A Report on

Fire Detection and Alert System

Submitted as a partial fulfillment of
the requirement for the Mini Project 1B Course of
Semester IV, AY 2023-2024

Submitted by

Jaden Fernandes	21
Ayush Gajbhiye	25
Soham Ghadigaonkar	29
Om Hindalekar	32

Supervised by
Ms. Freda Carvalho
Mr. Ankur Ganorkar

Department of Electronics & Telecommunication Engineering

The Bombay Salesian Society's

Don Bosco Institute of Technology, Mumbai-400 070
University of Mumbai

2023-24

The Bombay Salesian Society’s

Don Bosco Institute of Technology

(Affiliated to the University of Mumbai)
Premier Automobiles Road, Kurla, Mumbai - 400070

Certificate

This is to certify that the Mini project titled **Fire Detection and Alert System** is a work of

Jaden Fernandes	21
Ayush Gajbhiye	25
Soham Ghadigaonkar	29
Om Hindalekar	32

submitted as partial fulfilment of the requirement for the Mini Project 1B of **‘Semester IV”** in **‘Second Year of Engineering AY 2023-2024”**.

(**Ms. Freda Carvalho**)	(**Mr. Ankur Ganorkar**)	(**Ms. Namita Agarwal**)
Project Guide	Project Guide	HOD, EXTC

The Bombay Salesian Society's
Don Bosco Institute of Technology
(Affiliated to the University of Mumbai)
Premier Automobiles Road, Kurla, Mumbai - 400070

Mini Project 1B Report Approval

This Mini project report entitled **Fire Detection and Alert System** by **Jaden Fernandes, Ayush Gajbhiye, Soham Ghadigaonkar, Om Hindalekar** is approved for the completion of Mini Project 1B course **of Sem IV of AY 2023-2024 in Department of Electronics & Telecommunication Engineering.**

Examiners

1. ______________

2. ______________

Date : / /

Place : **Kurla, Mumbai**

Contents

Abstract

The ”Fire Detection and Alert System” is an innovative project designed to detect and alert the presence of fire in its early stages to ensure prompt action and mitigate potential hazards. The system integrates various components, including a DHT sensor for monitoring temperature and humidity, an IR flame sensor for fire detection, a smoke sensor for detecting smoke particles in the air, and a GSM module for real-time SMS alerting.

Chapter 1

Introduction

1.1 Project Motivation

Enhance Early Detection: The system is designed to detect fire incidents in their early stages, allowing for prompt action before the fire spreads and causes extensive damage.

Improve Alert Mechanisms: Real-time alerts are sent via SMS using the GSM module to predefined contacts, ensuring that occupants and emergency services are notified immediately.

Increase Accessibility: The use of affordable and readily available components makes the system accessible to a wider range of users, including households, small businesses, and community facilities.

1.2 Applications

Residential Buildings: Enhance fire safety in homes, apartments, and condominiums by providing early detection and immediate alerts to occupants and emergency services.

Commercial Establishments: Improve fire safety measures in offices, shops, warehouses, and industrial facilities to protect employees, customers, and valuable assets from potential fire hazards.

Educational Institutions: Safeguard schools, colleges, and universities by installing the fire detection and alert system in classrooms, laboratories, and other facilities to ensure quick evacuation and response during fire incidents.

Community Centers and Public Buildings: Implement the system in community centers, libraries, theaters, and other public buildings to protect the public and facilitate timely evacuation and response in case of fire emergencies.

1.3 Overview of Fire Incidents in Mumbai

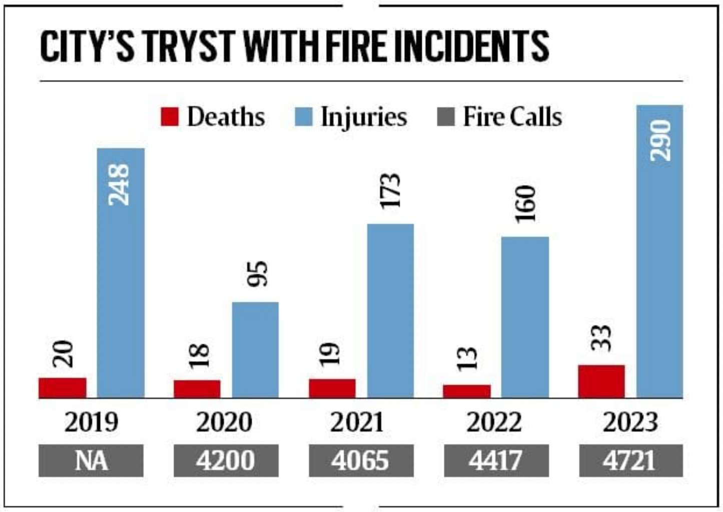


Figure 1.1: Fire Incidents in Mumbai Statistics

The statistical report shown in Figure 1.1. highlights the critical need for fire alert systems in urban areas of Mumbai. Over five years, it reveals a concerning frequency of fire incidents, resulting in deaths, injuries, and numerous fire calls. These statistics emphasize the urgent necessity of early detection and alert systems to mitigate the severity of fire emergencies. Without such systems in place, lives are at risk, and property damage is inevitable. Implementing fire alert systems is imperative to ensure prompt response, minimize casualties, and protect urban communities from the devastating impact of fires.

Chapter 2

Basic Biological Concepts

2.1 Project Objective and Outcomes

2.1.1 Project Objective

To design, develop, and implement a cost-effective and reliable fire detection and alert system using Arduino Uno, integrated with various sensors and a GSM module, to provide early detection of fire incidents, real-time alerts to occupants and emergency services, and precise location tracking, thereby enhancing fire safety measures in residential and commercial environments.

2.1.2 Project Outcomes

Effective Fire Detection: A fully functional fire detection system capable of accurately sensing and differentiating fire incidents from false alarms, ensuring prompt and reliable detection.

Immediate Alert System: A robust alerting mechanism that sends real-time SMS notifications to predefined contacts upon detecting a fire, enabling quick responses from occupants and emergency services.

Chapter 3

Project Implementation

3.1 Block Diagram

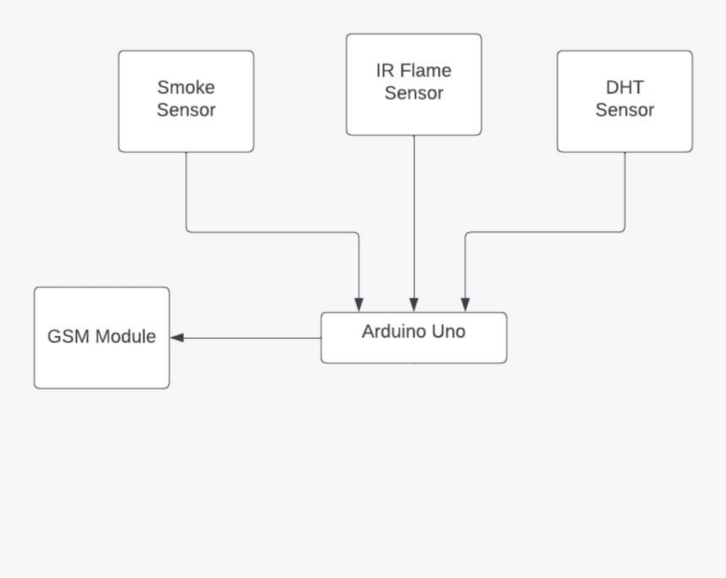


Figure 3.1: Block Diagram

Refering to Figure 3.1 This block diagram illustrates the design of the ”Fire Detection and Alert System,” A system focused on early fire detection and alert notification. The primary sensors connected to the Arduino include an IR Flame Sensor which detects the presence of flames by identifying the infrared (IR) radiation typically emitted by a fire. A Smoke Sensor that monitors the environment for the presence of smoke and gas. A DHT Sensor which measures temperature and humidity, providing crucial data that can indicate the likelihood of a fire hazard.

When any of these sensors detect fire indicators such as flames, smoke, or an unusual increase in temperature, the Arduino processes this information and triggers an alert mechanism. The alert is executed by a GSM module, which is configured to send an SMS notification to predefined phone numbers, providing immediate information about the potential fire situation. [1]

3.2 Circuit Diagram

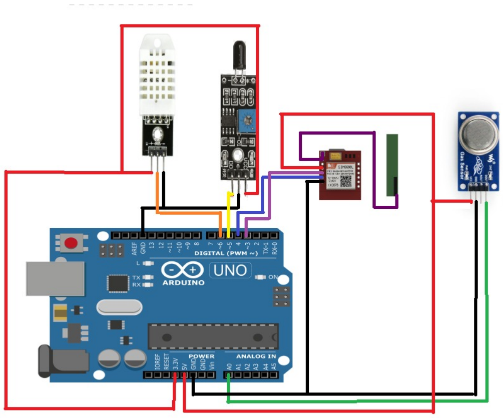


Figure 3.2: Circuit Diagram

The circuit diagram shown in Figure 3.2. depicting the detailed electrical connections and configuration of the System. At the heart of the system is the Arduino Uno, which serves as the central controller, interfacing with the IR flame sensor, smoke sensor, and DHT sensor to monitor fire-related indicators such as flames, smoke, and ambient temperature changes.

These sensors are connected to specific digital I/O pins on the Arduino, ensuring that signals are accurately captured and processed. The GSM module, connected via a serial interface to the Arduino, is configured to send SMS alerts when any of the sensors detect fire or smoke. [2]

3.3 List of Components

Table 3.1: List of Components

Component	Specification	Quantity	Price in Rs
IR Flame Sensor	-	04	42*4
Temperature Sensor	DHT11	01	50
GSM Module	SIM 800L	01	245
Smoke Sensor	MQ-2	01	80
Total			543

Chapter 4

Conclusion

The "Fire Detection and Alert System" project successfully addresses the critical need for a cost-effective, reliable, and efficient fire detection and alert system. By integrating the Arduino Uno microcontroller with DHT, IR flame, smoke sensors, and a GSM module, the system provides early detection of fire incidents, real-time SMS alerts to predefined contacts.

The project's objectives were met through the design and development of a comprehensive fire safety solution that is accessible, user-friendly, and capable of enhancing fire safety measures in various environments, including residential buildings, commercial establishments, educational institutions, healthcare facilities, and remote locations.

Bibliography

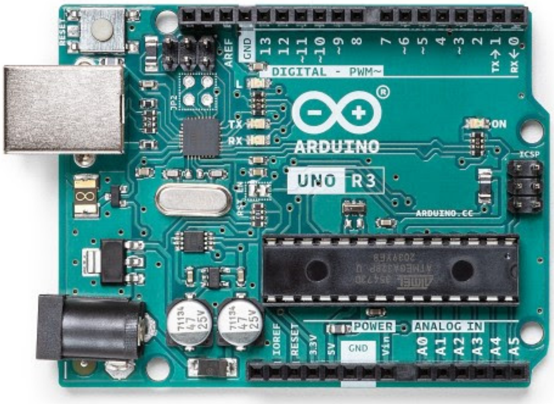
[1] G. Healey, D. Slater, T. Lin, B. Drda, and A. D. Goedeke, "A system for real-time fire detection," in *Proceedings of IEEE conference on computer vision and pattern recognition.* IEEE, 1993, pp. 605–606.

[2] S. Gupta, A. Mudgil, P. Bhardwaj, and M. Gupta, "Design and development of automatic fire alert system," in *2016 8th International Conference on Computational Intelligence and Communication Networks (CICN).* IEEE, 2016, pp. 632–636.

Appendix A

Datasheets

Product Reference Manual
SKU: A000066



Description

The Arduino UNO R3 is the perfect board to get familiar with electronics and coding. This versatile development board is equipped with the well-known ATmega328P and the ATMega 16U2 Processor.
This board will give you a great first experience within the world of Arduino.

Target areas:

Maker, introduction, industries

Features

- **ATMega328P** Processor
  - **Memory**
    - AVR CPU at up to 16 MHz
    - 32KB Flash
    - 2KB SRAM
    - 1KB EEPROM
  - **Security**
    - Power On Reset (POR)
    - Brown Out Detection (BOD)
  - **Peripherals**
    - 2x 8-bit Timer/Counter with a dedicated period register and compare channels
    - 1x 16-bit Timer/Counter with a dedicated period register, input capture and compare channels
    - 1x USART with fractional baud rate generator and start-of-frame detection
    - 1x controller/peripheral Serial Peripheral Interface (SPI)
    - 1x Dual mode controller/peripheral I2C
    - 1x Analog Comparator (AC) with a scalable reference input
    - Watchdog Timer with separate on-chip oscillator
    - Six PWM channels
    - Interrupt and wake-up on pin change
  - **ATMega16U2 Processor**
    - 8-bit AVR® RISC-based microcontroller
  - **Memory**
    - 16 KB ISP Flash
    - 512B EEPROM
    - 512B SRAM
    - debugWIRE interface for on-chip debugging and programming
  - **Power**
    - 2.7-5.5 volts

CONTENTS

1 The Board

1.1 Application Examples

The UNO board is the flagship product of Arduino. Regardless if you are new to the world of electronics or will use the UNO as a tool for education purposes or industry-related tasks, the UNO is likely to meet your needs.

First entry to electronics: If this is your first project within coding and electronics, get started with our most used and documented board; Arduino UNO. It is equipped with the well-known ATmega328P processor, 14 digital input/output pins, 6 analog inputs, USB connections, ICSP header and reset button. This board includes everything you will need for a great first experience with Arduino.

Industry-standard development board: Using the Arduino UNO R3 board in industries, there are a range of companies using the UNO board as the brain for their PLC's.

Education purposes: Although the UNO R3 board has been with us for about ten years, it is still widely used for various education purposes and scientific projects. The board's high standard and top quality performance makes it a great resource to capture real time from sensors and to trigger complex laboratory equipment to mention a few examples.

1.2 Related Products

- Starter Kit
- Arduino UNO R4 Minima
- Arduino UNO R4 WiFi
- Tinkerkit Braccio Robot

2 Ratings

2.1 Recommended Operating Conditions

Symbol	Description	Min	Max
	Conservative thermal limits for the whole board:	-40 °C (-40°F)	85 °C (185°F)

NOTE: In extreme temperatures, EEPROM, voltage regulator, and the crystal oscillator, might not work as expected.

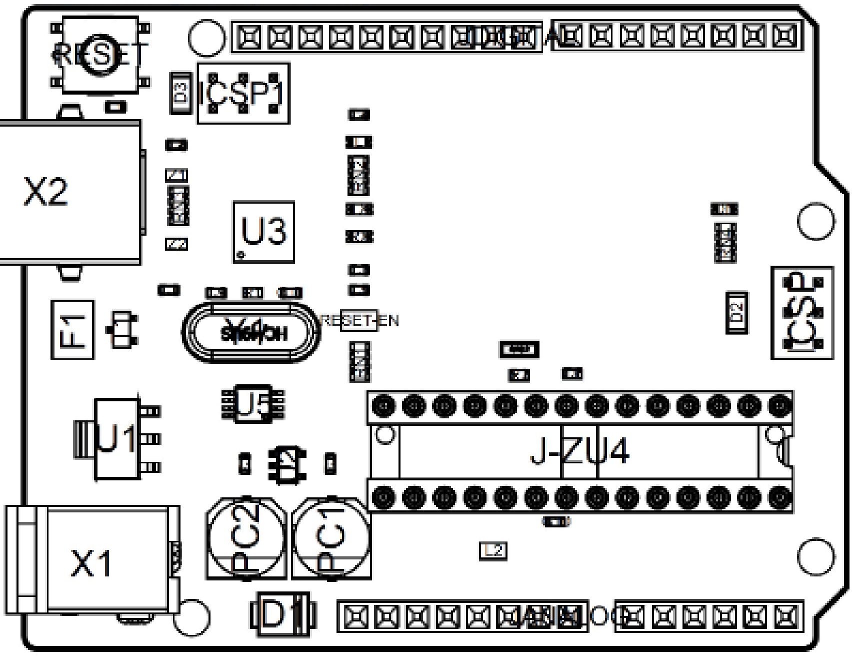
2.2 Power Consumption

Symbol	Description	Min	Typ	Max	Unit
VINMax	Maximum input voltage from VIN pad	6	-	20	V
VUSBMax	Maximum input voltage from USB connector		-	5.5	V
PMax	Maximum Power Consumption	-	-	xx	mA

3 Functional Overview

3.1 Board Topology

Top view



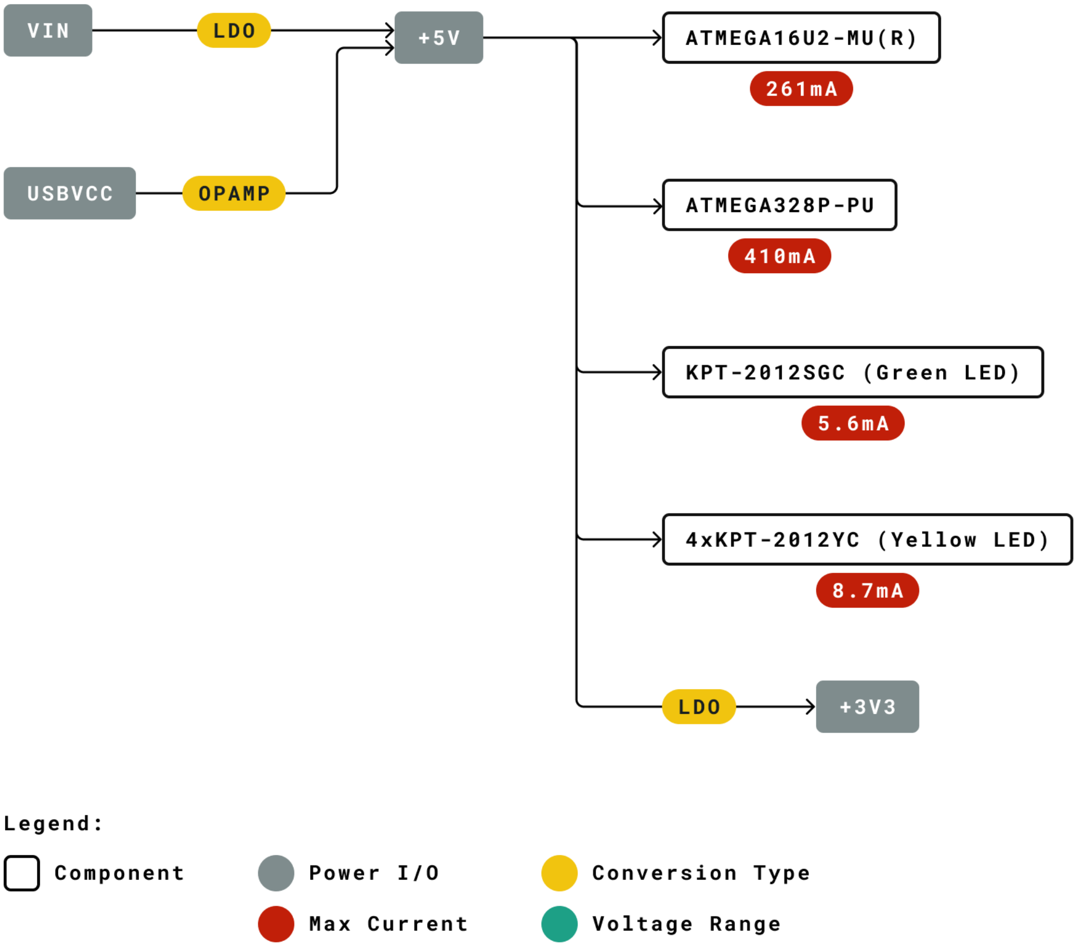
Board topology

Ref.	Description	Ref.	Description
X1	Power jack 2.1x5.5mm	U1	SPX1117M3-L-5 Regulator
X2	USB B Connector	U3	ATMEGA16U2 Module
PC1	EEE-1EA470WP 25V SMD Capacitor	U5	LMV358LIST-A.9 IC
PC2	EEE-1EA470WP 25V SMD Capacitor	F1	Chip Capacitor, High Density
D1	CGRA4007-G Rectifier	ICSP	Pin header connector (through hole 6)
J-ZU4	ATMEGA328P Module	ICSP1	Pin header connector (through hole 6)
Y1	ECS-160-20-4X-DU Oscillator		

3.2 Processor

The Main Processor is a ATmega328P running at up to 20 MHz. Most of its pins are connected to the external headers, however some are reserved for internal communication with the USB Bridge coprocessor.

3.3 Power Tree



Power tree

4 Board Operation

4.1 Getting Started - IDE

If you want to program your Arduino UNO R3 while offline you need to install the Arduino Desktop IDE [1] To connect the Arduino UNO to your computer, you'll need a USB-B cable. This also provides power to the board, as indicated by the LED.

4.2 Getting Started - Arduino Web Editor

All Arduino boards, including this one, work out-of-the-box on the Arduino Web Editor [2], by just installing a simple plugin.

The Arduino Web Editor is hosted online, therefore it will always be up-to-date with the latest features and support for all boards. Follow **[3]** to start coding on the browser and upload your sketches onto your board.

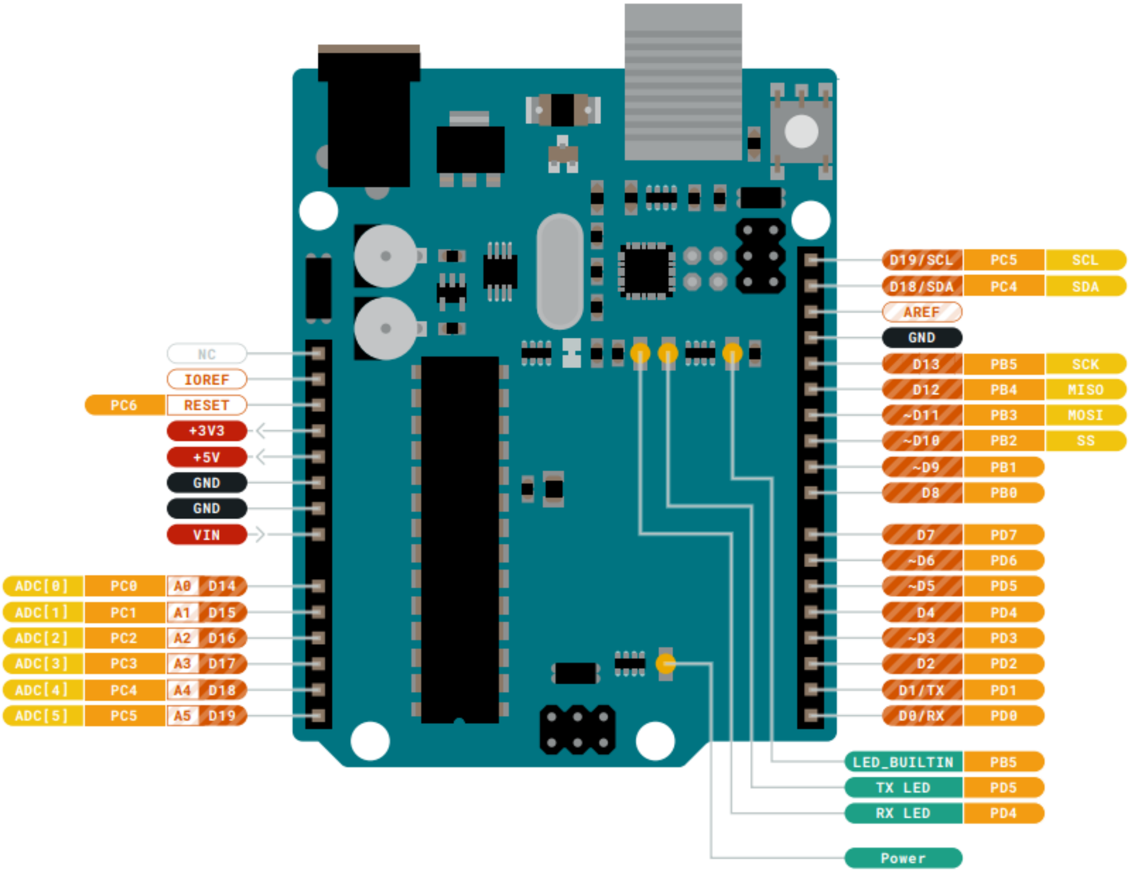
4.3 Sample Sketches

Sample sketches for the Arduino UNO R3 can be found either in the "Examples" menu in the Arduino IDE or in the "Documentation" section of the Arduino website [4]

4.4 Online Resources

Now that you have gone through the basics of what you can do with the board you can explore the endless possibilities it provides by checking exciting projects on Arduino Project Hub **[5]**, the Arduino Library Reference **[6]** and the online Arduino store **[7]** where you will be able to complement your board with sensors, actuators and more.

5 Connector Pinouts



Pinout

5.1 JANALOG

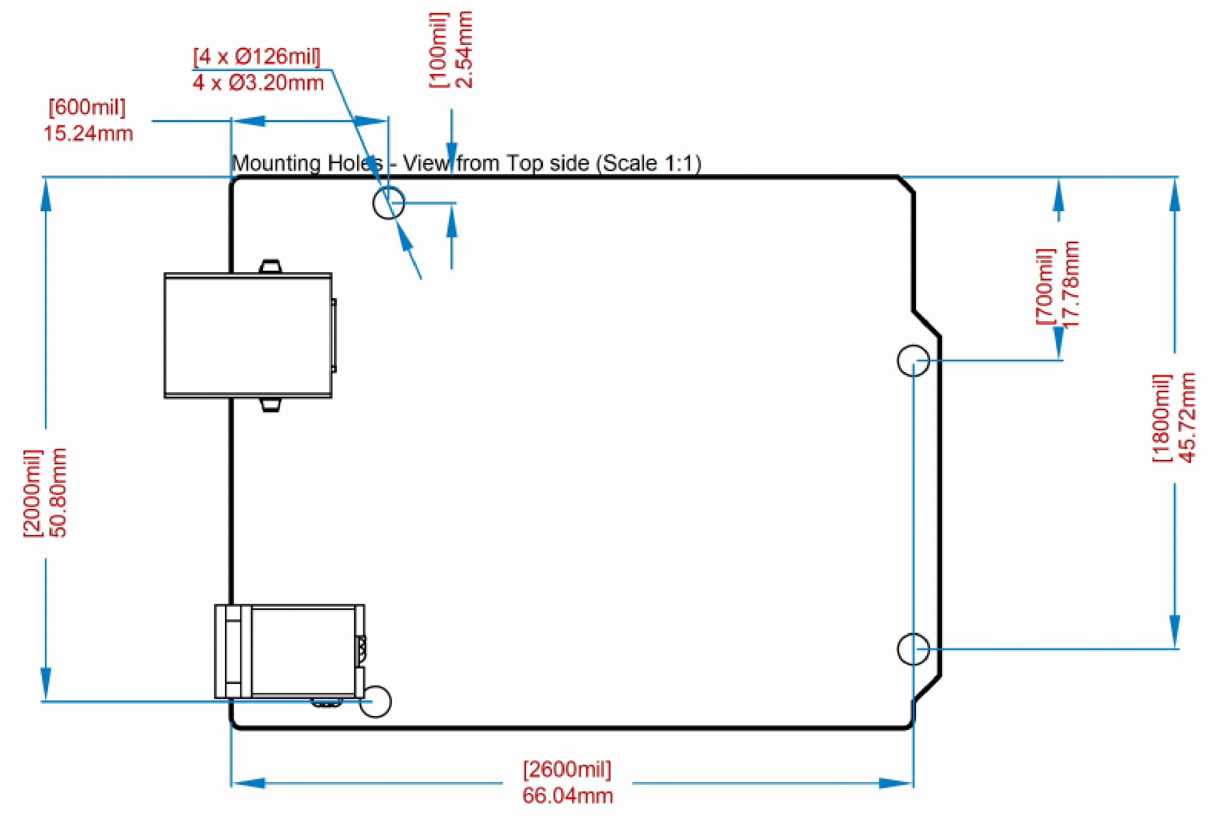
Pin	Function	Type	Description
1	NC	NC	Not connected
2	IOREF	IOREF	Reference for digital logic V - connected to 5V
3	Reset	Reset	Reset
4	+3V3	Power	+3V3 Power Rail
5	+5V	Power	+5V Power Rail
6	GND	Power	Ground
7	GND	Power	Ground
8	VIN	Power	Voltage Input
9	A0	Analog/GPIO	Analog input 0 /GPIO
10	A1	Analog/GPIO	Analog input 1 /GPIO
11	A2	Analog/GPIO	Analog input 2 /GPIO
12	A3	Analog/GPIO	Analog input 3 /GPIO
13	A4/SDA	Analog input/I2C	Analog input 4/I2C Data line
14	A5/SCL	Analog input/I2C	Analog input 5/I2C Clock line

5.2 JDIGITAL

Pin	Function	Type	Description
1	D0	Digital/GPIO	Digital pin 0/GPIO
2	D1	Digital/GPIO	Digital pin 1/GPIO
3	D2	Digital/GPIO	Digital pin 2/GPIO
4	D3	Digital/GPIO	Digital pin 3/GPIO
5	D4	Digital/GPIO	Digital pin 4/GPIO
6	D5	Digital/GPIO	Digital pin 5/GPIO
7	D6	Digital/GPIO	Digital pin 6/GPIO
8	D7	Digital/GPIO	Digital pin 7/GPIO
9	D8	Digital/GPIO	Digital pin 8/GPIO
10	D9	Digital/GPIO	Digital pin 9/GPIO
11	SS	Digital	SPI Chip Select
12	MOSI	Digital	SPI1 Main Out Secondary In
13	MISO	Digital	SPI Main In Secondary Out
14	SCK	Digital	SPI serial clock output
15	GND	Power	Ground
16	AREF	Digital	Analog reference voltage
17	A4/SD4	Digital	Analog input 4/I2C Data line (duplicated)
18	A5/SD5	Digital	Analog input 5/I2C Clock line (duplicated)

5.3 Mechanical Information

5.4 Board Outline & Mounting Holes



Board outline

7. Manufacturing

7.1. Top and Bottom View of SIM800L

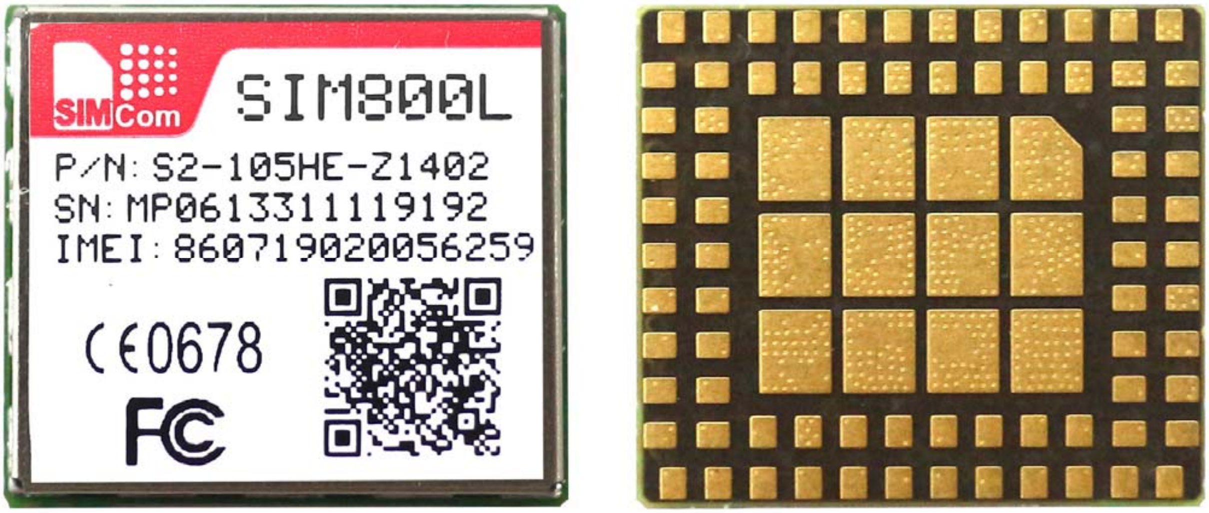
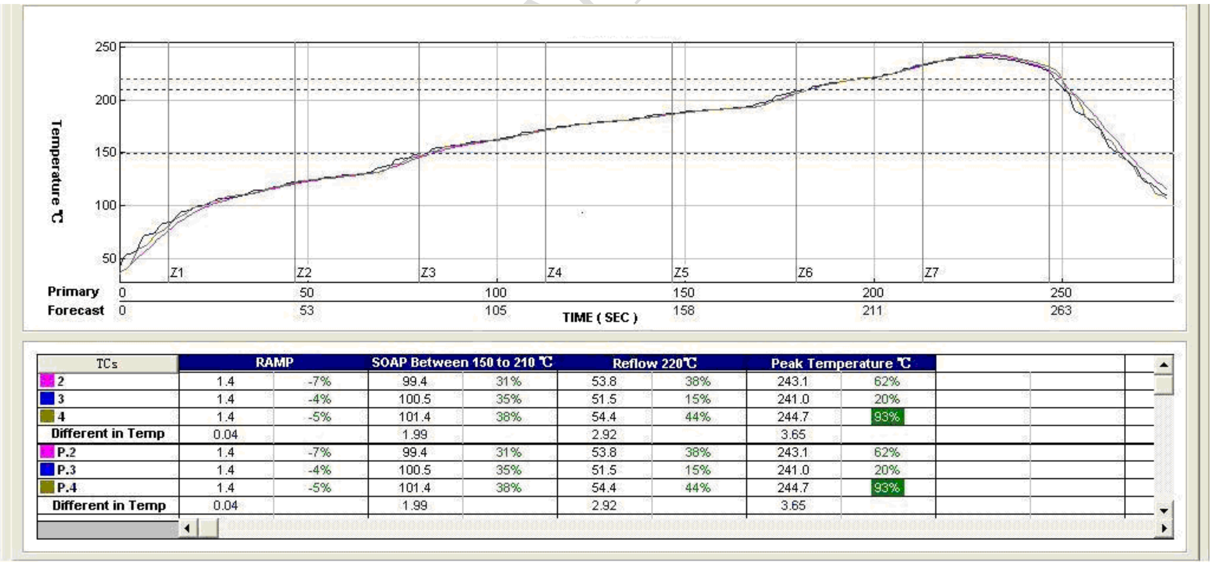


Figure 53: Top and bottom view of SIM800L

7.2. Typical Solder Reflow Profile



TCs	RAMP		SOAP Between 150 to 210 ℃		Reflow 220℃		Peak Temperature ℃	
🟪 2	1.4	-7%	99.4	31%	53.8	38%	243.1	62%
🟦 3	1.4	-4%	100.5	35%	51.5	15%	241.0	20%
🟫 4	1.4	-5%	101.4	38%	54.4	44%	244.7	93%
Different in Temp	0.04		1.99		2.92		3.65	
🟪 P.2	1.4	-7%	99.4	31%	53.8	38%	243.1	62%
🟦 P.3	1.4	-4%	100.5	35%	51.5	15%	241.0	20%
🟫 P.4	1.4	-5%	101.4	38%	54.4	44%	244.7	93%
Different in Temp	0.04		1.99		2.92		3.65	

Figure 54: Typical solder reflow profile

7.3. The Moisture Sensitivity Level

The moisture sensitivity level of SIM800L module is 3. The modules should be mounted within 168 hours after unpacking in the environmental conditions of temperature <30℃ and relative humidity of <60% (RH). It is necessary to bake the module if the above conditions are not met:

Table 54: Moisture sensitivity level and floor life

Moisture Sensitivity Level (MSL)	Floor Life (out of bag) at factory ambient≤30° C/60% RH or as stated
1	Unlimited at ≦30℃/85% RH
2	1 year
2a	4 weeks
3	168 hours
4	72 hours
5	48 hours
5a	24 hours
6	Mandatory bake before use. After bake, it must be reflowed within the time limit specified on the label.

NOTES:

For product handling, storage, processing, IPC / JEDEC J-STD-033 must be followed.

7.4. Baking Requirements

SIM800L modules are vacuum packaged, and guaranteed for 6 months storage without opening or leakage under the following conditions: the environment temperature is lower than 40℃, and the air humidity is less than 90%.

If the condition meets one of the following ones shown below, the modules should be baked sufficiently before re-flow soldering, and the baking condition is shown in below table; otherwise the module will be at the risk of permanent damage during re-flow soldering.

- If the vacuum package is broken or leakage;
- If the vacuum package is opened after 6 months since it’s been packed;
- If the vacuum package is opened within 6 months but out of its Floor Life at factory ambient≦30℃ /60%RH or as stated.

Table 55: Baking requirements

Baking temperature	Moisture	Time
40℃±5℃	<5%	192 hours
120℃±5℃	<5%	6 hours

NOTES：

Care should be taken if that plastic tray is not heat-resistant, the modules should be taken out for preheating, otherwise the tray may be damaged by high-temperature heating.

8. Appendix

I. Related Documents

Table 56: Related documents

SN	Document name	Remark
[1]	SIM800 Series_AT Command Manual	
[2]	ITU-T Draft new recommendation V.25ter：	Serial asynchronous automatic dialing and control
[3]	GSM 07.07：	Digital cellular telecommunications (Phase 2+); AT command set for GSM Mobile Equipment (ME)
[4]	GSM 07.10：	Support GSM 07.10 multiplexing protocol
[5]	GSM 07.05：	Digital cellular telecommunications (Phase 2+); Use of Data Terminal Equipment – Data Circuit terminating Equipment (DTE – DCE) interface for Short Message Service (SMS) and Cell Broadcast Service (CBS)
[6]	GSM 11.14：	Digital cellular telecommunications system (Phase 2+); Specification of the SIM Application Toolkit for the Subscriber Identity Module – Mobile Equipment (SIM – ME) interface
[7]	GSM 11.11：	Digital cellular telecommunications system (Phase 2+); Specification of the Subscriber Identity Module – Mobile Equipment (SIM – ME) interface
[8]	GSM 03.38：	Digital cellular telecommunications system (Phase 2+); Alphabets and language-specific information
[9]	GSM 11.10	Digital cellular telecommunications system (Phase 2)； Mobile Station (MS) conformance specification； Part 1： Conformance specification
[10]	AN_Serial Port	AN_Serial Port

II. Multiplexing Function

Table 57: Multiplexing function

Pin name	Pin number	Mode 0(default)	Mode 1	Mode 2	Mode 3
STATUS	4	STATUS	GPIO4		
SIMPRE	54	SIMPRE	GPIO5	EINT7	
COL0	20	COL0	GPIO6	-	-
COL1	25	COL1	GPIO7	-	-
COL2	22	COL2	GPIO8	-	-
COL3	21	COL3	GPIO9	-	-
COL4	24	COL4	GPIO10	EINT1	-
ROW0	62	ROW0	GPIO11	-	-
ROW1	60	ROW1	GPIO12	-	-
ROW2	61	ROW2	GPIO13	-	-
ROW3	23	ROW3	GPIO14	-	-
ROW4	63	ROW4	GPIO15	EINT3	-
PWM	26	PWM	GPIO16	EINT0	-
NETLIGHT	64	NETLIGHT	GPIO17	-	-
PCMCLK	29	PCMCLK	DISP_RSTB	MC3CM0	GPIO18
PCMOUT	30	PCMOUT	DISP_DI	MC3DA3	GPIO19
PCMSYNC	65	PCMSYNC	DISP_CEB	MC3CK	GPIO20
PCMIN	66	PCMIN	DISP_DA	MC3DA2	GPIO21
GPIO2	27	GPIO2	DISP_A0DA	MC3DA0	-
GPIO3	28	GPIO3	DISP_CLK	MC3DA1	-
UART_DTR	69	UART_DTR	GPIO22	EINT8	-
UART_RI	68	UART_RI	GPIO23	-	-
UART_DCD	70	UART_DCD	GPIO24	-	-
CTS	34	CTS	URXD2(IN)	-	-
RTS	33	RTS	UTXD2(OUT)	-	-
TXD	32	TXD	UTXD1(OUT)	-	-
RXD	31	RXD	URXD1(IN)	EINT4	-
SCL	74	SCL	GPIO25	-	-
SDA	75	SDA	GPIO26	-	-

Note: Multiplexing function need different software supply.

Appendix B

Codes

```
#include <SoftwareSerial.h>
#include <DHT.h>

#define SIM800L_RST 2
#define SIM800L_TXD 4
#define SIM800L_RXD 3

#define FLAME_SENSOR_PIN 5
#define MQ2_SENSOR_PIN 6

#define DHT_PIN 7
#define DHT_TYPE DHT22

SoftwareSerial sim800lSerial(SIM800L_TXD, SIM800L_RXD);
DHT dht(DHT_PIN, DHT_TYPE);

void setup() {
  Serial.begin(9600);
  sim800lSerial.begin(9600);
  pinMode(SIM800L_RST, OUTPUT);
  digitalWrite(SIM800L_RST, HIGH);
  delay(100);
  digitalWrite(SIM800L_RST, LOW);
  delay(1000);
  digitalWrite(SIM800L_RST, HIGH);
  delay(3000);

  pinMode(FLAME_SENSOR_PIN, INPUT);
  pinMode(MQ2_SENSOR_PIN, INPUT);

  dht.begin();

  // Adjusted sendSMS function call to include temperature information
  sendSMS("+91[redacted]", "System Started!", "");
}

void loop() {
  // Read sensor data
  int flameValue = digitalRead(FLAME_SENSOR_PIN);
  int mq2Value = digitalRead(MQ2_SENSOR_PIN);
  float temperature = dht.readTemperature();

  // Check for fire detection
  if (flameValue == HIGH) {
    // Adjusted sendSMS function call to include temperature information
    sendSMS("+91[redacted]", "Fire Detected!", "");
    delay(10000); // Delay to prevent sending multiple SMS in quick succession
  }

  // Check for gas detection
  if (mq2Value == HIGH) {
    // Adjusted sendSMS function call to include temperature information
    sendSMS("+91[redacted]", "Gas Detected!", "");
    delay(10000); // Delay to prevent sending multiple SMS in quick succession
  }

  // Check for temperature rise
  if (temperature > 30.0) { // Adjust threshold as needed
    // Convert temperature to a String object
    String temperatureString = String(temperature) + "°C";
    // Convert temperature String to a C-style string
    const char* temperatureCString = temperatureString.c_str();
    // Send SMS with temperature information
    sendSMS("+91[redacted]", "Temperature rise detected: ", temperatureCString);
    delay(10000); // Delay to prevent sending multiple SMS in quick succession
  }

  // Check for incoming SMS or other commands
  if (sim800lSerial.available()) {
    char c = sim800lSerial.read();
    Serial.write(c);
  }

  // Check for user input
  if (Serial.available()) {
    char c = Serial.read();
    sim800lSerial.write(c);
  }

  delay(1000); // Delay before checking sensors again
}

// Modified to handle temperature information as a separate argument
void sendSMS(const char* phoneNumber, const char* message, const char* temperature) {
  sim800lSerial.println("AT+CMGF=1");
  delay(1000);

  sim800lSerial.print("AT+CMGS=\"");
  sim800lSerial.print(phoneNumber);
  sim800lSerial.println("\"");
  delay(1000);

  // Send the message
  sim800lSerial.print(message);
  sim800lSerial.print(temperature); // Print temperature information
  delay(100);

  sim800lSerial.println((char)26);
  delay(1000);
}
```