"Heaven's Light is Our Guide"

Rajshahi University of Engineering & Technology, Rajshahi

Department of Electrical and Computer Engineering

Course Code: ECE-3112

Course Title: Microprocessor, Assembly Language & Interfacing Sessional

Date of Submission: 26.07.2025

Project Report

Submitted To,

Md. Faysal Ahamed
Lecturer
Dept. Of ECE, RUET

Submitted By,

Radia Ilhan Reza
Roll: 2110004
&
Md. Humayun Khalid
Roll: 2110015
Session: 2021-2022

Index of Chapters

Acknowledgment

I would like to express my sincere gratitude to my supervisors, **Md. Faysal Ahamed,** Lecturer in the Department of Electrical and Computer Engineering, for their invaluable guidance, encouragement, and support throughout the development of this project. Their insights and expertise have been instrumental in shaping the direction of this research. I am deeply appreciative of their patience and dedication, which have greatly contributed to the successful completion of this work.

I would also like to extend my heartfelt thanks to my friend and teammate, for his continuous support and collaboration throughout this project. His contributions, both in technical assistance and moral encouragement, played a significant role in overcoming challenges and completing the project successfully. His willingness to share ideas, provide constructive feedback, and assist in problem-solving made this journey truly rewarding.

Finally, I am grateful to all those who offered encouragement and support in various ways, helping make this project a reality. Thank you all for your invaluable contributions to this endeavor.

Abstract

The Smart Medicine Reminder System is an embedded healthcare solution designed to improve medication adherence among patients, particularly the elderly and those with memory challenges. Built around the ATmega32P microcontroller, the system integrates a Real-Time Clock (RTC) module for accurate timekeeping, a 16x2 LCD for user interaction, and alert mechanisms including a buzzer and LED indicators. Users can set multiple alarms using push buttons to receive timely audio-visual notifications reminding them to take their medication. The system operates as a standalone device requiring no internet connectivity, making it accessible in low-resource settings. This project demonstrates a cost-effective and user-friendly approach to automating medication reminders, thereby enhancing treatment compliance and patient safety.

1. Introduction

The Smart Medicine Reminder System is a microcontroller-based embedded device designed to assist patients in taking their medication on time, thereby reducing the risk of missed doses that can negatively affect treatment outcomes. This device is particularly useful for elderly individuals, patients with memory impairments, and those managing multiple medications daily.

At its core, the system employs an ATmega32P microcontroller integrated with a Real-Time Clock (RTC) module for precise timekeeping. User interaction is facilitated through a 16x2 LCD display, while a buzzer and a red LED indicator provide audible and visual alerts, respectively. The system also features four push buttons, enabling intuitive alarm setting without the need for external devices.

Key functionalities of the system include:

- Continuously displaying the current date and time on the LCD.
- Allowing users to set medication alarm times via simple button controls.
- Triggering audio and visual alerts (buzzer and LED) when it is time to take medicine.
- Displaying a comforting message, “Time to take your Meds :)”, on the screen during the alarm period.

By leveraging cost-effective components and a user-friendly interface, this system aims to make smart healthcare accessible within home environments, particularly in regions with limited healthcare technology infrastructure.

2. Objectives

- Develop an accessible and affordable reminder system to improve medication adherence.
- Utilize embedded systems knowledge in a real-world healthcare application.
- Eliminate caregiver dependency by allowing patients to independently manage medicine schedules.
- Design a simple user interface with visual and audible alerts.

3. Required Apparatus

Component	Quantity	Purpose
ATmega32P Microcontroller	1	Core controller to manage the entire system
DS1307 RTC Module	1	Keeps track of real time even when power is off
16x2 LCD Display	1	Displays time, alarm status, and countdown
Push Buttons (Mini)	4	Used to set alarm hour, minute, and stop alarm
Buzzer	1	Audio alert for medicine reminder
Red LED	1	Indicates active alarm / medicine due

Yellow LED	1	Optional: Warning (e.g., water low, future use)
Green LED	1	Indicates system is powered and functioning
10kΩ Resistor	1-2	Used for pull-up configuration for buttons or LCD contrast
Breadboard / PCB	1	Circuit assembly
Wires (Jumper wires)	Several	For connecting all components
Power Supply (5V/USB)	1	Powers the ATmega32P and components
Crystal Oscillator (16 MHz)	1	Used for ATmega32P clock timing
Capacitors (22pF)	2	For oscillator stability
3V Coin Cell Battery	1	Powers the DS1307 RTC module when main power is off

4. Working Principles

The Smart Medicine Reminder System is designed to alert users to take their medication at the correct time using a programmable alarm. Here's how the system works:

1. Time Tracking

- The DS1307 RTC (Real Time Clock) module keeps track of the current time even when the main power is off (thanks to its onboard battery).
- The ATmega32P communicates with the RTC using I2C protocol to fetch the current hour and minute continuously.

2. Alarm Setup

- The user can set up to three alarm times using push buttons (for hour, minute, and navigation).
- These alarm times are stored in the microcontroller’s memory and shown on the 16x2 LCD display.

3. Alert Mechanism

- When the current time matches any of the preset alarm times:
  - A buzzer sounds.
  - A red LED lights up to notify the user.
  - Optional: A voice reminder can be played using an external module.
- The user can press a button to stop the alarm, marking the medicine as “taken.”

4. Additional Features

- **Countdown timer** on LCD for next dose.
- **Green LED** indicates system is running normally.
- **Yellow LED** may indicate a warning (e.g., low water level in future upgrades).

5. Methodology

The development of the Smart Medicine Reminder System was carried out using a structured and modular approach. The primary goal was to design a system that can accurately keep track of time, allow the user to set medicine alarms, and notify them via visual and audio alerts. The following methodology outlines the step-by-step process of development:

1. Component Selection & Requirement Analysis

- Chose the ATmega32P microcontroller for its efficiency and flexibility.
- Selected DS1307 RTC to maintain accurate timekeeping.
- Chose an LCD (16x2) to display time and alarm messages.
- Integrated push buttons for setting alarm hours and minutes.
- Added LED indicators (Red, Yellow, Green) for visual notification.
- Used a buzzer for sound alerts.

2. Circuit Design and Simulation

- Designed the hardware circuit using Proteus to simulate all components.
- Ensured proper wiring between ATmega32P, RTC, LCD, buzzer, LEDs, and buttons.
- Simulated the time display and alarm triggering process.

3. Code Development

- Wrote the embedded C program to:
    - Initialize and communicate with RTC via I2C.
    - Read and display real-time data on the LCD.
    - Allow time setting through button inputs.
    - Store and compare alarm time with RTC time.
    - Trigger buzzer and LED when alarm time is matched.
    - Clear alarm on button press or after 1 minute.

4. Testing and Debugging

- Each module (RTC, LCD, buzzer, buttons) was tested individually.
- Debugged logic for alarm match condition and user input.
- Ensured no glitches in the loop and proper reset after each alarm.

5. Enhancements

- Implemented user interaction features like:
  - Holding a specific button to enter alarm set mode.
  - Saving time after setting.
  - Allowing the alarm to be canceled manually.
- Future scalability considered to add:
  - Multiple alarm slots
  - Voice output for medicine names
  - Data logging

6. Circuit Diagram

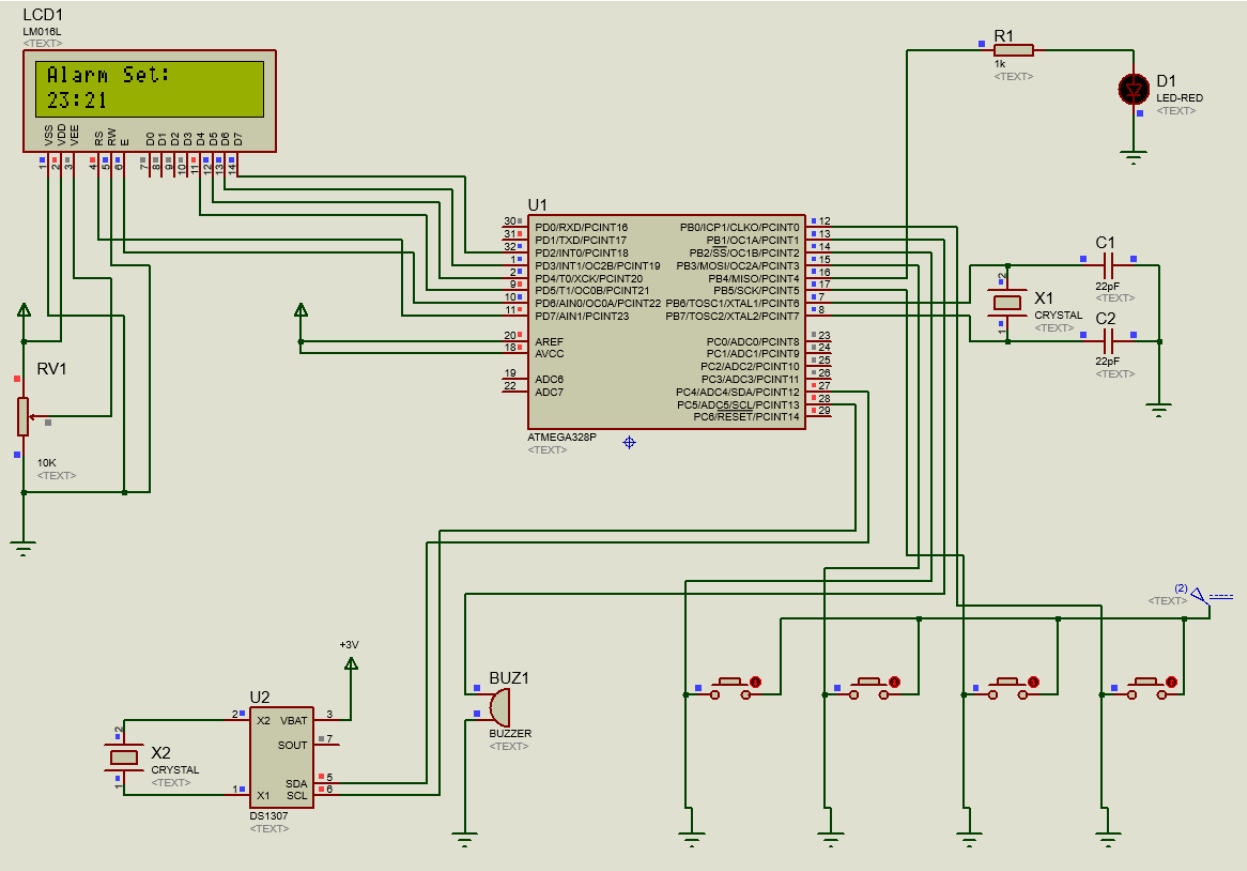

Fig 1: Circuit Diagram

7. Working Flow & Flowchart

System Working Steps:

1. System powers on, LCD initialized.
2. RTC starts showing real-time date & time.
3. User holds Button 4 to enter alarm set mode.
4. User presses Button 1 and 2 to set hour and minute.
5. Alarm time is saved upon releasing Button 4.
6. Microcontroller constantly checks if RTC matches alarm time.
7. If matched:
    a. Buzzer rings
    b. Red LED blinks
    c. LCD shows "Time to take your Meds :)"
8. Alarm stops after 1 minute or upon pressing Button 3.
9. Returns to real-time display.

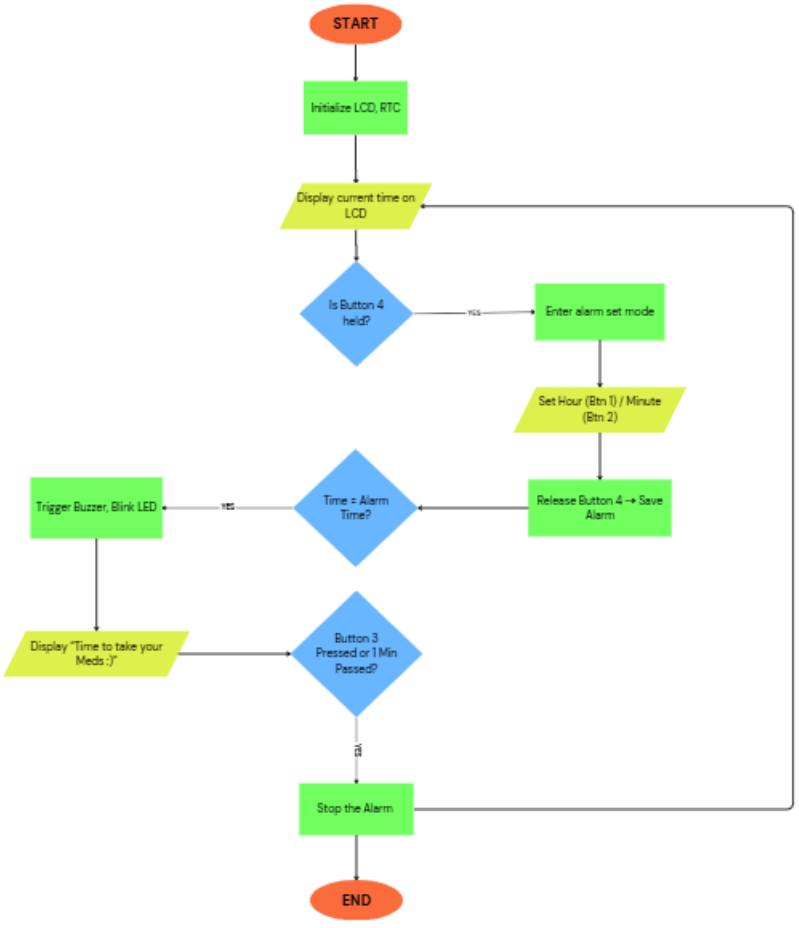

Fig: Flow chart

8. Output:

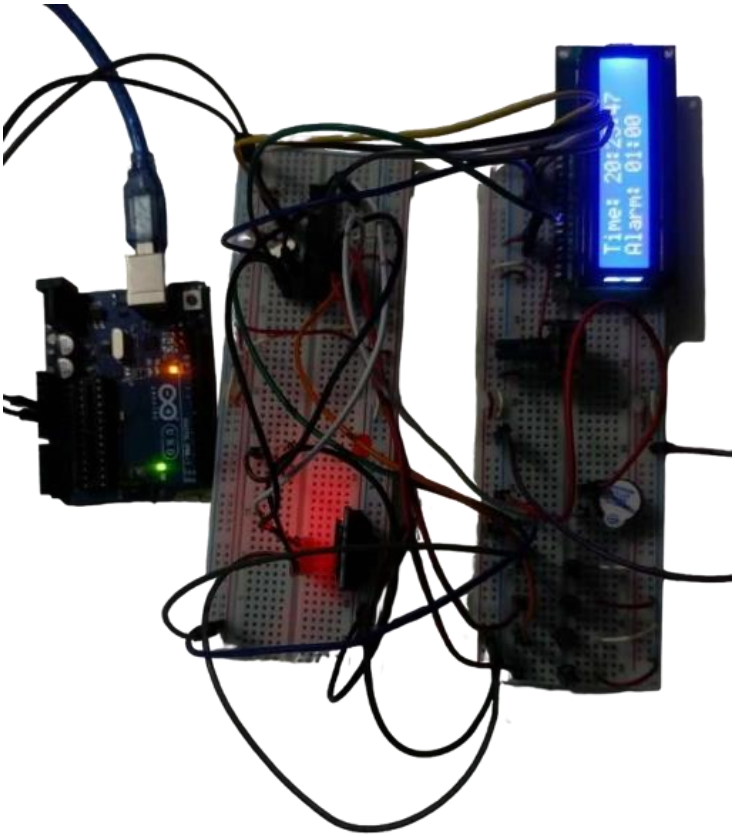

Fig: Output for Smart Medicine Reminder System

9. Hardware connections:

ATmega328P PIN MAPPING:

Arduino Pin	ATmega328P Physical Pin	Port
D0 (RX)	Pin 2	PD0
D1 (TX)	Pin 3	PD1
D2	Pin 4	PD2
D3	Pin 5	PD3
D4	Pin 6	PD4
D5	Pin 11	PD5
D6	Pin 12	PD6
D7	Pin 13	PD7
D8	Pin 14	PB0
D9	Pin 15	PB1
D10	Pin 16	PB2
D11	Pin 17	PB3
D12	Pin 18	PB4
D13	Pin 19	PB5
A4 (SDA)	Pin 27	PC4
A5 (SCL)	Pin 28	PC5

Table 1.1 ATmega328P connections

POWER CONNECTION:

ATmega328P Pin	Connection
VCC (Pin 7)	+5V
AVCC (Pin 20)	+5V
GND (Pins 8, 22)	GND
AREF (Pin 21)	Leave unconnected or to 5V (optional)
RESET (Pin 1)	10kΩ pull-up to +5V (with button to GND optional)
XTAL1/XTAL2 (Pins 9, 10)	16 MHz crystal + two 22pF capacitors to GND

Table 1.2 Power connections

LCD CONNECTION:

LCD Pin	Connect to ATmega328P
RS	PD7 (Pin 13)
EN	PD6 (Pin 12)
D4	PD5 (Pin 11)
D5	PD4 (Pin 6)
D6	PD3 (Pin 5)
D7	PD2 (Pin 4)
VSS	GND
VDD	+5V
VO	Potentiometer center pin (contrast)
RW	GND
A/K	+5V / GND for backlight

Table 1.3 LCD connections

RTC CONNECTION:

RTC Pin	Connect to ATmega328P
SDA	PC4 (A4 / Pin 27)
SCL	PC5 (A5 / Pin 28)
VCC	+5V
GND	GND
Battery	Coin cell

Table 1.4 RTC connections

PUSH BUTTONS:

RTC Pin	Connect to ATmega328P
SDA	PC4 (A4 / Pin 27)
SCL	PC5 (A5 / Pin 28)
VCC	+5V
GND	GND
Battery	Coin cell

Table 1.5 Push Button connections

BUZZER:

Buzzer Wire	Connect To
+ve	PB1 (D9 / Pin 15)
-ve	GND

Table 1.6 Buzzer connections

10. Cost Estimation:

Here's the total cost estimation for the project:

Component	Quantity	Unit Price (BDT)	Total (BDT)
ATmega32P Microcontroller	1	180	180
DS1307 RTC Module	1	120	120
16x2 LCD Display	1	150	150
Push Buttons	4	5	20
Red/Yellow/Green LEDs	3	2	6
Buzzer	1	20	20
10kΩ Resistors (for pull-up & buttons)	6	1	6
Breadboard	1	120	120
Jumper Wires	1 set	50	50
Crystal Oscillator (16MHz)	1	20	20
22pF Capacitors (for crystal)	2	2	4
Power Supply (5V adapter or battery)	1	80	80

Table 1.7 Cost Estimation

11. Challenge Faced:

During the development of the Smart Medicine Reminder System, several challenges were encountered:

1. RTC Communication Issues

- **Problem:** Initial failure in I2C communication between ATmega32P and DS1307.
- **Solution:** Verified pull-up resistors on SDA and SCL lines and ensured correct address usage.

2. Button Debouncing

- **Problem:** Button presses were detected multiple times due to mechanical bouncing.
- **Solution:** Implemented software debouncing using delay logic and state-checking.

3. Alarm Logic Malfunction

- **Problem:** Alarm triggered repeatedly or failed to reset after ringing.
- **Solution:** Added flags and timers to restrict alarm activation to once per match and stop after 1 minute or button press.

4. LCD Display Errors

- **Problem:** LCD sometimes displayed garbled text or no output.
- **Solution:** Adjusted wiring and delay timings, ensured LCD was initialized correctly after power reset.

5. Power Stability

- **Problem:** System sometimes failed to start reliably with external power.
- **Solution:** Added capacitor filtering and ensured VCC/GND stability.

6. ATmega32P Boot & Fuse Settings

- **Problem:** Code not running after programming standalone ATmega32P.
- **Solution:** Corrected fuse bits (for external crystal), and connected crystal oscillator and reset circuit properly.

12. Advantage:

Smart medicine reminder system is useful in the following ways:

- Improves medication adherence for patients.
- Simple user interface requiring no technical knowledge.
- Affordable and easy to build using readily available components.
- Compact and portable design.
- Provides both visual and audio alerts, ensuring accessibility for different patient needs.

13. Limitations:

While functional and reliable for academic purposes, the current prototype has a few limitations:

1. Limited Alarm Count

- Supports only **three alarms**. Users who need more reminders per day must reconfigure.

2. No Real-Time Logging

- Medicine intake times are not saved permanently (unless EEPROM or SD card logging is implemented).

3. No Internet Connectivity

- System does not send notifications via mobile app, SMS, or email (which is possible with Wi-Fi/Bluetooth upgrades).

4. Manual Input Required

- Alarm times must be set using physical buttons, which may be inconvenient for elderly or visually impaired users.

5. Fixed Power Supply

- The system currently relies on a fixed power source and has no battery backup for the microcontroller (though the RTC has one).

14. Conclusion:

The Smart Medicine Reminder System was successfully designed and implemented using the ATmega32P microcontroller, DS1307 RTC, LCD display, buzzer, LEDs, and push buttons. The system effectively allows users to set up to three medication alarms using intuitive button-based input and provides timely visual and audio notifications, enhancing medication adherence especially for elderly or forgetful individuals.

Throughout the development, several challenges such as accurate time synchronization, debouncing button input, and memory limitations of the microcontroller were addressed. Despite these, the final prototype demonstrated reliable performance in tracking time, setting alarms, and alerting users with voice prompts and visual signals.

This project not only showcases the potential of embedded systems in improving health care accessibility but also highlights practical integration of real-time hardware components, low-cost design strategies, and user-centric interfaces. Though limited by storage and audio processing capabilities, the system forms a strong foundation for future upgrades such as GSM alerts or mobile app integration.

In summary, this project has met its objectives and can serve as an assistive tool in daily healthcare routines. It reflects the effective application of microprocessor-based design in solving real-world problems with moderate complexity and high social value.

Appendices

Program Codes:

```
#include <LiquidCrystal.h>
#include <Wire.h>
#include <RTClib.h>

LiquidCrystal lcd(7, 6, 5, 4, 3, 2);
RTC_DS3231 rtc;

int ahours = 0;
int amins = 0;
bool alarmTriggered = false; // flag to track if alarm has been triggered

void setup()
{
  lcd.begin(16, 2);
  lcd.clear();

  Serial.begin(9600);

  // RTC INIT
  Wire.begin();
  if (!rtc.begin()) {
    lcd.print("Couldn't find RTC");
    while (1);
  }
  // Optional: Uncomment to set RTC time ONCE
   //rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));

  pinMode(13, INPUT_PULLUP); // Alarm stop button

  pinMode(11, INPUT);
  digitalWrite(11, HIGH);
```

```
  pinMode(10, INPUT);
  digitalWrite(10, HIGH);

  pinMode(8, INPUT);
  digitalWrite(8, HIGH);

  pinMode(A0, OUTPUT);
  digitalWrite(A0, HIGH);
  //output pins
  pinMode(9, OUTPUT);   // Buzzer
  pinMode(12, OUTPUT);  // LED
}
void loop()
{
  DateTime now = rtc.now(); // Get current time from RTC

   // ========== ALARM SET MODE ==========
  while (digitalRead(8) == LOW) {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Alarm : ");

  if (digitalRead(11) == LOW) {
    amins++;
    delay(200);
  } else if (digitalRead(10) == LOW) {
    ahours++;
    delay(200);
  }

    if (amins > 59) {
      amins = 0;
      ahours++;
    }
    if (ahours > 23) {
      ahours = 0;
    }
    // Display alarm time
    lcd.setCursor(6, 0);
    if (ahours < 10) lcd.print("0");
    lcd.print(ahours);
    lcd.print(":");
    if (amins < 10) lcd.print("0");
    lcd.print(amins);
    delay(500);
  }
```

```
 // ========== DISPLAY CURRENT TIME ==========
 lcd.setCursor(0, 0);
 lcd.print("Time: ");
 if (now.hour() < 10) lcd.print("0");
 lcd.print(now.hour());
 lcd.print(":");
 if (now.minute() < 10) lcd.print("0");
 lcd.print(now.minute());
 lcd.print(":");
 if (now.second() < 10) lcd.print("0");
 lcd.print(now.second());

// ========== DISPLAY ALARM TIME ==========
 lcd.setCursor(0, 1);
 lcd.print("Alarm: ");
 if (ahours < 10) lcd.print("0");
 lcd.print(ahours);
 lcd.print(":");
 if (amins < 10) lcd.print("0");
 lcd.print(amins);


 // ======= ALARM TRIGGER =========
if (now.hour() == ahours && now.minute() == amins && !alarmTriggered) {
   alarmTriggered = true; // mark that we've triggered it
 while (true) {
   lcd.clear();
   lcd.setCursor(0, 0);
   lcd.print("Time to take your");
   lcd.setCursor(0, 1);
   lcd.print("Meds :)");

   digitalWrite(12, HIGH);  // LED ON
   tone(9, 1000);           // Buzzer ON
   delay(500);
   noTone(9);
   digitalWrite(12, LOW);   // LED OFF
   delay(500);

   // Check if Stop Alarm Button is pressed
   if (digitalRead(13) == LOW) {
     noTone(9);
     digitalWrite(12, LOW);
     lcd.clear();
     delay(500);
```

```
      break; // Exit alarm loop
    }
   }
  }

  // ========== RESET alarmTriggered FLAG AFTER 1 MINUTE ==========
  if (now.minute() != amins) {
    alarmTriggered = false;  // reset so alarm can trigger again next day
  }

  // ========== DEBUG ==========
  Serial.print("Alarm Time: ");
  Serial.print(ahours);
  Serial.print(":");
  Serial.println(amins);
  Serial.print("Current Time: ");
  Serial.print(now.hour());
  Serial.print(":");
  Serial.println(now.minute());
  delay(500);
}
```

References:

1. A. A. Kulkarni and S. S. Sonavane, "Smart Medicine Reminder System Using IOT," *International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering*, vol. 6, no. 3, pp. 99–102, Mar. 2018.
2. M. B. Patel and P. R. Parate, "Automatic Pill Reminder System Using Arduino," *International Journal of Advanced Research in Computer Science*, vol. 9, no. 5, pp. 50–54, May 2018.
3. M. S. Islam, M. N. Uddin, and M. M. Rahman, "Design and Implementation of a Digital Medicine Reminder System for Aged People," *2016 2nd International Conference on Electrical, Computer & Telecommunication Engineering (ICECTE)*, Rajshahi, Bangladesh, 2016, pp. 1–4.
4. H. J. Shah, R. A. Prajapati and M. N. Patel, "Medicine Reminder and Monitoring System for Secure Health Using IOT," *International Journal of Computer Applications*, vol. 163, no. 5, pp. 32–35, Apr. 2017.
5. B. P. Pathan, K. D. Joshi and B. R. Joshi, "Microcontroller Based Medicine Reminder," *International Journal of Science, Engineering and Technology Research (IJSETR)*, vol. 6, no. 5, pp. 801–805, May 2017.
6. Adafruit Industries, “DS1307 Real Time Clock Breakout Board Kit,” [Online]. Available: https://learn.adafruit.com/ds1307-real-time-clock-breakout-board. [Accessed: Jul. 25, 2025].
7. Arduino, “ATmega328P Microcontroller Datasheet,” Microchip Technology Inc., 2016. [Online]. Available: https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-42735-8-bit-AVR-Microcontroller-ATmega328-328P_Datasheet.pdf. [Accessed: Jul. 25, 2025].

8. SparkFun Electronics, “Liquid Crystal Display (LCD) 16x2 Datasheet,” [Online]. Available: https://www.sparkfun.com/datasheets/LCD/HD44780.pdf. [Accessed: Jul. 25, 2025].
9. R. Kumar, A. Vyas and P. Jain, "Pill Dispenser System using Embedded System," *International Journal of Computer Applications*, vol. 180, no. 20, pp. 35–38, Mar. 2018.
10. D. R. Sisodia and D. Saini, "Design and Development of a Cost Effective Medicine Reminder System for Elderly People," *2017 International Conference on Computing, Communication and Automation (ICCCA)*, Greater Noida, India, 2017, pp. 1064–1069.