**TOPIC - SMART MEDICATION DISPENSER**

**EMBEDDED SYSTEMS ( BCSE305L )**

**DIGITAL ASSIGNMENT 3**

**SUBMITTED TO**

****

**SCHOOL OF COMPUTER SCIENCE & ENGINEERING**

**UNDER**

**FACULTY : DR, SHARON GIFTSY**

**SUBMITTED BY**

**NEHA RAMESH - 22BCE5201**

**JAYARHUPIKA - 22BCE1665**

**MOHAMED ASIF M - 22BCE1634**

**SAHANA S - 22BCE5195**

**TEAM INTRODUCTION**

1. **Neha Ramesh – 22BCE5201**

Contributions: Coded Arduino, Built dispenser, integrated circuits, tested hardware

1. **Jayarhupika – 22BCE1665**

Contributions: Built Bluetooth-connected Android app for scheduling & alerts. Motor control implementation,

1. **Mohamed Asif M – 22BCE1634**

Contributions: Circuit Assembly and System Debugging. Preparation of Report

1. **Sahana S – 22BCE5195**

Contributions: Arduino programming, serial communication, and buzzer alert logic.

**ABSTRACT:**

Medication non-adherence is a widespread issue in modern healthcare, particularly among the elderly, individuals with chronic illnesses, and patients on complex medication regimens. Traditional methods such as manual pill organizers or alarms often fail to provide a reliable and consistent solution, especially for those requiring frequent assistance or remote care giving. To address these challenges, this project proposes an **Automated Medication Dispensing System** that ensures accurate, timely, and user-friendly medication management. The system incorporates a programmable dispenser capable of **precise dose dispensing**, reducing the chances of human error. Furthermore, the system is integrated with a **mobile application** that enables **real-time tracking** and **remote monitoring**, allowing caregivers and healthcare providers to oversee adherence and receive alerts for missed doses. This comprehensive solution aims to enhance medication compliance, improve patient outcomes, and support independent living through intelligent automation and connected care.

**INTRODUCTION** Medication adherence—the degree to which a patient correctly follows medical prescriptions—is a cornerstone of effective healthcare management. Despite its critical importance, non-adherence remains a persistent and global problem, particularly among the elderly, individuals with chronic illnesses, and those requiring complex medication regimens. Studies indicate that nearly 50% of patients do not take their medications as prescribed, which can lead to a cascade of negative outcomes, including disease progression, avoidable hospitalizations, increased healthcare costs, and even mortality.

Several factors contribute to this widespread issue: forgetfulness, cognitive decline, physical disabilities, lack of awareness, and the absence of constant caregiver supervision. Moreover, in households where a caregiver is not always present, ensuring timely and correct medication intake becomes increasingly challenging. Traditional interventions such as pill organizers, printed schedules, or reminder alarms offer limited support, often failing to provide real-time tracking, dosage verification, or accountability.

To bridge these gaps, this project proposes the design and development of an **Automated Medication Dispensing System**, which integrates **precise mechanical dispensing** with **intelligent scheduling and mobile connectivity**. The system aims to dispense the correct dosage of medication at the prescribed time intervals, accompanied by **audio-visual alerts** that notify the user to take their medication. This approach minimizes reliance on memory and reduces the chance of dosage errors.

Additionally, the system is enhanced with a **mobile application** that enables users and caregivers to set and customize medication schedules, receive real-time adherence notifications, and monitor historical compliance data. This remote monitoring capability adds a layer of accountability and allows for proactive intervention by caregivers or healthcare providers in case of missed doses.

By combining **automation, personalization, and real-time monitoring**, the proposed solution not only enhances medication adherence but also promotes greater independence for patients, reduces caregiver burden, and contributes to improved health outcomes. This system represents a step forward in the integration of assistive technologies within patient-centered care, offering a scalable and effective solution to one of healthcare’s most pressing challenges.

**CODE**

#define IN1 9

#define IN2 8

#define ENA 10

#define BUZZER 8

int interval = 10000;

void setup() {

pinMode(IN1, OUTPUT);

pinMode(IN2, OUTPUT);

pinMode(ENA, OUTPUT);

pinMode(BUZZER, OUTPUT);

digitalWrite(IN1, LOW);

digitalWrite(IN2, LOW);

analogWrite(ENA, 0);

Serial.begin(9600);

}

void loop() {

if (Serial.available()) {

String data = Serial.readStringUntil('\n');

interval = data.toInt() \* 1000;\

Serial.println("Interval set to: " + String(interval));

}

runMotorAndBuzzer();

Serial.println("Time's up! Dispense the medicine.");

delay(interval);

}

void runMotorAndBuzzer() {

digitalWrite(IN1, HIGH);

digitalWrite(IN2, LOW);

analogWrite(ENA, 150);

delay(1000);

analogWrite(ENA, 0);

digitalWrite(BUZZER, HIGH);

delay(1000);

digitalWrite(BUZZER, LOW);

}

### **Software Workflow**

The software component of the medicine dispensing system was developed using the Arduino programming environment. It is designed to receive user-defined time intervals via Bluetooth communication and subsequently trigger the dispensing mechanism at the specified intervals. The software executes in two primary stages:

1. **system initialization (setup()) and**
2. **continuous operation (loop())**

#### **1. Initialization Phase (setup() Function)**

Upon powering up, the setup() function is executed once to configure the Arduino pins and initialize the system:

* **Pin Configuration**:  
   The motor control pins (IN1, IN2, and ENA) and the buzzer control pin (BUZZER) are configured as output pins using the pinMode() function.
* **Motor Initialization**:  
   The motor is stopped initially by setting both IN1 and IN2 to LOW, and the motor speed is set to zero using analogWrite(ENA, 0).
* **Serial Communication Setup**:  
   Serial communication is initialized using Serial.begin(9600). This enables the Arduino to receive time interval data from the Bluetooth module (HC-05), which is connected to the Serial RX/TX pins.

#### **2. Continuous Operation Phase (loop() Function)**

The loop() function runs repeatedly throughout the operation of the system. It performs the following tasks:

##### **a. Reading User Input via Bluetooth**

* The system first checks if there is any available data in the serial buffer using Serial.available().
* If data is available, it is read using Serial.readStringUntil('\n'), which reads the complete message until a newline character is detected.
* The received string is expected to be a numeric value (in seconds), which is converted into an integer and multiplied by 1000 to convert it into milliseconds. This value is then stored in the global interval variable.
* A confirmation message is sent back via the serial port using Serial.println() to notify the user of the new dispensing interval.

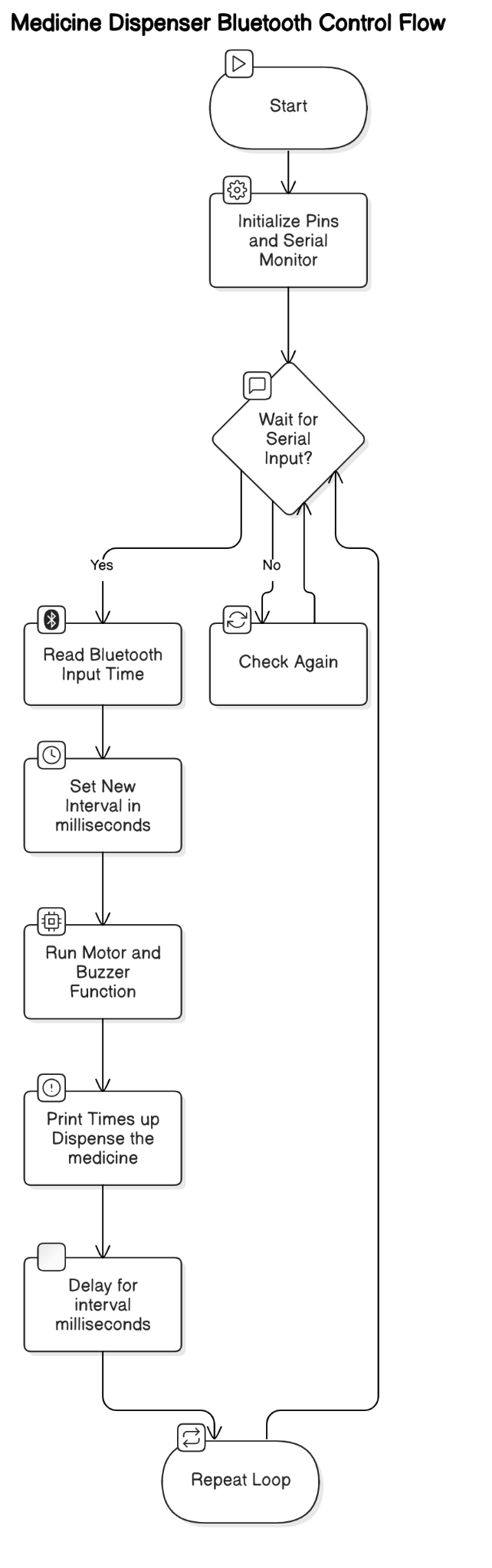
##### **b. Executing the Dispensing Action**

After processing any new input, the function runMotorAndBuzzer() is called to perform the medicine dispensing action. This includes:

* Activating the motor in a forward direction by setting IN1 to HIGH, IN2 to LOW, and applying PWM to the ENA pin (analogWrite(ENA, 150)) to control the speed.
* Allowing the motor to run for a duration of 1 second using delay(1000).
* Stopping the motor by setting the PWM value on ENA back to 0.
* Activating the buzzer by setting BUZZER to HIGH for 1 second, then turning it off.

##### **c. Interval Delay**

After each dispensing cycle, the system waits for the duration specified in the interval variable using the delay(interval) function. This delay ensures that the dispensing action is repeated only after the defined interval, allowing for timed and scheduled medicine dispensing.



## 

## 

## **WORKING METHODOLOGY**

The hardware components used in the development of this Bluetooth-controlled medicine dispensing system include the **Arduino UNO**, **L298N Motor Driver Module**, **DC Motor**, **Bluetooth Module (HC-05)**, **Power Supply**, **Buzzer**. Each of these components plays a crucial role in enabling wireless control, timed execution, and physical dispensing of medicine. The details of each hardware component are described below:

### **Microcontroller Setup - Arduino UNO**

The system is built around the Arduino UNO, a reliable and widely used microcontroller board. It acts as the brain of the entire setup, responsible for receiving user input, processing logic, and activating other components like the motor and buzzer. The Arduino is programmed using the Arduino IDE and was initially powered via USB during development, later through an external power adapter for standalone operation.

**Role:** Controls the logic of the system based on Bluetooth input.  
 **Connections:**

* Receives power via USB or Vin pin.
* Sends control signals to the motor driver via digital pins (e.g., IN1, IN2, ENA).
* Communicates with the Bluetooth module via RX and TX pins.

### **Wireless Communication Integration - HC-05 Bluetooth Module**

A DC motor attached to a wheel was used to simulate the dispensing action. This motor was driven using an L298N Motor Driver Module, which receives control signals from the Arduino to turn the motor ON or OFF and determine the direction of rotation. The motor driver helps handle the higher current required by the motor that the Arduino cannot supply directly.

**Role:** Receives input wirelessly from the mobile app and forwards it to the Arduino.  
 **Connections:**

* VCC to 5V on Arduino.
* GND to GND.
* TX to RX (Pin 0) on Arduino (disconnect while uploading code).
* RX to TX (Pin 1) on Arduino via voltage divider (to step down 5V to 3.3V).

### **Motor Control Mechanism - L298N Motor Driver + DC Motor**

A DC motor attached to a wheel was used to simulate the dispensing action. This motor was driven using an L298N Motor Driver Module, which receives control signals from the Arduino to turn the motor ON or OFF and determine the direction of rotation. The motor driver helps handle the higher current required by the motor that the Arduino cannot supply directly.

**Role:** Controls the motor’s direction and speed using signals from Arduino.  
 **Connections:**

* IN1, IN2 to Arduino digital pins (e.g., D9 and D8).
* ENA to PWM pin (e.g., D10) for speed control.
* VCC to external power supply (9–12V).
* GND to Arduino GND.
* OUT1 and OUT2 to DC motor terminals.

### **Power Supply**

The entire setup was powered using an adapter connected to the Arduino’s power jack to enable standalone operation. Proper precautions were taken during connections using a breadboard and jumper wires to ensure stable operation. Copper wires were occasionally used for direct connections where standard jumper wires were insufficient.

**Role:** Powers the Arduino and motor driver during standalone operation.  
 **Connections:**

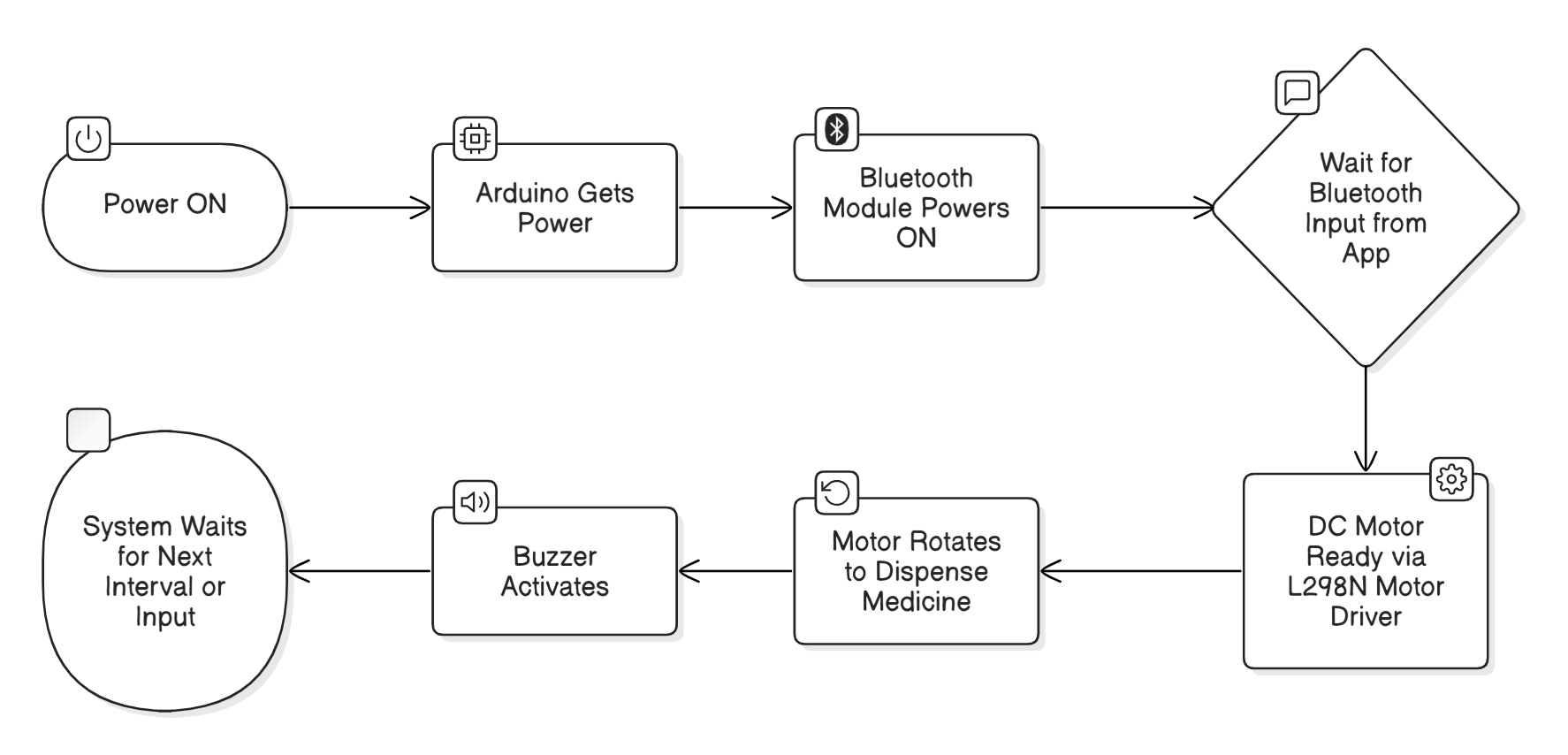
* Connected to the power input terminal of the L298N module or Arduino Vin pin.

### **Alert Mechanism - Buzzer**

To enhance feedback, a buzzer was added to the system. After each dispensing action, the buzzer sounds briefly to notify the user. The buzzer was connected to one of the Arduino's digital output pins and was triggered immediately after the motor action.

**Role:** Gives an audible alert when medicine is dispensed.  
 **Connections:**

* One terminal to a digital output pin on Arduino (e.g., D7 or D8).
* Other terminal to GND.

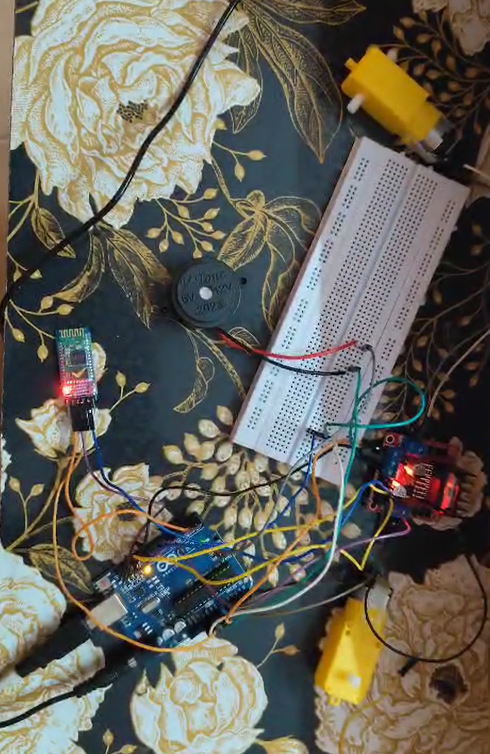


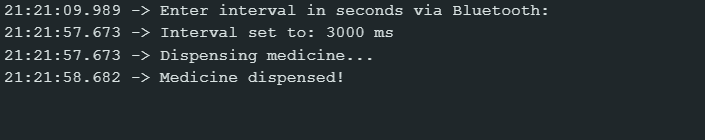
## **RESULTS :**

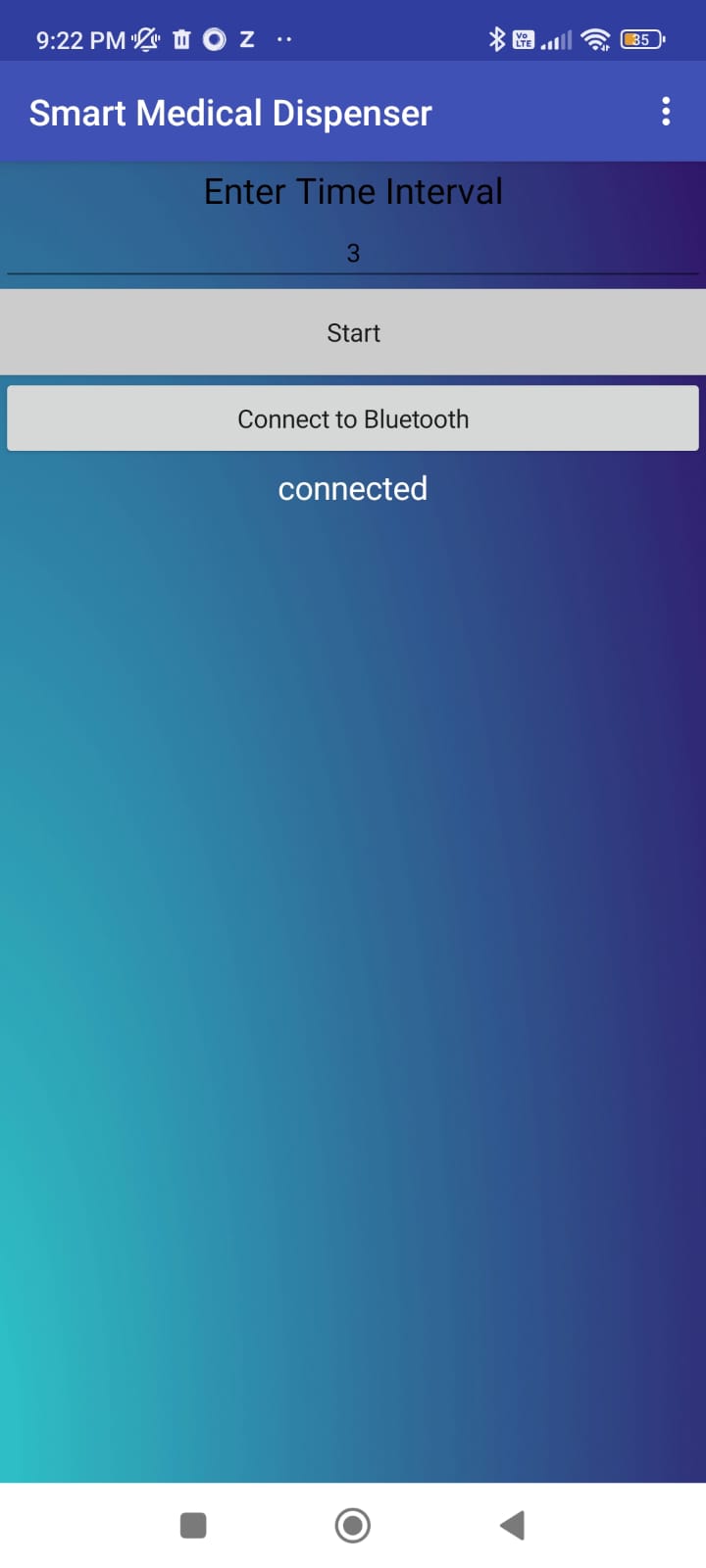
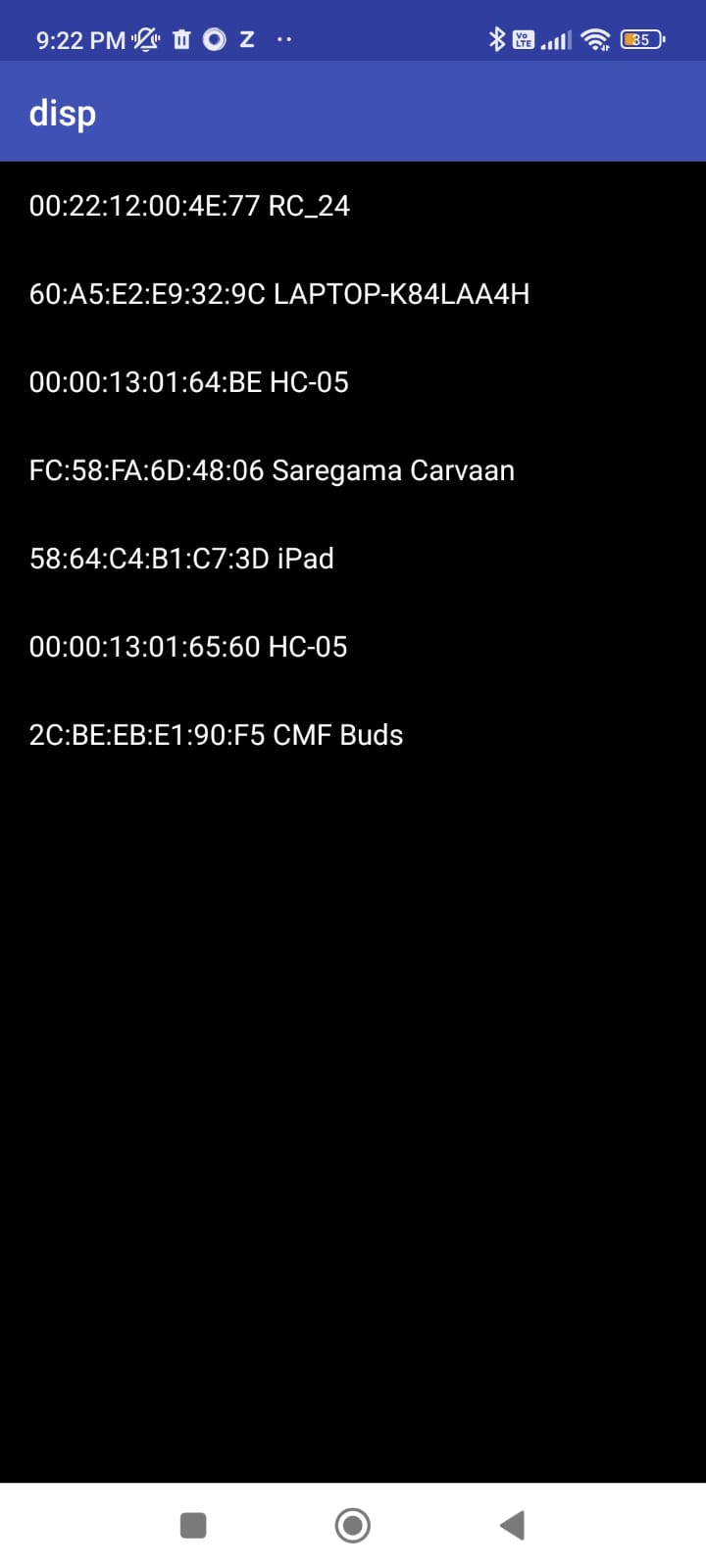
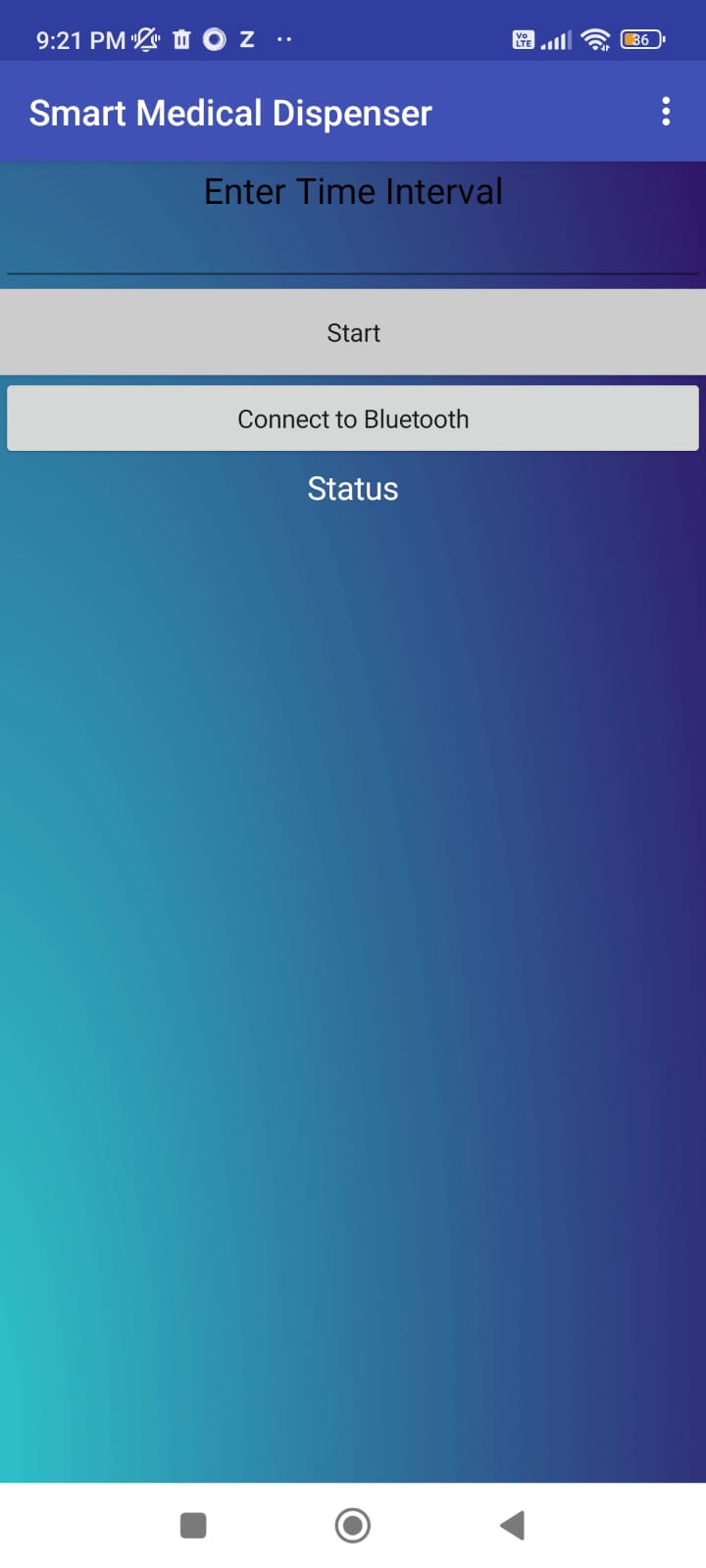
The Bluetooth-controlled automated medicine dispenser was successfully designed and tested to demonstrate scheduled and remote dispensing functionalities. The prototype was developed using an Arduino Uno microcontroller, DC motor, L298N motor driver, HC-05 Bluetooth module, and a buzzer. The Arduino code facilitated wireless interval setting via a mobile application, simulating real-world scenarios where patients or caregivers can remotely control medication timing.

During simulation, values transmitted through the HC-05 module (using the Serial Monitor as a test interface) were interpreted as time intervals in seconds. The system responded by:

* Parsing the input correctly via serial communication.
* Displaying the received interval and dispensing message.
* Simulating the dispensing mechanism using printed output in the Serial Monitor.
* Activating the buzzer simulation through printed feedback to notify the dispensing event.

****





Following successful simulation, the system was implemented and tested on physical hardware. The Arduino Uno received commands via the HC-05 Bluetooth module from a mobile application. The DC motor, controlled by the L298N module, successfully simulated medicine dispensing by rotating a tray or wheel. After each dispensing event, the buzzer activated, providing audible confirmation.

The hardware worked as expected:

* The motor rotated briefly at the defined interval, simulating the dispensing of medicine.
* The buzzer provided a clear audio alert after dispensing.
* Communication between the mobile app and Arduino via Bluetooth was stable and responsive.

The system successfully met its intended objectives, reliably executing scheduled dispensing events controlled wirelessly, confirming both the functional logic and hardware robustness.

**FUTURE IMPROVEMENTS**

**1. Enhanced Connectivity & IoT Integration**

Currently, our system uses **Bluetooth (HC-05**) for wireless control, which limits the range to a few meters. To make it more practical for real-world use, we can integrate **Wi-Fi (ESP8266/ESP32)** or **cellular connectivity (SIM800L).** This would allow remote monitoring via a mobile app where caregivers can adjust schedules, receive missed dose alerts, and track adherence history. Additionally, storing data in the cloud (like **Firebase or AWS IoT**) would enable long-term analytics and seamless access from anywhere.

**2. Multi-Dose & Multi-Compartment Dispensing**

Right now, our prototype dispenses a single type of medication at fixed intervals. However, many patients require multiple pills at different times. A rotary carousel with separate compartments could store different medicines and dispense them as per the schedule. We could also add weight sensors or a small camera with **OpenCV-based pill recognition** to ensure the correct pill is dispensed, reducing errors.

**3. Voice & Touchscreen Interface**

While our current system relies on serial input for scheduling, a **touchscreen display (TFT LCD)** would make it more user-friendly, especially for elderly users. Adding voice control (like Alexa or Google Assistant) would further improve accessibility—imagine just saying, “Dispense my morning pills,” and the system responds. This would make the device more intuitive and reduce dependency on smartphones.

**4. Advanced Security & Authentication**

Since medications can be sensitive, we need to ensure only authorized users can access them. Adding a fingerprint sensor (R307) or facial recognition (**using Raspberry Pi + camera**) would prevent misuse. Alternatively, a simple PIN-based lock could be implemented to restrict access to settings, ensuring only caregivers can modify dispensing schedules.

**5. Power Backup & Low-Power Modes**

Power outages can disrupt medication schedules, which is risky for patients. Integrating a rechargeable battery (like **LiPo**) with solar charging would ensure uninterrupted operation. Additionally, implementing deep sleep modes in the microcontroller would save power when the system is idle, making it more energy-efficient for long-term use.

**6. Emergency & Telemedicine Features**

In critical situations, users might need immediate help. Adding an **SOS button** that alerts caregivers via SMS or app notifications could be lifesaving. We could also integrate telemedicine support, allowing doctors to remotely update prescriptions or check adherence logs, making healthcare more proactive.

**7. AI-Based Predictive Analytics**

Instead of just following a fixed schedule, the dispenser could learn user behavior using **Machine Learning**. For example, if a patient frequently missed doses at a certain time, the system could automatically adjust reminders or send stronger alerts. It could also predict when medication stocks are low and auto-order refills from pharmacies.

**8. Compliance with Medical Standards**

For real-world deployment, the device must meet healthcare regulations like **HIPAA (for data privacy) and FDA/CE certifications** (for safety). Implementing secure data encryption and rigorous testing would ensure reliability, making it a trusted medical device rather than just a DIY project.

**CONCLUSION**

The development of this **Smart Medication Dispenser** demonstrates how embedded systems can be leveraged to address real-world healthcare challenges, particularly medication non-adherence. By integrating an Arduino-based control system with Bluetooth connectivity, we have created a functional prototype that automates **medicine dispensing** at scheduled intervals while providing auditory alerts through a **buzzer.** This system not only reduces dependency on manual reminders but also **minimizes human error**, ensuring timely and accurate medication intake. The use of motor control mechanisms and wireless communication highlights the potential of IoT in healthcare, offering a scalable solution for elderly care, chronic disease management, and assisted living environments.

Looking ahead, this project lays the foundation for future enhancements, such as **multi-compartment dispensing, cloud-based monitoring, and AI-driven scheduling**. Expanding the system with **biometric authentication, telemedicine integration, and emergency SOS features** could further improve reliability and accessibility. By refining hardware efficiency and incorporating user-friendly interfaces, this dispenser could evolve into a commercially viable healthcare device, bridging the gap between technology and patient care. Ultimately, this project underscores the transformative role of embedded systems in improving medication adherence and enhancing quality of life for patients worldwide.

**REFERENCES**

1. Rayan, Rehab & Τsagkaris, Christos & Romash, Iryna. (2021). The Internet of Things for Healthcare: Applications, Selected Cases and Challenges. 10.1007/978-981-15-9897-5\_1.
2. Rejeb, A., Rejeb, K., Treiblmaier, H., Appolloni, A., Alghamdi, S., Alhasawi, Y., & Iranmanesh, M. (2023). The Internet of Things (IoT) in healthcare: Taking stock and moving forward. Technological Forecasting and Social Change, 191, 122556
3. Yin, Y., Zeng, Y., Chen, X., & Fan, Y. (2016). The Internet of Things in healthcare: An overview. Journal of Industrial Information Integration, 1, 3–13.
4. Huang, C., Wang, J., Wang, S., & Zhang, Y. (2023). Internet of Medical Things: A systematic review. Computer Methods and Programs in Biomedicine, 239, 107518.
5. Mukund, S. (2012). Design of Automatic Medication Dispenser. Computer Science & Information Technology. 2. 251-257. 10.5121/csit.2012.2324.
6. W. Antoun, A. Abdo, S. Al-Yaman, A. Kassem, M. Hamad and C. El-Moucary, "Smart Medicine Dispenser (SMD)," 2018 IEEE 4th Middle East Conference on Biomedical Engineering (MECBME), Tunis, Tunisia, 2018
7. Guerrero Ulloa, Gleiston & Hornos, Miguel & Rodríguez-Domínguez, Carlos & Fernández-Coello, Ma. (2020). IoT-Based Smart Medicine Dispenser to Control and Supervise Medication Intake. 10.3233/AISE200021.
8. Pak, J., & Park, K. (2012). Construction of a smart medication dispenser with high degree of scalability and remote manageability. Journal of biomedicine & biotechnology, 2012, 381493
9. C. H. Patil, N. Lightwala, M. Sherdiwala, A. D. Vibhute, S. A. Naik and S. M. Mali, "An IoT based Smart Medicine Dispenser Model for Healthcare," 2022 IEEE World Conference on Applied Intelligence and Computing (AIC), Sonbhadra, India, 2022
10. Ahadani, M. A., De Silva, L. C., Petra, I., Hameed, M. F. A., & Wong, T. S. (2012). Low cost robotic medicine dispenser. Procedia Engineering, 41, 1193–1199