Al-Quds University

Faculty of Engineering

Computer Engineering Department

Elderly Health Monitor

Embedded Systems

(0702466)

Project Members:

Masa Aljayyousi (22012430)

Huthayfa Shaheen (22011389)

Alaa Abdalqader (22011034)

Supervised by: Dr. Imad Hamadeh

May/15/2024

Table of Contents

Table of Figures

Table of Tables

- ## Introduction:

The “Elderly Health Monitor” project aims to provide continuous health monitoring and medication appointment management for elderly individuals who lack consistent personal care. This innovative system utilizes sensors to track vital health indicators such as temperature and heart rate. The system can detect critical health events like fainting and deviation from normal health parameters by analyzing these metrics.

- ## Objectives:

1- **Improve Health Awareness:**
Implement a system that monitors the heart rate, and temperature of elderly individuals.
Develop alerts for caregivers when readings fall outside pre-defined normal ranges.

2- **Enhance Medication Adherence:**
Implement a medication appointment management system allowing caregivers to set medication time schedules.
Implement a confirmation button for elderly individuals to acknowledge medication intake.
Send alerts to caregivers if medication confirmation is not received.

3- **Promote Timely Intervention:**
Implement sensor technology to detect potential health emergencies like fainting episodes.
Establish a communication system to send immediate alerts to caregivers upon detecting such emergencies.

4- **Increase Caregiver Peace of Mind:**
Provide real-time health data and medication adherence information to caregivers.

- **Project Schedule:**

#Week	Start Date	End Date	Task	Status
1	27/MAR/2024	29/MAR/2024	Research and gather information on health monitoring requirements for the elderly.	Completed
	30/MAR/2024	2/APR/2024	Define system architecture and sensor requirements.	Completed
2	13/APR/2024	16/APR/2024	develop the medication alarm system part for the demo	Completed
	16/APR/2024	18/APR/2024	Prepare a document for the initial demo and present it with an explanation of the code.	Completed
3	19/APR/2024	20/APR/2024	Set up communication channels (WhatsApp and SMS) for caregiver alerts.	Completed
	21/APR/2024	22/APR/2024	Implement sensor integration for temperature, and heart rate monitoring.	Completed
	21/APR/2024	23/APR/2024	Write fall detection, health monitor and read Serial code.	Completed
	23/APR/2024	24/APR/2024	Test and calibrate sensor readings.	Completed
4	25/APR/2024	25/APR/2024	Integrate sensor data with notification systems.	Completed
	26/APR/2024	27/APR/2024	Conduct initial system testing and debugging.	Completed
	29/APR/2024	30/MAY/2024	Address any identified issues and perform optimization.	Completed
5	2/MAY/2024	3/MAY/2024	Finalize project.	Completed

	5/MAY/2024	7/MAY/2024	Conduct final demonstration and presentation	Completed
6	8/MAY/2024	10/MAY/2024	Prepare Web page for the project	Completed
	11/MAY/2024	13/MAY/2024	Prepare project report.	Completed

Table (1): Project Schedule

- ## Non-Recurring Engineering (NRE):

#Week/Task	W1	W2	W3	W4	W5	W6
Research and gather information on health monitoring requirements for the elderly.	H M					
Define system architecture and sensor requirements.	A H M					
develop the medication alarm system part for the demo		A H M				
Prepare a document for the initial demo and present it with an explanation of the code.		A H M				
Set up communication channels (WhatsApp and SMS) for caregiver alerts.			A H			
Implement sensor integration for temperature, and heart rate monitoring.			A			
Write fall detection, health monitor and read Sireal code			H M			
Test and calibrate sensor readings.			A			
Integrate sensor data with notification systems.				A		
Conduct initial system testing and debugging.				A		
Address any identified issues and perform optimization.				H M		
Finalize project.					A H M	
Conduct final demonstration and presentation					A H M	
Prepare Web page for the project						H
Prepare project report.						M

Table (2): Time sheet

Key map → A: Alaa, H: Huthayfa, M: Masa

Task (1) Research and gather information on health monitoring requirements for the elderly:

Huthayfa: 1Week * 2 Day * 2 hours per day * 5 ₪ per hour = 20₪.

Masa: 1 Week * 2 Day * 2 hours per day * 5 ₪ per hour = 20₪.

Task (2) Define system architecture and sensor requirements:

Alaa: 1 Week * 3 Day * 2 hours per day * 5 ₪ per hour = 30₪.

Huthayfa: 1 Week * 2 Day * 2 hours per day * 5 ₪ per hour = 20₪.

Masa: 1 Week * 2 Day * 2 hours per day * 5 ₪ per hour = 20₪.

Task (3) Develop the medication alarm system part for the demo:

Alaa: 1Week * 4 Day * 3 hours per day * 5 ₪ per hour = 60₪.

Huthayfa: 1Week * 4 Day * 3 hours per day * 5 ₪ per hour = 60₪.

Masa: 1Week * 4 Day * 3 hours per day * 5 ₪ per hour = 60₪.

Task (4) Prepare a document for the initial demo and present it with an explanation of the code:

Alaa: 1Week * 3 Day * 3 hours per day * 5 ₪ per hour = 45₪.

Huthayfa: 1Week * 3 Day * 3 hours per day * 5 ₪ per hour = 45₪.

Masa: 1Week * 3 Day * 3 hours per day * 5 ₪ per hour = 45₪.

Task (5) Set up communication channels (WhatsApp and SMS) for caregiver alerts:

Alaa: 1Week * 2 Day * 2 hours per day * 5 ₪ per hour = 20₪.

Huthayfa: 1Week * 2 Day * 2 hours per day * 5 ₪ per hour = 20₪.

Task (6) Implement sensor integration for temperature, and heart rate monitoring:

Alaa: 1Week * 2 Day * 2 hours per day * 5 ₪ per hour = 20₪.

Task (7) Write fall detection, health monitor and read Serial code:

Huthayfa: 1Week * 3 Day * 3 hours per day * 5 ₪ per hour = 30₪.

Masa: 1Week * 3 Day * 3 hours per day * 5 ₪ per hour = 30₪.

Task (8) Test and calibrate sensor readings:

Alaa: 1Week * 2 Day * 3 hours per day * 5 ₪ per hour = 30₪.

Task (9) Integrate sensor data with notification systems:

Alaa: 1Week * 1 Day * 3 hours per day * 5 ₪ per hour = 15₪.

Task (10) Conduct initial system testing and debugging:

Alaa: 1Week * 2 Day * 3 hours per day * 5 ₪ per hour = 30₪.

Task (11) Address any identified issues and perform optimization:

Huthayfa: 1Week * 2 Day * 3 hours per day * 5 ₪ per hour = 30₪.

Masa: 1Week * 2 Day * 3 hours per day * 5 ₪ per hour = 30₪.

Task (12) Finalize project:

Alaa: 1Week * 2 Day * 1 hours per day * 5 ₪ per hour = 10₪.

Huthayfa: 1Week * 2 Day * 1 hours per day * 5 ₪ per hour = 10₪.

Masa: 1Week * 2 Day * 1 hours per day * 5 ₪ per hour = 10₪.

Task (13) Conduct final demonstration and presentation:

Alaa: 1Week * 3 Day * 3 hours per day * 5 ₪ per hour = 45₪.

Huthayfa: 1Week * 3 Day * 3 hours per day * 5 ₪ per hour = 45₪.

Masa: 1Week * 3 Day * 3 hours per day * 5 ₪ per hour = 45₪.

Task (14) Prepare Web page for the project:

Huthayfa: 1Week * 3 Day * 2 hours per day * 5 ₪ per hour = 30₪.

Task (15) Prepare project report:

Masa: 1Week * 3 Day * 3 hours per day * 5 ₪ per hour = 45₪.

Total (Costs for all functions): 125₪.

Total cost for Huthayfa: 310₪.

Total cost for Masa: 305₪.

Total cost for Alaa: 305₪.

- ## Bill of Materials:

Name of the component	Cost
Arduino Uno	35₪
I2C LCD	79₪ (borrowed from the university)
Keypad	49₪ (borrowed from the university)
RTC DS1302	25₪ (borrowed from the university)
Full Breadboard	16₪
Half Breadboard	8₪
Jumper wires	40₪
Buzzer	8₪
MAX30102 heart rate sensor	50₪
ESP32	50₪
MPU6050 accelerometer sensor	40₪
Total Cost: 400 – 153 = 247₪	

Table (3): Bill of Material

- ## Key features:

1. **Health Monitoring:**
   - Sensors continuously monitor vital signs such as temperature, heart rate, and detect falls.
   - Abnormal reading or fall detection trigger alerts to caregivers via WhatsApp and SMS, ensuring prompt intervention in case of emergencies.
2. **Medication Appointment Management:**
   - An integrated alarm system reminds the elderly individual to take medication at prescribed times.
   - Upon taking the medication, the individual confirms by pressing a button. Failure to confirm triggers alerts to caregivers to ensure medication adherence.

- ## Boot code:

A set of instruction which is part of the firmware, the main function of the boot code is to turn on the essential hardware component, load the main application software into memory for execution, and update the firmware when there is a new version of firmware stored.

- **Purpose of the Boot Code:** The boot code for the Elderly Care Project is designed to initialize the ESP32 and Arduino microcontroller, ensuring that all peripherals and sensors are correctly configured and ready for use.
- **Hardware Initialization**: Configures GPIO pins, sets up sensor interfaces, and initializes the LCD screen.
  *GPIO: General Purpose Input/Output.
- **Network Setup:** Establishes Wi-Fi connections necessary for communicating with caregivers or medical systems.
- **Configuration Loading:** Reads initial settings from non-volatile memory to set up alarms, user preferences, and other critical parameters.
- **Testing and Validation:** The boot code is rigorously tested to ensure it performs correctly under various scenarios, including power failures, interrupted updates, and hardware malfunctions. This testing helps guarantee reliability and stability of the system during critical operations.

- **Bootloader:**

  Is the program that instructs the computer to start running the operating system or firmware, it is the initial software component that runs when the system is powered on or reset.

  - **ESP32 Bootloader:**

    The ESP32 bootloader is crucial for hardware setup and ensuring the device executes the correct software. It manages firmware updates and ensures that only valid and intact firmware runs on the device. This capability is essential for maintaining the reliability and security of the device, especially for a project like the Elderly Care Project where uptime and correct functioning are critical.
  - **Arduino Bootloader:**

    The Arduino also has a bootloader, which performs a similar function. It initializes the hardware when the Arduino powers up and loads the main application that's been uploaded to the board. The Arduino bootloader is particularly user-friendly, as it simplifies the process of programming the board without the need for an external programmer. It checks if new code has been sent to the board and installs it if available.

- **Initializing hardware:**
  Setting up the CPU, memory, and peripheral devices to ensure everything is ready for the main application to run.
- **Loading firmware:**
  Reading the latest firmware from memory and loading it into the processor. This firmware could include updates to improve sensor reading accuracy or implement new features.
- **Firmware Updates:**
  Arduino typically requires connection to a computer for updates, whereas ESP32 can support OTA updates over Wi-Fi.
  *OTA: Over-The-Air.

Functional requirements:

This project offers several key functions to assist elderly individuals:

1- **Setting Time and Date:** Users can set the current time and date by pressing the (*) button.

2- **Medicine Reminder:** Three alarms (A, B, C) can be set using the keypad to remind users about medication times.
A buzzer sounds and a message is displayed on the LCD to remind the user to take their medication.

3- **Medication Confirmation**: The buzzer continues to sound until the user presses the (D) button to confirm they have taken their medication.

4- **Heart Rate Check:** Users can measure their heart rate by placing their finger on the sensor.

5- **Temperature Check:** The temperature can also be measured using the same sensor.

6- **Fainting Detection:** The system can detect if the user has fainted.

7- **Communication with Caregiver:**
The system sends messages to the caregiver when it’s time for medication and when the user confirms medication intake.

8- **Fall Detection:**
If the system detects a fall, it immediately sends a message to the caregivers through WhatsApp.

- **Non-Functional Requirements:**

  1. **User Interface:** The system should have a clear, easy-to-read LCD display with large text and simple menus. Each function, like setting alarms or checking health metrics, should provide visual and auditory feedback to confirm actions.
  2. **Accessibility:** Designed with the elderly in mind, the system must be operable with minimal physical effort and cognitive load. Features like audible alerts and spoken instructions should support those with visual impairments.
  3. **Reliability:** It's crucial that the system operates reliably, particularly for critical functions such as fall detection and emergency alerts. A backup system should ensure these alerts are sent even if the main system encounters a fault.
  4. **Scalability:** The design should accommodate future upgrades, whether adding new health monitoring sensors or expanding communication methods to include additional caregivers or medical professionals.
  5. **Maintenance:** The system should be easy to maintain, with software that can be updated remotely and hardware components that can be easily replaced. This ensures the system stays operational and up-to-date with minimal hassle.
  6. **Security and Privacy:** Security measures must protect personal and health data rigorously. Data transmission should be encrypted, and the system must comply with privacy laws to restrict access to authorized users only.
  7. **Battery Life and Energy Efficiency:** The system should be energy-efficient, with a long-lasting battery to ensure it remains operational between charges. Users should receive clear, timely notifications when the battery is low to avoid interruptions in monitoring.

- **Design and Implementation:**

  - **System Architecture:** The system architecture for the Elderly Health Monitor utilizes components such as an RTC DS1302, Keypad, LCD, ESP32, MPU6050, MAX30102, and Arduino Uno. These components are integrated to facilitate efficient medication reminders and confirmation processes tailored for elderly individuals, as well as to detect falls, and measure heart rate and temperature.

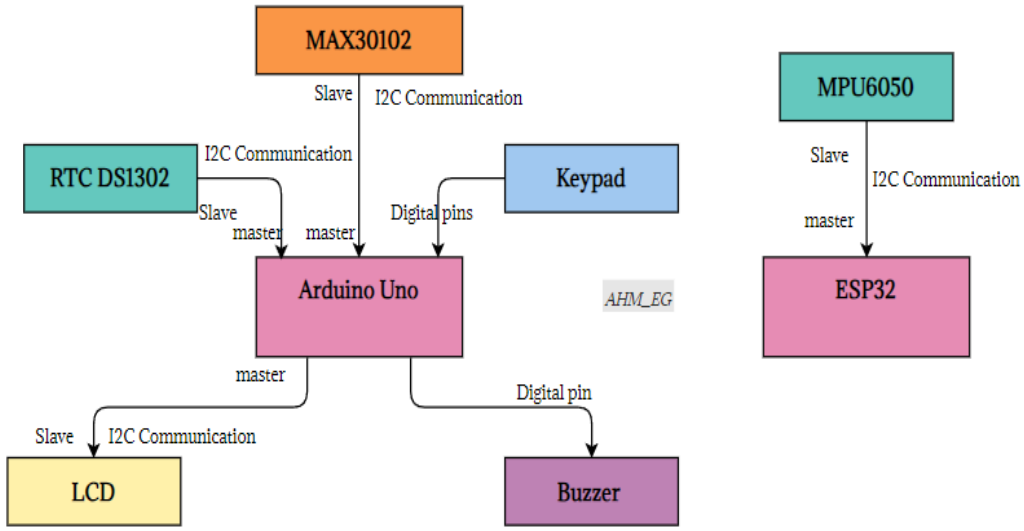


Figure (1): System Architecture

- ## Hardware Components:

1. **Arduino Uno:** It is an open-source microcontroller board based on the ATmega328P processor. It is responsible for controlling sensor inputs, managing alarms, and interfacing with other hardware components.

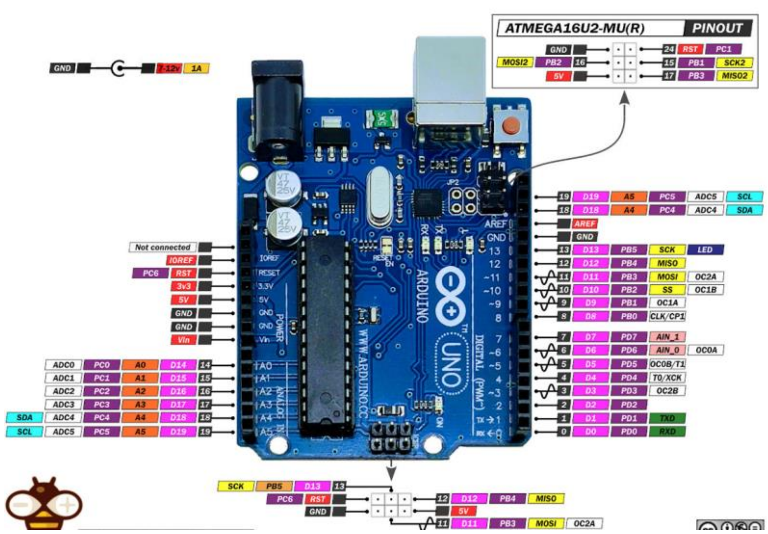


Figure (2): Arduino Uno

2. **ESP32:** ESP32 is a series of low-cost, low-power system on a chip microcontroller with integrated Wi-Fi and dual-mode Bluetooth.

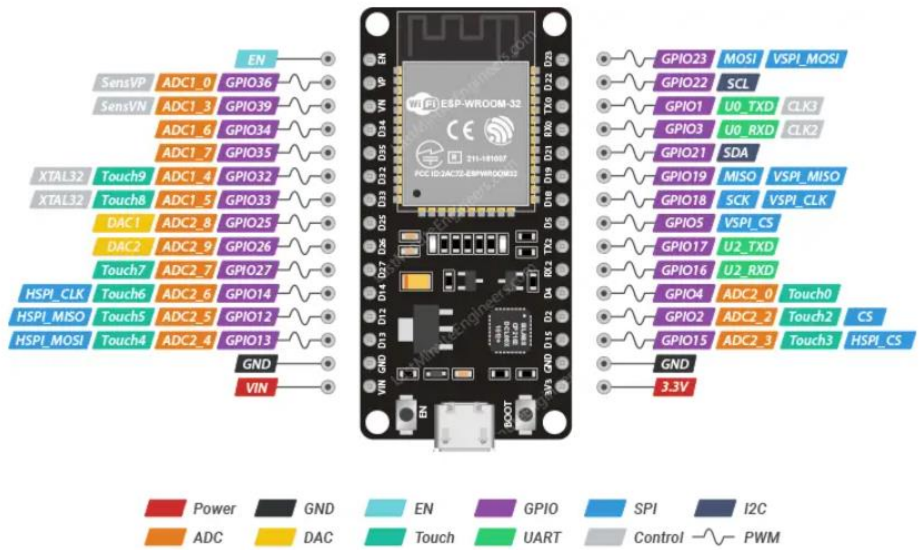


Figure (3): ESP32

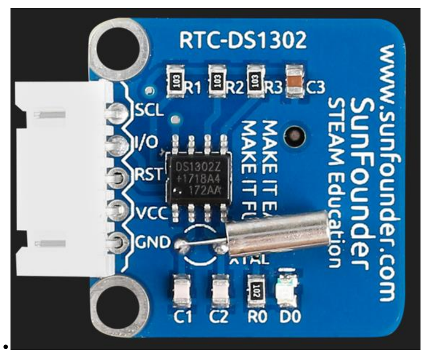
3. **MAX30102 Heart Rate and Temperature Sensor:** integrated pulse oximetry and heart-rate monitor module, with built in temperature sensor. It includes internal LEDs, photodetectors, optical elements, and low-noise electronics with ambient light rejection.

Figure (4): MAX30102

4. **MPU6050 Accelerometer:** MPU6050 sensor module is complete 6-axis Motion Tracking Device. It combines 3-axis Gyroscope, 3-axis Accelerometer and Digital Motion Processor all in small package. It has I2C bus interface to communicate with the microcontrollers.

Figure (5): MPU6050

5. **RTC DS1302 (Real-Time Clock):** The RTC DS1302 is a real-time clock module utilized for precise timekeeping and date tracking within the system. It communicates with the Arduino Uno through the I2C (Inter-Integrated Circuit) protocol, ensuring accurate time synchronization for medication reminders and health monitoring.



.

Figure (6): RTC DS1302

6. **Keypad**: The keypad is an input device equipped with multiple buttons, facilitating user interaction and control. It enables users to set alarms, confirm medication intake, and interact with the system through intuitive button presses.

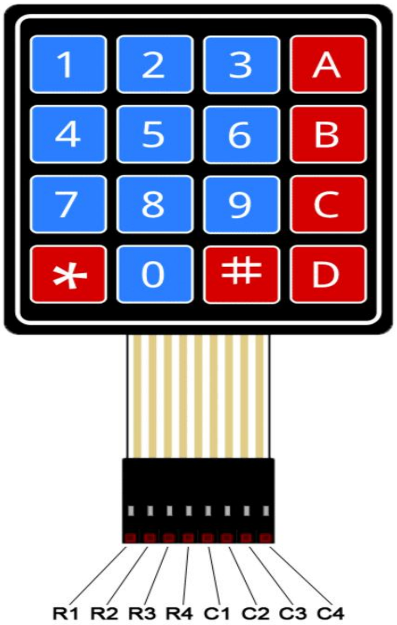


Figure (7): Keypad

7. **I2C Liquid Crystal Display 16x2**: The I2C LCD module combines a standard LCD display with an I2C (Inter-Integrated Circuit) backpack or interface board, facilitating easy interfacing with microcontrollers such as Arduino via the I2C communication protocol. This configuration simplifies wiring and reduces the number of required I/O pins, making it particularly suitable for projects where space and simplicity are essential considerations. The LCD module serves as the visual output interface for the system, displaying vital sign readings, medication reminders, system status, and prompts for user interaction. This allows elderly users to easily access important information displayed on the LCD screen.

Figure (8): I2C

8. **Buzzer**: The buzzer serves as an audio signaling device, emitting sound alerts to notify the user when it's time to take medication. It assists in drawing the attention of the elderly user to important notifications displayed on the LCD screen.

Figure (9): Buzzer

9. **Full and Half Breadboard**:

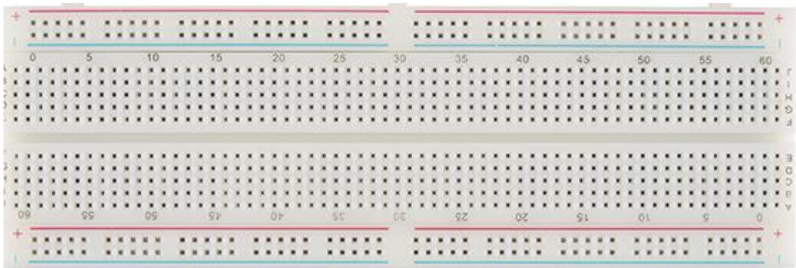


Figure (10): Breadboard

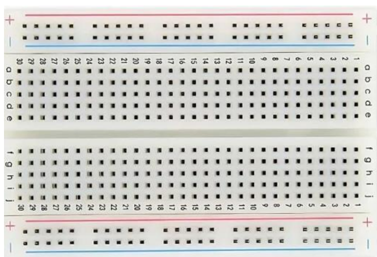


Figure (11): Half Breadboard

10. **Jumper wires**:

Figure (12): Jumper wires

- ## Circuit Design:

Connection:

Arduino Uno	**RTC**
Port D2 (D2)	SCL
Port D3 (D3)	I/O
Port D4 (D4)	RST
GND	GND
5 V	VCC

Table (4): Connection Arduino with RTC

Arduino Uno	**I2C LCD**
Port C5 SCL (Pin 19)	SCL
Port C4 SDL (Pin 18)	SDL
GND	GND
5 V	VCC

Table (5): Connection Arduino with I2C LCD

Arduino Uno	**Keypad**
Port D5 (D5)	C1
Port D6 (D6)	C2
Port D7 (D7)	C3
Port B0 (D8)	C4
Port B1 (D9)	R1
Port B2 (D10)	R2
Port B3 (D11)	R3
Port B4 (D12)	R4

Table (6): Connection Arduino with Keypad

Arduino Uno	Buzzer
Port B5 (D13)	VCC
GND	GND

Table (7): Connection Arduino with the Buzzer

Arduino Uno	MAX30102
VCC	VCC
GND	GND
Port C4(A4)	SDA
Port C5(A5)	SCL

Table (8): Connection Arduino with MAX30102 sensor

ESP32	MPU6050
3.3V	VCC
GND	GND
D21(SDA)	SDA
D22(SCL)	SCL

Table (9): Connection ESP32 with MPU6050 sensor

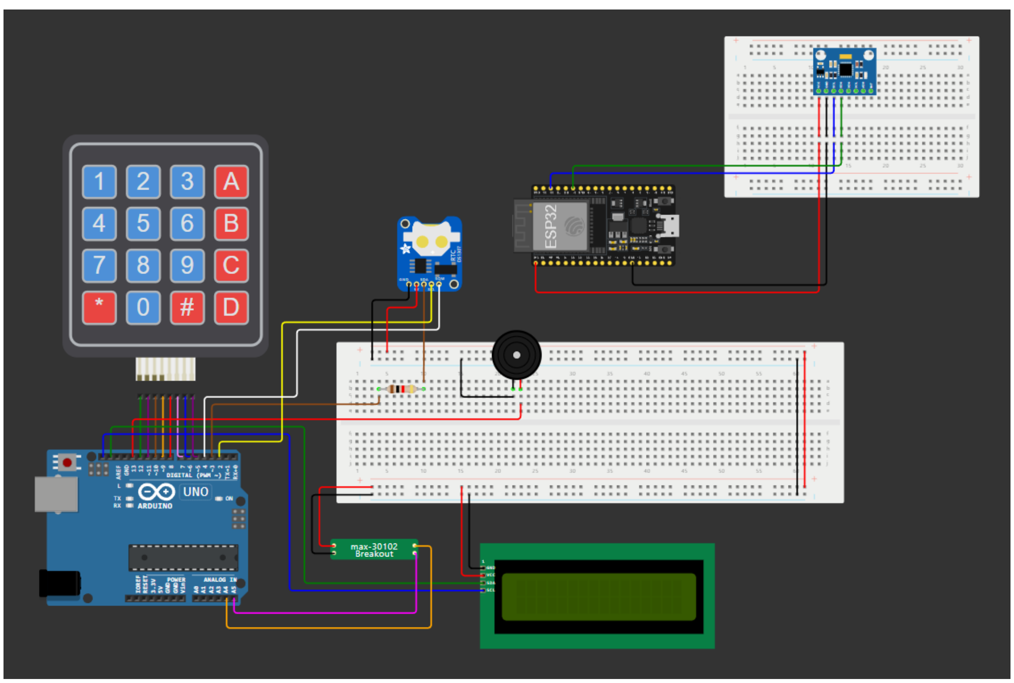


Figure (13): breadboard layout

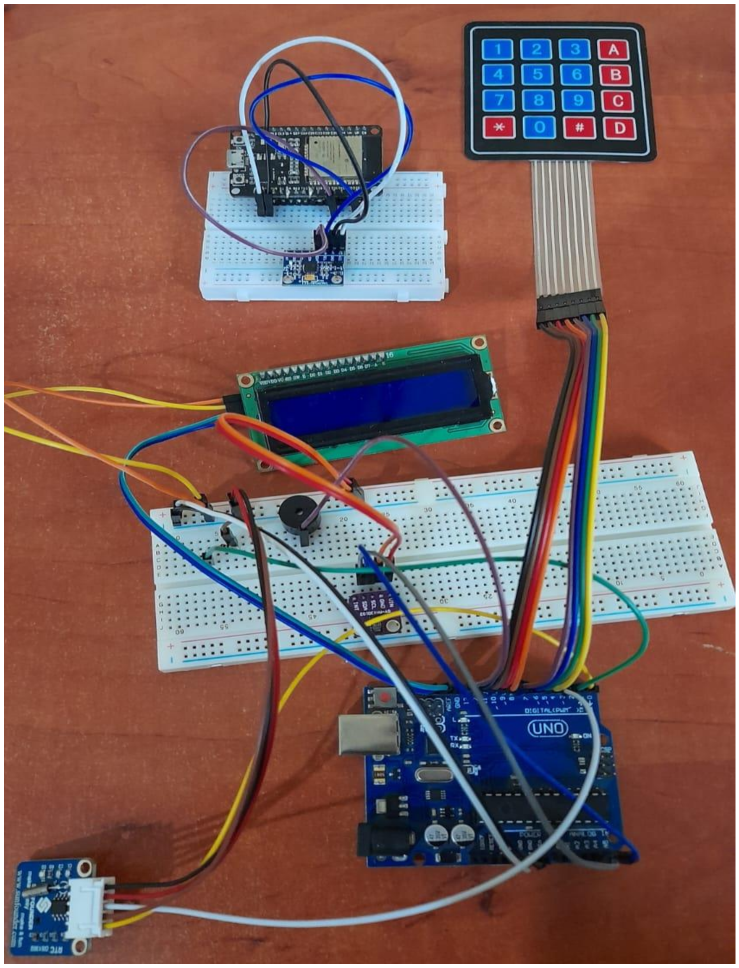


Figure (14): Circuit design for the system

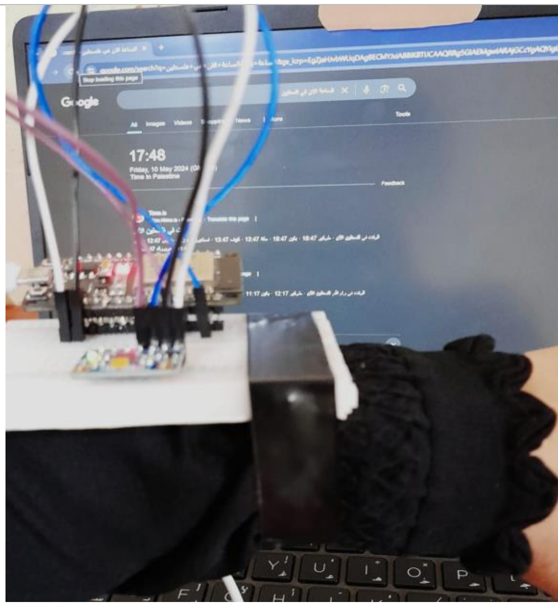
- ## Software Requirements & Code:

  The software tool we used to perform our coding is Arduino IDE and PyCharm, the code was written in C and Python language.

  Here is the link of the code for the project:

  https://github.com/LoloAbd/AHM_EG.git

- ## System Testing Result:


Google
17:48

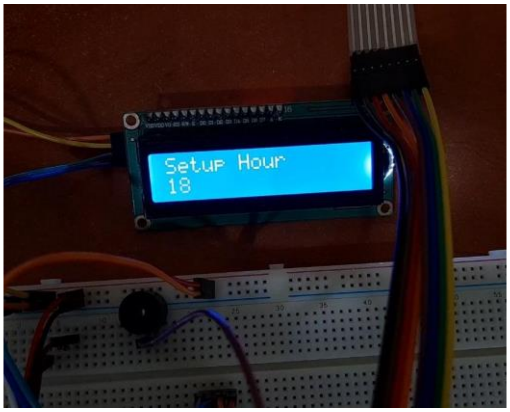

Boot
Sunday
Messages and calls are end-to-end encrypted. No one outside of this chat, not even WhatsApp, can read or listen to them. Tap to learn more.
Today
FALL DETECTED! 17:44
FALL DETECTED! 17:44
FALL DETECTED! 17:48
Message

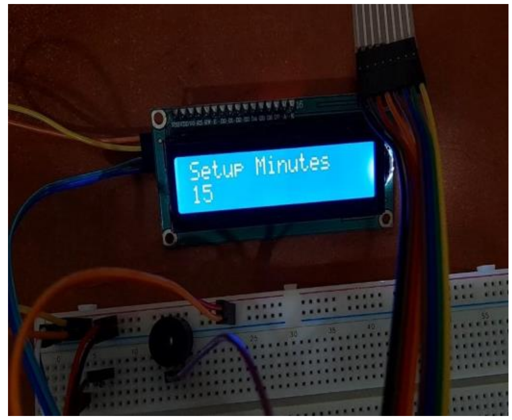

Setup


Setup Year
2024

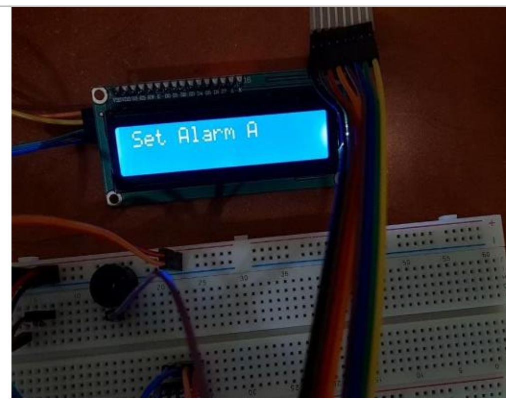

Setup Month
05


Setup Day
10


Setup Hour
18

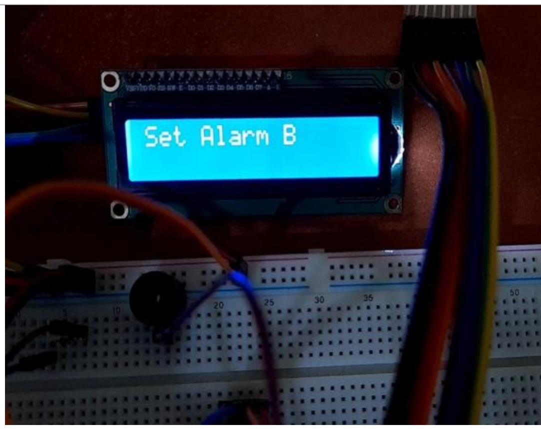

Setup Minutes
15


10/5/2024
18:15


Set Alarm A


Set Alarm Hour
18

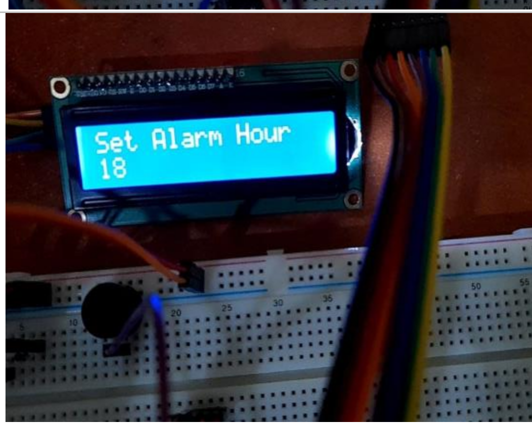

Set Alarm Minute
34


Set Alarm B

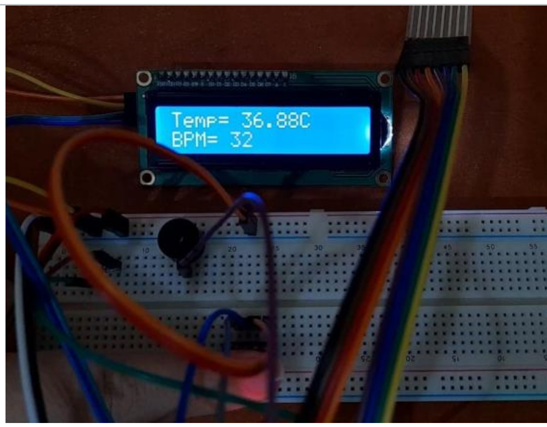

Set Alarm Hour
18

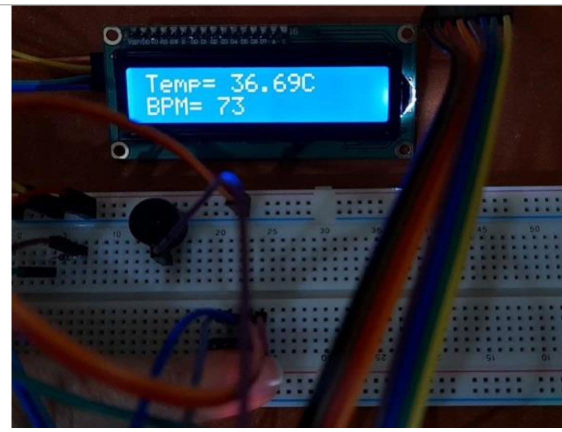

Set Alarm Minute
35


Set Alarm C


Set Alarm Hour
18

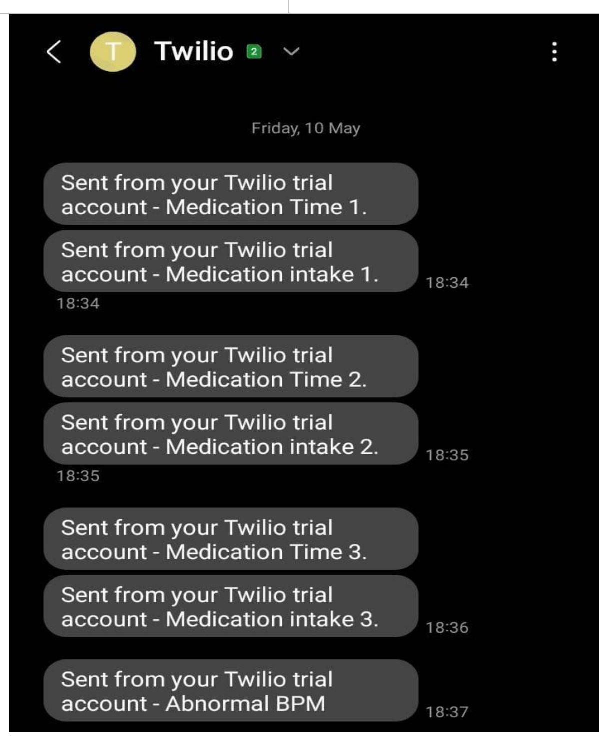

Set Alarm Minute
36


Temp= 36.88C
BPM= 32


Temp= 36.69C
BPM= 73


Medication Time


Alarm deactivate


Twilio
Friday, 10 May
Sent from your Twilio trial account - Medication Time 1.
Sent from your Twilio trial account - Medication intake 1.
18:34
18:34
Sent from your Twilio trial account - Medication Time 2.
Sent from your Twilio trial account - Medication intake 2.
18:35
18:35
Sent from your Twilio trial account - Medication Time 3.
Sent from your Twilio trial account - Medication intake 3.
18:36
Sent from your Twilio trial account - Abnormal BPM
18:37

- **System evolution:**

1. **Enhanced Health Monitoring Features:** Future iterations of the system could include more sophisticated health monitoring capabilities, such as blood pressure tracking, sleep quality analysis, and oxygen saturation levels. These additional metrics would provide a more comprehensive health overview, enabling early detection of potential health issues.

2. **Integration with Health Systems:** The system could evolve to integrate seamlessly with electronic health records (EHR) systems used by healthcare providers. This integration would allow for real-time data sharing with doctors, facilitating more informed and timely medical interventions.

3. **Advanced Machine Learning Algorithms:** Incorporating machine learning algorithms could enhance the system's ability to predict potential health issues based on historical data and trends. For example, it could predict the risk of falls or serious health declines, prompting preemptive actions.

4. **Improved User Interaction:** To further simplify user interaction, future versions could include voice recognition technology, allowing users to operate the system through voice commands. This would make the system even more accessible, particularly for users with mobility or vision impairments.

- **Conclusion:**

The Elderly Health Monitoring System is designed to improve the safety and independence of elderly users by offering easy-to-use, reliable features. It includes medication reminders, heart rate and temperature checks, fainting detection, and direct communication with caregivers. The interface is straightforward, ensuring elderly users can operate it without confusion.

This system is also built to be robust and dependable, with critical alerts like fall detection having fail-safe mechanisms to ensure caregivers are always notified in emergencies. It's scalable for future upgrades and designed for easy maintenance, allowing for simple software updates and hardware repairs. Security and privacy are top priorities, with stringent measures to protect sensitive data and ensure compliance with privacy laws. The system's energy efficiency and extended battery life guarantee continuous operation, offering continuous support and reassurance to both users and their families. This monitoring system is a vital tool in managing elderly care, combining functionality with simplicity and safety.