Automated Pill Dispenser (Pillsnap)

ECE445 Design Document

Team: 66 TA: Chi Zhang

Qingyu Li, Wennan Zhai, Shengyu Ge

1. Introduction

1.1 Background

Taking medications as instructed and punctually can be a hard task for patients who need many kinds of medicine, especially for senior people with memory problems. According to the National Survey on Drug Use and Health, most of the people in America took the drugs as prescribed, but 19 million people misuse their drugs. They often get drugs from their friends or relatives and took more than their doctor had prescribed [1]. Given the number of different types of medicine, it is often difficult and troublesome for the patients to keep track of the correct dose and consumption time for each type. It is also very common that many old people need notifications to remind them of taking the pills.

There are some companies trying to build the pill dispenser to solve this problem and make profits. However, it costs a lot of money and many people might not be able to afford that. For instance, The Hero Health Pill Dispenser is one of the most popular products on the market, but it sells for $99.99 and a membership fee (including app and some other services) $29.99 per month.

We will try our best to lower the cost as much as possible. At least, we want our app services to be free for all the users since people are likely to prefer the free software services that come with the devices they buy.

1.2 Objective

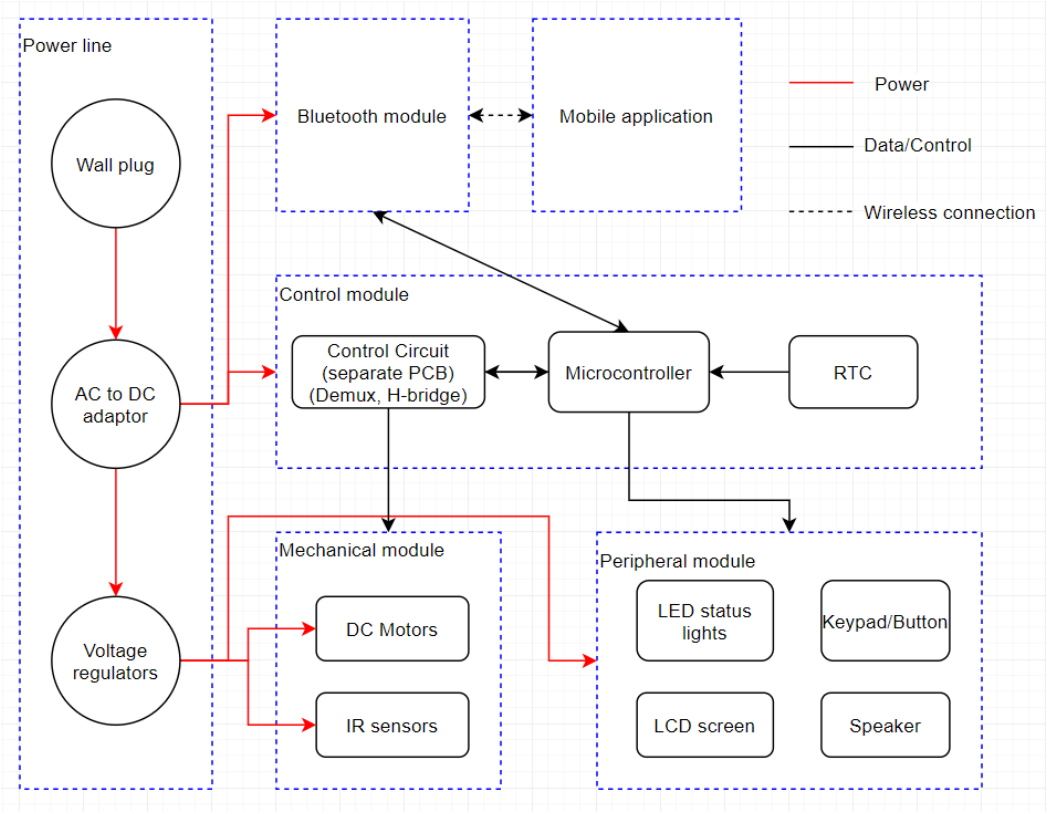
We want to design an automatic pill dispenser that can alert the user to take medicine on time and automatically dispense the correct type and dose of the pills. It will also show the instructions for those pills on a screen. To make the dispenser user-friendly, we plan to develop a mobile application for registering/keeping the information of the medicines and setting dose and consumption time. It can also send notification to the user at the set time. Ideally there will be different profiles for different users for easier family usage. The goal of this pill dispenser is to make it more convenient for people who need to take different kinds of medicine regularly without memorizing the exact dose and consumption.

1.3 High-level Requirements List

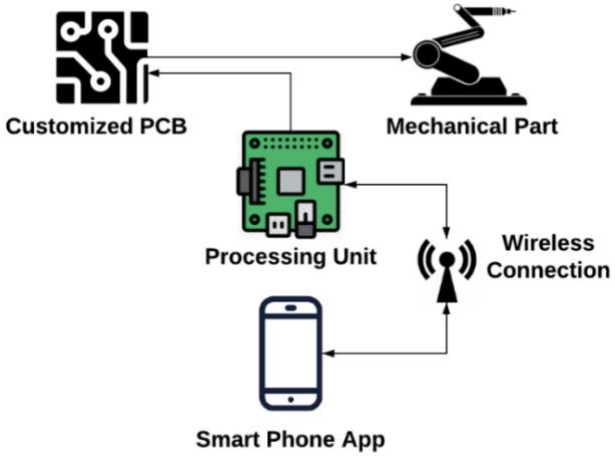
- Be able to control mechanical parts to dispense the pill.
- Be able to keep track of the correct mappings among pills, compartments and dispensing information which can be displayed on an LCD screen.
- User should be able to set up the dispenser's schedule through the mobile app and receive a notification at the right time.

2. Design

2.1 Block Diagram



2.2 Physical Design



2.3 Subsystems

2.3.1 Control Module

The control module consists of a microcontroller, an RTC chip and a control circuit on a separate PCB. The software which will be running on the microcontroller consists of the main program, driver for peripherals and driver for Bluetooth module. The control module is responsible for sending to and receiving data from the mobile app via Bluetooth module and interacting with peripherals such as motors and LCD screen. It also stores the dosage information in the on-chip memory.

Requirement	Verification
1. Software can correctly store and access data in the on-chip memory. 2. Control circuit can select correct output signal and send to the mechanical module. 3. Correctly interact with peripherals using software drivers.	1. A. Stores data at some memory address. B. Fetch at the address and print the result to see if the console's output is correct. 2. A. Turn on the oscilloscope. B. Turn on the monitor for input x. C. Probe the PWM pins on the PCB and press the auto scale button. D. The shown voltage wave should match square wave function and the peak voltage should be 3.3V-5V. 3. A. Connect the LCD display with the correct pins on PCB and test printing "Test" to see if the displayed data is correct.

2.3.2 Peripheral Module

The peripheral module contains the power/data ports for peripherals, software driver for peripherals (overlapping with control module) and peripherals such as LCD screen.

Requirement	Verification

1. Motors can be controlled by the output signal from control module. 2. LCD screen can display the correct information. 3. LEDs can be controlled by the control module.	1. A. Power the motor and connect to the microcontroller. B. Change the PWM output signals from the microcontroller and see if the motor reacts. 2. A. Write code to print “Test”. B. See if LCD can display the text properly and correctly at the top left corner. 3. A. Set digital pins for LED on microcontroller to HIGH or LOW. B. See if LEDs react.

2.3.3 Bluetooth Module

The ESP32 is used to enable Bluetooth connection so that we can exchange information between microcontroller and smart phone app.

Requirement	Verification
1. Can pair with the phone. 2. Can enable the microcontroller to communicate with the phone.	1. A. Connect pin VCC and pin EN to 3.3V input and pin GND and pin IO0 to the GND. Then interconnect RX and TX to TX and RX on the serial-USB converter. B. Upload the example program ‘BLE_server” from the library with the device name “Long name works now”. C. Disconnect the ESP32 with the serial-USB converter and urn on the Bluetooth on Phone to search for the Bluetooth device with the same name. 2. A. Connect pin VCC and pin EN to 3.3V input and pin GND and pin IO0 to the GND. Then interconnect RX and TX to TX and RX on the serial-USB converter, connect LCD display to an Arduino. B. Upload the example program ‘BLE_write” from the library and in loop() add lines of code to transmit the data to the Arduino. C. Disconnect the ESP32 with the serial-USB converter. D. Download and install an BLE scanner app on the phone, connect to “MyESP32”

	and go to CUSTOM CHARACTERISTIC in CUSTOM SERVICE to write something. E. See if the entered text appears on the LCD display.

2.3.4 Application for Phone

The app is the key to inform the user of any update or status. Our current plan is to build an iOS app connected via Bluetooth protocols. The user will have to set the dosing information through the app. Then the message will get transmitted to our processing unit via wireless protocols. Besides, the app will also listen from our processing unit for any update of pill dispensing or notification.

Requirement	Verification
1. The app should be able to display notification with the correct dosing information at the right time. 2. The app should be able to set user's profile including dosing information, pill kinds and etc. 3. The app should be able to send and receive data through Bluetooth.	1. A. Open the app on the phone, set compartment 1 to be dispensed with water one minute later. B. See if there is a notification shown on top of the screen with information "taken with water". 2. A. Continued from the following procedure, we change the first reminder by clicking on the edit button and changed the time to be one minute later from now and add another reminder by clicking on the add button in the app, then setting compartment 2 to be dispensed 2 minutes later without water. B. After one minute and 2 minutes later, check if we can get the corresponding notification. 3. A. On the ESP32, connect pin VCC and pin EN to 3.3V input and pin GND and pin

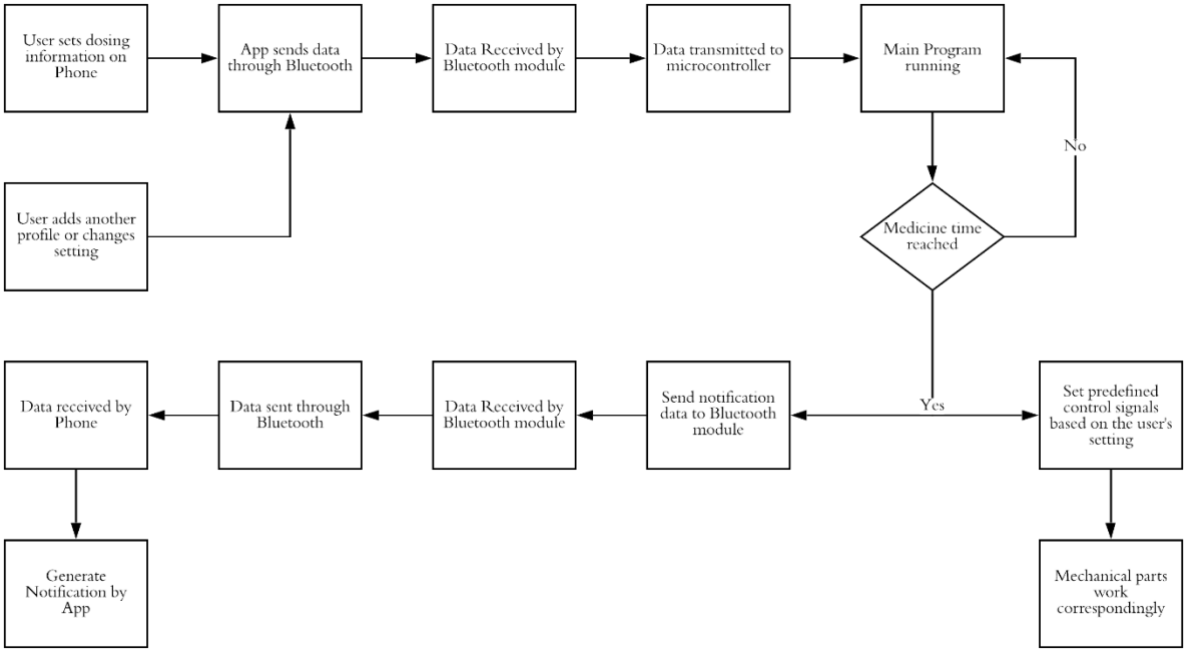
	IO0 to the GND. Then interconnect RX and TX to TX and RX on the serial-USB converter, connect LCD display to an Arduino. B. Upload the example program 'BLE_write" from the library and in loop() add lines of code to transmit the data to the Arduino. C. Disconnect the ESP32 with the serial-USB converter. D. On the phone, connect to "MyESP32", open our iOS app on the phone and click on the print button added specifically for debugging and testing. E. See if the "Hello World!" text appears on the LCD display.

2.3.5 Power Line

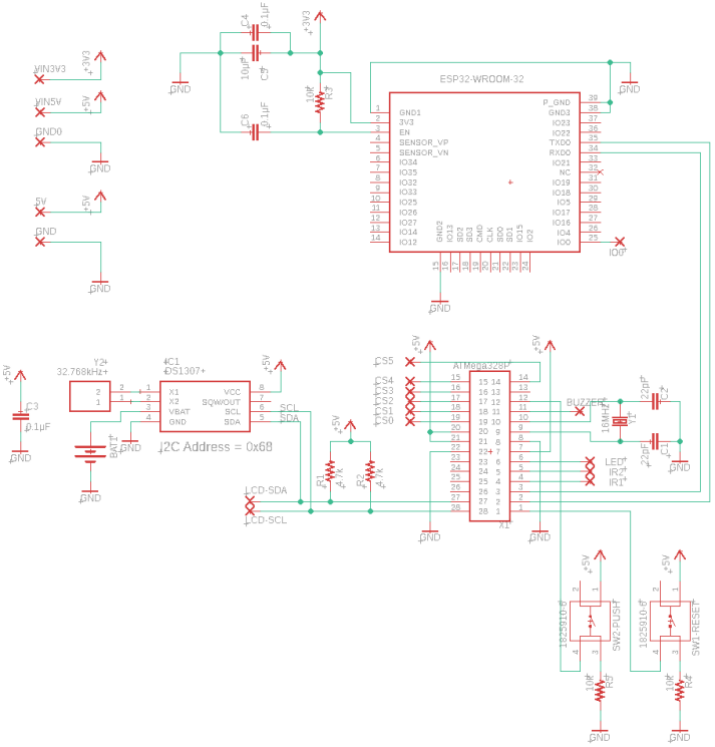
This unit is responsible for delivering power to the control module, motors and other peripherals.

Requirement	Verification
1. Power the components on the PCB (microcontroller, Bluetooth chip). Use the low-dropout voltage regulators to provides 5.5V +/- 5% from a 6-6.6V source and supply the current within 0-500mA 2. Power for peripherals (motors, LCD screen). Note: The power supply should be able to drive the mechanical parts including a 12V/2A stepper motor, 3 5V/0.5A servo motors, 3 5V/50mA IR sensors, a 24V/20mA keypad, a 3.5V/35mA buzzer and several 5V LEDs for now. For now, the power estimates to be not larger than 50W. Changes may apply due to the pending design of ME team.	1. A. Connect the AC adaptor and connect the output of the voltage regulator to the microcontroller and Bluetooth. B. The microcontroller and Bluetooth chip can be turned on. 2. A. Connect the AC adaptor. B. Connect the motors' power and set the control signals to see if motors work. C. Connect to other peripherals' (LCD display) power and see if they turn on.

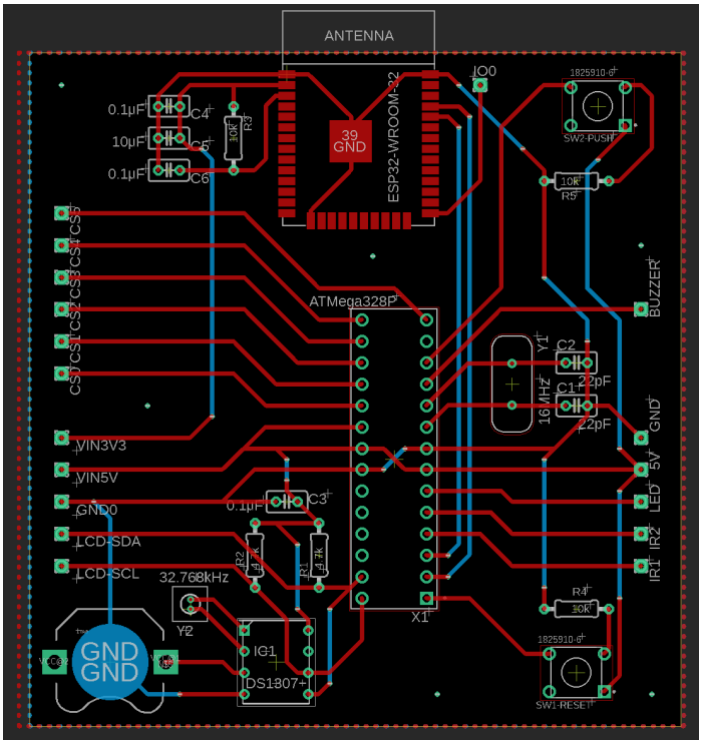
2.4 Data Flow


User sets dosing information on Phone
App sends data through Bluetooth
Data Received by Bluetooth module
Data transmitted to microcontroller
Main Program running
User adds another profile or changes setting
Medicine time reached
No
Yes
Data received by Phone
Data sent through Bluetooth
Data Received by Bluetooth module
Send notification data to Bluetooth module
Set predefined control signals based on the user's setting
Generate Notification by App
Mechanical parts work correspondingly

2.5 Schematics



2.6 PCB design



3. Cost and Schedule

3.1 Cost Analysis:

Labor:
Our development costs are roughly estimated to be $20 per hour, 15 hours of work per week for 3 people and 10 weeks in total that we can contribute to this project.

$$3 * 20\$/hr * 15hr/week * 10week * 2.5 = \$22{,}500$$

Parts:
Microcontroller (ATMega328P) - $2.08
Customized PCB – around $40
Power Supply - around $20
Bluetooth Module (ESP32/WRL-1367) - $6.95
LED - Given
Capacitors, resistors and wires – around $20
(Motors, drivers and other parts from ME team)
Total: $89.03

3.2 Schedule

	Qingyu Li	Wennan Zhai	Shengyu Ge
Week 1	PCB design	PCB design	PCB design
Week 2	Coding main program and Bluetooth set up with breadboard	Coding main program and generating correct control signals to any specific pin with breadboard	Generating correct control signals to any specific pin with breadboard and Bluetooth channel set up
Week 3	Unit test program output	Unit test circuit output	Unit test circuit output
Week 4	Start developing iOS app development	Integrating with ME team for various motors' control signals	Integrating with ME team for various motors' control signals
Week 5	iOS app development	iOS app development	Other collaboration details with ME team
Week 6	Any unfinished task from above and final integration	Any unfinished task from above and final integration	Any unfinished task from above and final integration
Week 7	Any unfinished task from above and final integration	Any unfinished task from above and final integration	Any unfinished task from above and final integration

Week 8	Prepare mock-up (if needed) and prototype refining	Prepare mock-up (if needed) and prototype refining	Prepare mock-up (if needed) and prototype refining

4. Tolerance Analysis

For the ESP32, we are looking for regulator that can output a minimum of 3V in order for the chip to work, as compared to the 3.3V specified on the datasheet and the minimum current will be 500mA. If we are using a linear regulator, then there will be a requirement for the resistor inside the linear regulator to be no more than:

$$R = \frac{36V - VCC}{0.5A} = 66\ \Omega$$

5. Ethics and Safety

There might be some potential safety problems with our projects. If the pill dispenser doesn't give pills at the proper time, it could be detrimental. Besides, little children might also find this device very interesting. They might take the pills as sugar and that could possibly have a negative effect on their health.

If this dispenser is exposed in a moist room, that will cause damage to the circuits. In order to avoid this, we need to tell the user to keep this device in dry area.

We thoroughly went over the 10 ethics mentioned on the IEEE Code of Ethics and we firmly believe that we will obey the rules of these ethics.

1. "to hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices, and to disclose promptly factors that might endanger the public or the environment;" [2]
    - Our project will not affect the safety of the public. It uses electricity as its main power supply, so it will not have a negative effect on the environment.
2. "to avoid real or perceived conflicts of interest whenever possible, and to disclose them to affected parties when they do exist;" [2]
    - Our project will not have conflict of interest and even if the conflict exists, we will inform the affected parties.

References

[1] Harris, Richard. "Federal Survey Finds 119 Million Americans Use Prescription Drugs." NPR, NPR, 8 Sept. 2016, www.npr.org/2016/09/08/493157917/federal-survey-finds-119-million-americans-use-prescription-drugs.

[2] Ieee.org, "IEEE IEEE Code of Ethics", 2016. [Online]. Available: http://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 29- Feb- 2016].