for our system. FCM is a cross-platform solution for messages and notifications for Android, iOS and web applications which currently can be used at no cost. The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronised in realtime to every connected client. When you build cross-platform apps with our iOS, Android, and JavaScript SDKs, all of your clients share one Realtime Database instance and automatically receive updates with the newest data.

1. Google Map API

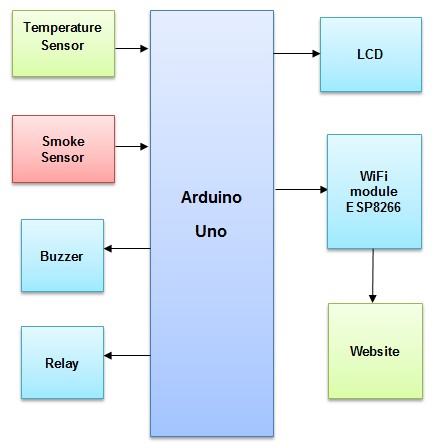
For displaying the location which fire happen, we used Google Maps Android API which can show the map based on Google Maps data to our application. The API automatically handles access to Google Maps servers, data downloading, map display and response to map gestures. Also, developers can use API calls to add markers, polygons, and overlays to a basic map, and to change the user’s view of a particular map area. These objects provide additional information for map locations, and allow user interaction with the map. Because our project was already registered Firebase console account, we can get google map API key by just linking our project on the Google Cloud Platform webpage.

Circuit diagram



Methodology

Block Diagram



The architecture diagram is almost self explanatory. The sensors and the camera act as input devices that gives input data to the Arduino Uno which then processes the information and deduces through the unique algorithm whether there is a fire or not. The buzzer and spikler are close vicinity warning and countermeasure system while the wifi is used to send a message to the security guard and the nearest fire department.

Coding

#include <ESP8266WiFi.h>

const char\* ssid = "Enter your ssid";    
const char\* password = "enter your password";    
char server[] = "mail.smtp2go.com";    
const int flame = D0;   
const int buzz = D1;  
WiFiClient Client;              //define wifi client as client

void setup() {

  pinMode(flame, INPUT);  
  pinMode(buzz, OUTPUT);

  Serial.begin(115200);           
  Serial.println("");  
  Serial.print("Connecting To: ");  
  Serial.println(ssid);  
  WiFi.begin(ssid, password);

  while (WiFi.status() != WL\_CONNECTED)  
  {  
    delay(500);  
    Serial.print(".");  
  }  
  Serial.println("");  
  Serial.println("WiFi Connected.");  
  Serial.print("IP address: ");  
  Serial.println(WiFi.localIP());

}

void loop() {

  int t = digitalRead(flame);

  Serial.println(t);  
   if (t==0) {             
  digitalWrite(buzz, HIGH);  
    sendEmail();  
    Serial.print("Mail sent to:");   
    Serial.println(" The recipient");  
    Serial.println("");  
  }  
digitalWrite(buzz, LOW);  
}

byte sendEmail()  
{  
  if (Client.connect(server, 2525) == 1)        // connect to smtp server with port address 2525  
  {  
    Serial.println(F("connected to server"));  
  }   
  else   
  {  
    Serial.println(F("connection failed"));  
    return 0;  
  }  
  if (!emailResp())         // if connection failed return now  
    return 0;  
  //  
  Serial.println(F("Sending EHLO"));  
  Client.println("EHLO www.example.com");        
  if (!emailResp())   
    return 0;  
      
  Serial.println(F("Sending auth login"));  
  Client.println("AUTH LOGIN");  
  if (!emailResp())   
    return 0;  
    
  Serial.println(F("Sending User"));  
  Client.println("c2VuZGVyQHh5ei5jb20="); //base64, ASCII encoded SMTP Username     
  if (!emailResp())   
    return 0;

  Serial.println(F("Sending Password"));  
  Client.println("cGFzc3dvcmQ=");   //base64, ASCII encoded SMTP Password      
  if (!emailResp())   
    return 0;  
    
  Serial.println(F("Sending From"));  
    Client.println(F("MAIL From: sender@xyz.com"));         
  if (!emailResp())    
    return 0;  
  // change to recipient address  
  Serial.println(F("Sending To"));  
  Client.println(F("RCPT To: receiver@xyz.com"));      
    
  if (!emailResp())   
    return 0;  
    
  Serial.println(F("Sending DATA"));  
  Client.println(F("DATA"));  
  if (!emailResp())   
    return 0;  
  Serial.println(F("Sending email"));  
    
  Client.println(F("To:  receiver@xyz.com "));                  
    
  Client.println(F("From: sender@xyz.com "));                   
  Client.println(F("Subject: Fire Alarm\r\n"));  
  Client.println(F("Attention: Fire Detected.\n"));  
  Client.println(F("."));  
  if (!emailResp())   
    return 0;  
  Serial.println(F("Sending QUIT"));  
  Client.println(F("QUIT"));  
  if (!emailResp())   
    return 0;  
  Client.stop();  
  Serial.println(F("disconnected"));  
  return 1;  
}

byte emailResp()  
{  
  byte responseCode;  
  byte readByte;  
  int loopCount = 0;

  while (!Client.available())   
  {  
    delay(1);  
    loopCount++;  
    // Wait for 20 seconds and if nothing is received, stop.  
    if (loopCount > 20000)   
    {  
      Client.stop();  
      Serial.println(F("\r\nTimeout"));  
      return 0;  
    }  
  }

  responseCode = Client.peek();  
  while (Client.available())  
  {  
    readByte = Client.read();  
    Serial.write(readByte);  
  }

  if (responseCode >= '4')  
  {  
    //  efail();  
    return 0;  
  }  
  return 1;  
}

Implementation/Target Audience

Fire breakouts create serious health and Infrastructure hazard, associated with it is unavoidable injuries or loss of lives in one hand, partial or complete damage to properties. This loss is inestimably enormous; hence this paper proposes the development of IOT based fire department alerting system. This model constantly monitoring the fire signal and will send warning to alert the user and nearest fire station. This application targets people who don’t have someone to be at home, office or any other workplace so when they away from their place they are notified about the fire problems if any. Using this application will help these people as they will be informed quickly about the incident and also the nearest fire department will be notified in an effective way. The application has a notification feature which notifies the user and the nearest fire station plus the domestic help so that a quick action can be taken. It is also very efficient and hence very easy to use. This system has tried to solve almost every problem related to the safety of homes and its assets.

Concludsion

There is an enormous need of usage of programmed fire recognizing framework to shield lives and resources from flame dangers. In this paper full fire assurance frameworks are clarified. Utilization of continuous control by means of the Internet or remote system will broaden the observing and control of flame security frameworks outside of the building. The status of the fire wellbeing framework and other building frameworks can be observed whenever and from anyplace by means of the Internet or remote system. The fire wellbeing frameworks situated in numerous structures will be controlled from one focal office. This will expand the productivity and diminish costs for building administration operations, all the more proficiently separate amongst flame and non-fire dangers and increment the time accessible for property and life security. Be that as it may, Internet based observing and control of building administration frameworks will require security assurance to avert false fire data being given to building proprietors and fire units.

References

<https://iotdesignpro.com/projects/iot-based-fire-alarm-project-using-esp8266>

<https://www.projectsof8051.com/iot-based-fire-alerting-system-project/>

<https://www.ijitee.org/wp-content/uploads/papers/v8i6s3/F10200486S319.pdf>

<https://iotdesignpro.com/projects/iot-based-fire-alarm-project-using-esp8266>