RESEARCH ARTICLE OPEN ACCESS

Fire Detection System Using an Arduino Uno Microcontroller

Dr.Tarig ibrahim Osman Ahmed
Almustaqbal University in KSA of Computer-Engineering College-

Abstract: To summarize, there are basic functions / techniques that have been implemented to ensure that the prototype is used to achieve its primary objectives of reducing the rates of false alarms and their sub-targets being shipped and making fire detection devices “smart”. They are as follows:
1. Use of smoke, temperature and light intensity as fire detection parameters.
2. Use GSM instead of Wi-Fi to send text notifications.

Date of Submission: 13-03-2020 Date Of Acceptance: 28-03-2020

I. INTRODUCTION

Types of fire-fighting systems used in large buildings and companies and type of fire alarm systems.

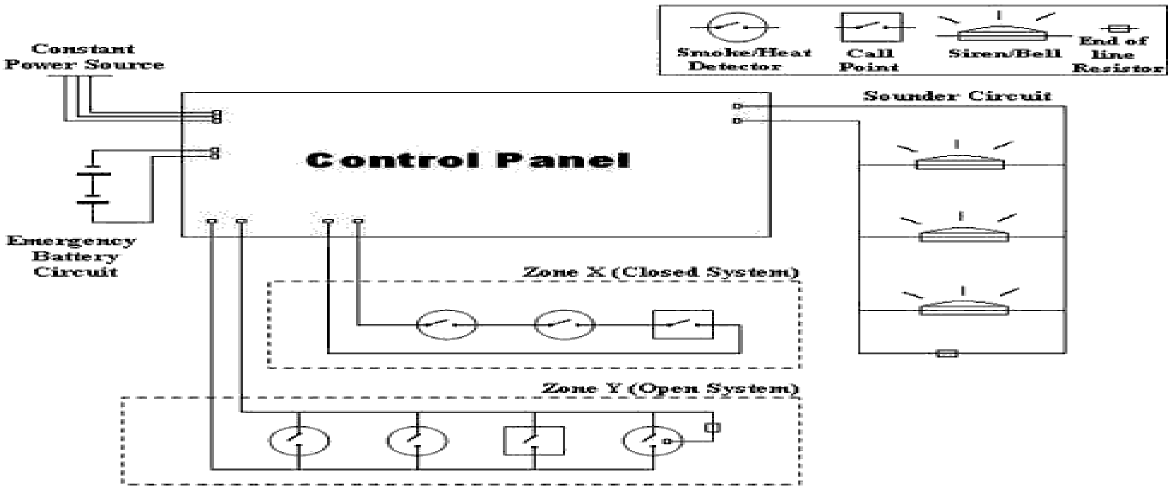
The paper title was chosen based on common fire-fighting systems used to improve the speed of alerting the civil defense department.

Most fire-fighting systems do not support the use of gps or SMS to inform the civil defense department

A **fire alarm control panel** (FACP), **fire alarm control unit** (FACU), or simply **fire alarm panel** is the controlling component of a fire alarm system. The panel receives information from devices designed to detect and report fires, monitors their operational integrity and provides for automatic control of equipment, and transmission of information necessary to prepare the facility for fire based on a predetermined sequence. The panel may also supply electrical energy to operate any associated initiating device, notification appliance, control, transmitter, or relay. There are four basic types of panels: coded panels, conventional panels, addressable panels, and multiplex systems.

Today, there are two types of fire panels:

- Conventional fire alarm panel;
- Addressable fire alarm panel



A wiring diagram for a simple fire alarm system consisting of two input loops (one closed, one open)

II. THE ARDUINO

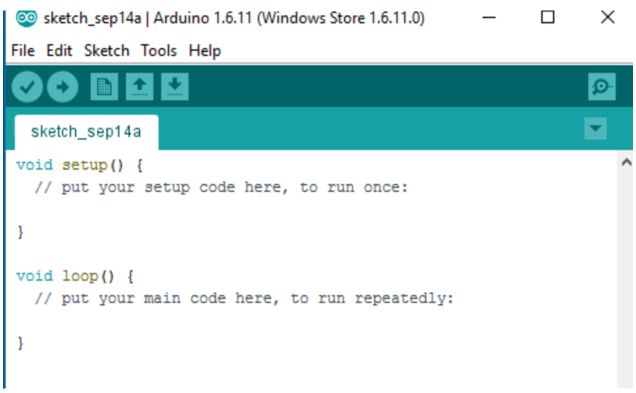
Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.

There are many other microcontrollers and microcontroller platforms available for physical computing. Parallax Basic Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard, and many others offer similar functionality. All of these tools take the messy details of microcontroller programming and wrap it up in an easy-to-use package. Arduino also simplifies the process of working with microcontrollers, but it offers some advantage for teachers, students, and interested amateurs over other systems:

- **Inexpensive** - Arduino boards are relatively inexpensive compared to other microcontroller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than $50
- **Cross-platform** - The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.
- **Simple**, clear programming environment - The Arduino Software (IDE) is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works.
- **Open source and extensible software** - The Arduino software is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries, and people wanting to understand the technical details can make the leap from Arduino to the AVR C programming language on which it's based. Similarly, you can add AVR-C code directly into your Arduino programs if you want to.
- **Open source and extensible hardware** - The plans of the Arduino boards are published under a Creative Commons license, so experienced circuit designers can make their own version of the module, extending it and improving it. Even relatively inexperienced users can build the breadboard version of the module in order to understand how it works and save money.

ARDUINO = TWO THINGS

Software for programing

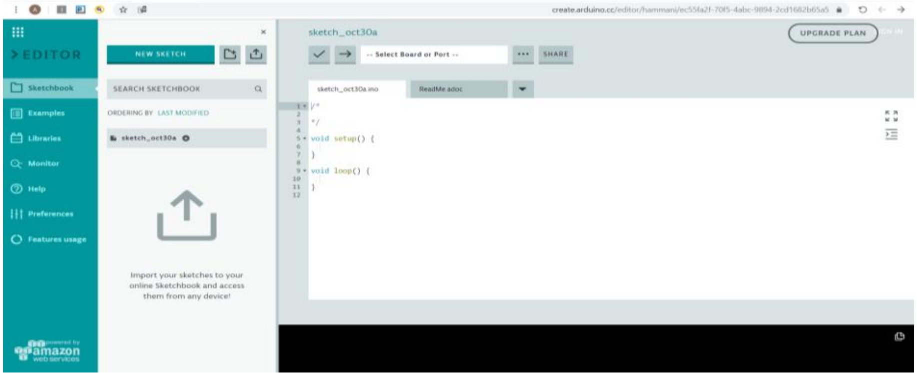

sketch_sep14a | Arduino 1.6.11 (Windows Store 1.6.11.0)
File Edit Sketch Tools Help
sketch_sep14a

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

ARDUINO EDITOR

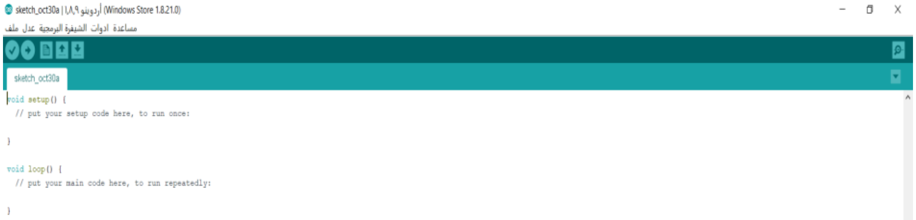
Two ways to editor :

1-arduino web editor


create.arduino.cc/editor/hammani/ec55fa2f-70f5-4abc-9894-2cd1682b65a5
EDITOR
Sketchbook
Examples
Libraries
Monitor
Help
Preferences
Features usage
powered by amazon web services
NEW SKETCH
SEARCH SKETCHBOOK
ORDERING BY LAST MODIFIED
sketch_oct30a
Import your sketches to your online Sketchbook and access them from any device!
sketch_oct30a
-- Select Board or Port --
SHARE
UPGRADE PLAN
sketch_oct30a.ino
ReadMe.adoc

```
/*

*/

void setup() {

}

void loop() {

}
```

2-arduino application editor

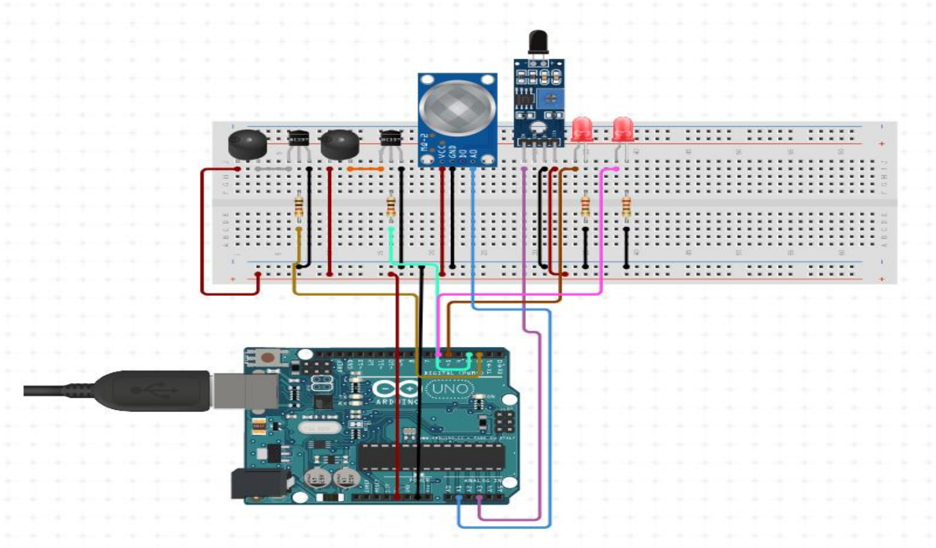

sketch_oct30a | أردوينو ١,٨,٩ (Windows Store 1.8.21.0)
مساعدة ادوات الشيفرة البرمجية عدل ملف
sketch_oct30a

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Our papers used

Arduino uno

Structure



Code

```
int FlamePin = 2;
int Flame = HIGH;
int smokeA0 = A5;
int redled = 8;
int redled1 = 9;
int buzzer = 8;
int sensorThresF = 200;
int sensorThresS = 10;
void setup(){
  pinMode(buzzer, OUTPUT);
  pinMode(8, OUTPUT);
  pinMode(redled1, OUTPUT);
  pinMode(9, OUTPUT);
  pinMode(smokeA0, INPUT);
  pinMode(FlamePin, INPUT);
  digitalWrite(8, HIGH);
  digitalWrite(9, HIGH);
  Serial.begin(9600);
}
void loop()
{
  Flame = digitalRead(FlamePin);
  if (Flame== LOW)
  {
    Serial.println("HIGH FLAME");
        tone(buzzer, 1000,200);
// Send 1KHz sound signal...

         delay(100);

        noTone(buzzer);
                        delay(100);
          digitalWrite(redled1, HIGH);
                    delay(10);
             digitalWrite(redled1, LOW);
                delay(10);
  }
  else
  {
    noTone(buzzer);
       digitalWrite(redled1, LOW);
       }
int analogSensor = analogRead(smokeA0);
  Serial.print("Pin A5: ");
  Serial.println(analogSensor);
  // Checks if it has reached the threshold value
  if (analogSensor > sensorThresS)
  {
          tone(buzzer, 1000,200);
// Send 1KHz sound signal...

         delay(100);
        noTone(buzzer);

         delay(100);// ...for 1 sec
    // ...for 1sec
              digitalWrite(redled1, HIGH);
              delay(10);
              digitalWrite(redled1, LOW);
              delay(10);
  }
  else
      noTone(buzzer);
              digitalWrite(redled1, LOW);

}
```

Recommendation

The following are recommendation based on the research results that were analyzed by the researcher and challenges faced during implementation:

i. A Mobile based application can be developed to aid the home user to manage the device for instance check if its up and running.

ii. The device can be extended to enable it use rechargeable batteries as a power source.

iii. A web application based on the Google maps API for navigation by the fire department once an alert is sent can be developed.

i. The use of a flame sensor instead of light sensor would make the device more accurate.

APPENDICES AND REFERENCES

[1]. LancashireFire (http://www.youtube.com/watch?v=ezJ6SorlpJo. 7 March 2013). *Bedroom Fire Test* [MP4]. Retrieved from

[2]. Wikipedia. *YUV*. http://en.wikipedia.org/wiki/YUV.

[3]. Shafkat Hossain (9 April 2014). Fire detection using matlab one source [MP4].Retrieved from http://www.youtube.com/watch?v=GgIDLeO60yc

[4]. Arduino Uno, http://arduino.cc/en/Main/arduinoBoardUno.

Tarig Ibrahim Osman Ahmed

1. PhD in Computer Engineering, Sudan Academic of Science and Technology 2012.
2. M.Sc. in Computer Engineering, Omdurman Islamic University 2005.
3. Honors Bachelor in Computer Engineering, Omdurman Islamic University 1999.

Sudanese – Date Born 1974

Assistant Professor - at Almustaqbal University-KSA

Computer Engineering