HEART VOICE IOT GUARDIAN

A project report submitted

in partial fulfillment of requirement for the award of degree

BACHELOR OF TECHNOLOGY

in

ELECTRONICS & COMMUNICATION ENGINEERING

by

R. VARUNTEJA	**2105A41212**
M.ANIL KUMAR	**2105A41199**
J. NAVEEN REDDY	**2105A41197**
T. CHANDU	**2105A41088**

Under the guidance of

Dr.N.Ram Kumar

Assistant Professor, Department of ECE.

Department of Electronics and Communication Engineering

SR UNIVERSITY

Ananthasagar, Warangal.

SR UNIVERSITY

CERTIFICATE

This is to certify that this project entitled **“HEART VOICE IOT GUARDIAN**" is the bonafied work carried out by **R. VARUNTEJA**, **J. NAVEEN REDDY, M. ANIL KUMAR and T. CHANDU** as a Capstone project for the partial fulfillment to award the degree **BACHELOR OF TECHNOLOGY** in **ELECTRONICS & COMMUNICATION ENGINEERING** during the academic year 2024-2025 under our guidance and Supervision.

Dr.N. Ram Kumar
Asst.Professor
SR University,
Ananthasagar, Warangal.

Dr. Sandip Bhattacharya
Prof. & HOD (ECE),
SR University,
Ananthasagar, Warangal.

ACKNOWLEDGEMENT

We wish to take this opportunity to express our sincere gratitude and deep sense of respect to our beloved Vice Chancellor, **Prof. DEEPAK GARG,** for his continuous support and guidance to complete this project in the institute.

We would like to extend our gratitude to **Dr. LMI Leo Joseph**, **Assoc. Professor,** the Dean of the School of Engineering, and **Dr. Sandip Bhattacharya**, **Professor** and Head of the ECE Department, for their encouragement and support.

We owe an enormous debt of gratitude to our project guide **N. Ram Kumar, Asst.Prof** as well as project coordinators **Dr. LMI Leo Joseph, Assoc. Prof., Dr. J. Ajayan, Prof.**, for guiding us from the beginning through the end of the Major project with their intellectual advices and insightful suggestions. We truly value their consistent feedback on our progress, which was always constructive, and encouraging and ultimately drove us in the right direction.

Finally, we express our thanks to all the teaching and non-teaching staff of the department for their suggestions and timely support.

ABSTRACT

The project titled “**Heart Voice IOT Guardian”** is an Internet of Things (IoT)-based cardiac health monitoring and emergency alerting system that enables continuous, real-time monitoring and prompt emergency responses. To monitor position and heart health, the system combines an Arduino Uno, GPS module, GSM 900 module, and HW827 heart rate sensor. The Arduino analyzes the heart rate data continuously collected by the HW827 sensor to look for any irregularities. An emergency alarm is triggered if the heart rate exceeds or falls below safe standards. The GSM module provides information about the cardiac irregularity via an SMS warning sent to pre-configured contacts, such as family members or medical professionals, in reaction to abnormal readings. The GPS module simultaneously sends the user's current location, which is part of the alert, In order to aid in immediate action, the GPS module automatically broadcasts the user's current location, which is included in the warning.

Finally, Heart Voice IoT Guardian is a cost-effective and scalable IoT-based system that improves safety and responsiveness for those at cardiovascular risk by guaranteeing continuous heart health monitoring and prompt emergency assistance.

CONTENTS

1. INTRODUCTION

1.1 OVERVIEW

Deaths from cardiovascular disease (CVD) jumped globally from 12.1 million in 1990 to 20.5 million in 2021, according to a new report from the World Heart Federation (WHF). The highest CVD death rates occur in the Central Europe, Eastern Europe, and Central Asia region. Though the overall number of CVD deaths increased in the last three decades—largely due to an ageing and growing population—the CVD death rate fell globally from 354.5 deaths per 100,000 people in 1990 to 239.9 deaths per 100,000 people in 2019. This decline was however uneven across regions; the fastest decline in death rate was experienced in high-income countries.

All death rates used are “age-standardized,” In recent years, India has witnessed an upsurge of heart attacks among young adults and people in their 30s and 40s. According to a comprehensive medical study published in October 2023, heart attacks were the underlying cause of up to 45% of deaths in the 40-69 year age group. As of today, 1 in 5 heart attack patients are younger than 40 years of age.

Since cardiovascular diseases (CVDs) rank among the world's top causes of death, it is imperative that those who are at risk receive ongoing monitoring and prompt management. Remote health monitoring systems, which offer real-time data and instant alarms that can save lives in emergencies, have become more widely available and efficient with the development of Internet of Things (IoT) technology.

The goal of the Heart Voice IoT Guardian project is to use IoT technology to develop an effective and reasonably priced cardiac health monitoring and alerting system. The goal of the Heart Voice IoT Guardian project is to use IoT technology to develop an effective and reasonably priced cardiac health monitoring and alerting system. The Heart Voice IoT Guardian is made especially for rapid emergency response and ongoing heart rate monitoring. Important parts like an Arduino Uno, a GPS module, a GSM 900 module, and a HW827 heart rate sensor are all included into this system. The system can evaluate a user's cardiac health in real time by recording heart rate data. The device automatically notifies selected contacts, such family members or medical professionals, by SMS whenever the heart rate rises or falls below a predetermined threshold. Responders can find the user more quickly because to the GPS module, which adds the user's location to the alert. For those who are at danger, especially in remote locations, this Internet of Things-based system offers a scalable and affordable method of heart health monitoring.

The Heart Voice IoT Guardian seeks to improve patient safety, enable prompt action, and ultimately raise the standard of living for people with cardiovascular disorders by fusing real-time monitoring with automated notifications.

2. LITERATURE SURVEY

2.1 EXISTING METHODS

Emergency response systems and cardiac health monitoring employ a variety of wearable technology, smartphone apps, and specialized medical equipment. A few popular techniques are as follows:

2.1.1 Remote Monitoring System

These systems are modern medical solutions that continually monitor a patient's vital signs in real-time, including heart rate, electrocardiogram, and other important metrics. In order to enable prompt action in the event of abnormal readings, patients wear specialized monitoring devices that wirelessly send data to healthcare specialists. Because these systems allow proactive treatment and potentially lower emergency hospital visits, they are especially helpful for high-risk patients. They are less practical for daily usage outside of clinical supervision, though, because they are typically costly and require hospital-grade equipment and continuous connectivity.

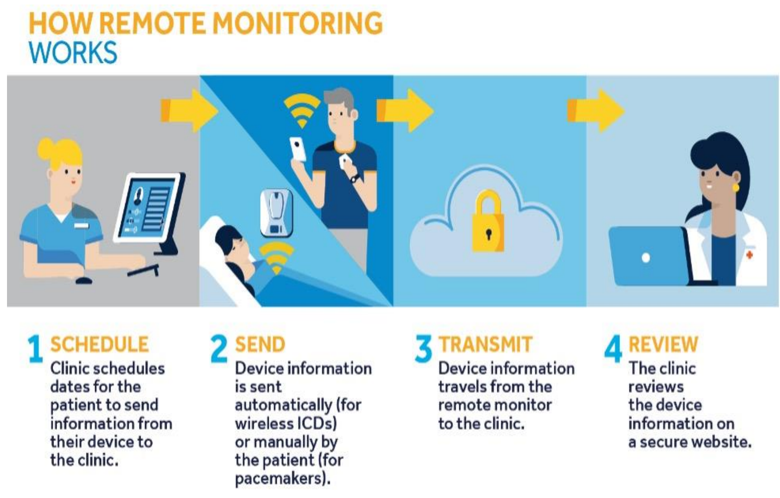


Fig-2.1.1: Remote Monitoring System

2.1.2 Wearable Health Trackers

These gadgets that are worn on the body, like fitness bands and smartwatches, and that track health indicators like heart rate, physical activity, sleep patterns, and occasionally even electrocardiograms. Well-known companies like Fitbit, Garmin, and Apple Watch use optical sensors to give users real-time data. They can also integrate with smartphone apps to track long-term health trends. Despite their convenience and growing accuracy, the majority of wearable trackers have limited emergency response capabilities and usually rely on cellphones to communicate data with healthcare providers or send alarms.

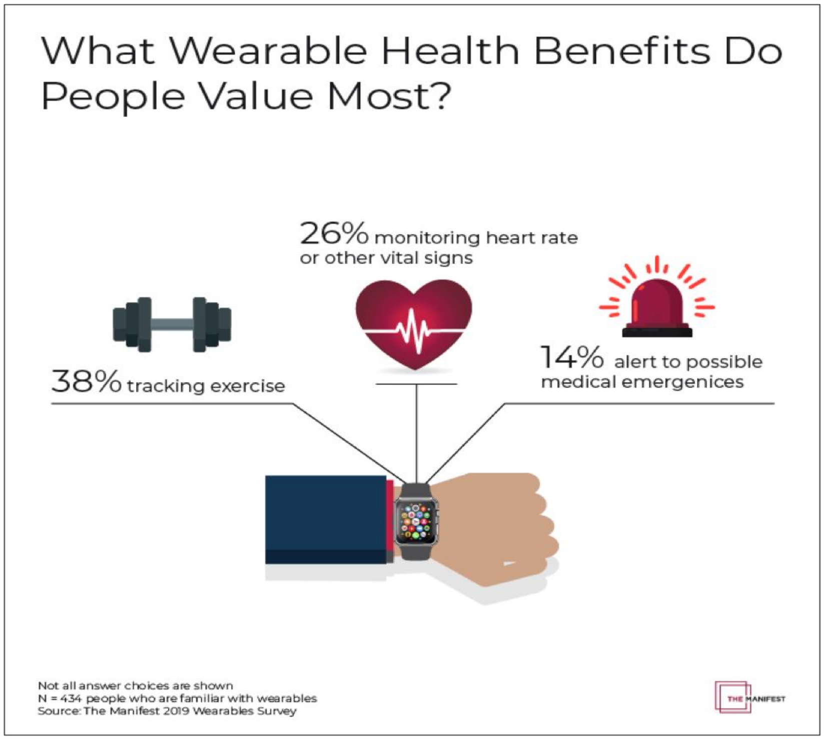


Fig-2.1.2: Wearable Health Trackers

2.1.3 Mobile Health Apps

These are smartphone applications designed to track various health metrics, including heart rate, physical activity, and general wellness. Some apps use the phone's camera and photoplethysmography (PPG) technology to measure heart rate by detecting changes in blood flow, while others integrate with wearable devices for more comprehensive tracking. These apps offer convenience and accessibility, making health monitoring available to a wide audience. However, they often rely on smartphone availability, may lack accuracy compared to dedicated medical devices, and typically do not include automated emergency alert capabilities, limiting their use in critical situations.

Fig-2.1.3: Mobile Health Apps

2.2 MOTIVATION

The necessity for a dependable, real-time ventricular monitoring system that guarantees prompt assistance in the event of a cardiac event is the driving force behind the Heart Voice IoT Guardian project. Prompt action during cardiac emergencies can save lives because cardiovascular disorders are one of the world's leading causes of death. Wearables, smartphone apps, and medical alert systems are examples of current options, but they have drawbacks. For example, they frequently need smartphone connectivity, need manual activation, or lack accurate GPS tracking and automatic notifications. By developing an accessible, stand-alone gadget that automatically checks heart health, identifies anomalies, and notifies specified contacts with GPS coordinates—even in places where smartphones are not widely available—this project seeks to remove these obstacles. Patients and their families benefit from this system's independence and peace of mind, especially in rural or isolated areas where it would be more difficult to get emergency medical help.

SCOPE OF THE WORK

A complete Internet of Things-based cardiac monitoring system that incorporates an Arduino Uno, GSM 900 module, GPS module, LCD display, and heart rate sensor is part of the project's scope. The system is set up to continuously check heart rate, identify anomalous readings, and automatically send SMS alarm messages to pre-configured contacts with the user's location. In order to improve the device's usability, the project also entails creating an intuitive LCD interface that shows the system status and heart rate in real time. With GPS for accurate position monitoring and GSM for dependable communication, the Heart Voice IoT Guardian guarantees a useful, affordable solution that can be used in a variety of settings, including distant locations where access to healthcare may be restricted.

2.3 Problem Statement

The lack of dependable, self-sufficient, and reasonably priced cardiac monitoring systems that can offer prompt aid during cardiac emergencies—especially for individuals with heart conditions—is the issue that the Heart Voice IoT Guardian project aims to solve. One of the main causes of death worldwide is cardiovascular disease, and preventing deaths requires immediate action. Many of the health monitoring systems that are now in use, however, have limited capabilities since they depend on human activation, smartphone connectivity, or external internet access—all of which might not always be available in emergency situations. Particularly in rural or isolated locations with limited healthcare resources, this gap in current solutions results in delays in emergency response and subpar care.

Existing mobile apps and wearable health gadgets that monitor vital signs like heart rate frequently rely on cellphones or internet access to work. These solutions have inherent limitations even though they offer insightful information about health. When a distressing incident occurs, the majority of gadgets need the user to actively activate emergency notifications. During a cardiac emergency, when the patient may be debilitated, this might be very difficult or impossible, delaying assistance and raising the possibility of serious health repercussions. Additionally, a lot of devices don't track precise location, which is necessary for emergency personnel to get to the patient as soon as possible.

In order to solve these issues, the Heart Voice IoT Guardian provides a standalone, self-contained device that can continuously monitor heart rate without requiring a smartphone or internet connection. When the system identifies aberrant heart rate measurements, it immediately sends out an alert to notify healthcare practitioners or specified contacts. In the event of a crisis, the patient's condition will be rapidly transmitted thanks to this automation, which eliminates the need for manual intervention. Additionally, the addition of a GPS module allows for real-time position tracking, which helps emergency personnel find the patient and administer care promptly.

The high expense and complication of the current remote monitoring systems present another difficulty. Although remote cardiac monitoring is available from many healthcare providers, these systems are frequently costly and need certain equipment. Because of this, many people cannot use them, particularly those who live in rural or lower-income areas. The Heart Voice IoT Guardian is intended to be an inexpensive, user-friendly substitute that can be set up and used by anybody with access to basic parts. The system is designed to satisfy the requirements of regular people without requiring costly medical facilities.

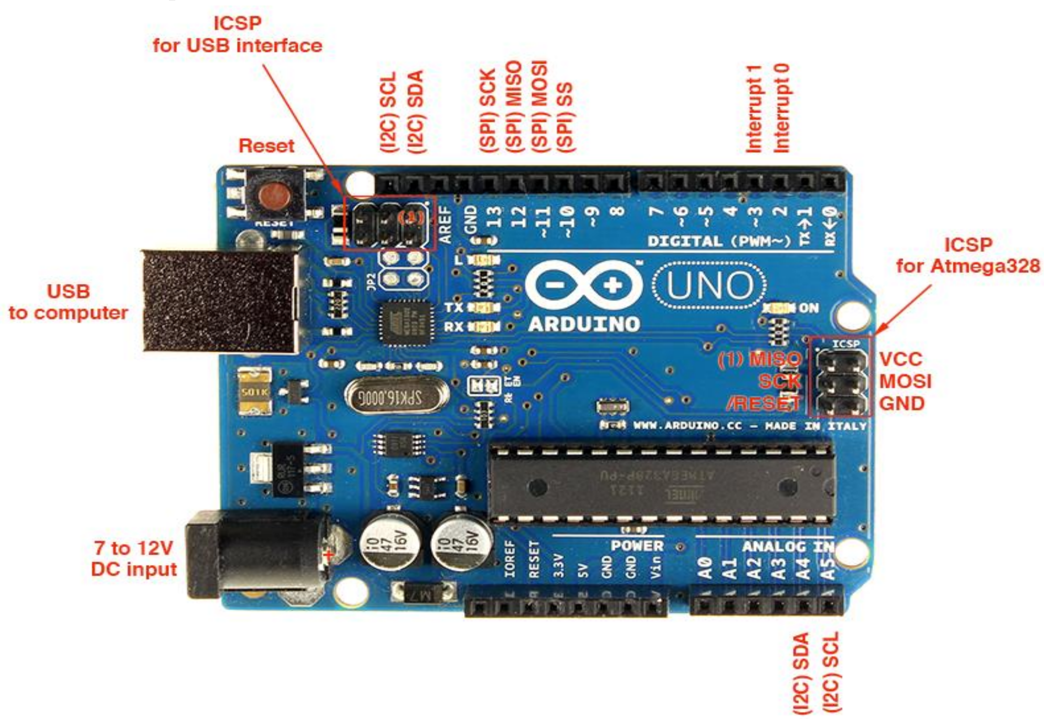
The lack of an autonomous, dependable, and reasonably priced cardiac monitoring system that offers location tracking and emergency alarms without relying on cellphones or human involvement is, in summary, the issue this project attempts to solve. By making sure that quick assistance is always only a message away, the Heart Voice IoT Guardian offers a solution that empowers people with heart issues, boosting their safety and providing peace of mind. By resolving the drawbacks of current monitoring technologies, the system enhances patient outcomes and becomes a viable and affordable choice for everyone in need of cardiac care.

3. PROPOSED METHODOLOGY

1. DESCRIPTION OF THE COMPONENTS:

i) Arduino Uno

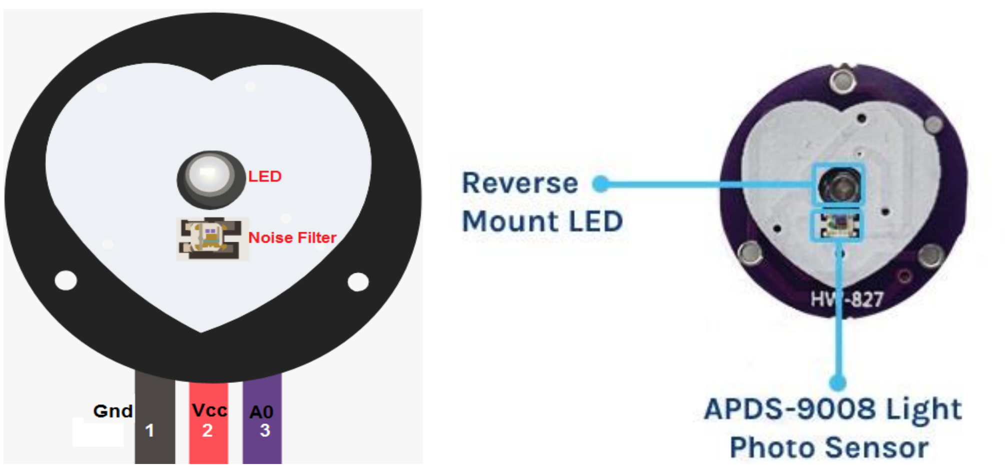
- **Overview**: The **Arduino Uno** is an open-source microcontroller board based on the **ATmega328P** microchip. It is one of the most popular development boards due to its simplicity and ease of use, offering an ideal platform for embedded systems.
- **Features**:
  - **Microcontroller**: ATmega328P, with 32 KB of flash memory, 2 KB SRAM, and 1 KB EEPROM.
  - **Digital I/O Pins**: 14 (of which 6 can be used as PWM outputs).
  - **Analog Inputs**: 6 (10-bit ADC).
  - **Communication**: UART, I2C, and SPI support for external devices like sensors, modules, and displays.
  - **Power**: Can be powered via USB or external 9V DC power.
- **Role in the project**:
  - The Arduino Uno serves as the central processing unit that handles input from the **HW827 heart rate sensor** and **GPS module** and sends output to the **LCD display** and **SIM900A GSM module**.
  - It runs the logic that processes the heart rate data and triggers emergency alerts when required.



Figure

ii) HW827 Heart Rate Sensor

- **Overview**: The **HW827 heart rate sensor** is a photoplethysmogram (PPG)-based sensor that detects the user's heart rate by measuring the intensity of light reflected by the skin. The sensor uses infrared light to detect blood flow changes with each heartbeat.
- **Features**:
  - **Heart Rate Detection**: Measures the user's heart rate in beats per minute (BPM).
  - **Sensor Type**: Analog, with output typically in the range of 0 to 5V depending on the heart rate.
  - **Operation Principle**: It uses infrared light and photodiodes to detect variations in light absorption due to blood flow.
- **Role in the project**:
  - The **HW827 sensor** continuously detects the user's heart rate and provides this information as an analog signal to the **Arduino Uno**. The Arduino then processes this signal and converts it into beats per minute (BPM).
  - If the heart rate exceeds a predefined threshold (e.g., 100 BPM), the Arduino triggers an emergency response by sending alerts via the **GSM module**.



iii) SIM900A GSM Module

- **Overview**: The **SIM900A** GSM module is a widely-used, low-cost GSM communication module designed for sending SMS messages, making voice calls, and using the GSM network for wireless communication.
- **Features**:
    - **Communication**: Supports GSM/GPRS, capable of sending and receiving SMS and voice calls.
    - **Interface**: Serial communication (using **SoftwareSerial** in this case), which allows connection to the **Arduino** on digital pins (pins 2 and 3 for this project).
    - **SMS Functions**: Can send and receive SMS messages and provide status updates (e.g., network registration).
    - **Call Functions**: Supports making and receiving calls.
- **Role in the project**:
    - The **SIM900A GSM module** is used to send an SMS message containing the user's heart rate and GPS coordinates to emergency contacts when the heart rate exceeds the predefined threshold.
    - It also supports making emergency calls to a predefined number, providing an additional method for contacting help.

iv) GPS Module (e.g., NEO-6M)

- **Overview**: The **GPS module** (such as the **NEO-6M**) is used to provide real-time GPS coordinates (latitude and longitude) by communicating with GPS satellites. It outputs location data in the NMEA sentence format, which the Arduino can process.
- **Features**:
  - **Location Data**: Provides latitude, longitude, altitude, and time information.
  - **Accuracy**: Typically provides an accuracy of 2-5 meters under ideal conditions.
  - **Communication**: Uses **UART** (serial communication) to send data to the **Arduino**.
  - **Power Supply**: Operates at 3.3V or 5V, depending on the module's design.
- **Role in the project**:
  - The **GPS module** provides the location data that is included in the emergency SMS message. When an emergency occurs (e.g., high heart rate), the **Arduino** retrieves the latitude and longitude from the GPS module and includes this information in the alert sent via the **SIM900A GSM module**.
  - It allows emergency responders or caregivers to pinpoint the user's location quickly.

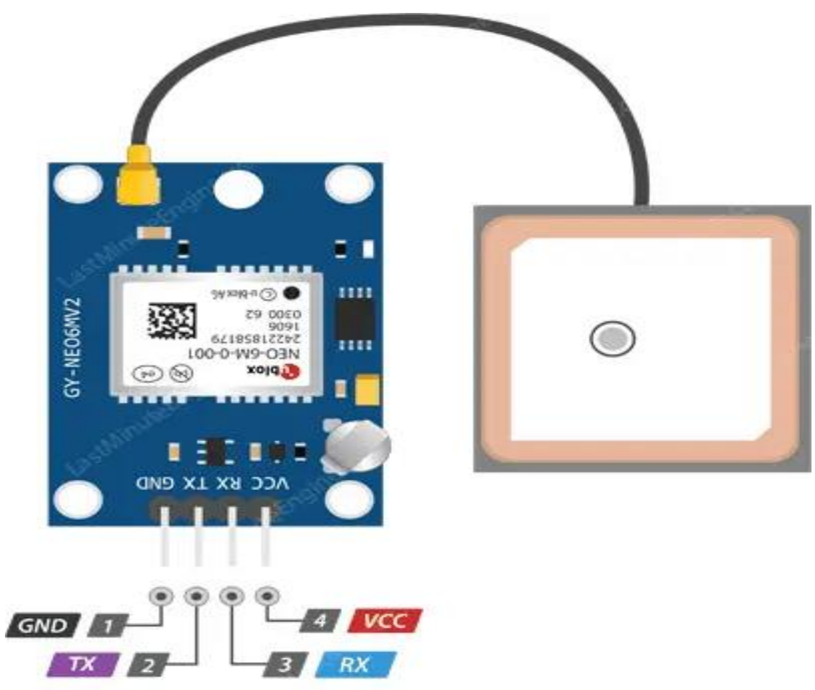


Fig:NEO-6M Pinout

v) LCD Display (I2C 16x2)

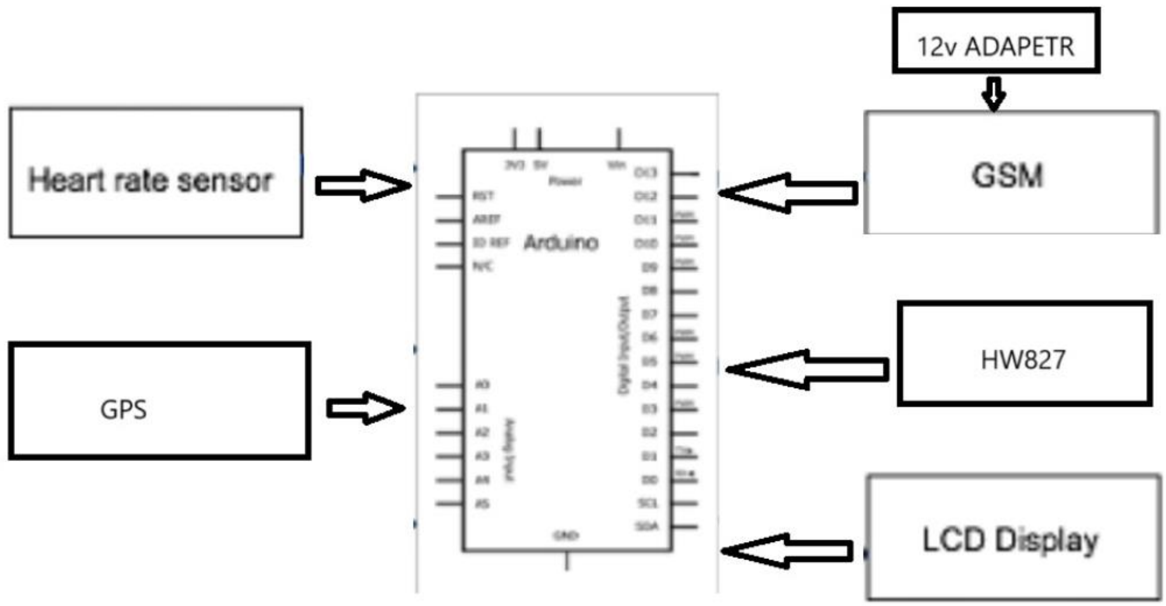
- **Overview**: The **LCD display** used in this project is a **16x2 I2C** screen, which uses an I2C interface to reduce the number of required pins on the **Arduino**. It can display text and simple graphics, making it ideal for real-time data feedback in embedded systems.
- **Features**:
  - **I2C Interface**: Uses only two data pins (SDA and SCL) for communication, making it ideal for projects with limited I/O pins.
  - **Display Size**: 16 columns and 2 rows, capable of displaying alphanumeric characters.
  - **Backlight**: Comes with an adjustable backlight to improve readability in low-light conditions.
- **Role in the project**:
  - The **LCD display** is used to provide live feedback to the user about their heart rate, the system's status, and emergency alerts.
  - It shows the current heart rate, and if the heart rate exceeds the threshold, it can display messages like "Heart Rate High" to alert the user.
  - It also shows the system's operational status, such as GPS signal strength or GSM network registration.

Fig:I2C LCD Pinout

3.1 System Design and Architecture

The Heart Voice IoT Guardian system is intended to show real-time statistics, continuously monitor the user's heart rate, and notify emergency contacts of the user's position in the event that critical thresholds are exceeded. The HW827 heart rate sensor, SIM900A GSM module, GPS module, and LCD display are all integrated via I2C into the system, which is primarily controlled by an Arduino Uno. Every component processes data; the Arduino reads heart rate readings, processes GPS locations, shows information on the LCD, and sends out GSM warnings if any unusual readings are found. These procedures are handled successively by a flowchart in the code logic, guaranteeing seamless module interaction. Every part is set up and adjusted to guarantee precise GPS location tracking, accurate heart rate monitoring, and dependable GSM notifications.

3.1.1 Block Diagram:



3.2 Hardware Setup and Configuration

1. HW827 Heart Rate Sensor
    - Signal Pin: Connects to A0 (Analog Input) on the Arduino Uno.
    - VCC: Connects to 5V on the Arduino Uno.
    - GND: Connects to GND on the Arduino Uno.
2. SIM900A GSM Module
    - TX (Transmit): Connects to D2 on the Arduino Uno (via SoftwareSerial).
    - RX (Receive): Connects to D3 on the Arduino Uno (via SoftwareSerial).
    - VCC: Connects to an external 5V power supply (as the SIM900A module can draw significant current).
    - GND: Connects to GND on the Arduino Uno.
3. NEO-6M GPS Module
    - TX (Transmit): Connects to D5 on the Arduino Uno (via SoftwareSerial).
    - RX (Receive): Connects to D6 on the Arduino Uno (via SoftwareSerial).
    - VCC: Connects to 3.3V or 5V on the Arduino Uno (based on module specifications).
    - GND: Connects to GND on the Arduino Uno.
4. 16x2 I2C LCD Display
    - SDA: Connects to A4 on the Arduino Uno (I2C data line).
    - SCL: Connects to A5 on the Arduino Uno (I2C clock line).
    - VCC: Connects to 5V on the Arduino Uno.
    - GND: Connects to GND on the Arduino Uno.

Summary Table

Component	Arduino Pin	Function
HW827 Heart Rate	A0	Analog Input
	5V, GND	Power and Ground
SIM900A GSM	D2 (TX), D3 (RX)	SoftwareSerial
	External 5V, GND	Power and Ground
NEO-6M GPS	D5 (TX), D6 (RX)	SoftwareSerial
	3.3V/5V, GND	Power and Ground
I2C LCD	A4 (SDA), A5 (SCL)	I2C Data/Clock
	5V, GND	Power and Ground

Ensure that you use a stable power source for the GSM module, as it may cause power fluctuations on the Arduino if powered directly. Additionally, use SoftwareSerial for the GSM and GPS modules to avoid conflicts with the default serial communication on the Arduino Uno's pins (D0 and D1).

3.3 Software Development and Code Breakdown

3.3.1 Code Structure Overview

This project's software code is designed to manage several functions, including calling, sending SMS alerts, updating the LCD display, reading the heart rate, and retrieving GPS data. It makes use of libraries like LiquidCrystal_I2C for LCD screen control, Wire for I2C communication with the LCD, and TinyGPS++ for GPS data handling.

Code Explanation

Setup Section

The setup initializes serial connections, the LCD display, and displays a welcome message:

```cpp
Serial.begin(9600); // Initialize serial for monitoring
sgps.begin(9600);   // Start GPS communication
sgsm.begin(9600);   // Start GSM communication
lcd.init();         // Initialize LCD
lcd.backlight();    // Turn on LCD backlight
lcd.print("Heart Monitor"); // Welcome message
```

Loop Section:

The loop continuously reads data from the GPS and heart rate sensor, displays the heart rate, and sends alerts if the threshold is exceeded:

1.Reading GPS Data: The GPS data is processed with TinyGPS++, checking for valid coordinates. This ensures location data is accurate:

```cpp
  sgps.listen(); // Listen to GPS
  while (sgps.available()) {
    char c = sgps.read();
    gps.encode(c); // Parse GPS data
  }
  if (gps.location.isValid()) {
    gpslat = gps.location.lat();
    gpslon = gps.location.lng();
  } else {
    gpslat = 0.0; // Default values if GPS is invalid
    gpslon = 0.0
 }
```

2. Heart Rate Monitoring: The heart rate sensor data is read and mapped to beats per minute (BPM), displayed on the LCD, and checked against the threshold:

```cpp
int heartRate = analogRead(heartRatePin); // Read raw heart rate data
heartRate = map(heartRate, 0, 1023, 60, 180); // Map to BPM
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Heart Rate:");
lcd.setCursor(0, 1);
lcd.print(heartRate);
lcd.print(" BPM");
```

3. Emergency Alert Logic: If the heart rate exceeds 100 BPM, an emergency alert is triggered:

```cpp
if (heartRate > heartRateThreshold) {
  sgsm.listen();
  sgsm.print("AT+CMGF=1\r"); // Set SMS mode to text
  delay(1000);
  sgsm.print("AT+CMGS=\"+918977622003\"\r"); // Phone number
  delay(1000);
  sgsm.print("Heart Rate: ");
  sgsm.print(heartRate);
  sgsm.print(" bpm\nLatitude: ");
  sgsm.print(gpslat, 6);
  sgsm.print("\nLongitude: ");
  sgsm.print(gpslon, 6);
  sgsm.write(0x1A); // Send SMS (Ctrl+Z)
  delay(5000);

  // Make a call
  sgsm.print("ATD+918977622003;\r"); // Dial emergency number
  delay(30000); // Call duration 30 seconds
  sgsm.print("ATH\r"); // Hang up
  delay(60000); // Delay to prevent retriggers
}
```

4. Loop Delay: A delay at the end of the loop limits the refresh rate to optimize performance:

```cpp
delay(2000); // Update rate
```

3.4 Testing and Validation

Testing entails examining each module separately before evaluating the system as a whole. Verifying heart rate accuracy, GPS precision, and SMS delivery under various network conditions are examples of individual testing. Following component reliability verification, integration testing makes sure the modules operate without hiccups. In order to verify emergency alert functionality and make sure SMS and call features work as intended, scenarios such as simulated elevated heart rate measurements are used. In order to improve usability for older or high-risk consumers, user feedback is also gathered.

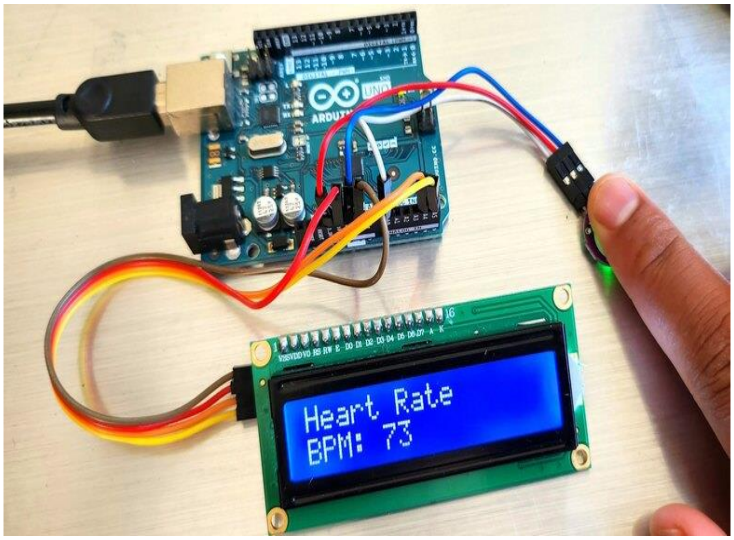
CODE:

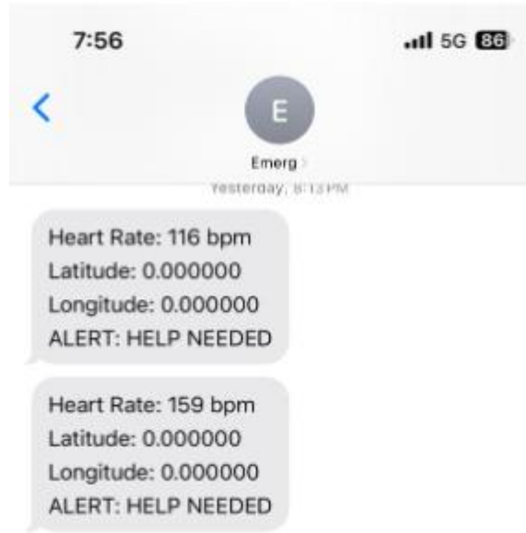
```
#include <SoftwareSerial.h>
#include <TinyGPS++.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// GPS setup
SoftwareSerial sgps(5, 6); // GPS SoftwareSerial pins tx=5, rx=6
TinyGPSPlus gps; // GPS object
float gpslat = 0.0; // Latitude variable
float gpslon = 0.0; // Longitude variable

// GSM setup
SoftwareSerial sgsm(2, 3); // GSM SoftwareSerial pins tx=2, rx=3

// Heart Rate Sensor setup
const int heartRatePin = A0; // Heart rate sensor pin
const int heartRateThreshold = 100; // Threshold for high heart rate

// LCD setup
LiquidCrystal_I2C lcd(0x27, 16, 2); // I2C address 0x27, 16 columns, 2 rows

void setup() {
  Serial.begin(9600); // Serial monitor
  sgps.begin(9600);   // GPS
  sgsm.begin(9600);   // GSM

  // Initialize LCD
  lcd.init();
  lcd.backlight(); // Turn on backlight

  // Initial message on the LCD
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Heart Monitor");
  delay(2000); // Display for 2 seconds
  lcd.clear();
}
```

```
void loop() {
  // Read GPS data
  sgps.listen(); // Listen to GPS module
  while (sgps.available()) {
    char c = sgps.read(); // Read GPS data
    gps.encode(c);        // Process GPS data
  }

  // Get the latest GPS coordinates
  if (gps.location.isValid()) {
    gpslat = gps.location.lat();
    gpslon = gps.location.lng();
  } else {
    gpslat = 0.0;
    gpslon = 0.0;
  }

  // Read heart rate sensor data
  int heartRate = analogRead(heartRatePin); // Read raw data from heart rate sensor
  heartRate = map(heartRate, 0, 1023, 60, 180); // Adjust mapping based on sensor range

  // Display heart rate on LCD
  lcd.clear(); // Clear LCD before updating
  lcd.setCursor(0, 0);
  lcd.print("Heart Rate:");

  lcd.setCursor(0, 1);
  lcd.print(heartRate);
  lcd.print(" BPM"); // Display "BPM" for clarity

  // Print GPS and Heart Rate data to Serial Monitor
  Serial.print("Latitude: ");
  Serial.println(gpslat, 6);
  Serial.print("Longitude: ");
  Serial.println(gpslon, 6);
  Serial.print("Heart Rate: ");
  Serial.println(heartRate);

  // Send SMS and make a call if heart rate exceeds threshold
  if (heartRate > heartRateThreshold) {
```

```
    sgsm.listen(); // Switch to GSM module
    Serial.println("Emergency! Sending SMS and making call...");

    // Send SMS
    sgsm.print("AT+CMGF=1\r"); // Set SMS to text mode
    delay(1000);
    sgsm.print("AT+CMGS=\"+918977622003\"\r"); // Replace with your phone number
    delay(1000);
    sgsm.print("Heart Rate: ");
    sgsm.print(heartRate);
    sgsm.print(" bpm\nLatitude: ");
    sgsm.print(gpslat, 6);
    sgsm.print("\nLongitude: ");
    sgsm.println(gpslon, 6);
    sgsm.print("ALERT: HELP NEEDED");
    sgsm.print("18.08889770107394, 79.4683602045776");
    sgsm.write(0x1A); // Send the SMS (Ctrl+Z)
    delay(5000);

    // Make a call
    sgsm.print("ATD+918977622003;\r"); // Replace with your phone number
    delay(30000); // Call duration (30 seconds)
    sgsm.print("ATH\r"); // Hang up the call

    // Optional: Delay to prevent multiple triggers
    delay(60000); // Wait for 1 minute before checking again
  }

  delay(2000); // Loop delay for updating readings
}
```

RESULTS:


ARDUINO
UNO
Heart Rate
BPM: 73


7:56
5G 86
E
Emerg
Heart Rate: 116 bpm
Latitude: 0.000000
Longitude: 0.000000
ALERT: HELP NEEDED
Heart Rate: 159 bpm
Latitude: 0.000000
Longitude: 0.000000
ALERT: HELP NEEDED

CONCLUSION:

In order to sum up, the Heart Voice IoT Guardian offers a useful and approachable way to monitor your heart in real time and respond to emergencies. The goal of this project is to aid people with heart issues, especially those who might require assistance when alone or in remote locations, by providing automated alarms and continuous heart rate monitoring. Faster reaction times are made possible by the smooth data flow that is made possible by the combination of the HW827 heart rate sensor, SIM900A GSM module, GPS module, and LCD display. This enables for the detection of important health events and the notification of emergency contacts with position data.

FUTURE SCOPE:

Adding power-saving capabilities, such sleep modes, to prolong battery life is one possible enhancement. Future revisions are anticipated to include device reduction, additional sensors (such as SPO2 or temperature), and cloud integration for real-time data logging and historical tracking. To make the gadget more adaptable and user-friendly, a mobile app interface might be added to track health history and set alarm settings.

BIBLIOGRAPHY:

1. Arduino Uno Documentation

   Arduino.cc. (n.d.). Arduino Uno Rev3.

   DOI:https://www.arduino.cc/en/Guide/ArduinoUno

2. HW827 Heart Rate Sensor Specifications

   Heart Rate Sensor HW827. (n.d.). HW827 Heart Rate Sensor - Specifications and Datasheet. DOI:https://example-sensor-website.com/hw827

3. SIM900A GSM Module User Manual

   SIMCom. (n.d.). SIM900A GSM Module User Manual and AT Commands Guide.

   DOI:https://simcom.ee/documents

4. TinyGPS++ Library Documentation

   Mikal Hart. (n.d.). TinyGPS++ Library for Arduino.

   DOI:https://github.com/mikalhart/TinyGPSPlus

5. LiquidCrystal_I2C Library for LCD

   Arduino Libraries. (n.d.). LiquidCrystal_I2C Library for Arduino.

   DOI:https://github.com/johnrickman/LiquidCrystal_I2C

6. LCD I2C 16x2 Module Specifications

   Electronics Hub. (n.d.). Interfacing I2C LCD Module with Arduino.

   DOI:https://www.electronicshub.org/interfacing-i2c-lcd-arduino/

7. GPS Module (NEO-6M) Datasheet

   U-blox. (n.d.). NEO-6M GPS Module - Datasheet and User Guide.

   DOI:https://www.u-blox.com/en/product/neo-6-series

8. SoftwareSerial Library

   Arduino.cc. (n.d.). DOI:https://www.arduino.cc/en/Reference/SoftwareSerial

9. IoT for Healthcare

   Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. Future Generation Computer Systems, 29(7), 1645-1660.

10. Wearable Health Monitoring SystemsIslam, S. M. R., Kwak, D., Kabir, M. H., Hossain, M., & Kwak, K. S. (2015). The Internet of Things for Health Care*:* A Comprehensive Survey. IEEE Access, 3, 678–708.