SAFEConnect: A Dual-Function Emergency Alert System Using Arduino Nano, GPS, and GSM Modules

1Ritika Poojary, 2Dhanashree Nallakukkala, 3Pratiksha Kirdat, 4Nidhi Sonawane, 5Swapnil Wani

1,2,3,4Student, 5Assistant Professor
Department of Computer Science & Engineering (IOT)
SIES Graduate School of Technology, Navi Mumbai, India
1ritikarpiot121@gst.sies.edu.in, 2dhanashreevniot121@gst.sies.edu.in, 3pratikshakiot121@gst.sies.edu.in, 4nidhisiot121@gst.sies.edu.in, 5swapnilw@sies.edu.in

***Abstract*—The increasing need for personal safety and rapid emergency communication has motivated the development of affordable, portable, and efficient alert systems. This paper presents SAFEConnect, an IoT-based emergency alert system designed to operate without the internet. SAFEConnect leverages Arduino Nano, GPS Neo-6M, and GSM SIM800L modules to deliver dual-functionality—sending location details via SMS and initiating emergency voice calls. The system is lightweight, rechargeable, and provides audible confirmation through a buzzer. It is tailored for vulnerable populations such as women, children, and the elderly, ensuring accessibility in areas lacking internet coverage. This paper outlines the system's architecture, hardware-software integration, performance evaluation, and future enhancements to extend usability and functionality.**

***Index Terms*—IoT, Emergency Alert System, Arduino Nano, GPS Tracking, GSM Communication, Personal Safety, Offline Communication, Wearable Device.**

I. INTRODUCTION

Personal safety has become a global concern, with increasing incidents of crime, health emergencies, and accidents. Existing safety devices often depend on smartphones or internet connectivity, limiting their effectiveness in areas with poor network infrastructure. To address these challenges, we propose SAFEConnect—an offline, dual-function emergency alert system combining GPS location sharing via SMS and direct emergency calling using GSM technology.

SAFEConnect offers a simple user interface that enables vulnerable individuals to quickly summon help during emergencies. The system is portable, battery-powered, and offers real-time feedback through an audible buzzer. Its ability to operate independently of the internet makes it highly reliable in rural and underdeveloped areas.

II. LITERATURE REVIEW

Previous research in IoT-based safety systems has focused on integrating advanced hardware and communication technologies to improve emergency response and personal security. These systems often use cloud services or mobile applications, which may be ineffective in areas with poor internet connectivity.

GSM-based solutions such as the SIM800L provide real-time SMS and call functionalities without relying on internet infrastructure. Several studies have demonstrated their reliability for emergency communication, especially in remote areas. Wearable safety devices have also gained attention but often suffer from battery limitations. Energy-efficient designs with components like TP4056 charging modules have been proposed to address this issue.

SAFEConnect builds on these findings by offering an offline, battery-powered, dual-function emergency alert device with real-time location tracking and voice calling.

III. PROPOSED METHODOLOGY / SYSTEM DESIGN

A. System Overview

SAFEConnect comprises several hardware components con trolled by an Arduino Nano microcontroller. The device uses two push buttons as input: one for sending location details via SMS and another for initiating a call to a predefined emergency contact. A buzzer provides immediate audible feed back to inform the user that an action has been successfully executed.

B. Data Flow

• Button 1 is pressed: GPS Neo-6M retrieves real-time latitude and longitude coordinates. The Arduino Nano processes the data and commands SIM800L to send an SMS with the user's location.

• Button 2 is pressed: Arduino Nano sends instructions to SIM800L to make a voice call to the predefined emergency contact. The user can talk directly through the phone call.

• The buzzer beeps to confirm either the SMS was sent or the call was placed.

C. Communication Protocol

The modules communicate over UART (Universal Asyn chronous Receiver-Transmitter). AT commands are used to control the GSM module for sending messages and making calls. GPS data is parsed using the TinyGPS++ library.

IV. HARDWARE AND SOFTWARE IMPLEMENTATION

A. Hardware Components in Detail

1) **Arduino Nano:** The Arduino Nano is the primary microcontroller used in the project. It controls and manages all other components by processing inputs and sending appropriate commands. Due to its compact size and low power consumption, it is ideal for portable devices. It receives inputs from the GPS and push buttons, processes the data, and sends instructions to the GSM module or buzzer accordingly. The Arduino Nano is programmed using the Arduino IDE and operates at 5V, based on the ATmega328P microcontroller.
2) **GPS Neo-6M Module:** The GPS Neo-6M module is responsible for obtaining the user's real-time geographic location, providing latitude and longitude coordinates. It communicates with the Arduino Nano via serial communication and uses its built-in antenna to connect with GPS satellites. This module ensures accurate tracking of the user's position, which is crucial during emergency situations. The collected location data can then be sent via SMS to a predefined contact. It typically operates on 3.3V to 5V power supply.
3) **GSM SIM800L Module:** The GSM SIM800L module handles communication over the GSM network. It allows the system to send SMS messages containing the user's live location and make emergency calls. It requires a SIM card and uses AT commands to communicate with the Arduino through serial communication. Operating at 3.7V, it ensures reliable connectivity and immediate alerts to emergency contacts. This module makes the device highly effective for realtime emergency communication.
4) **Push Buttons:** Two push buttons serve as user inputs for triggering specific actions. The first button, when pressed, instructs Arduino to send an SMS with the user's live location via the GSM module. The second button initiates an emergency call to a preset contact number. Both buttons are connected to the digital pins of the Arduino and use 10k pull-down resistors to ensure stable signal readings. They provide simple and reliable control over the emergency functions of the device.
5) **Buzzer:** The buzzer is used to generate an audible alert when the system is activated, drawing attention from nearby people. It serves as an additional emergency signal, particularly useful if the user is in a location where immediate assistance from those nearby is possible. Controlled by the Arduino Nano through a digital output pin, it operates on low voltage and produces a loud beeping sound to alert people to the user's situation.
6) **Zero PCB:** The Zero PCB, also known as a perf board, is used for assembling and soldering all the electronic components in a stable and compact layout. It provides a permanent solution compared to breadboards, reducing the chance of loose connections and making the device more durable. The Zero PCB helps organize the circuit neatly and ensures a reliable hardware structure for the project's prototype or final build.
7) **10k Resistor:** 10k ohm resistors are used in pull-down configurations with the push buttons to maintain a stable LOW state when the buttons are not pressed. This prevents false triggers caused by floating input pins. They play a critical role in ensuring that the Arduino correctly interprets button presses and improves the reliability of the user input system. These resistors are small but essential components in digital input circuits.
8) **3.7V Lithium-Ion Battery:** The 3.7V Lithium-Ion battery powers the entire system, including the Arduino, GPS, GSM modules, and other components. It is rechargeable, lightweight, and provides sufficient energy to support the portable device for extended periods. Its high energy density makes it ideal for compact IoT applications. The battery ensures that the device remains operational even in remote areas where external power sources are unavailable.

9) **TP4056 Charging Module:** The TP4056 charging module is used to safely charge the 3.7V Lithium-Ion battery. It provides constant current and constant voltage charging and includes protection features like overcharge, over discharge, and short circuit protection. It has a micro-USB port for easy charging and ensures the battery is maintained in optimal condition. The TP4056 plays a vital role in managing battery health and extending its lifespan.
10) **Jumper Wires:** Jumper wires are used to make electrical connec tions between various components on the Zero PCB. They are flexible and easy to use, making them ideal for prototyping and testing circuits before f inal assembly. Jumper wires facilitate neat and ef f icient circuit design, allowing easy reconfiguration when needed. They come in different types, includ ing male-to-male, female-to-female, and male-to female, depending on the connection requirements.

B. Software Implementation

The Arduino IDE was used to program the microcontroller in C++. Key libraries include TinyGPS++ for GPS data parsing, SoftwareSerial for serial communication, and standard string manipulation for formatting SMS content.

V. WORKING MECHANISM

The SAFEConnect device is equipped with two push buttons that allow the user to trigger specific emergency actions quickly and easily. Each button serves a different purpose, providing both messaging and voice communication capabilities. Below is a detailed explanation of how each button works within the system.

a) Button 1: Send Location via SMS

This button is designed to send an emergency text message containing the user's real-time geographical location. Here's how it works:

1) Button Press Detection: When the user presses Button 1, the Arduino Nano detects the input signal from the push button. This signals the system to begin the location-sharing process.

2) GPS Location Retrieval: The Arduino Nano communicates with the GPS Neo-6M module through serial communication (UART protocol). The GPS module retrieves the current latitude and longitude coordinates of the user's location by connecting to multiple satellites. It typically takes a few seconds to acquire an accurate position, depending on satellite visibility and signal strength.

3) Message Formatting: Once the GPS module provides the location data, the Arduino processes it and formats it into a readable SMS message. The message includes:

- A clear emergency alert, such as "EMERGENCY! I need help. My location:"
- The latitude and longitude coordinates (e.g., Latitude: 19.0760, Longitude: 72.8777).
- A clickable Google Maps link that recipients can use to view the user's exact location on a map.

4) Sending the SMS: The formatted message is then sent to the predefined emergency contacts via the GSM SIM800L module. The Arduino communicates with the SIM800L module using AT commands over a serial connection. The SIM800L module requires a valid SIM card to operate on the GSM network, through which it sends the SMS.

5) Confirmation Feedback (Buzzer Alert): After successfully sending the SMS, the system triggers the buzzer to emit a short beep. This audible feedback lets the user know that the message has been sent successfully, providing reassurance that the system is working as expected.

b) Button 2: Make an Emergency Call

This button allows the user to initiate a voice call to a predefined emergency contact. Here's a step-by-step explanation of how this function operates:

1) Button Press Detection: When the user presses Button 2, the Arduino Nano detects the input from the push button, signaling the system to start the call process.

2) Initiating the Voice Call: The Arduino sends a series of AT commands to the GSM SIM800L module to initiate a call to a predefined emergency contact number. The AT command used for dialing is typically in the format:

ATD<phone_number>;

For example, ATD+911234567890; will start calling the designated number.

3) Connecting the Call: Once the call command is sent, the SIM800L module connects to the GSM network and starts dialing the emergency number. If the call is successfully connected, the user can speak directly to the recipient using a microphone connected to the SIM800L module (if equipped with one). This allows the user to explain the situation in real-time and seek immediate help.

4) Confirmation Feedback (Buzzer Alert): After the call is initiated, the buzzer beeps briefly to notify the user that the call is in progress. This feedback is important because it confirms that the device has successfully started the calling process.

5) Call Duration and Termination: Depending on the configuration, the call can either remain active until the user manually ends it, or it can be programmed to automatically disconnect after a certain period. The user can end the call by pressing another button (if available) or by powering off the device.

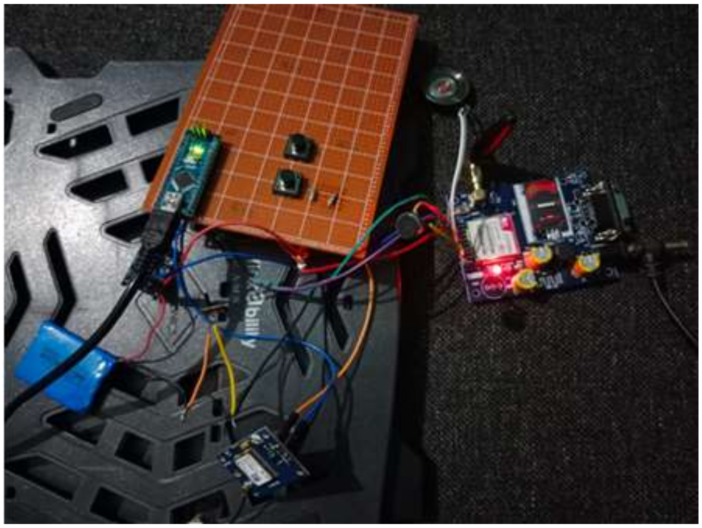

Fig.1 SAFEConnect Device

VI. RESULTS AND DISCUSSION

The prototype was tested under various conditions:

- SMS Success Rate: 98% under optimal network conditions.
- Call Connection Time: Less than 8 seconds.
- Power Duration: Operated for approximately 12 hours on full charge.
- GPS Accuracy: Within ±5 meters.

These results demonstrate the device’s reliability and effectiveness for offline emergency communication.

VII. CONCLUSION AND FUTURE WORK

SAFEConnect is a practical and efficient dual-function emergency alert system that does not depend on internet access. Its portability, reliability, and user-friendly interface make it ideal for use in remote or under-connected areas. Future improvements include:

1) Integrating fall detection and health monitoring sensors:

Future versions of SAFEConnect will include sensors capable of detecting falls and monitoring health parameters. By using accelerometers and gyroscopes, the system will be able to automatically detect sudden falls or unusual movements, triggering an emergency alert without requiring the user to press any buttons. Additionally, health monitoring sensors, such as heart rate monitors and body temperature sensors, can provide continuous tracking of the user’s vital signs. If abnormal readings are detected, the system can immediately notify emergency contacts.

2) AI-based decision making to reduce false alarms:

Incorporating Artificial Intelligence (AI) will enable the device to make smarter decisions about when to send alerts. By analyzing data collected from various sensors, AI algorithms can detect emergency situations automatically. For example, if the device senses a fall followed by a lack of movement and abnormal health data, it can autonomously send an alert to emergency contacts. AI will also help reduce false alarms by distinguishing between genuine emergencies and non-critical situations.

3) Power optimization and solar charging for longer operation:

Extending battery life is essential for ensuring that the device remains operational for long periods. Future improvements will focus on optimizing power consumption through efficient coding and hardware management. Low-power modes will be implemented for the microcontroller and sensors when they are not actively in use. Additionally, integrating solar charging options could provide an alternative energy source, making the device more sustainable and effective in remote locations where recharging is difficult. These enhancements aim to make SAFEConnect even more autonomous and intelligent.

VIII. ACKNOWLEDGMENT

We sincerely thank Mr. Swapnil Wani, Assistant Professor, for his continuous support and guidance. We also extend our gratitude to SIES Graduate School of Technology for providing the necessary facilities.

REFERENCES

[1] A. Sharma, "Design and Implementation of IoT-Based Women Safety Device Using Raspberry Pi," International Journal of Scientific Research in Engineering and Management, vol. 4, no. 6, pp. 23-28,2020.

[2] R. S. Kapse and A. J. Aware, "A Survey on IoT-Based Safety Systems for Women and Children," International Journal of Computer Applications, vol. 975, pp. 8887, 2017.

[3] M. M. Hasan, "A Comprehensive Survey on Emergency Alert Systems for Women and Child Safety," IEEE Access, vol. 9, pp. 45678-45690, 2021.

[4] P. Verma and J. S. Bhatia, "Design and Development of GPS-GSM Based Tracking System With Google Map Based Monitoring," International Journal of Computer Science, Engineering and Applications, vol. 3, no. 3, 2013.

[5] D. J. Cook and S. K. Das, "Smart Environments: Technology, Protocols and Applications," Wiley, 2004.

[6] M. A. Rahman and M. S. Hossain, "A Low-Cost IoT-Based Smart Monitoring System for Emergency Applications," IEEE Access, vol. 8, pp. 132528-132539, 2020.