Smoke Detection System Using Arduino Uno [Report]

KYLE ANTHONY BEDFORD 577599; KAMOGELO PHIRI 577418; JUSTIN STEYN 57830A7;
NKOSINATHI REMORATILE SIBUSISO MOGOROSI 577739;

IOT 2781

10/5/23

Table of Contents

Abstract

The Smoke Detections System project aims to demonstrate the integration of an Arduino Uno with sensors and actuators to create an efficient smoke detection and alarm system. The system makes use of an MQ-2 gas sensor, a piezo buzzer, a red LED, a 16x2 LCD, and a normally open pushbutton. When dangerous levels of smoke are detected, the system triggers the red LED to flash, the buzzer is then activated to sound an alarm and then the systems status is then communicated via the LCD. The push button is used to reset the system as well as silence the alarm. This report presents the circuit design, Arduino sketch and lots of detailed explanations of the system's operations.

Results, observations, and potential enhancements are discussed, highlighting the practical applications and market-readiness of the smoke detection system. This project showcases how effective Arduino-based solutions can be when used in real world scenarios, particularly in terms of safety and security through a smart sensor.

Introduction

Smart technology is becoming more and more progressive, the integration of microcontroller platforms like the Arduino Uno with sensors and actuators has ushered in a new era of innovation. One compelling example of this marriage between technology and practicality is the "Smoke Detection System." This project exemplifies the synergy between hardware and software, as it employs the versatile Arduino Uno microcontroller in conjunction with a range of sensors and actuators to create an efficient and responsive smoke detection and alarm system.

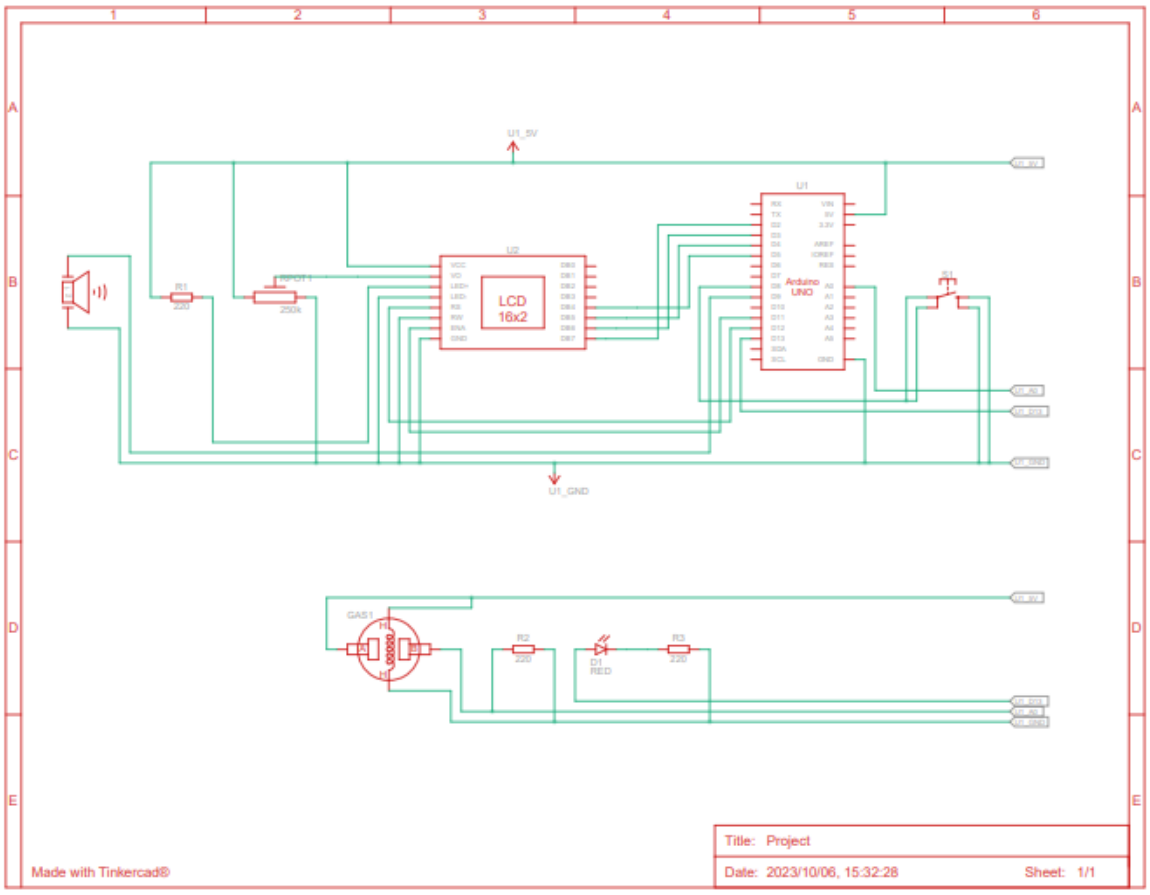
The core components of this system include the MQ-2 gas sensor, a piezo buzzer, a vivid red LED, a 16x2 LCD screen, and a simple yet effective normally open pushbutton. The primary objective of the system is to detect hazardous levels of smoke and respond promptly to mitigate potential risks swiftly and accurately. Upon detecting the presence of smoke, the system orchestrates a sequence of actions: the red LED flashes as a visual alert, the piezo buzzer sounds an audible alarm, and the status of the system is instantly communicated through the LCD display. To ensure user-friendliness and control, a conveniently placed pushbutton allows for system reset and alarm silencing.

In this comprehensive report, we delve into the intricate details of the circuit design, the intricacies of the Arduino sketch, and provide in-depth explanations of the system's operations. Furthermore, we present the findings, observations, and contemplate potential enhancements that can elevate this system's practicality and readiness for various applications. This project serves as a testament to the potency of Arduino-based solutions when deployed in real-world scenarios, particularly in the realm of safety and security through the seamless integration of intelligent sensors.

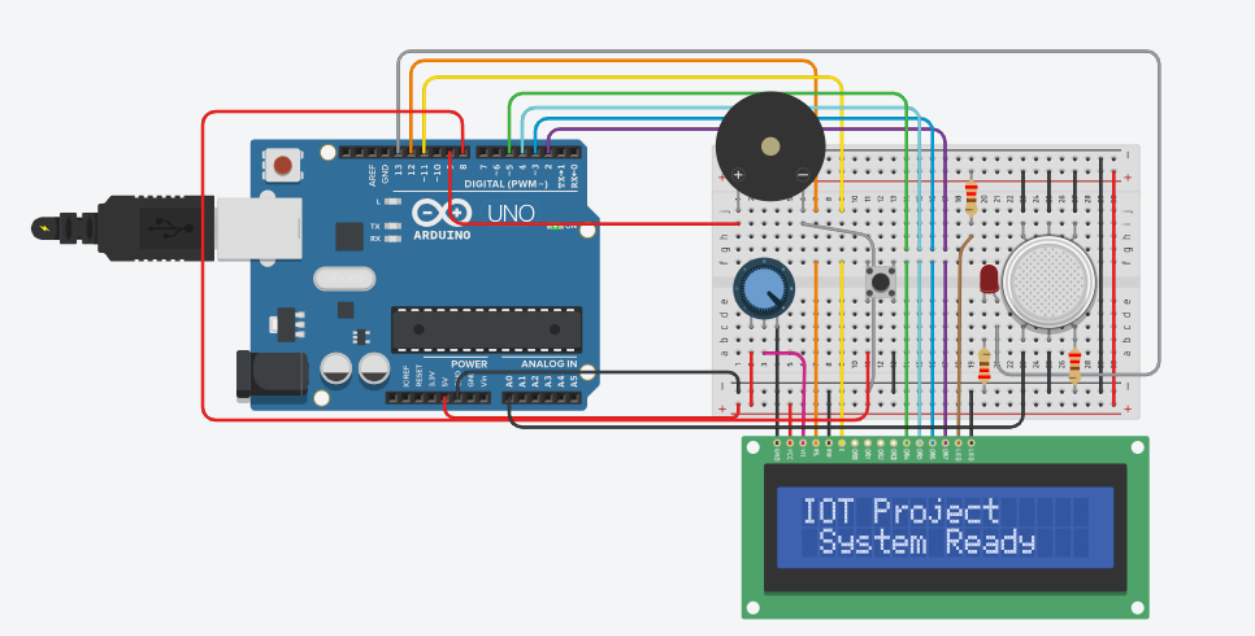
We are going to unveil the inner workings of the Smoke Detection System, illustrating the power of Arduino Uno as a cornerstone for innovative and practical solutions.

Lists of Figures and Tables

Schematic Drawing:



Circuit:



Arduino Sketch (Code)

```
// C++ code
```

```
//
/*

  The circuit:
  * LCD RS pin to digital pin 12
  * LCD Enable pin to digital pin 11
  * LCD D4 pin to digital pin 5
  * LCD D5 pin to digital pin 4
  * LCD D6 pin to digital pin 3
  * LCD D7 pin to digital pin 2
  * LCD R/W pin to ground
  * LCD VSS pin to ground
  * LCD VCC pin to 5V
  * 10K resistor:
  * ends to +5V and ground
  * wiper to LCD VO pin (pin 3)

*/

#include <LiquidCrystal.h>
int seconds = 0;

LiquidCrystal lcd_1(12, 11, 5, 4, 3, 2);
const int gas_input = A0;
int gas = 0;
const int RedLed = 13;
const int buzzer = 9;
const int buttonpin = 8;
int buttonState = 0;
bool alarmSilenced = false;
```

```
void setup()
{
  lcd_1.begin(16, 2); // Set up the number of columns and rows on the LCD.
  pinMode(RedLed, OUTPUT);
  pinMode(buzzer, OUTPUT);
  pinMode(buttonpin, INPUT_PULLUP);



  lcd_1.print("IOT Project");


  Serial.begin(9600);


  // Print a message to the LCD.
  lcd_1.setCursor(1, 1);
  lcd_1.print("System Ready");
  delay(3000);
}

void loop()
{
  Serial.begin(9600);
  //Serial.println(gas);
  // set the cursor to column 0, line 1
  // (note: line 1 is the second row, since counting
  // begins with 0):
  lcd_1.setCursor(0, 1);
  // print the gas values:
  lcd_1.print("Gas Value: ");


  lcd_1.print(gas);
```

```
 delay(1000); // Wait for 1000 millisecond(s)


 gas = analogRead(gas_input);
  Serial.println(gas);


  buttonState = digitalRead(buttonpin);




  if (buttonState == LOW) {
     alarmSilenced = true;
     noTone(buzzer);
     digitalWrite(RedLed, LOW);
     lcd_1.clear();
     lcd_1.print("System Silenced");
     delay(1000);
  }
  else if
(gas > 30 && !alarmSilenced) {
     int led_out = map(gas, 80, 400, 0, 255);
     digitalWrite(RedLed, HIGH);
     tone(buzzer, led_out, 100);
     delay(1000);
     tone(buzzer, led_out, 9000);
     delay(250);
     lcd_1.clear();
     lcd_1.print("Alert: Smoke!!!");
    lcd_1.setCursor(0, 1);
    lcd_1.print("Gas Value: ");
           lcd_1.print(gas);
    delay(2000);
```

```
  lcd_1.clear();
  lcd_1.print("Button to mute");
 }
```

Results and Observations

The MQ-2 gas sensor reliably detected the presence of smoke and hazardous gas levels.

Upon detection, the red LED flashed as a visual alert, providing a clear indication of the system's response.

Simultaneously, the piezo buzzer sounded an audible alarm, ensuring that the alert is noticed even in noisy environments.

The 16x2 LCD displayed real-time information about the system's status.

When smoke was detected, the LCD showed a warning message, providing immediate feedback to the user.

The display also featured instructions for using the pushbutton to reset the system and silence the alarm.

It is very user friendly which makes it more likely to use in real-world scenarios.

Pushing the pushbutton turned the alarm off and resent the alarm to its normal state as intended.

The smoke detection system is very versatile as well is it is market ready.

You can use it in homes, industrial environments, offices and integrate with larger smart building systems.

Its ease of use allows It to be valuable in enhancing safety and security.

The project leaves room for scalability and potential enhancements, such as connectivity to external systems or cloud-based monitoring.

Additional features like remote notifications or integration with home automation platforms can be explored.

Conclusion

The Smoke detection system project has illuminated the potential and versatility of microcontroller-based solutions in addressing real world problems, The Arduino – uno can be used in safety and security as we were able to build a functioning smoke detection system that makes use of sensors and actuators.

We were able to see that all the components worked together seamlessly and that the end goal was achieved in making a working smoke detection system, the functionalities include a working LED that provides a clear visual alert, the piezo buzzer emits an alarm that is easily heard, and the LCD display communicates the status of the system, everything can easily be reset via the pushbutton.

The results and observations gathered during the testing phase underscore the system's reliability and effectiveness. Its potential applications extend far beyond the scope of this project, ranging from enhancing home safety to safeguarding industrial environments and even integrating with larger smart home and building automation systems. This versatility positions the Smoke Detection System as a market-ready solution with the capacity to enhance safety and security in diverse settings.

References