**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

TRƯỜNG ĐẠI HỌC BÁCH KHOA

KHOA ĐIỆN – ĐIỆN TỬ

**BỘ MÔN ĐIỆN TỬ**

---------------o0o---------------

****

**ĐỒ ÁN MÔN HỌC**

**THIẾT BỊ GIÁM SÁT**

**LỊCH TRÌNH SỬ DỤNG THUỐC**

**GVHD: Lưu Phú**

**Sinh viên: MSSV:**

**Phan Văn Đức 1811981**

**Nguyễn Văn Đức 1811974**

**TP. HỒ CHÍ MINH, THÁNG 06 NĂM 2021**

***LỜI CẢM ƠN***

Chúng em xin gửi lời cảm ơn chân thành đến các thầy cô trường Đại học Bách Khoa TP. Hồ Chí Minh, các thầy cô khoa Điện – Điện tử của trường đã tạo điều kiện cho chúng em được thực hiện đồ án.

Trong thời gian học tập tại trường, chúng em đã thu được rất nhiều kiến thức và bài báo cáo này là kết quả của quá trình học tập, rèn luyện và nghiên cứu dưới sự dạy bảo của quý thầy cô. Đặc biệt, chúng em xin gửi lời cảm ơn chân thành đến Thầy Lưu Phú , người đã tận tình hướng dẫn và đóng góp ý kiến trong thời gian qua, giúp chúng em hoàn thành báo cáo một cách tốt nhất. Tuy nhiên, do kinh nghiệm bản thân còn hạn chế cũng như kiến thức còn hạn hẹp nên bài báo cáo không thể tránh khỏi những thiếu sót, chúng em rất mong nhận được những đóng góp của quý thầy cô để chúng em tiếp thu chỉnh sửa, tích lỹ thêm nhiều kinh nghiệm cho bản thân và sẽ hoàn thành tốt hơn ở những dự án sau này.

Cuối cùng, chúng em xin chúc quý thầy cô dồi dào sức khoẻ và thành công trong sự nghiệp của mình. Kính chúc Thầy Lưu Phú luôn có sức khoẻ tốt, đạt được nhiều thành công trong công việc và trong cuộc sống.

Chúng em xin chân thành cảm ơn!

*Tp. Hồ Chí Minh, ngày 10 tháng 06 năm 2021*

**TÓM TẮT ĐỒ ÁN**

Hiện nay lĩnh vực y tế đang đối mặt với các vấn đề liên quan đến thuốc rất phổ biến ở người cao tuổi, đó là sự không hiệu quả của thuốc, tác động phụ của thuốc, dùng quá liều, dùng chưa đủ liều thuốc và tương tác thuốc. Trong đó, nguyên nhân chủ quan ảnh hưởng rất lớn đến hiệu quả của thuốc. Sự tuân thủ điều trị bị ảnh hưởng bởi nhiều yếu tố, đặc biệt là do tuổi tác.

Năm bắt được vấn đề trên, cùng với các kiến thức hiện có, chúng em đã tiến hành nghiên cứu về nhu cầu thực tế, tham khảo ý kiến thầy cô về tính khả thi và thực hiện dự án Thiết bị giám sát lịch sử dụng thuốc. Thiết bị gồm 4 ngăn kéo chứa thuốc có khoá. Thiết bị sẽ chỉ mở ngăn thuốc tại thời điểm được cài đặt sẵn, đồng thời tín hiệu đến người dùng, giúp họ sử dụng thuốc một cách hiệu quả nhất.

Nội dung của báo cáo sẽ trình bày về cách thức hoạt động của thiết bị cũng như những thành phần cấu tạo nên thiết bị, bao gồm cả phần cứng lẫn phần mềm. Báo cáo nhằm mục đích tạo ra cái nhìn tổng quát về vấn đề, những chức năng và thiếu sót của thiết bị để từ đó đưa ra những giải pháp cần thiết để cải tiến hoặc đưa ra hướng đi mới cho thiết bị.

**Báo cáo gồm 4 phần chính:**

Phần 1 của báo cáo sẽ trình bày về những vấn đề và định hướng của đề tài, tiếp đó là giới thiệu cơ bản về những thiết bị, phần mềm để tạo nên thiết bị. Phần này gồm 2 chương:

* Chương 1: Giới thiệu về tác hại và nguyên nhân của việc sử dụng thuốc không đúng lịch trình của người cao tuổi. Qua đó đưa ra giải pháp là thiết kế thiết bị hỗ trợ người cao tuổi. Chương này còn bao gồm nhiệm vụ đề tài và phân chia công việc.
* Chương 2: Bao gồm lý thuyết về các thiết bị phần cứng, ứng dụng phần mềm phục vụ cho việc thiết kế sản phẩm.

Phần 2 sẽ là thiết kế phần cứng và phần mềm, tạo ra thiết bị hoàn chỉnh. Phần này gồm 2 chương:

* Chương 3: Thiết kế mô phỏng phần cứng trên phần mềm Proteus và thực hiện phần cứng sau khi thiết kế, ngoài ra còn có phần thiết kế mô hình sản phẩm.
* Chương 4: Thiết kế phần mềm dựa trên các công cụ: Keil uVision 5, Aduino IDE, STM32 CubeMx, Blynk.

Phần 3 sẽ trình bày kết quả thực hiện và đưa ra đánh giá về sản phẩm. Phần này gồm 2 chương:

* Chương 5: Trình bày kết quả thực hiện, cách thức đo đạc và đánh giá sản phẩm.
* Chương 6: Đánh giá ưu, khuyết điểm của sản phẩm, qua đó đánh giá tính thực tiễn và hướng phát triển của sản phẩm.

**MỤC LỤC**

[1. GIỚI THIỆU 1](#_Toc73804544)

[1.1 Tổng quan 1](#_Toc73804545)

[1.2 Nhiệm vụ đề tài 2](#_Toc73804546)

[1.3 Phân chia công việc trong nhóm 3](#_Toc73804547)

[2. LÝ THUYẾT 5](#_Toc73804548)

[2.1 Thiết bị phần cứng 5](#_Toc73804549)

[2.1.1 Kit phát triển STM32F103C8T6 Blue Pill ARM Cortex-M3 5](#_Toc73804550)

[2.1.2 Mạch thu phát Wifi ESP8266 UART ESP-01S Ai-Thinker 8](#_Toc73804551)

[2.1.3 L9110S 4 kênh điều khiển động cơ DC 10](#_Toc73804552)

[2.2 Công cụ phần mềm 11](#_Toc73804553)

[2.2.1 Keil C uVision 5 11](#_Toc73804554)

[2.2.2 STM32CubeMX 12](#_Toc73804555)

[2.2.3 Arduino IDE 13](#_Toc73804556)

[2.2.4 Blynk App 14](#_Toc73804557)

[2.2.5 Proteus 8 15](#_Toc73804558)

[2.2.6 Ultra Serial Port Monitor 15](#_Toc73804559)

[3 THIẾT KẾ VÀ THỰC HIỆN PHẦN CỨNG 16](#_Toc73804560)

[3.2 Yêu cầu thiết kế 16](#_Toc73804561)

[3.3 Phân tích thiết kế 17](#_Toc73804562)

[3.4 Sơ đồ khối 20](#_Toc73804563)

[3.5 Sơ đồ mạch chi tiết 22](#_Toc73804564)

[4 THIẾT KẾ VÀ THỰC HIỆN PHẦN MỀM 27](#_Toc73804565)

[4.1 Yêu cầu đặt ra cho phần mềm 27](#_Toc73804566)

[4.2 Phân tích 28](#_Toc73804567)

[4.3 Lưu đồ giải thuật tổng quát 28](#_Toc73804568)

[4.4 Lưu đồ giải thuật chi tiết 28](#_Toc73804569)

[5 KẾT QUẢ THỰC HIỆN 28](#_Toc73804570)

[6 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 29](#_Toc73804571)

[6.1 Kết luận 29](#_Toc73804572)

[6.2 Hướng phát triển 29](#_Toc73804573)

[7 TÀI LIỆU THAM KHẢO 29](#_Toc73804574)

[8 PHỤ LỤC 31](#_Toc73804575)

DANH SÁCH HÌNH MINH HỌA

[Hình 1: Sơ đồ chân của STM32F103C8T6 Blue Pill 7](file:///E:\Đồ%20án\Báo%20cáo%20Thiết%20bị%20giám%20sát%20lịch%20sử%20dụng%20thuốc.docx#_Toc73804596)

[Hình 2: Mô tả chức năng bảng mạch 8](file:///E:\Đồ%20án\Báo%20cáo%20Thiết%20bị%20giám%20sát%20lịch%20sử%20dụng%20thuốc.docx#_Toc73804597)

[Hình 3: Mạch thu phát wifi ESP8266-01S 8](file:///E:\Đồ%20án\Báo%20cáo%20Thiết%20bị%20giám%20sát%20lịch%20sử%20dụng%20thuốc.docx#_Toc73804598)

[Hình 4: Sơ đồ chân ESP8266-01S 9](file:///E:\Đồ%20án\Báo%20cáo%20Thiết%20bị%20giám%20sát%20lịch%20sử%20dụng%20thuốc.docx#_Toc73804599)

[Hình 5: L9110S 4 kênh điều khiển 10](file:///E:\Đồ%20án\Báo%20cáo%20Thiết%20bị%20giám%20sát%20lịch%20sử%20dụng%20thuốc.docx#_Toc73804600)

[Hình 6: Phần mềm Keil uVision 5 11](#_Toc73804601)

[Hình 7: Phần mềm STM32CubeMx 12](#_Toc73804602)

[Hình 8: Biểu tượng Arduino IDE 13](#_Toc73804603)

[Hình 9: Blynk và ứng dụng 14](#_Toc73804604)

[Hình 10: Phần mềm Proteus 8 15](#_Toc73804605)

[Hình 11: Biểu tượng Ultra Serial Port Monitor 15](#_Toc73804606)

[Hình 12: Sơ đồ khối chi tiết 20](#_Toc73804607)

[Hình 13: Các khối chức năng 22](#_Toc73804608)

[Hình 14: Thiết kế Reset 22](#_Toc73804609)

[Hình 15: Thiết kế Wifi 23](#_Toc73804610)

[Hình 16: Sensor by button 23](#_Toc73804611)

[Hình 17: Thiết kế LCD 24](#_Toc73804612)

[Hình 18: Khối Khoá 25](#_Toc73804613)

[Hình 19: Khối Buzzer 26](#_Toc73804614)

[Hình 20: Thiết kế Led 26](#_Toc73804615)

**DANH SÁCH BẢNG SỐ LIỆU**

[Bảng 1: Phân chia công việc 3](#_Toc73804616)

[Bảng 2: Các loại khoá 18](#_Toc73804617)

[Bảng 3: Các phương pháp chọn cảm biến 19](#_Toc73804618)

[Bảng 4: Danh sách phần cứng 19](#_Toc73804619)

[Bảng 5: Yêu cầu lựa chọn Vi điều khiển 20](#_Toc73804620)

[Bảng 6: Giao tiếp LCD với Vi điều khiển 24](#_Toc73804621)

[Bảng 7 Thông số hệ thống 29](#_Toc73804622)

# GIỚI THIỆU

## Tổng quan

Trong quá trình điều trị bệnh bằng thuốc, việc gặp phải tác dụng phụ của thuốc là không thể tránh khỏi. Những tác dụng phụ của thuốc được kể đến như an thần quá mức, nhầm lẫn, ảo giác, té ngã, và xuất huyết. Những tác dụng phụ ấy có thể chỉ gây không thoải mái, nhưng cũng có thể gây nguy hiểm. Trong số người bệnh ≥ 65 tuổi, tác dụng phụ bất lợi của thuốc xảy ra với tỉ lệ khoảng 50 lần/ 1000 người-năm. Tỷ lệ nhập viện do các tác dụng bất lợi của thuốc cao hơn 4 lần ở bệnh nhân cao tuổi (about khoảng 17%) so với ở bệnh nhân trẻ tuổi (4%) [1]. Những nguyên nhân gây ra tác dụng phụ của thuốc bao gồm sự không hiệu quả của thuốc, tác động phụ của thuốc, dùng quá liều, dùng chưa đủ liều thuốc và tương tác thuốc. Tuy nhiên, đối với người cao tuổi, nguyên nhân chủ quan (tuân thủ điều trị kém) là một vấn đề lớn, ảnh hưởng trực tiếp đến hiệu quả của thuốc. Có tới một nửa số bệnh nhân cao tuổi không dùng thuốc đúng theo chỉ dẫn, thường là dùng thuốc dưới mức quy định. Ngoài ra, còn do các yếu tố sau đây:

* Các vấn đề về nhận thức, có thể làm cho việc uống thuốc theo hướng dẫn khó khăn.
* Sử dụng quá nhiều loại thuốc.
* Việc sử dụng các loại thuốc phải uống nhiều lần trong ngày hoặc theo cách cụ thể cho từng loại
* Không có sự giúp đỡ thường xuyên của người có chuyên môn, hoặc người thân.

Như vậy, sau khi xem xét các nguyên nhân, chúng em đưa ra các hướng giải pháp giúp nhắc nhở người cao tuổi sử dụng thuốc, cụ thể là nghiên cứu, chế tạo thiết bị giúp hỗ trợ người cao tuổi sử dụng thuốc một cách chính xác, hiệu quả: Thiết bị giám sát lịch trình sử dụng thuốc. Để thực hiện được mục đích trên, thiết bị cần đáp ứng được các vấn đề sau:

* Nhập dữ liệu về thời gian sử dụng thuốc, số lượng thuốc, số lượng thuốc sử dụng mỗi lần qua máy tính (trực tiếp) và qua thiết bị điện thoại có wifi (gián tiếp).
* Phát tín hiệu cảnh báo tới người dùng mỗi khi tới thời gian uống thuốc (thời gian có sai lệch không quá 1giây so với thời gian xác định).
* Có khoá đóng mở, LCD hiển thị ngăn thuốc được sử dụng.
* Có khả năng kiểm soát số lượng thuốc được sử dụng và báo cáo về thiết bị điện thoại.

Ngoài ra, thiết bị cần phải có những yêu cầu phụ như:

* Thiết bị nhỏ gọn, dễ sử dụng.
* Có khả năng lưu trữ, bảo quản thuốc.
* Thiết bị có giá từ 400.000đ đến 500.000đ.
* Có độ bền cao.

## Nhiệm vụ đề tài

Để thiết bị đạt được những giá trị cao về mặt tiện ích, chúng em đã tiến hành nghiên cứu kỹ lưỡng về các vấn đề sau:

Nội dung 1: Tìm hiểu và đề ra nguyên lý hoạt động của thiết bị sao cho phù hợp với chức năng đề ra, đồng thời tìm hiểu về phương thức truyền nhận dữ liệu giữa điện thoại và thiết bị

Nội dung 2: Lựa chọn các thiết bị phần cứng, bao gồm:

* Lựa chọn vi xử lý trung tâm (ưu tiên loại vi xử lý có khả năng giao tiếp qua wifi, có tích hợp chức năng thời gian thực).
* Thiết bị giao tiếp qua wifi, thiết bị thời gian thực (nhỏ gọn nếu không được tích hợp trong vi điều khiển)
* Khoá hộp thuốc: nhỏ gọn, có khả gắn vào từng ngăn thuốc, đảm bảo đóng mở chính xác.

Nội dung 3: Tìm hiểu về cảm biến, vi xử lý, các module điều khiển:

* Kit phát triển STM32F103C8T6 Blue Pill ARM Cortex-M3: phương pháp lập trình, xuất/nhập ở các port, truyền nhận dữ liệu bằng UART, chức năng thời gian thực RTC.
* Mạch thu phát Wifi ESP8266 UART ESP-01S Ai-Thinker: tìm hiểu cách thức lập trình giao tiếp giữa dòng vi điều khiển STM32 với sever Blynk.
* Module L9110S 4 kênh điều khiển động cơ DC.

Nội dung 4: Thiết kế phần cứng, mô hình

* Từ các linh kiện được lựa chọn, thiết kế đủ chức năng cho thiết bị, đảm bảo về công suất tiêu thụ của các linh kiện.
* Thiết kế mô hình hộp thuốc nhỏ gọn, chắc chắn.

Nội dung 5: Tìm hiểu về các phần mềm hỗ trợ lập trình và lập trình.

* Keil C uVision 5: lập trình cho vi điều khiển STM32F103C8T6.
* STM32CubeMX: công cụ hỗ trợ cấu hình và sinh code cho MCU STM32
* Arduino IDE: trình soạn thảo văn bản giúp viết code để nạp vào Mạch thu phát Wifi ESP8266.
* Blynk: ứng dụng trên điện thoại hỗ trợ cho các thiết bị IoT.
* Ultra Serial Port Monitor: phần mềm hỗ trợ truyền dữ liệu qua máy tính.
* Proteus 8: Phần mềm thiết kế mô phỏng phần cứng.

## Phân chia công việc trong nhóm

Bảng 1: Phân chia công việc

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ngày  (năm 2021) | Nội dung họp | Phân chia công việc | | Thời hạn |
| Nguyễn Văn Đức | Phan Văn Đức |
| 30/03 | Chọn đề tài, xác định chính xác yêu cầu của đề tài, lập sơ đồ khối chức năng. | Cùng nhau thực hiện. | |  |
| 10/04 | - Lựa chọn các linh kiện phần cứng  - Vẽ sơ đồ mạch phần cứng. | Lựa chọn linh kiện. | Vẽ mạch phần cứng. | 17/04 |
| 17/04 | - Tìm hiểu về các phần mềm hỗ trợ lập trình.  - Tiếp tục hoàn thiện sơ đồ mạch. | Tìm hiểu về vi điều khiển STM32 và công cụ lập trình Keil C. | - Chỉnh sử mạch phần cứng.  - Tìm hiểu về Blynk app. | 20/04 |
| 24/04 | - Hoàn chỉnh sơ đồ mạch.  - Tiếp tục tìm hiểu về các công cụ lập trình. | - Hoàn chỉnh sơ đồ mạch.  - Tìm hiểu về ESP8266 và Adruino IDE. | - Hoàn chỉnh sơ đồ mạch.  - Tìm hiểu về STM32CubeMX. | 27/04 |
| 27/04 | - Lập lưu đồ giải thuật, phân chia công việc lập trình. | - Vẽ lưu đồ giải thuật  - Lập trình giao tiếp vi điều khiển qua ESP8266 và Blynk.  - Nắm nhiệm vụ chính trong lập trình. | - Vẽ lưu đồ giải thuật  - Lập trình giao tiếp uart qua máy tính.  - Tìm hiểu phương pháp kiểm soát số lượng thuốc. | 04/05 |
| 04/05 | - Thống nhất phương pháp khoá ngăn thuốc, kiểm soát số lượng thuốc. | Tiếp tục công việc lập trình | Tiếp tục công việc lập trình |  |
| 15/05 | - Thiêt kế mạch in, hàn linh kiện.  - Thiết kế mô hình. | Thiết kế mạch và hàn linh kiện. | Thiết kế mô hình hộp thuốc. | 22/05 |
| 25/05 | Hoàn chỉnh phần mềm |  | |  |
| … | Chạy thử nghiệm, |  | |  |
|  |  |  |  |  |

# LÝ THUYẾT

## Thiết bị phần cứng

### Kit phát triển STM32F103C8T6 Blue Pill ARM Cortex-M3

**Về vi điều khiển STM32F103C8T6**

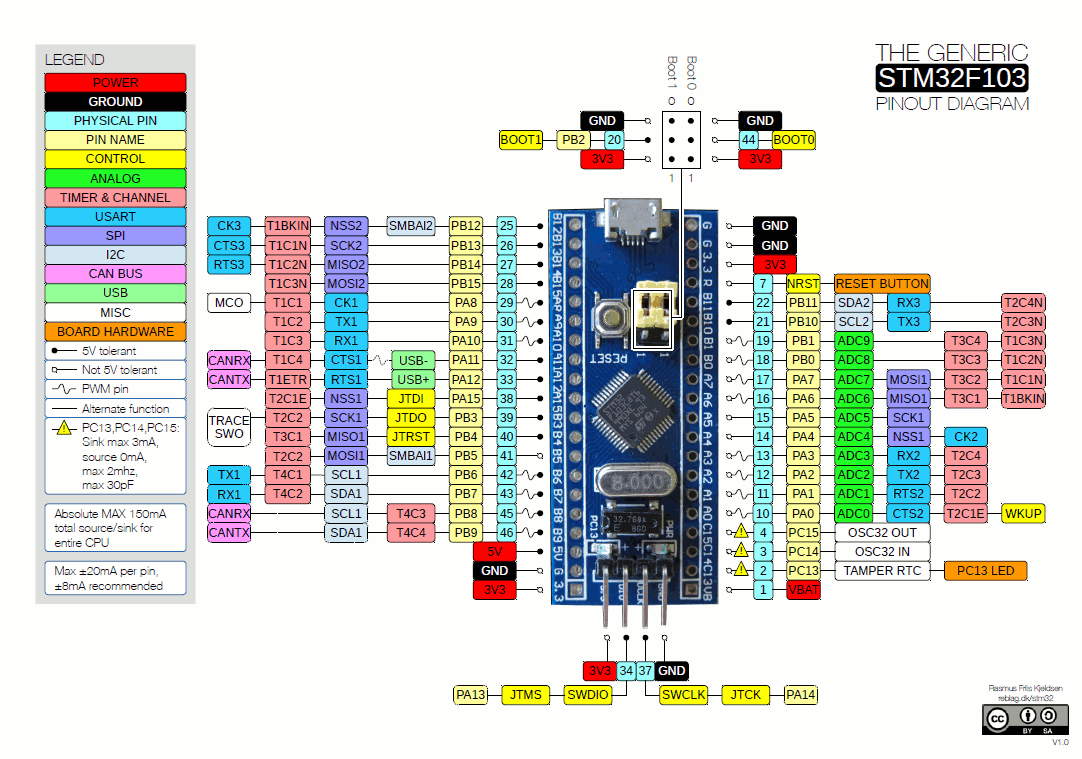
STM32 là một trong những dòng chip phổ biến của ST với nhiều họ thông dụng như F0, F1, F2, F3, F4, …Stm32F103 thuộc họ F1 với lõi là ARM COTEX M3. STM32F103 là vi điều khiển 32 bit, tốc độ tối đa là 72Mhz.

Cấu hình chi tiết của STM32F103C8T6:

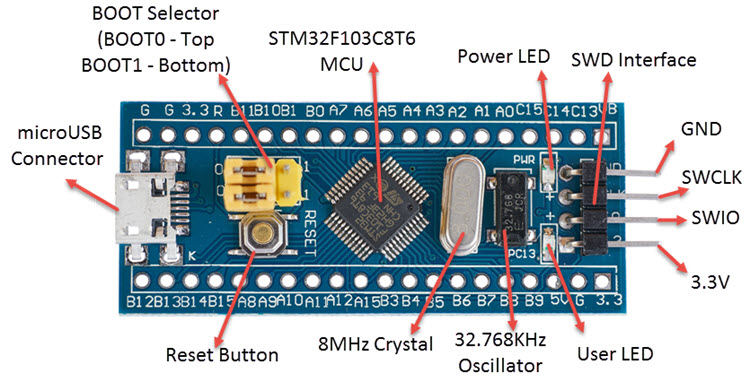
* ARM 32-bit Cortex M3 với clock max là 72Mhz.
* Bộ nhớ:
* 64 kbytes bộ nhớ Flash (bộ nhớ lập trình).
* 20kbytes SRAM.
* Clock, reset và quản lý nguồn.
* Điện áp hoạt động 2.0V -> 3.6V.
* Power on reset (POR), Power down reset (PDR) và programmable voltage detector (PVD).
* Sử dụng thạch anh ngoài từ 4Mhz -> 20Mhz.
* Thạch anh nội dùng dao động RC ở mode 8Mhz hoặc 40khz.
* Sử dụng thạch anh ngoài 32.768khz được sử dụng cho RTC.
* Trong trường hợp điện áp thấp:
* Có các mode: ngủ, ngừng hoạt động hoặc hoạt động ở chế độ chờ.
* Cấp nguồn ở chân Vbat bằng pin để hoạt động bộ RTC và sử dụng lưu trữ data khi mất nguồn cấp chính.
* 2 bộ ADC 12 bit với 9 kênh cho mỗi bộ.
* Khoảng giá trị chuyển đổi từ 0 – 3.6V.
* Lấy mẫu nhiều kênh hoặc 1 kênh.
* Có cảm biến nhiệt độ nội.
* DMA: bộ chuyển đổi này giúp tăng tốc độ xử lý do không có sự can thiệp quá sâu của CPU.
* 7 kênh DMA.
* Hỗ trợ DMA cho ADC, I2C, SPI, UART.
* TIMER: gồm 7 Timer
* 3 TIMER 16 bit hỗ trợ các mode IC/OC/PWM.
* 1 timer 16 bit hỗ trợ để điều khiển động cơ với các mode bảo vệ như ngắt input, dead-time…
* 2 watdog timer dùng để bảo vệ và kiểm tra lỗi.
* 1 sysTick timer 24 bit đếm xuống dùng cho các ứng dụng như hàm Delay….
* Hỗ trợ 9 kênh giao tiếp bao gồm:
* 2 bộ I2C(SMBus/PMBus).
* 3 bộ USART (ISO 7816 interface, LIN, IrDA capability, modem control).
* 2 SPIs (18 Mbit/s).
* 1 bộ CAN interface (2.0B Active).
* USB 2.0 full-speed interface
* Kiểm tra lỗi CRC và 96-bit ID.
* Phần mềm lập trình: có khá nhiều trình biên dịch cho STM32 như IAR Embedded Workbench, Keil C…
* Thư viện lập trình: STM32snippets, STM32Cube LL, STM32Cube HAL, Standard Peripheral Libraries, Mbed core.

**Về Kit Phát Triển STM32F103C8T6 BluePill**

KIT STM32F103C8T6 Mini thuộc loại kit phát triển là Kit phát triển được thiết kế với đơn giản, kít ra đầy đủ chân của vi điều khiển, có cổng giao tiếp USB và cổng nạp SWD, sử dụng dòng vi điều khiển 32 Bit của dòng ST.

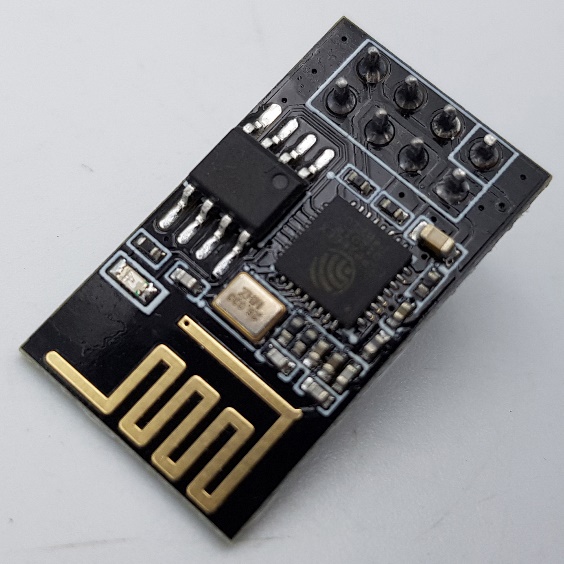
Thông số kỹ thuật:

Hình : Sơ đồ chân của STM32F103C8T6 Blue Pill

* Vi điều khiển: STM32F103C8T6.
* Điện áp cấp 5VDC qua cổng Micro USB sẽ được chuyển đổi thành 3.3VDC qua IC nguồn và cấp cho Vi điều khiển chính.
* Tích hợp sẵn thạch anh 8Mhz.
* Tích hợp sẵn thạnh anh 32Khz cho các ứng dụng RTC.
* Ra chân đầy đủ tất cả các GPIO và giao tiếp: CAN, I2C, SPI, UART, USB,…
* Tích hợp Led trạng thái nguồn, Led PC13, Nút Reset.
* Kích thước: 53.34 x 15.24mm
* Sử dụng với các mạch nạp:
* ST-Link Mini
* J-link
* USB TO COM
* Kết nối chân khi nạp bằng ST-Link Mini
* Nạp theo chuẩn SWD
* TCK — SWCLK
* TMS — SWDIO
* GND — GND
* 3.3V — 3.3V

Hình : Mô tả chức năng bảng mạch

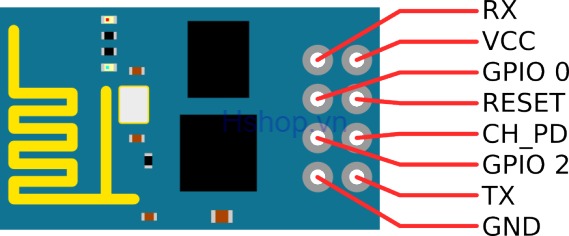
### Mạch thu phát Wifi ESP8266 UART ESP-01S Ai-Thinker

Mạch thu phát Wifi ESP8266 UART ESP-01S Ai-Thinker được sản xuất bởi Ai-Thinker sử dụng IC Wifi SoC ESP8266 của hãng Espressif, được sử dụng để kết nối với vi điều khiển thực hiện chức năng truyền nhận dữ liệu qua Wifi, mạch có thiết kế nhỏ gọn, sử dụng giao tiếp UART trong các ứng dụng IoT và điều khiển thiết bị qua Wifi

Hình : Mạch thu phát wifi ESP8266-01S

Thông số kỹ thuật:

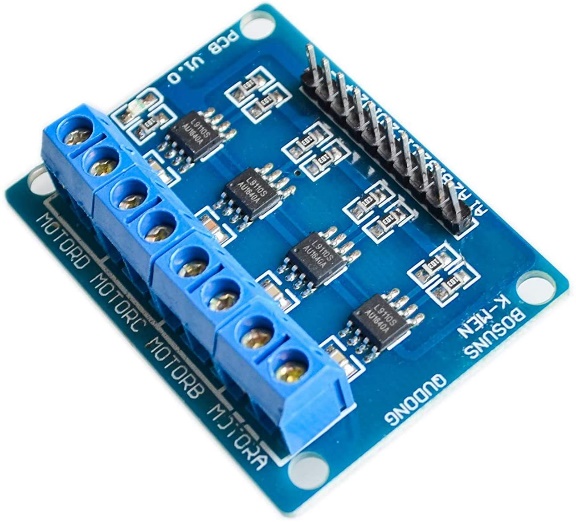
* Model: ESP8266 UART ESP-01S Ai-Thinker
* Điện áp sử dụng: 3.0V~3.6V (Optimal 3.3V)
* Dòng tiêu thụ: Max 320mA (nên sử dụng [module cấp nguồn riêng](http://hshop.vn/products/mach-giamop-3-3v-ams1117) cho mạch).
* Hỗ trợ chuẩn 802.11 b/g/n.
* Wi-Fi 2.4 GHz, hỗ trợ các chuẩn bảo mật như: OPEN, WEP, WPA\_PSK, WPA2\_PSK, WPA\_WPA2\_PSK.
* Hỗ trợ cả 2 giao tiếp TCP và UDP.
* Chuẩn giao tiếp UART với Firmware hỗ trợ bộ tập lệnh AT Command, tốc độ Baudrate mặc định 9600 hoặc 115200.
* Có 3 chế độ hoạt động: Client, Access Point, Both Client and Access Point.
* Kích thước: 24.8 x 14.3mm

Chức năng các chân:

Hình : Sơ đồ chân ESP8266-01S

1. **VCC: 3.3V, dòng có thể lên 300mA vì thế cần mạch nguồn riêng.**
2. GND: 0V.
3. Tx: Chân Tx của giao thức UART, kết nối đến chân Rx của vi điều khiển.
4. Rx: Chân Rx của giao thức UART, kết nối đến chân Tx của vi điều khiển.
5. RST:  chân reset, kéo xuống mass để reset.
6. CH\_PD: **kéo lên mức cao module sẽ bắt đầu thu phát wifi, kéo xuống mức thấp module dừng phát wifi.**
7. GPIO0: kéo xuống thấp cho chế độ upgrade firmware.
8. GPIO2: không sử dụng**.**

### L9110S 4 kênh điều khiển động cơ DC

 L9110S là một mạch điều khiển động cơ DC sử dụng IC L9110S, có thể điều khiển 4 động cơ DC cùng lúc để thực hiện chuyển động quay thuận, quay ngược và điều khiển tốc độ động cơ. Mạch được thiết kế nhỏ gọn, hoạt động ổn định, tiêu thụ điện năng thấp.

Thông số kỹ thuật:

* Chip sử dụng: L9110S

Hình : L9110S 4 kênh điều khiển

* Điện áp cấp: 2.5 - 12V
* Dòng tải: 0.8A/Kênh
* Sử dụng điều khiển 4 động cơ DC hoặc 2 động cơ bước 2 phase 4 dây

Mô tả 10 chân đen:

* + 1. A1 cổng IO điều khiển Motor A
    2. A2 cổng IO điều khiển Motor A
    3. B1 cổng IO điều khiển Motor B
    4. B2 cổng IO điều khiển Motor B
    5. C1 cổng IO điều khiển Motor C
    6. C2 cổng IO điều khiển Motor C
    7. D1 cổng IO điều khiển Motor D
    8. D2 cổng IO điều khiển Motor D
    9. + nối nguồn 2.5V-12V
    10. – nối GND

Mô tả cổng ra (8 chân):

* + 1. MOTORA kết nối DC Motor 2 chân
    2. MOTORB kết nối DC Motor 2 chân
    3. MOTORC kết nối DC Motor 2 chân
    4. MOTORD kết nối DC Motor 2 chân

## Công cụ phần mềm

### Keil C uVision 5

Keil MicroVision là một phần mềm miễn phí giải quyết nhiều vấn đề khó khăn cho một nhà phát triển chương trình nhúng. Phần mềm này là một môi trường phát triển tích hợp (IDE), tích hợp một trình soạn thảo văn bản để viết chương trình, một trình biên dịch và nó cũng sẽ chuyển đổi mã nguồn sàng các tệp hex. Keil uVision có những chức năng cơ bản sau:

* Viết chương trình bằng C / C ++ hoặc hợp ngữ.
* Chương trình biên dịch và lắp ráp.
* Chương trình gỡ lỗi.
* Tạo tệp Hex và Axf.
* Kiểm tra chương trình mà không cần Phần cứng thực (Chế độ giả lập).



Hình 6: Phần mềm Keil uVision 5

Sử dụng Keil C uVision 5 cho lập trình STM32F103C8 ta cần cấu hình một số chức năng theo các bước sau:

1. Tải và cài đặt Keil MDK uVision5 tại:

<https://www.keil.com/download/product/>

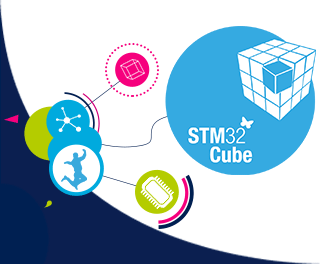
1. Cài đặt gói STM32 phục vụ cho lập trình và biên dịch, sửa lỗi.
2. Cấu hình cho Keil C: Chọn thiết bị cần lập trình trong Keil C, lựa chọn đầu ra của file lập trình, thiết bị nạp chương trình, …
3. Tạo file lập trình bằng ngôn nhữ C, Sau đó lập trình.
4. Biên dịch chương trình, sửa lỗi và nạp chương trình. [2]

Để có cái nhìn tổng quan về cách sử dụng, hãy tham khảo tại liên kết:

<http://www.ocfreaks.com/create-new-stm32-project-keil-uvision-5-tutorial/>

### STM32CubeMX

STM32CubeMX là một công cụ đồ họa cho phép cấu hình rất dễ dàng bộ vi điều khiển và bộ vi xử lý STM32, cũng như tạo mã khởi tạo C tương ứng cho lõi Arm Cortex-M hoặc một phần Linux Device Tree cho lõi Arm Cortex.



Hình 7: Phần mềm STM32CubeMx

Liên kết tải phần mềm:

<https://www.st.com/en/development-tools/stm32cubemx.html#get-software>

Các bước sử dụng STM32 CubeMx:

1. Bước đầu tiên bao gồm chọn: vi điều khiển STMicrolectronics STM32, vi xử lý hoặc nền tảng phát triển phù hợp.
2. Bước thứ hai cho phép định cấu hình GPIO và thiết lập Clock cho toàn bộ hệ thống, đồng thời gán cấu hình các thiết bị ngoại vi cho Arm Cortex-M.
3. Bước thứ 3, người dùng chuyển đổi những cấu hình đã chọn thành mã khởi tạo C cho Arm Cortex-M, sẵn sàng được sử dụng trong một số môi trường phát triển như Keil C uVision 5, … [3]

Để có cái nhìn tổng quan về cách sử dụng, hãy tham khảo tại liên kết:

<https://www.electronicshub.org/program-stm32f103c8t6-using-keil-uvision/>

### Arduino IDE

Arduino IDE là một phần mềm mã nguồn mở chủ yếu được sử dụng để viết và biên dịch mã vào module Arduino. Đây là một phần mềm Arduino chính thức, giúp cho việc biên dịch mã trở nên dễ dàng mà ngay cả một người bình thường không có kiến thức kỹ thuật cũng có thể làm được. Nó có các phiên bản cho các hệ điều hành như MAC, Windows, Linux và chạy trên nền tảng Java đi kèm với các chức năng và lệnh có sẵn đóng vai trò quan trọng để gỡ lỗi, chỉnh sửa và biên dịch mã trong môi trường.



Hình 8: Biểu tượng Arduino IDE

Để viết chương trình cho mạch thu phát Wifi ESP8266, chúng ta cần:

1. Tải phần mềm Aduino IDE (đối với máy tính sử dụng hệ điều hành Window, mở Microsoft Store, tìm kiếm phần mềm Aduino IDE và cài đặt)
2. Tạo file lập trình mới và tiến hành lập trình
3. Biên dịch và sửa lỗi.
4. Nạp chương trình: nạp chương trình cho ESP8266-01 thông qua board mạch Adruino Uno, tham khảo tại liên kết: [4]

<https://create.arduino.cc/projecthub/pratikdesai/how-to-program-esp8266-esp-01-module-with-arduino-uno-598166>

### Blynk App

Blynk là một phần mềm mã nguồn mở được thiết kế cho các ứng dụng IoT (Internet of Things). Ứng dụng giúp người dùng điều khiển phần cứng từ xa, có thể hiển thị dữ liệu cảm biến, lưu trữ dữ liệu, biến đổi dữ liệu hoặc làm nhiều việc khác.



Hình 9: Blynk và ứng dụng

Nền tảng Blynk bao gồm 3 phần chính:

* Blynk App – Ứng dụng Blynk cho phép khởi tạo giao diện cho các dự án.
* Blynk Server – Chịu trách nhiệm giao tiếp qua lại hai chiều  giữa điện thoại và phần cứng.
* Blynk Library – Thư viện chứa các nền tảng phổ biến , giúp việc giao tiếp phần cứng với Server dễ dàng hơn

Sử dụng Blynk:

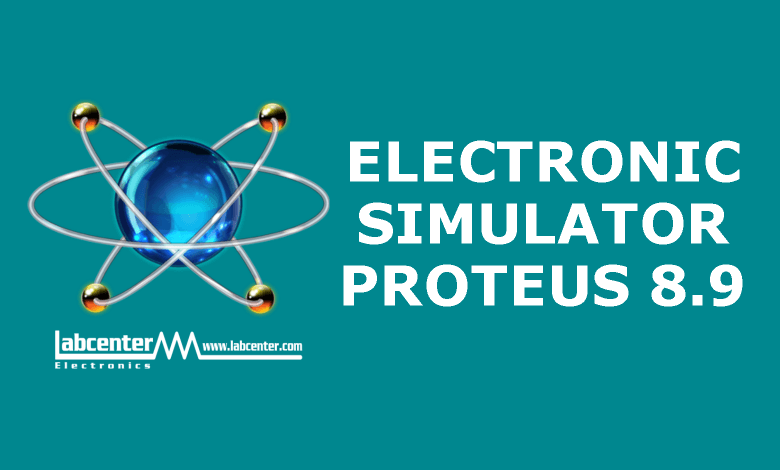
1. Tải Blynk về điện thoại với thiết bị Android.
2. Mở app Blynk, tạo tài khoản mới.
3. Tạo một project mới, chọn vi điều khiển phần cứng.
4. Thêm các thiết bị vào project, cấu hình từng thiết bị.
5. Chạy chương trình. [5]

Để có cái nhìn tổng quan về cách sử dụng, hãy tham khảo tại liên kết:

<https://docs.blynk.cc/>

### Proteus 8

Proteus 8 Professional là phần mềm dùng để vẽ lược đồ, phác thảo PCB và mô phỏng sơ đồ. Phần mềm này được phát triển bởi Labcenter Electronic Ltd. Phần mềm có thể mô tả hầu hết các [linh](https://chotroihn.vn/) kiện điện tử thông dụng hiện nay. Proteus có khả năng mô phỏng hoạt động của các mạch điện tử bao gồm phần thiết như kế mạch và viết trình điều khiển cho các loại vi điều khiển như MCS-51, AVR, PIC… [6]



Hình 10: Phần mềm Proteus 8

### [Ultra Serial Port Monitor](https://www.softpedia.com/get/Network-Tools/Network-Monitoring/Ultra-Serial-Port-Monitor.shtml)

Ultra Serial Port Monitor là một tiện ích hệ thống chuyên nghiệp và mạnh mẽ để giám sát các cổng COM RS232/422/485. Chương trình giám sát, hiển thị, phân tích tất cả hoạt động của cổng nối tiếp trong hệ thống, là một công cụ rất hữu ích cho tất cả các nhà phát triển phần mềm và phần cứng làm việc với các cổng nối tiếp. Ultra Serial Port Monitor cho phép nhận, hiển thị, ghi và phân tích tất cả dữ liệu được trao đổi giữa thiết bị nối tiếp. Và theo nhiều cách khác nhau để gửi dữ liệu đến thiết bị nối tiếp để đạt được quyền kiểm soát thiết bị. [7]

Ultra Serial Port Monitor

Hình 11: Biểu tượng Ultra Serial Port Monitor

# THIẾT KẾ VÀ THỰC HIỆN PHẦN CỨNG

## Yêu cầu thiết kế

* Name: Thiết bị giám sát lịch trình sử dụng thuốc
* Pupose: Nhận dữ liệu về thời gian để mở các ngăn thuốc đồng thời kiểm soát được số lượng thuốc trong ngăn.
* Inputs:
* Dữ liệu về thời gian sử dụng thuốc bao gồm: giờ, phút, các thứ trong tuần.
* Dữ liệu về thời gian thực: thứ, ngày, tháng, năm, giờ, phút, giây.
* Nút nhấn phát hiện chuyển động của ngăn thuốc.
* Nút nhấn reset để xoá dữ liệu về thời gian sử dụng thuốc
* Outputs:
* LCD hiển thị: ngày giờ, ngăn thuốc được sử dụng, dữ liệu thời gian sử dụng thuốc.
* Motor khoá, mở ngăn thuốc.
* Còi báo, Led khi ngăn thuốc được mở.
* Functions:
* Nhận thời gian thực qua ứng dụng điện thoại, hiển thị trực tiếp lên LCD
* Nhận thời gian sử dụng thuốc qua máy tính hoặc qua điện thoại. Khi đến thời gian sử dụng, sẽ mở ngăn thuốc, bật các cảnh báo (hiển thị lên LCD, bật còi, đèn led). Nếu trong khoảng 15 phút từ lúc mở, người dùng kéo, đóng ngăn thuốc sẽ được tính 1 lần sử dụng và khoá ngăn thuốc, tắt các cảnh báo. Nếu không sử dụng trong 15 phút, khoá sẽ tự động khoá, tắt cảnh báo và không được tính sử dụng thuốc.
* Số lượng thuốc mỗi ngăn sẽ được hiển thị về điện thoại.
* Performance:
* Khi cấp nguồn cần chờ 5s để thiết bị khởi động.
* Thời gian thực sẽ được hiển thị trên LCD với sai lệch không quá 1ms.
* Thời gian mở ngăn thuốc sai lệch không quá 1 giây.
* Manufacturing cost: chi phí sản xuất vào khoảng 200.000đ.
* Power: Thiết bị cần được cấp nguồn 5 VDC liên tục
* Physical size and weight:
* Kích thước 3 chiều: 20x30x20 cm.
* Khối lượng: 800 gram.

## Phân tích thiết kế

* Màn hình hiển thị:
* Mô tả: Hiển thị thời gian theo cú pháp: dòng 1 hiển thị giờ, dòng 2 hiển thị ngày và hiển thị ngăn mở.
* Yêu cầu:
* Giao tiếp 4 bit
* Hiển thị ít nhất 2 dòng với 16 ký tự.
* Lựa chọn thiết bị: LCD 1602.
* Thiết bị cảnh báo:
* Mô tả: cảnh báo cho người dùng đến giờ sử dụng thuốc
* Yêu cầu: mỗi khi cảnh báo sẽ bật led tương ứng với từng ngăn và bật còi. Tắt tự động sau 15 phút hoặc tắt do người dùng.
* Từ yêu cầu đặt ra, chọn 4 đèn Led cho 4 ngăn và 1 còi
* Khoá ngăn thuốc:
* Mô tả: Khoá/mở ngăn thuốc tự động.
* Yêu cầu: Nhỏ gọn; khoá/mở chắc chắn.
* Các loại khoá:

Bảng 2: Các loại khoá

|  |  |  |
| --- | --- | --- |
| **Phương pháp** | **Mô tả** | **Đặc điểm** |
| Khoá chốt điện | Sử dụng cơ chế hút của nam châm điện, đẩy của lò xo để điều khiển chốt khoá | Sản phẩm ngoài thị trường giá thành rất cao (khoảng 60.000đ)  Sản phẩm tự làm không chắc chắn, cồng kềnh. |
| Khoá xoay | Sử dụng góc xoay của motor đề khoá | Thiết kế đơn giản, khoá chắc chắn |
| Băng chuyền | Dùng băng chuyển để di chuyển kéo, mở ngăn thuốc | Tác động ngoại lực sẽ làm hỏng kết cấu băng chuyển. |

Qua đó chọn khoá sử dụng motor điện.

* Modul Wifi:
* Mô tả: thu/nhận dữ liệu giữa thiết bị và điện thoại.
* Yêu cầu: nhỏ gọn, thực hiện đúng chức năng, thu nhận dữ liệu chính xác.
* Chọn mạch Wifi: do yêu cầu mạch wifi chỉ dùng để truyền dữ liệu nên chọn mạch ESP8266-01.
* Mạch nạp dữ liệu:
* Mô tả: nhận dữ liệu qua máy tính.
* Yêu cầu: nhỏ gọn, truyền dữ liệu chính xác.
* Chọn mạch nạp: Chọn mạch nạp USB CH340 vì thiết kế nhỏ gọn, giá thành rẻ (18.000đ)
* Nút nhấn Reset:
* Mô tả: khi được nhấn sẽ truyền tín hiệu về vi điều khiển để reset dữ liệu.
* Yêu cầu: nhỏ gọn, bền.
* Cảm biến số lượng thuốc:
* Mô tả: cảm biến đếm số lượng thuốc trong ngăn
* Yêu cầu: nhỏ gọn, cảm biến chính xác.
* Chọn cảm biến:

Bảng 3: Các phương pháp chọn cảm biến

|  |  |  |
| --- | --- | --- |
| **Phương pháp** | **Mô tả** | **Đặc điểm** |
| Cảm biến khối lượng | Sử dụng cảm biến đo khối lượng ngăn thuốc, so sánh sự thay đổi khối lượng để tìm số lượng thuốc. | Giá thành cao, lắp đặt khó khăn, khó để cảm biến vật có khối lượng nhỏ (thuốc). |
| Cảm biến chuyển động | Sử dụng mạch thu phát hồng ngoại để xác định số lần mở ngăn thuốc (mỗi lần mở là một lần dùng thuốc) | Giá thành cao, thiết bị có thể tích khá lớn. Cảm biến có thể không chính xác nếu người dùng mở ngăn mà không dùng thuốc. |
| Nút nhấn | Sử dụng nút nhấn để xác định số lần đóng ngăn thuốc | Thiết bị nhỏ gọn, cần thêm lò xo đẩy, tránh trường hợp kẹt nút nhấn. Cảm biến có thể không chính xác nếu người dùng mở ngăn mà không dùng thuốc. |

Qua đó chọn nút nhấn để làm cảm biến.

* **Danh sách phần cứng:**

Bảng 4: Danh sách phần cứng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Thiết bị | Phần cứng | Mô tả | Mức độ quan trọng | Giá |
| LCD 1602 | Màn hình hiển thị | Giao tiếp 4 bit  16 ký tự x 2 dòng  VCC: 3,3 VDC | Quan trọng | 48.000đ |
| Buzzer | Thiết bị cảnh báo | VCC: 3-24 VDC  Âm thanh: ~85dB | Quan trọng | 10.000đ |
| Led | Thiết bị cảnh báo | Led green  VCC: 3,3 VDC | Quan trọng | 250đ |
| Motor | Khoá ngăn thuốc | VCC: 1-6 VDC  Tốc độ tại 3V: 18000rpm | Quan trọng | 4.000đ |
| L9110S | Khoá ngăn thuốc | 8 ngõ vào tín hiệu  8 ngõ ra điều khiển 4 motor | Quan trọng | 44.000đ |
| ESP8266-01 | Module wifi | Tốc độ baud: 115200  Wifi 2.4GHz  VCC: 3,3 VDC | Quan trọng | 45.000đ |
| Mạch nạp USB CH340 | Mạch nạp dữ liệu | Tốc độ baud: 115200 | Quan trọng | 18.000đ |
| Button | Cảm biến chuyển động, nút reset | Nút nhấn nhỏ gọn | Quan trọng | 500đ |

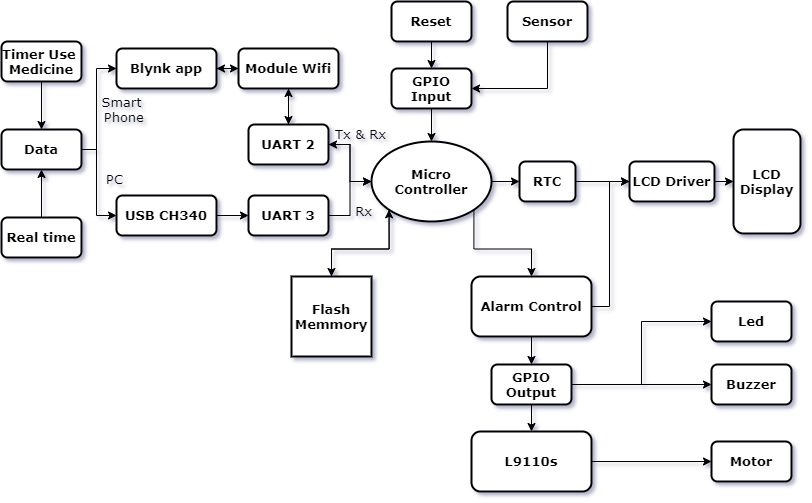
* Vi điều khiển:

Bảng 5: Yêu cầu lựa chọn Vi điều khiển

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Ngoại vi | Giao diện | Số lượng chân | Yêu cầu |
| 1 | LCD 1602 | GPIO | 7 |  |
| 2 | Buttons | GPIO | 5 |  |
| 3 | Led | GPIO | 4 |  |
| 4 | Buzzer | GPIO | 1 |  |
| 5 | Motor và L9110s | GPIO | 8 |  |
| 6 | ESP8266 | UART | 2 | Data 8 bit |
| 7 | USB CH340 | UART | 2 | Data 8 bit |
| 8 |  | Real time clock |  | Tích hợp sẵn RTC |
| 9 |  | Flash memory |  | Tích hợp Flash memory |

Qua các yêu cầu trên, lựa chọn vi điều khiển STM32F103C8T6.

## Sơ đồ khối

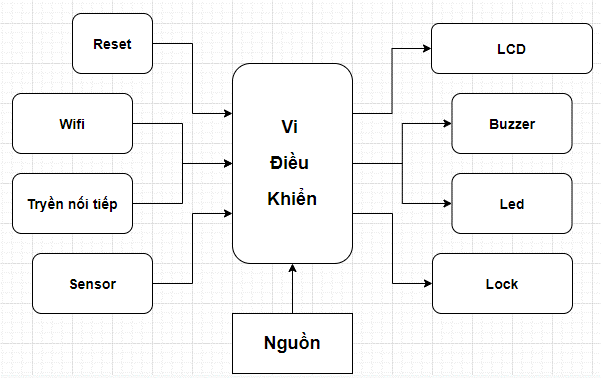


Hình 12: Sơ đồ khối chi tiết

Giải thích:

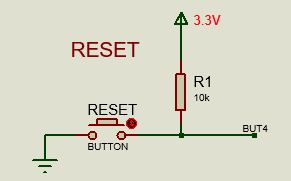
* Dữ liệu thời gian thực (Real time) được nhập từ Blynk app và truyền qua mạch Wifi thông qua UART và tới vi điều khiển. Dữ liệu được xử lý, đưa vào bộ RTC của vi điều khiển để xác lập thời gian ban đầu. Thời gian từ RTC của Vi điều khiển sẽ đưa ra LCD hiển thị.
* Dữ liệu thời gian, số thuốc (Data) được nhận bằng 2 cách:
* Nhập từ Blynk app, mạch Wifi, tới UART của vi điều khiển.
* Nhập từ máy tính, qua USB CH340, tới UART của vi điều khiển.
* Dữ liệu thời gian, số thuốc sẽ được xử lý, lưu vào Flash Memory tránh trường hợp mất nguồn dẫn đến mất dữ liệu
* Dữ liệu thời gian sẽ được so sánh với RTC và tác động tới Alarm Control.
* Nhận được tín hiệu từ vi điều khiển, khối Alarm Control sẽ bật cảnh báo Led, Buzzer, LCD hiển thị và mở khoá ngăn thuốc (Lock). Nếu không nhận được tín hiệu tắt, các báo động sẽ được tắt trong 15 phút
* Nếu Sensor phát hiện cảm biến ở ngăn thuốc, Sensor tín hiệu đến vi điều khiển, sau đó khối Alarm Control sẽ tắt các cảnh báo và khoá ngăn thuốc.
* Nếu nhấn Reset, vi xử lý sẽ xoá thời gian dùng thuốc và số lượng thuốc.

## Sơ đồ mạch chi tiết



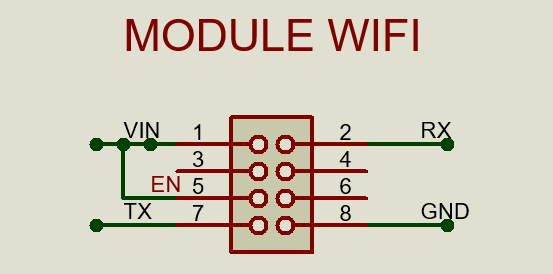
Hình : Các khối chức năng

* Reset:



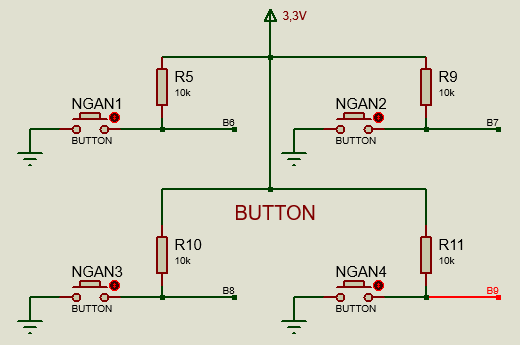
Hình : Thiết kế Reset

* Nút nhấn được thiết kế theo dạng điện trở kéo lên (10k). Khi chưa nhấn, vi điều khiển sẽ được giữ ở mức cao (3V). Khi nhấn, điện áp ở GPIO sẽ ở mức thấp (0V).
* Nút reset được đọc bởi chân B4 của Vi điều khiển
* Khối Wifi



Hình : Thiết kế Wifi

* Mạch Wifi ESP8266-01 được cấp nguồn 3,3V vào chân VCC và chân EN (chân Enable); GND vào GND.
* Chân RX nối với chân A2 (TX của Vi điều khiển).
* Chân RX nối với chân A3 (RX của Vi điều khiển).
* Truyền nối tiếp:
* USB CH340 được nối TX vào chân B11, RX vào chân B10 của Vi điều khiển.
* Công suất tiêu thụ: dùng nguồn từ máy tính.
* Sensor:

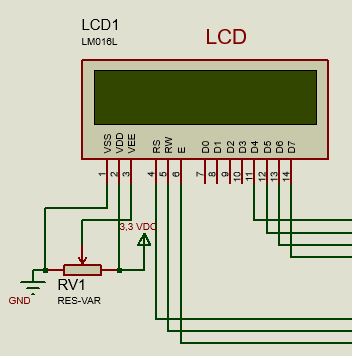


Hình 16: Sensor by button

* Ngõ ra 4 nút nhấn của 4 ngăn ở được nối với chân B6, B7, B8, B9 của vi điều khiển.
* Tương tự nút nhấn reset, nút nhấn đọc số thuốc cũng là điện trở kéo lên.
* Khối LCD:

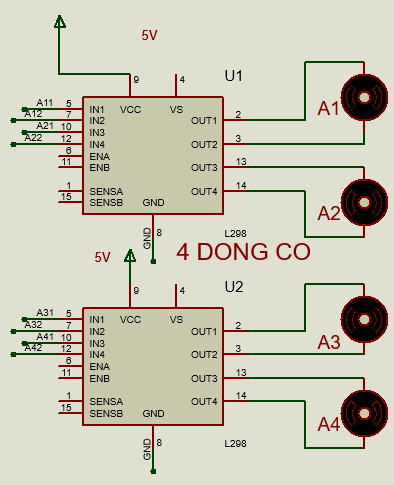
Bảng 6: Giao tiếp LCD với Vi điều khiển

|  |  |  |
| --- | --- | --- |
| Ký hiệu | Mô tả | Vi điều khiển |
| VSS |  | GND |
| VDD |  | 3,3V |
| VEE | Độ tương phản | Nối qua biến trở 10k tới 3,3V |
| RS | Lựa chọn thanh ghi | A12 |
| RW | Lựa chọn thanh ghi đọc/ viết | A15 |
| E | Enable | B3 |
| D4,D5,D6,D7 | Chân truyền dữ liệu | B12, B13, B14, B15 |
| A | Led | 3,3 V |
| K | Led | GND |



Hình : Thiết kế LCD

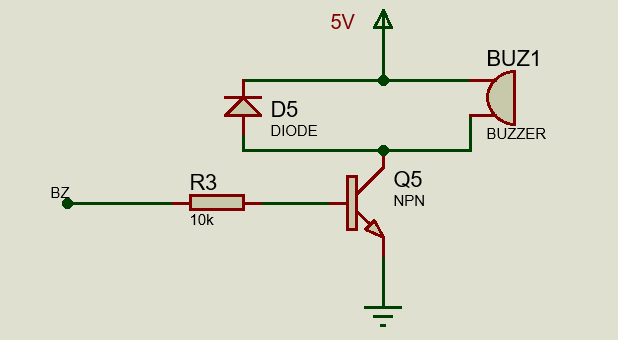
* Lock



Hình : Khối Lock

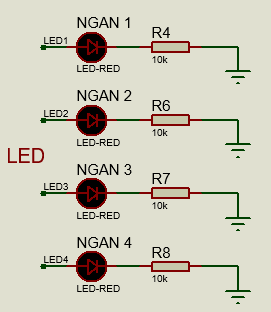
(Hình trên sử dụng 2 mạch điều khiển mô phỏng cho mạch L9110s).

* Khối khoá ngăn thuốc gồm 1 mạch điều khiển động cơ L9110s và 4 motor cho 4 ngăn.
* Các motor được cấp nguồn 5V gián tiếp qua mạch điều khiển L9110s.
* Khi nhận được tín hiệu điều khiển, các motor sẽ được cấp nguồn 5V, 0,8A qua các cặp chân ngõ ra của vi điều khiển qua mạch điều khiển L9110s.
* Motor 1: cặp chân A8 và A6
* Motor 2: cặp chân A9 và A5
* Motor 3: cặp chân A20 và B0
* Motor 4: cặp chân A11 và B1
* Buzzer



Hình 19: Khối Buzzer

* Buzzer nối với chân B5, khi có nguồn vào chân B của thyristor NPN sẽ có dòng từ 5V xuống GND làm buzzer hoạt động. Diod D5 giúp tránh trường hợp xuất hiện ngược dòng, bảo vệ buzzer.
* Led
* Các Led nối tiếp với điện trở 10k trước khi nối vào chân A0, A1, A4, A5 của vi điều khiển.



Hình 20: Thiết kế Led

* Khối nguồn:

Bảng 7: Thống kê và thiết kế nguồn

|  |  |  |  |
| --- | --- | --- | --- |
| **Khối chức năng** | **Điện áp (V)** | **Dòng (mA)** | **Công suất tiêu thụ (mW)** |
| Reset | Không đáng kể | | |
| Wifi | 3,3 | 300 | 990 |
| Truyền nối tiếp | Dùng nguồn từ máy tính | | |
| Sensor | Không đáng kể | | |
| LCD | 3,3 | 220 | 726 |
| Buzzer | 5 | 30 | 150 |
| Led | 3,3 | 75 | 250 |
| Lock | 5 | 400 | 2.000 |
| Vi điều khiển | 5 | 150 | 750 |
| **Tổng công suất tiêu thụ** |  |  | 4.866 |
| **Nguồn cấp** | 5 | 1 | 5.000 |

Dựa trên tiêu chí về tổng công suất tiêu thụ, sử dụng nguồn 5V, 1A cho toàn bộ hệ thống.

**Sơ đồ mạch chi tiết được đính kèm ở phần Phục lục.**

# THIẾT KẾ VÀ THỰC HIỆN PHẦN MỀM

## Yêu cầu đặt ra cho phần mềm

Thiết bị phải có những yêu cầu sau đây:

1. Có khả năng hiển thị thời gian thực và dữ liệu nhập vào thiết bị:

Với vấn đề này, chúng em sử dụng thư viện hiển thị LCD có sẵn để hiển thị dữ liệu.

1. Cho phép nhận dữ liệu từ máy tính, điện thoại trong quá trình thiết bị hoạt động.

Ở phần mở đầu chương trình, khởi động 2 bộ UART. Khi dữ liệu nhận bị tràn sẽ thực thi hàm ngắt: khởi động lại việc nhận dữ liệu. Sử dụng 2 bộ giao tiếp UART để tránh xung đột trong việc nhận dữ liệu từ 2 thiết bị

1. Có bộ nhớ Flash lưu tránh trường hợp thiết bị tắt đột xuất.

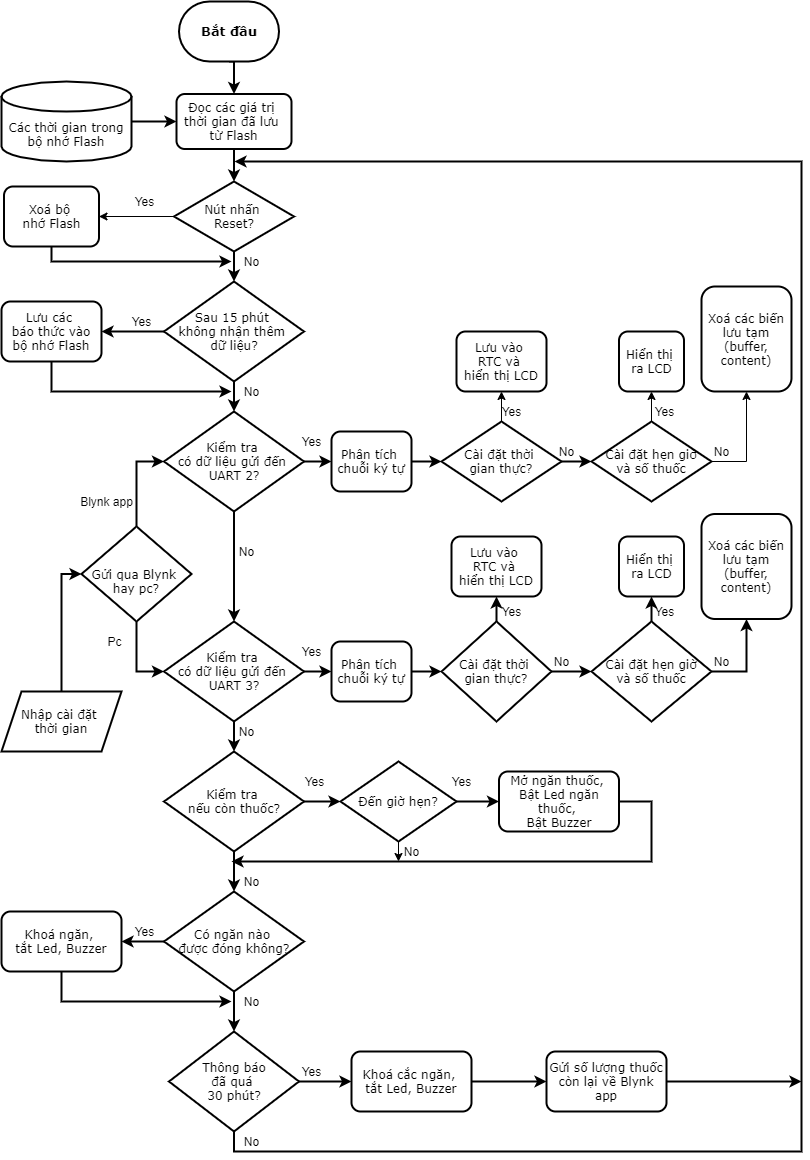
Dữ liệu nhận được sẽ được xử lý theo cú pháp định sẵn và được lưu vào bộ nhớ Flash. Mỗi khi chương trình hoạt động, sẽ thực hiện lấy dữ liệu từ bộ nhớ Flash ở đầu chương trình.

1. Thiết lập so sánh giữa thời gian thực và thời gian uống thuốc, sau đó đưa ra các tín hiệu cảnh báo:

Trong vòng lặp while của chương trình chính: So sánh thời gian thực với thời gian uống thuốc, qua đó quyết định cho phép dùng thuốc hay không, đồng thời xuất tín hiệu ra màn hình LCD, Led, Buzzer.

1. Có công cụ định thời 30 phút kể từ lúc cho phép dùng thuốc. Sau 30 phút sẽ tự động đóng ngăn thuốc, tắt báo động. Thời gian đếm ngược được sử dụng là chức năng Systick.
2. Nếu trong trạng thái cho phép sử dụng thuốc, liên tục đọc cảm biến để đóng ngăn thuốc, tắt cảnh báo. Nếu cảm biến được đọc trong khi ngăn thuốc được mở, sẽ đóng ngăn thuốc, tắt cảnh báo và bộ đếm Systick.

## Lưu đồ giải thuật chi tiết



# KẾT QUẢ THỰC HIỆN

Trong phần này, sinh viên mô tả:

* Trình bày **cách thức đo đạc, thử nghiệm** 
  + Ghi rõ các thiết bị sử dụng và sơ đồ kết nối trong việc thử nghiệm
  + Ghi rõ các phần mềm sử dụng trong việc viết và thực thi chương trình
  + Ghi rõ cách bước tiến hành thử nghiệm (phần cứng và phần mềm)
* Trình bày số liệu đo đạc
  + Thực hiện thu thập số liệu trong nhiều trường hợp
  + Ghi rõ số liệu đo đạc thu được dưới hình thức bảng biểu, đồ thị …
* **Giải thích và phân tích về kết quả thu được**
  + Cần giải thích rõ ràng số liệu thu được trên các bảng biểu, đồ thị, dạng sóng …
  + Phân tích các số liệu để biết kết quả đã thực hiện là phù hợp, đạt yêu cầu

Nếu những bảng số liệu và kết quả mô phỏng quá nhiều, sinh viên có thể trình bày đưa vào phần Phụ Lục.

Ví dụ về hình minh họa: (dùng chức năng **Insert Caption** để tạo liên kết cho Danh sách hình minh họa)

Ví dụ về Bảng số liệu

Bảng 8 Thông số hệ thống

|  |  |  |  |
| --- | --- | --- | --- |
| Thông số 1 | Thông số 2 | Thông số 3 | Thông số 4 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

* **Đánh giá về kết quả làm việc nhóm** (Nếu chỉ có 1 sinh viên làm đề tài thì không trình bày phần này)

Sinh viên trình bày kết quả công việc của từng thành viên, các ý kiến trong khi thiết kế, thời hạn hoàn thành thiết kế của các thành viên. Mục tiêu chung của đề tài có đạt được không? Có thành viên nào không tích cực?

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết luận

Sinh viên tóm tắt những điều rút ra được từ kết quả đề tài, những kinh nghiệm có được sau khi thực hiện đề tài. **Ưu và khuyết điểm** của kết quả nghiên cứu đề tài cũng được trình bày trong mục này. Sinh viên cần so sánh với mục tiêu đặt ra trong chương 1.

## Hướng phát triển

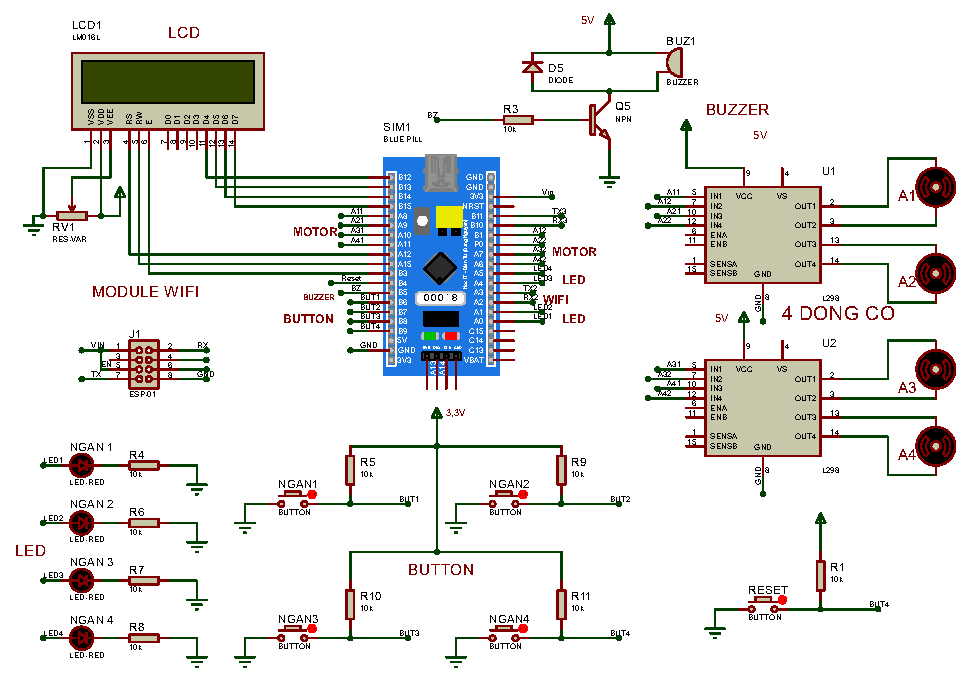
Sinh viên trình bày hướng phát triển và khả năng ứng dụng của đề tài

1. TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | J. Mark Ruscin; Sunny A. Linnebur, "MSD MANUAL," Merck Sharp & Dohme Corp, June 2016. [Online]. Available: https://www.msdmanuals.com/vi/chuy%C3%AAn-gia/l%C3%A3o-khoa/%C4%91i%E1%BB%81u-tr%E1%BB%8B-thu%E1%BB%91c-%E1%BB%9F-ng%C6%B0%E1%BB%9Di-cao-tu%E1%BB%95i/c%C3%A1c-v%E1%BA%A5n-%C4%91%E1%BB%81-li%C3%AAn-quan-%C4%91%E1%BA%BFn-thu%E1%BB%91c-%E1%BB%9F-ng%C6%B0%E1%. [Accessed 28 May 2021]. |
| [2] | U. Gajera, "OCFreaks," 25 July 2018. [Online]. Available: http://www.ocfreaks.com/create-new-stm32-project-keil-uvision-5-tutorial/. [Accessed 28 May 2021]. |
| [3] | Anusha V; Ravi Teja; Bala, "Electronicshub," 20 April 2020. [Online]. Available: https://www.electronicshub.org/program-stm32f103c8t6-using-keil-uvision/. [Accessed 28 May 2021]. |
| [4] | P. Desai, "Project Hub," 13 January 2019. [Online]. Available: https://create.arduino.cc/projecthub/pratikdesai/how-to-program-esp8266-esp-01-module-with-arduino-uno-598166. [Accessed 5 May 2021]. |
| [5] | Blynk team, "Blynk," Blynk Inc, 10 October 2015. [Online]. Available: https://docs.blynk.cc/. [Accessed 28 May 2021]. |
| [6] | S. Suresh, "Techulator," SpiderWorks Technologies, 28 April 2016. [Online]. Available: https://www.techulator.com/resources/15432-Proteus-8-Professional-a-useful-tool-for-electronic-engineers.aspx. [Accessed 28 May 2021]. |
| [7] | S. team, "Softpedia," SoftNews NET SRL, 3 November 2009. [Online]. Available: https://www.softpedia.com/get/Network-Tools/Network-Monitoring/Ultra-Serial-Port-Monitor.shtml. [Accessed 28 May 2021]. |

# PHỤ LỤC

Sơ đồ mạch chi tiết.



Trong phần này, sinh viên có thể trình bày:

* Những kết quả nghiên cứu bổ sung mà trong phần Kết quả luận văn chưa trình bày hết.
* Phần mã nguồn chương trình, sinh viên cũng có thể trình bày trong mục này. Để ngắn gọn, sinh viên chỉ đưa những mã nguồn chính vào phần Phụ lục.
* Sơ đồ toàn mạch chi tiết