BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH

KHOA ĐIỆN - ĐIỆN TỬ

BỘ MÔN KĨ THUẬT MÁY TÍNH



**ĐỒ ÁN MÔN HỌC 1**

**KHÓA CỬA ĐIỆN TỬ SỬ DỤNG MÃ BẢO MẬT VÀ RFID**

GVHD : ThS. Trần Thị Quỳnh Như

SVTH : NGUYỄN QUANG BÌNH - 20119063

PHAN HOÀNG PHÚC - 20119088

*TP. Hồ Chí Minh, 5/2023*

# LỜI CẢM ƠN

Lời đầu tiên, chúng em xin gửi đến Cô Trần Thị Huỳnh Như lời cảm ơn chân thành và sâu sắc nhất. Nhờ có sự hướng dẫn và giúp đỡ tận tình của Cô trong suốt thời gian qua, chúng em đã có thể thực hiện và hoàn thành Đồ Án 1 . Những lời nhận xét, góp ý và hướng dẫn tận tình của Cô đã giúp em có một định hướng đúng đắn trong suốt quá trình thực hiện Đề tài, giúp em nhìn ra được những ưu khuyết điểm của Đề tài và từng bước hoàn thiện

hơn.

Đồng thời, chúng em xin trân trọng cảm ơn các Thầy Cô của Trường đại học Sư phạm kỹ thuât nói chung và của khoa Điện - Điện Tử nói riêng đã dạy dỗ chúng em suốt quãng thời gian qua .

# TÓM TẮT ĐỒ ÁN

Ngày nay với sự phát triển mạnh mẽ của thế giới về mọi mặt, trong đó khoa đó khoa học công nghệ nói chung và ngành công nghệ kỹ thuật điện tử nói riêng có nhiều phát triển vượt bậc, góp phần làm cho thế giới ngày càng hiện đại và văn minh hơn. Sự phát triển của kỹ thuật điện tử đã tạo ra hàng loạt những thiết bị với các đặc điểm như sự chính xác cao, tốc độ nhanh, gọn nhẹ và hoạt động ổn định. Là những yếu tố cần thiết làm cho hoạt động con người đạt hiệu quả cao. Hiện nay với sự bùng nổ về các thiết bị IOT, đặc biệt là sự phát triển vượt bậc về các hệ thống chống trộm và bảo vệ nhà cửa . Nhiều loại khóa điện tử thông minh ngày càng được phát triển.Nhận thấy được tính ưu việt và xu hướng của thực tế , vì vậy nhóm chúng em đã chọn đề tài “**KHÓA ĐIỆN TỬ SỬ DỤNG MÃ BẢO MẬT VÀ RFID** ” để tìm hiểu và nghiên cứu. Nội dung báo cáo này gồm 4 chương:

**CHƯƠNG I: GIỚI THIỆU YÊU CẦU – GIỚI HẠN**

**CHƯƠNG II: THIẾT KẾ SƠ ĐỒ KHỐI**

**CHƯƠNG III: THI CÔNG MẠCH – VIẾT CHƯƠNG TRÌNH**

**CHƯƠNG IV: KẾT QUẢ THỰC HIỆN**

Để hoàn thành đồ án “KHÓA ĐIỆN TỬ SỬ DỤNG MÃ BẢO MẬT VÀ RFID” em xin chân thành cảm ơn sự hướng dẫn tận tình của cô Trần Thị Quỳnh Như – Giảng viên khoa Điện – Điện Tử, Trường Đại Học Sư Phạm Kỹ Thuật Tp.HCM. Cùng với sự giúp đỡ tận tình của các thầy cô bộ môn và các bạn trong lớp đã giúp đỡ em hoàn thành tốt đồ án này.

Trong quá trình nghiên cứu, tìm hiểu về đề tài và thi công mạch không tránh khỏi những sai sót. Em mong Cô cùng các bạn góp ý để đề tài được hoàn thiện hơn và có thể ứng dụng nhiều trong thực tế.

Một lần nữa chúng em xin chân thành cảm ơn!

# NHẬN XÉT CỦA GIẢNG VIÊN

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………

Ký tên

Trần Thị Quỳnh Như

**Mục Lục**

[**LỜI CẢM ƠN 2**](#_Toc136183217)

[**TÓM TẮT ĐỒ ÁN 3**](#_Toc136183218)

[**NHẬN XÉT CỦA GIẢNG VIÊN 4**](#_Toc136183219)

[**CHƯƠNG 1: GIỚI THIỆU YÊU CẦU – GIỚI HẠN 6**](#_Toc136183220)

[1.1 GIỚI THIỆU: 6](#_Toc136183221)

[1.2 MỤC ĐÍCH, YÊU CẦU SẢN PHẨM: 6](#_Toc136183222)

[1.3 GIỚI HẠN: 7](#_Toc136183223)

[**CHƯƠNG 2: THIẾT KẾ SƠ ĐỒ KHỐI 8**](#_Toc136183224)

[2.1 GIỚI THIỆU: 8](#_Toc136183225)

[2.2 THIẾT KẾ SƠ ĐỒ KHỐI: 8](#_Toc136183226)

[2.2.1 SƠ ĐỒ KHỐI: 8](#_Toc136183227)

[2.2.2 CHỨC NĂNG VÀ NHIỆM VỤ TỪNG KHỐI: 9](#_Toc136183228)

[2.3 THIẾT KẾ SƠ ĐỒ NGUYÊN LÝ: 9](#_Toc136183229)

[2.3.1 GIỚI THIỆU VỀ ARDUINO UNO : 9](#_Toc136183230)

[2.3.2 GIỚI THIỆU VỀ MODULE RFID RC522 16](#_Toc136183231)

[2.3.3 KHỐI MA TRẬN PHÍM 4X3 : 22](#_Toc136183232)

[2.3.4 KHỐI HIỂN THỊ LCD – LIQUID CRYTAL DISPLAY 24](#_Toc136183233)

[2.3.5 KHỐI RELAY VÀ KHÓA ĐIỆN TỬ 33](#_Toc136183234)

[2.3.6 KHỐI NGUỒN: 35](#_Toc136183235)

[2.4 SƠ ĐỒ NGUYÊN LÝ TOÀN MẠCH: 39](#_Toc136183236)

[2.5 NGUYÊN LÝ HOẠT ĐỘNG: 42](#_Toc136183237)

[**CHƯƠNG 3 : KẾT QUẢ THỰC HIỆN 43**](#_Toc136183238)

[3.1 VẼ MẠCH PCB 43](#_Toc136183239)

[3.2 MÔ HÌNH THỰC TẾ 45](#_Toc136183240)

[**CHƯƠNG 4: KẾT LUẬN VÀ ĐỊNH HƯỚNG ĐỀ TÀI 46**](#_Toc136183241)

[4.1 KẾT QUẢ ĐẠT ĐƯỢC 46](#_Toc136183242)

[4.2 HẠN CHẾ 46](#_Toc136183243)

[4.3 HƯỚNG PHÁT TRIỂN 46](#_Toc136183244)

[**TÀI LIỆU THAM KHẢO 48**](#_Toc136183245)

# CHƯƠNG 1: GIỚI THIỆU YÊU CẦU – GIỚI HẠN

# 1.1 GIỚI THIỆU:

Ngày nay với sự phát triển của công nghệ đặc biệt là sự nổi trội của hệ thống IOT , đòi hỏi con người phải nâng cao về bảo mật nhà cửa .Vì thế những thiết bị thông minh như khóa bảo mật , hệ thống còi báo hay camera chống trộm ra đời . Tuy nhiên hiện nay khóa cơ học còn được sử dụng khá nhiều trên thị trường và đặc biệt các khóa này dễ bị cạy hay bẻ khóa .

Dựa trên khảo sát thực tế chúng em quyết định thiết kế khóa điện tử như sau :

+ Sử dụng khóa số và thẻ RFID để mở cửa tự động ( phòng trường hợp quên mật khẩu của khóa ngoài ) . Có thể chỉnh sửa hoặc xóa thẻ.

+ Sử dụng ma trận phím 4x3 để Chọn chế độ : Chế độ 1 – mở khóa bằng mật khẩu , Chế độ 2 – mở khóa bằng RFID , nhập mật khẩu và thay đổi mật khẩu.

+ Hiển thị kí tự password ( mã hóa số thành kí tự \* VD : nhập “1234” thì hiển thị “\*\*\*\*” ) .

+ Khi nhập password đúng thì hệ sẽ mở cửa cùng còi báo và tự động đóng lại sao 10s.

+ Khi nhập sai password hay quẹt sai thẻ thì hệ thống sẽ yêu cầu nhập lại , nếu sai quá 3 lần hệ thống sẽ báo còi.

# 1.2 MỤC ĐÍCH, YÊU CẦU SẢN PHẨM:

* Mục đích của khóa thông minh ứng dụng nhiều trong hệ thống IOT trong nhà thông minh , tính bảo mật cao hơn so với khóa vật lý thông thường , dễ dàng phát triển .
* Yêu cầu:
* Đơn giản, gọn, giá thành rẻ.
* Bộ phận hiển thị phải rõ ràng.
* Mạch phải hoạt động tốt khi có hoặc không có tín hiệu.
* Lập trình và điều khiển: Lập trình Arduino Uno R3 để điều khiển các chức năng của khóa thông minh, bao gồm giao tiếp với ma trận nút và LCD, kiểm tra và xử lý dữ liệu từ nút, hiển thị thông tin lên LCD, và thực hiện các chức năng an ninh như mở/khóa cửa.
* Nắm vững ngôn ngữ lập trình C.
* Kiểm soát bảo mật.

# 1.3 GIỚI HẠN:

* Sản phẩm còn khá đơn giản cần phát triển nhiều hơn .
* Chốt khóa điện nên khá dễ hư .
* Chưa kiểm soát được người truy cập vào.

# 

# CHƯƠNG 2: THIẾT KẾ SƠ ĐỒ KHỐI

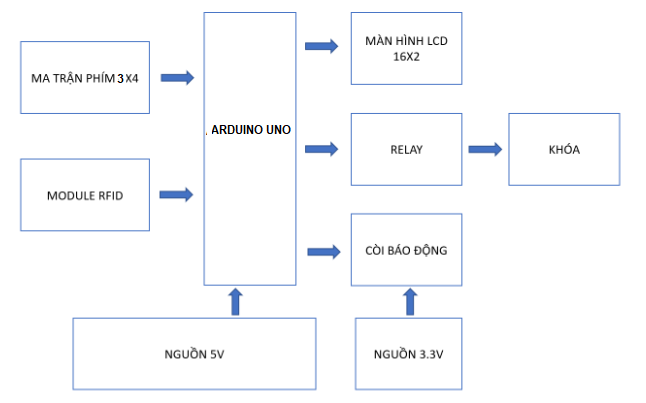
# 2.1 GIỚI THIỆU:

Khóa thông minh phải hoạt động một cách chính xác, đơn giản, gọn nhẹ. Với mỗi chế độ bấm nhập mật khẩu bằng phím hay quét thẻ từ đòi hỏi bộ phần điều khiển và hiển thị phải hoạt động theo yêu cầu đã đề ra.

# THIẾT KẾ SƠ ĐỒ KHỐI:

## **2.2.1 SƠ ĐỒ KHỐI:**

Sau thời gian khảo sát và tìm hiểu về đề tài, dựa trên các yêu cầu của mạch sơ đồ khối được thiết kế và trình bày như hình sau:



Hình 1.1: Sơ đồ khối khóa bảo mật

## **2.2.2 CHỨC NĂNG VÀ NHIỆM VỤ TỪNG KHỐI:**

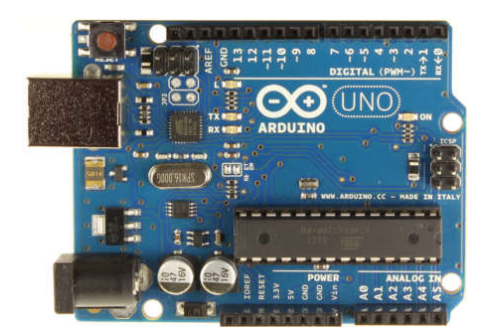
* **Khối nguồn:** Tạo ra dòng điện ổn định cung cấp cho toàn mạch hoạt động.
* **Khối ma trận nút**: Có chức năng như một bộ điều khiển , Chọn Mode và nhập mật khẩu.
* **Khối Relay**: Có chức năng kích hoạt chốt khóa khi nhận được tín hiệu mở khóa từ vi điều khiển .
* **Khối RFID**: Có chức năng quét thẻ từ để kích relay mở khóa
* **Khối hiển thị**: Hiển thị kết quả ra LCD

# THIẾT KẾ SƠ ĐỒ NGUYÊN LÝ:

## **2.3.1 GIỚI THIỆU VỀ ARDUINO UNO :**

**a. Tổng quan.**

Arduino Uno là bo mạch vi xử lý hoạt động dựa trên ATmega328. Bo mạch này có 14 chân input/output digital (trong đó có 6 chân được dùng cho điều chế xung đầu ra PWM), 6 đầu vào analog, tần số giao động thạch anh là 16MHz, kết nối USB, jack cắm nguồn, chân tiêu đề ICSP, một nút reset. Bo mạch này chứa tất cả các tính năng cần thiết để hỗ trợ kết nối với các vi điều khiển khác. Nguồn sử dụng cho bo mạch có thể qua USB, sử dụng pin hoặc nguồn thông qua bộ chuyển đổi AC–DC.



Hình 2.1: Bo mạch Arduino Uno

Arduino Uno khác với tất cả các bo mạch khác ở chỗ nó không sử dụng chip điều khiển nối tiếp FTDI USB. Thay vào đó các tính năng của ATmega16U2 được lập trình để chuyển đổi USB nối tiếp.

Ở phiên bản thứ hai: bo mạch Uno có điện trở nối đường 8U2 HWB với đất, do đó ta dễ dàng hơn trong việc thiết lập chế độ DFU. Ở phiên bản sửa đổi thứ 3 của bo mạch có các tính năng mới dưới đây:

- Sơ đồ chân 1.0: Thêm các chân SDA và SCL gần với chân AREF và 2 chân mới được đặt gần chân RESET, IOREF cho phép Shield nhận nguồn cấp từ bo mạch. Trong tương lai, Shield sẽ tương thích với các bo mạch có sử dụng AVR, có điện áp hoạt động là 5V và Arduino Due hoạt động ở 3.3V. Một số chân trong bo mạch không được kết nối gì để dành cho mục đích trong tương lai.

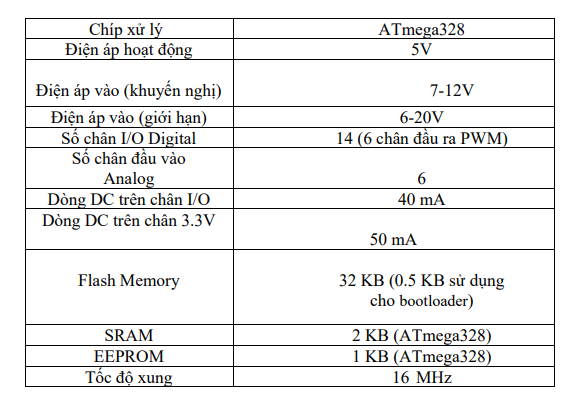
- Mạch Reset mạnh mẽ hơn.

- ATmega 16U2 thay thế cho 8U2.

"Uno" là từ trong tiếng Ý và được đặt tên để đánh dấu việc phát hành phiên bản Arduino 1.0. Uno và phiên bản 1.0 sẽ là phiên bản tham khảo của Arduino, và luôn có sự cải tiến. Uno là một trong những bo mạch mới nhất trong một loạt các bo mạch USB Arduino, và là mô hình tham chiếu nền tảng của Arduino, để so sánh với phiên bản trước đó, xem các thông số của bo mạch Arduino.

**b. Tóm tắt các thông số.**

Bảng 1: thông số chính bo mạch Arduino Uno



**c. Nguồn cấp.**

Arduino Unocó thể được cấp nguồn thông qua kết nối USB hoặc với một nguồn cung cấp điện bên ngoài. Nguồn điện được chọn một cách tự động. Nguồn cấp bên ngoài (không phải là USB) có thể lấy từ bộ chuyển đổi AC-DC hoặc nguồn pin. Các bộ chuyển đổi có thể được kết nối bằng cách cắm chân cắm đường kính 2.1mm vào lỗ cắm điện trên bo mạch. Nếu nguồn lấy từ pin có thể được lắp vào 2 đầu GND và Vin chân tiêu đề của kết nối POWER.

Bo mạch có thể hoạt động với các nguồn cấp ngoài từ 6 đến 20 volt. Nếu nguồn ít nhất thường là 7V, tuy nhiên, các chân 5V có thể được cấp nguồn bé hơn 5V nhưng khi đó mạch có thể hoạt động không ổn định. Nếu sử dụng hơn 12V, bộ ổn áp bị nóng và hỏng mạch, khuyến nghị nên sử dụng ở khoảng 7 đến 12 volts.

Nguồn cấp của các chân như sau:

- Vin: Điện áp đầu vào của bo mạch Arduino khi nó sử dụng nguồn cấp ngoài (khác với các nguồn cấp 5V từ kết nối USB và các nguồn điện theo quy định). Ta có thể cấp nguồn qua chân Vin hoặc cấp nguồn thông qua các jack cắm kết nối với chân này.

- 5V: Chân đầu ra được quy đinh là 5V. Bo mạch có thể có thể được cấp nguồn điện từ các jack (7-12V), kết nối USB (5V), hoặc chân Vin của bo mạch (7- 12V). Cung cấp điện áp thông qua chân 5V hoặc 3.3V bỏ qua các khuyến cáo có thể gây hỏng mạch. Không nên sử dụng nó.

- 3.3V: Nguồn cấp 3.3V được quy định trên bo mạch. Dòng cấp tối đa là 50mA. - GND Chân nối đất. - IOREF. Chân này cấp điện áp tham chiếu cho vi điểu khiển hoạt động. Bộ hỗ trợ cấu hình chuẩn đọc điện áp trên chân OIREF và lựa chọn nguồn cấp thích hợp hoặc kích hoạt dịch điện áp trên đầu ra để làm việc với các nguồn 5V hoặc 3.3V.

**d. Đầu vào đầu ra.**

Trên bo mạch có 14 chân digital có thể được sử dụng cho mạch đích vào hoặc ra, sử dụng các hàm pinMode(), digitalWrite(), và digitalRead(). Chúng hoạt động ở mức điện áp 5V. mỗi chân có thể cung cấp hoặc nhận dòng cực đại là 40mA và được nối với điện trở mặc định từ 20-50 K. Ngoài ra còn có một số chân có chức năng đặc biệt:

- Chân nối tiếp: 0 (RX) và 1 (TX). Sử dụng để nhận và truyền dữ liệu TTL. Chân này được nối với chân tương ứng của ATmega82U. - Chân ngắt ngoài: 2 và 3. Chân này có thể được thiết lập để thực hiện ngắt khi điện áp quá thấp hoặc thay đổi đột biến giá trị điện áp. Thường sử dụng hàm ngắt attachInterrupt().

- PWM: 3, 5, 6, 9 và 10. Xung PWM có độ rộng là 8 bits. Khi điều chế xung đầu ra sử dụng hàm analogWrite().

- SPI: 10(SS), 11 (MOSI), 12 (MISO), 13 (SCK). Các chân này hỗ trợ truyền dẫn SPI sử dụng thư viện SPI.

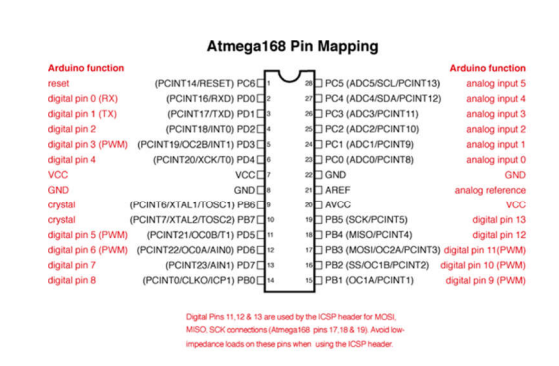
- LED: 13. Có một đèn LED được nối với chân 13. Khi điện áp ở mức cao LED sáng và ở mức thấp thì LED tắt.

Uno có 6 đầu ra analog, có nhãn là A0 đến A5, mỗi chân được cung cấp 10 bits dữ liệu (tương ứng với 1024 giá trị khác nhau). Trên bo mạch Ethernet có 6 chân đầu vào tương tự, có nhãn từ A0 đến A5, mỗi chân sử dụng 10 bits (tức là có 1024 giá trị khác nhau). Nguồn cấp cho các chân này là từ 0-5V, mặc dù nó có thể thay đổi phạm vi hoạt động của chúng bằng cách sử dụng chân AREF và hàm analogReference(). Ngoài ra, một sô chân có chức năng chuyên biệt:

- TWI: A4 (SDA) và A5 (SCL). Hỗ trợ truyền thông TWI sử dụng thư viện Wire. Còn có hai chân khác trên bo mạch là:

- AREF: Điện áp tham chiếu cho đầu vào tương tự. Sử dụng với hàm analogReference().

- Reset. Được dùng để thiết lập lại vi điều khiển. Thường sử dụng nút reset để hỗ trợ các khối trên bo mạch. Ta có thể xem kết nối giữa các chân Arduino và ATmega328 như sau:



Hình 2.2: Sơ đồ chân Atmea168

**e. Các thông số khác.**

* Bộ nhớ.

ATmega328 có 32KB (với 0.5KB dành cho bootloader). Nó có 2KB bộ nhớ SRAM và 1KB bộ nhớ EEPROM (có thể đọc và ghi thông qua thư viện EEPROM library).

* Truyền dẫn.

Arduino Uno có một số phương thức để giao tiếp với máy tính và các Arduino khác, hoặc các dòng vi xử lý khác. ATmega328 cung cấp giao tiếp nối tiếp UART TTL (5V), trong đó có sẵn trên các chân digital 0 (RX) và 1 (TX). ATmega16U2 trên các kênh giao tiếp nối tiếp với bo mạch này thông qua cổng USB và xuất hiện cổng COM ảo kết nối với phần mềm máy tính. 16U2 sử dụng các trình điều khiển tiêu chuẩn phần cứng USB COM. Tuy nhiên khi thực hiện trên Window, một tập tin được yêu cầu. Phần mềm Arduino bao gồm một màn hình cho ta nhìn thấy được dữ liệu được chuyển đến bo mạch Arduino. LED RX và TX sẽ nhấp nháy khi dữ liệu được chuyển thông qua kết nối cổng USB của chip và cổng USB của máy tính. (không thực hiện giao tiếp nối tiếp trên chân 0 và 1).

Thư viện SoftwareSerial cho phép giao tiếp nối tiếp trên bất kỳ chân digital nào của Uno. ATmega328 cũng hỗ trợ giao tiếp I2C (TWI) và SPI. Phần mềm 19 Arduino bao gồm một thư viện wire library để đơn giản hóa việc sử dụng bus I2C. Với truyền dẫn SPI, sử dụng thư viện SPI.

* Lập trình.

Arduino Uno có thể được lập trình với các phần mềm Arduino. Chọn " Arduino Uno từ thanh công cụ Tools Board menu (theo vi điều khiển trên bo mạch). Để biết chi tiết , xem tài liệu tham khảo và hướng dẫn.

ATmega328 trên Arduino Uno đi kèm với một bộ nạp khởi động preburned cho phép tải code mới lên mà không cần sử dụng lập trình phần cứng bên ngoài. Nó giao tiếp bằng cách sử dụng giao thức ban đầu STK500 (tham chiếu, tập tin tiêu đề viết bằng ngôn ngữ C). Ta cũng có thể bỏ qua bộ nạp khởi động và chương trình vi điều khiển thông qua ICSP (In-Circuit Serial Programming). Ở các dòng ATmega16U2 (hoặc 8U2) mã nguồn phần mềm có sẵn. Các ATmega16U2/8U2 được nạp với một bộ nạp khởi động DFU, mà có thể được kích hoạt bằng cách:

- Trên thế hệ bo mạch thứ 1: kết nối jumper được hàn ở mặt sau của bo mạch và sau đó cài đặt lại 8U2.

- Trên thế hệ bo mạch thứ 2 hoặc cuối cùng: có một điện trở nối HWB 8U2/16U2 với đất, làm cho nó dễ dàng hơn trong việc đặt chế độ DFU.

Sau đó có thể sử dụng phần mềm FLIP Atmel (Windows) hoặc lập trình DFU (Mac OS X và Linux) để tải một phần mềm mới. Hoặc có thể sử dụng header ISP lập trình ngoài (ghi đè lên các bộ nạp khởi động DFU).

* Tự động reset (phần mềm).

Thay vì reset vật lý trước khi tải dữ liệu, Arduino Uno được thiết kế để cho phép thay thế bằng một phần mềm chạy trên máy tính khi được kết nối. Một trong những dòng điều khiển phần cứng (DTR) của ATmega8U2/16U2 được kết nối với đường dây được thiết lập của ATmega328 thông qua một tụ điện 100 nF. Khi dòng này được xác định (thấp nhất) , dòng reset rơi đủ lớn để reset lại chip. Phần mềm Arduino sử dụng khả năng này để cho phép tải code lên bằng cách nhấn nút upload trong môi trường Arduino. Điều này có nghĩa là bộ nạp khởi động có thể có một thời gian chờ ngắn hơn, giảm thiểu DTR có thể phối hợp khi bắt đầu tải lên.

Thiết lập này có ý nghĩa khác. Khi Uno được kết nối với một trong hai máy tính chạy Mac OS X hoặc Linux, nó reset mỗi khi kết nối được thực hiện từ phần mềm (thông qua cổng USB). Trong nửa giây hoặc lâu hơn, bộ nạp khởi động chạy trên Uno. Trong khi nó được lập trình để bỏ qua dữ liệu bị thay đổi, nó sẽ ngăn chặn các byte đầu tiên của dữ liệu gửi đến bo mạch khi kết nối được thiết lập. Nếu 20 một bản sketch chạy trên bo mạch nhận được cấu hình một lần hoặc các dữ liệu khác khi lần đầu tiên bắt đầu, chắc chắn rằng phần mềm mà bo mạch giao tiếp chờ đợi một giây sau khi mở kết nối và trước khi gửi dữ liệu này.

* Bảo vệ quá dòng USB.

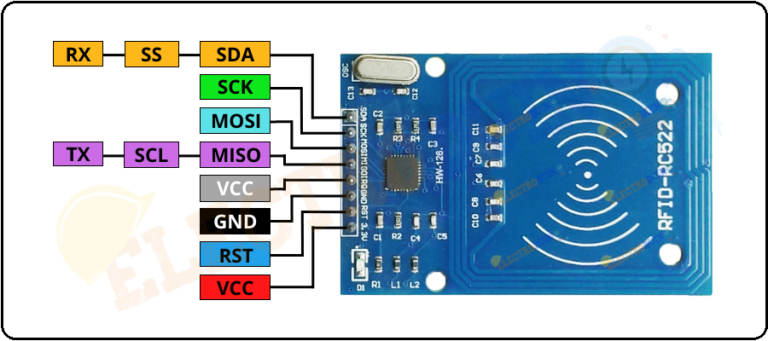
Arduino Uno có một bảng thiết lập được sử dụng để bảo vệ cổng USB của máy tính khi ngắn mạch và quá dòng. Mặc dù hầu hết các máy tính cung cấp chế độ bảo vệ nội bộ, nhưng còn có thêm các cầu chì, các cầu chì này có thêm một lớp bảo vệ. Nếu dòng cao hơn 500 mA được áp dụng cho các cổng USB, cầu chì sẽ tự động phá vỡ các kết nối khi ngắn mạch và quá dòng xảy ra.

Đặc tính vật lý.

Chiều dài tối đa và chiều rộng tương ứng của Uno PCB là 2,7 và 2,1 inch, với các kết nối USB và jack nguồn thì sẽ vượt ra ngoài kích thước cũ. Bốn lỗ vít trên bo mạch dùng để gắn vào một bề mặt trong một số trường hợp. Lưu ý rằng khoảng cách giữa chân số 7 và 8 là 160 mil (0,16 ").

## **2.3.2 GIỚI THIỆU VỀ MODULE RFID RC522**

1. **Tổng quan**



Hình 2.3: Module RFID RC522

Mô-đun đọc/ghi RFID RC522 (Transceiver) dựa trên vi mạch đọc/ghi tích hợp cao MFRC522 của công ty NXP. Nó được sử dụng cho việc truyền thông không tiếp xúc tại tần số 13.56 MHz. RFID viết tắt của Radio Frequency Identification, tức là Nhận diện Tần số Radio. Mô-đun này sử dụng sóng điện từ trong tần số radio để truyền dữ liệu (đọc/ghi). Nó có thể đọc/ghi tất cả các loại Transponder (thẻ RFID và thẻ key fob) có bộ nhớ 1KB và tương thích với tần số 13.56 MHz. Đây là một mô-đun có điện áp thấp, giá rẻ, kích thước nhỏ và đi kèm giao thức SPI, cho phép nó dễ dàng giao tiếp với hầu hết các vi điều khiển như ATTiny, Arduino, ESP8266, Raspberry Pi và các bo mạch phát triển tiên tiến khác.

Mô-đun đọc/ghi RFID RC522 bao gồm 3 thành phần chính, đó là vi mạch MFRC522, dao động tinh thể 27.12 MHz và ăng-ten :

* Vi mạch MFRC522/Chip: Mô-đun đọc RFID RC522 dựa trên vi mạch MFRC522/Chip. Đây là vi mạch đọc thẻ RFID tích hợp cao được thiết kế bởi công ty NXP, hoạt động trong giao tiếp không tiếp xúc ở tần số 13.56MHz. Đó là một vi mạch đọc và ghi thẻ nhỏ gọn, tiêu thụ công suất thấp và giá thành rẻ. Vi mạch MFRC522 hỗ trợ nhiều loại thẻ RFID như MIFARE 1K, MIFARE 4K, MIFARE Mini và các loại thẻ và tag tuân thủ chuẩn giao thức ISO / IEC 14443 khác. Ngoài ra, nó hỗ trợ giao tiếp không tiếp xúc tốc độ cao của chuỗi Mifare, tốc độ giao tiếp song song lên đến 424 kb/s. Vi mạch MFRC522 hoạt động ở tần số 13.46MHz với phạm vi hoạt động lên đến 50mm tùy thuộc vào kích thước và điều chỉnh của ăng-ten. Vi mạch MFRC522 hỗ trợ giao tiếp chuỗi SPI, UART và I2C với máy chủ (vi điều khiển như Arduino).
* Dao động tinh thể 27.12MHz: Một tinh thể quartz 27.12MHz được kết nối với chân OSCIN và OSCOUT của vi mạch để tạo ra dao động nội. Xung đồng hồ 13.56MHz được phát sinh từ xung đồng hồ 27.12MHz của tinh thể quartz 27.12MHz chia cho 2.
* Ăng-ten: Một cuộn NFC được nhúng trong bo mạch của mô-đun. Đây là một ăng-ten phát ra một trường elektromagnet tần số cao 13.56MHz. Nó hỗ trợ các thành phần không hoạt động 13.56MHz.

Module đọc thẻ RC522 có thể đọc được tất cả loại thẻ không dây như NCF , thẻ từ ( thẻ giảm giá, thẻ xe bus, tàu điện ngâm,..)

1. **Cơ chế đọc/ghi của RFID RC522**

Trước hết, mô-đun liên tục phát ra các sóng điện từ mà bao phủ một chiều cao là 60 mm trên các bề mặt cùng điện tiềm. Vi mạch MFRC522 ở trạng thái chờ đợi mà không có bất kỳ sự xao lạc nào trong trường điện từ. Ngay khi một thẻ tiếp xúc với khu vực sóng điện từ, vi mạch sẽ phát ra một tín hiệu ngắt có thể đọc trên chân IRQ. Hơn nữa, nó cũng cung cấp tín hiệu cho vi mạch, cho biết rằng có một thẻ hoặc tag nào đó.

Khi một thẻ tiếp xúc với trường điện từ này, chúng trở nên hoạt động. Vì thẻ và tag không có nguồn điện, chúng tạo ra EMF/sự chênh lệch tiềm năng/điện áp với trường điện từ này và tạo ra một trường điện từ đối lập trong khu vực đó. Vi mạch nhận ra sự xao lạc trong trường từ và cuộn bắt đầu nhận các biến động trong trường điện từ đối lập. Phương pháp này được gọi là back scattering. Thẻ hoặc tag có một vi mạch đặc biệt không làm nhiều công việc. Chúng chỉ hoạt động trên điện áp được tạo ra bởi cuộn được nhúng vào chúng.

Khi vi mạch trong thẻ hoạt động, nó bắt đầu tạo ra các sóng điện từ đối lập đại diện cho dữ liệu được lưu trữ trong thẻ. Những biến động này sau đó được đầu đọc nhận và giải mã thành dữ liệu chính xác dưới dạng có thể đọc được bởi con người. Trong quá trình truyền dữ liệu, đầu đọc không phát ra sóng, mà chỉ đọc chúng vì nó phải giải mã những sóng này thành dữ liệu. Chắc chắn, có đủ công suất để thẻ truyền toàn bộ dữ liệu. Có những băm cố định chỉ ra điểm bắt đầu và kết thúc dữ liệu, dữ liệu không hoàn chỉnh hoặc không đủ sẽ gây ra lỗi.

Mặc dù mô-đun có thể được cấu hình bằng bất kỳ một trong ba cách đã đề cập, cách thông dụng và phổ biến nhất là giao tiếp SPI. Trong giao tiếp này, ngoài các dòng nguồn, có 4 chân nguồn điện được sử dụng, chân IRQ là tùy chọn và có thể được sử dụng cho các tình huống khác hoặc ngắt. MISO, MOSI, SS và SCK là các chân cần được kết nối với cổng SPI của vi điều khiển để truyền thông dữ liệu qua lại. Tuy nhiên, bạn phải đảm bảo rằng vi điều khiển bạn đang sử dụng có cùng mức logic như mô-đun, nếu không, nó có thể gây ra một số vấn đề.

Đối với vấn đề này, bạn có thể sử dụng một bộ điều chỉnh mức logic (logic level shifter), điều này rất dễ dàng tìm thấy. Tôi khuyến nghị điều này vì tín hiệu quá áp có thể làm hỏng mô-đun của bạn, và tôi hy vọng bạn không muốn điều đó xảy ra. MFRC522 liên tục tạo ra Mặt Trường Điện từ (Sóng Radio) tần số 13.56MHz, đó là tần số truyền thông (môi trường). Vi mạch sẽ ở trạng thái chờ đợi cho đến khi một thẻ tiếp xúc với trường điện từ được phát ra bởi cuộn của mô-đun.

1. **Tóm tắt thông số RFID RC522**

**A screenshot of a computer

Description automatically generated with medium confidence**

Hình 2.4: Sơ đồ mạch điện RFID RC522

Cấu hình chân :

Bảng 2 : Cấu hình chân RFID RC522

|  |  |
| --- | --- |
| **Kí tự chân** | **Mô tả chân** |
| VCC | Đây là chân cấp nguồn cho module. Module hoạt động trong dải điện áp từ 2.5 đến 3.3 volt. Chúng ta có thể kết nối nó với chân đầu ra 3.3V của vi điều khiển (Arduino). |
| RST | Chân reset là chân vào để đặt lại và tắt nguồn của module. Khi chân này đi xuống, chế độ tắt nguồn được kích hoạt, sau đó tất cả các nguồn dòng điện bên trong, bao gồm cả dao động và các chân vào của module, sẽ được ngắt kết nối với thế giới bên ngoài. |
| GND | chân mát của module |
| IRQ | chân ngắt để báo cho vi điều khiển biết khi module đọc thẻ RFID trong phạm vi của nó. Nó giúp module vào chế độ ngủ để tiết kiệm năng lượng |
| MISO / SCL / Tx | Khi giao diện SPI được kích hoạt, chân này hoạt động như Master-In-Slave-Out. Khi giao diện I2C được kích hoạt, chân này hoạt động như xung đồng hồ tuần tự, và khi giao diện UART được kích hoạt, nó hoạt động như đầu ra dữ liệu tuần tự. |
| MOSI | chân Master Out Slave In cho giao tiếp SPI. |
| SCK | Chân xung đồng hồ tuần tự. Nó nhận các xung đồng hồ được cung cấp bởi bus SPI Master, tức là vi điều khiển (Arduino). |
| SS / SDA / RX | Chân này hoạt động như đầu vào dữ liệu tuần tự (SS) trong giao tiếp SPI, SDA trong giao tiếp I2C, và Rx trong giao tiếp UART. |

Thông số kỹ thuật :

* Điện áp hoạt động: 2.5V~3.3V.
* Dòng hoạt động: 13 - 26mA ở 3.3V.
* Dòng chờ: 10 - 13mA ở 3.3V.
* Dòng ngủ: <80uA.
* Dòng cao nhất: <30mA.
* Tần số hoạt động: 13.56MHz.
* Tương thích với thẻ ISO 14443A và thẻ MIFARE 1K, cũng như thẻ key fob.
* Hỗ trợ tốc độ truyền thông cao hơn ISO/IEC 14443A: lên đến 848 KBd.
* Hỗ trợ tốc độ bus SPI: lên đến 10Mbit/s.
* Hỗ trợ giao diện I2C-bus: ở chế độ Fast mode tốc độ: lên đến 400 kBd, ở chế độ High-speed mode: lên đến 3400 kBd.
* Tốc độ RS232 Serial UART: lên đến 1228.8 kBd, với mức điện áp phụ thuộc vào nguồn cung cấp của chân.
* Phạm vi hoạt động điển hình: lên đến 50mm ở chế độ Đọc/Ghi, tùy thuộc vào kích thước và điều chỉnh của ăng-ten.
* Nhiệt độ hoạt động: -20 độ C đến 80 độ C.
* Nhiệt độ lưu trữ: -40 độ C đến 85 độ C.
* Độ ẩm: Độ ẩm liên quan từ 5% đến 95%.
* Kích thước bo mạch: 40mm × 60mm.
* Khoảng cách hoạt động 0~60mm

## **2.3.3 KHỐI MA TRẬN PHÍM 4X3 :**

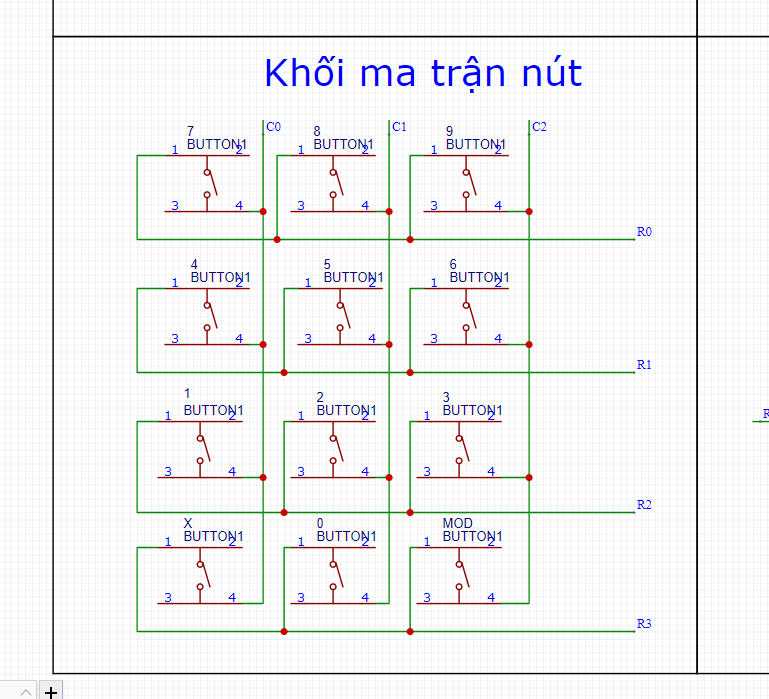
Khối ma trận phím 4x3 là một thành phần phổ biến được sử dụng trong các ứng dụng điện tử và điều khiển. Nó bao gồm một ma trận gồm 4 hàng và 3 cột các phím, tổng cộng có 12 phím.

Mỗi phím trên ma trận có thể được kết nối đến một đường dẫn điện tử riêng, tạo thành một mạng lưới ma trận. Khi một phím được nhấn, nó tạo ra một kết nối giữa hàng và cột tương ứng trên ma trận.

Các phím trên ma trận phổ biến được đánh số từ 0 đến 9 và có thể bao gồm các phím chức năng bổ sung như \*, # hoặc các phím điều hướng (up, down, left, right). Mỗi phím được đại diện bởi một cặp giá trị hàng-cột trên ma trận.

Khi một phím được nhấn, hệ thống điện tử đọc các tín hiệu tương ứng trên ma trận. Bằng cách phân tích hàng và cột kích hoạt, nó xác định phím cụ thể được nhấn. Thông qua việc giao tiếp với vi điều khiển hoặc mạch điện tử khác, thông tin về phím nhấn có thể được sử dụng để thực hiện các hành động hoặc kiểm soát các chức năng khác nhau trong hệ thống.

Khối ma trận phím 4x3 thường được sử dụng trong các ứng dụng như bàn phím điện thoại, bàn điều khiển từ xa, màn hình cảm ứng và các hệ thống điều khiển đơn giản khác. Nó cung cấp một cách thuận tiện và tiết kiệm không gian để tương tác với các thiết bị điện tử.



Hình 2.4. Sơ đồ khối ma trận nút bấm 4x3

## **2.3.4 KHỐI HIỂN THỊ LCD – LIQUID CRYTAL DISPLAY**

1. **LCD 1602**

Màn hình text LCD1602 xanh lá có khả năng hiển thị 2 dòng với mỗi dòng 16 ký tự. Khi sản xuất LCD, nhà xản xuất đã tích hợp chíp điều khiển (HD44780) bên trong lớp vỏ và đưa các chân giao tiếp cần thiết



Hình 2.5 : Màn hình LCD 1602

Thông số kĩ thuật :

* Điện áp hoạt động: 4.5V - 5.5V (thường là 5V)..
* Kích thước: 80 x 36 x 12.5 mm
* Số ký tự hiển thị: 16 ký tự trên 2 dòng.
* Chữ đen, nền xanh lá
* Khoảng cách giữa hai chân kết nối là 0.1 inch tiện dụng khi kết nối với Breadboard.
* Nhiệt độ hoạt động: -20°C đến 70°C.
* Tên các chân được ghi ở mặt sau của màn hình LCD hổ trợ việc kết nối, đi dây điện.
* Có đèn led nền, có thể dùng biến trở hoặc PWM điều chình độ sáng để sử dụng ít điện năng hơn. Có thể được điều khiển với 6 dây tín hiệu
* Driver của LCD1602 là một vi mạch điều khiển (controller) có mã số HD44780 hoặc tương tự. Nó giúp điều khiển hiển thị và giao tiếp giữa Arduino hoặc các vi điều khiển khác với màn hình LCD1602.
* Có bộ ký tự được xây dựng hổ trợ tiếng Anh và tiếng Nhật

Sơ đồ nôi chân LCD:

Bảng 3: Bảng chân của LCD

|  |  |  |
| --- | --- | --- |
| **Số chân** | **Ký hiệu chân** | **Mô tả chân** |
| 1 | Vss | Cấp điện 0v |
| 2 | Vcc | Cấp điện 5v |
| 3 | V0 | Chỉnh độ tương phản |
| 4 | RS | Lựa chọn thanh ghi địa chỉ hay dữ liệu. Nối chân RS với logic “ 1 “ (VCC) để chọn thanh ghi ghi dữ liệu , logic “0” (GND) Ghi dữ liệu. |
| 5 | RW | Lựa chọn thanh ghi Đọc hay Viết |
| 6 | EN | Cho phép xuất dữ liệu |
| 7 | D0 | Đường truyền dữ liệu 0 |
| 8 | D1 | Đường truyền dữ liệu 1 |
| 9 | D2 | Đường truyền dữ liệu 2 |
| 10 | D3 | Đường truyền dữ liệu 3 |
| 11 | D4 | Đường truyền dữ liệu 4 |
| 12 | D5 | Đường truyền dữ liệu 5 |
| 13 | D6 | Đường truyền dữ liệu 6 |
| 14 | D7 | Đường truyền dữ liệu 7 |
| 15 | A | Chân dương đèn màn hình |
| 16 | K | Chân âm đèn màn hình |

1. **Địa chỉ vùng nhớ LCD**

Địa chỉ ba vùng nhớ (hay còn gọi là địa chỉ bộ nhớ) của một màn hình LCD (Liquid Crystal Display) thường được sử dụng trong các dự án nhúng và điều khiển LCD thông qua giao diện song song. Các vùng nhớ này là:

* Địa chỉ vùng nhớ dòng (DDRAM - Display Data RAM): Đây là vùng nhớ chứa dữ liệu hiển thị trên màn hình LCD, bao gồm các ký tự, biểu tượng, v.v. Mỗi ô nhớ trong DDRAM được định đạc bằng một địa chỉ duy nhất, từ 0x00 đến 0x4F (tương ứng với 80 cột và 2 dòng trên màn hình LCD 16x2).
* Địa chỉ vùng nhớ lệnh (IR - Instruction Register): Đây là vùng nhớ dùng để gửi các lệnh điều khiển cho màn hình LCD, chẳng hạn như lệnh điều khiển hiển thị, lệnh di chuyển con trỏ, lệnh xóa màn hình, v.v. Các lệnh được gửi đến IR thông qua các chân điều khiển của LCD.
* Địa chỉ vùng nhớ bộ nhớ ký tự (CGRAM - Character Generator RAM): Đây là vùng nhớ dùng để lưu trữ các dữ liệu ký tự đặc biệt, được người dùng tự định nghĩa. Có thể dùng để tạo ra các ký tự đặc biệt hoặc biểu tượng riêng của người dùng trên màn hình LCD.

1. **Các lệnh điều khiển**

Table

Description automatically generated

Hình 2.6 : Tập lệnh trên LCD

* Lệnh thiết lập chức năng giao tiếp "Function set":
* Bit DL (data length) = 1 cho phép giao tiếp 8 đường data D7 ÷ D0, nếu bằng 0 thì cho phép giao tiếp 4 đường D7 ÷ D4.
* Bit N (number of line) = 1 cho phép hiển thị 2 hàng, nếu bằng 0 thì cho phép hiển thị 1 hàng.
* Bit F (font) = 1 cho phép hiển thị với ma trận 5×8, nếu bằng 0 thì cho phép hiển thị với ma trận 5×11.
* Lệnh xoá màn hình "Clear display":
* Khi thực hiện lệnh này, thì LCD sẽ bị xoá và bộ đếm địa chỉ được xoá về 0.
* Lệnh di chuyển con trỏ về đầu màn hình "Cursor Home":
* Khi thực hiện lệnh này, bộ đếm địa chỉ được xoá về 0, phần hiển thị trở về vị trí gốc đã bị dịch trước đó. Nội dung bộ nhớ RAM hiển thị DDRAM không bị thay đổi.
* Lệnh thiết lập lối vào "Entry mode set":
* Lệnh này dùng để thiết lập lối vào cho các kí tự hiển thị.
* Bit I/D = 1 thì con trỏ tự động tăng lên 1 mỗi khi có 1 byte dữ liệu ghi vào bộ hiển thị, khi I/D = 0 thì con trỏ sẽ tự động giảm đi 1 mỗi khi có 1 byte dữ liệu ghi vào bộ hiển thị.
* Bit S = 1 thì cho phép dịch chuyển dữ liệu mỗi khi nhận 1 byte hiển thị.
* Lệnh điều khiển con trỏ hiển thị "Display Control":
* Bit D: cho phép LCD hiển thị thì D = 1, không cho hiển thị thì bit D = 0.
* Bit C: cho phép con trỏ hiển thị thì C= 1, không cho hiển thị con trỏ thì bit C = 0.
* Bit B: cho phép con trỏ nhấp nháy thì B= 1, không cho con trỏ nhấp nháy thì bit B = 0.
* Với các bit như trên thì để hiển thị phải cho D = 1, 2 bit còn lại thì tùy chọn, trong thư viện thì cho 2 bit đều bằng 0, không cho phép mở con trỏ và nhấp nháy, nếu bạn không thích thì có thể hiệu chỉnh lại các bit trong thư viện để cho phép mở con trỏ và nhấp nháy nếu bạn muốn.
* Lệnh di chuyển con trỏ "Cursor / Display Shift" được sử dụng để điều khiển di chuyển con trỏ hiển thị dịch chuyển.
* Bit SC: SC = 1 cho phép dịch chuyển, SC = 0 thì không cho phép.
* Bit RL xác định hướng dịch chuyển: RL = 1 thì dịch phải, RL = 0 thì dịch trái. Nội dung bộ nhớ DDRAM vẫn không đổi.
* Vậy khi cho phép dịch chuyển, bạn có 2 tùy chọn: dịch trái và dịch phải.
* Lệnh thiết lập địa chỉ cho bộ nhớ RAM phát kí tự "Set CGRAM Addr" được sử dụng để thiết lập địa chỉ cho bộ nhớ RAM phát kí tự.
* Lệnh thiết lập địa chỉ cho bộ nhớ RAM hiển thị "Set DDRAM Addr" được sử dụng để thiết lập địa chỉ cho bộ nhớ RAM lưu trữ các dữ liệu hiển thị.
* Hai lệnh cuối cùng là lệnh đọc và lệnh ghi dữ liệu LCD, được sử dụng để đọc dữ liệu từ bộ nhớ RAM hoặc ghi dữ liệu vào bộ nhớ RAM của LCD.

1. **Giao tiếp I2C**

I2C (Inter-Integrated Circuit) là một chuẩn giao tiếp hai dây được phát triển bởi Philips (nay là NXP) dùng để kết nối giữa các thiết bị điện tử trong một hệ thống, ví dụ như giữa vi điều khiển (microcontroller) và các thiết bị ngoại vi như cảm biến, bộ nhớ, hiển thị, IC điều khiển, và nhiều loại thiết bị khác.

Cách hoạt động của giao tiếp I2C dựa trên nguyên tắc truyền dẫn dữ liệu theo kiểu chuỗi (serial) trên hai dây: SDA (Serial Data) và SCL (Serial Clock). Dưới đây là quá trình hoạt động cơ bản của giao tiếp I2C:

* Khởi đầu (Start): Trong quá trình hoạt động, master (thiết bị điều khiển chính) sẽ gởi tín hiệu khởi đầu (Start) bằng cách giữ dây SDA ở mức cao và rồi kéo dây SDA xuống mức thấp trong khi giữ dây SCL ở mức cao.
* Gởi địa chỉ (Address): Master sau đó gởi địa chỉ của thiết bị nô lệ (thiết bị nhận dữ liệu) trên dây SDA, bao gồm cả bit R/W để chỉ định hoạt động đọc (Read) hay ghi (Write). Thiết bị nô lệ nào có địa chỉ trùng khớp với địa chỉ gởi sẽ phản hồi lại.
* Truyền dữ liệu (Data Transfer): Sau khi đã giao tiếp với thiết bị nô lệ cụ thể, master có thể truyền hoặc nhận dữ liệu. Trong quá trình truyền, master gởi dữ liệu trên dây SDA và đợi phản hồi từ thiết bị nô lệ để xác nhận dữ liệu đã được nhận đúng (ACK) hoặc không (NACK). Trong quá trình nhận, master nhận dữ liệu trên dây SDA và phản hồi lại bằng tín hiệu ACK hoặc NACK để thông báo cho thiết bị nô lệ truyền dữ liệu tiếp theo hoặc kết thúc quá trình truyền.
* Kết thúc (Stop): Sau khi hoàn thành quá trình truyền dữ liệu, master gởi tín hiệu kết thúc (Stop) bằng cách đưa dây SDA từ mức thấp lên mức cao trong khi giữ dây SCL ở mức cao.

1. **PCF8574**



Hình 2.7 : IC PCF8574

PCF8574: là một bộ đổi mức độ cao/độ thấp (I/O expander) dựa trên giao tiếp I2C (Inter-Integrated Circuit) của hãng NXP (trước đây là Philips). Nó cho phép mở rộng số lượng đầu vào/đầu ra của vi điều khiển (ví dụ như Arduino) thông qua giao tiếp I2C, giúp giảm số lượng chân kết nối cần thiết để điều khiển các thiết bị ngoại vi.

Thông số kỹ thuật chung của PCF8574 như sau:

* Điện áp hoạt động: 2,5V - 6V.
* Tín hiệu giao tiếp: I2C (tốc độ tối đa 400kHz).
* Số lượng đầu vào/đầu ra: 8 chân.
* Công suất tiêu thụ thấp: 100 μA (đỉnh) / 400 nA (chế độ chờ).
* Dòng tối đa trên mỗi chân: 25 mA
* Dòng tối đa của toàn IC: 125 mA
* Điện trở kéo lên nội bộ: 100 kΩ (tương đương với đầu vào cao).
* Nhiệt độ hoạt động: -40°C đến +85°C.

Diagram

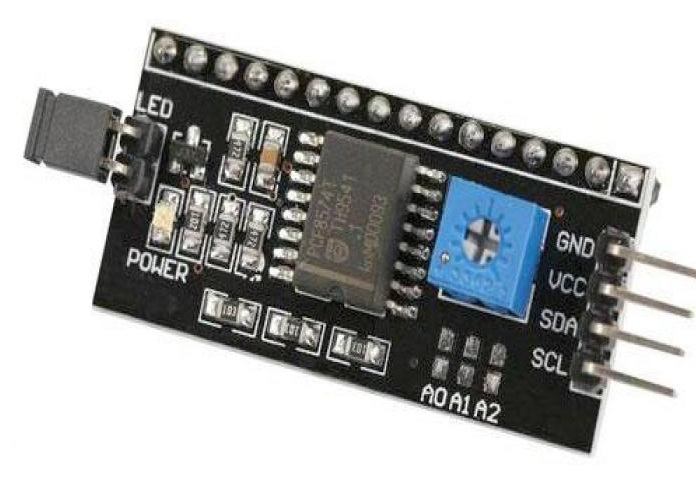
Description automatically generated with low confidence

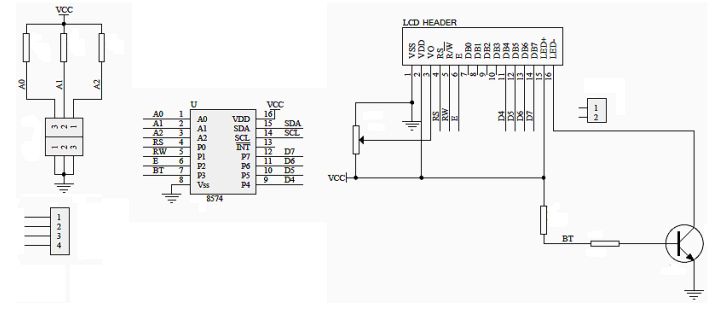
Hình 2.8 : Sơ đồ chân PCF8574

Trong đó:

* P0 đến P7 là các chân đầu vào/đầu ra của PCF8574, tương ứng với 8 chân I/O mở rộng.
* SDA và SCL là các chân dữ liệu và xung clock của giao tiếp I2C.
* A2, A1, A0 là các chân địa chỉ để định địa chỉ I2C của PCF8574, cho phép kết nối nhiều PCF8574 cùng lúc trên cùng một dòng I2C.
* VCC là nguồn cấp 5V, GND là đất.

1. **Mô đun chuyển đổi I2C**





Hình 2.9 : Mô đun chuyển chuyển đổi I2C HW-061

## **2.3.5 KHỐI RELAY VÀ KHÓA ĐIỆN TỬ**

1. **Khối Relay**

A picture containing diagram, text, line, plan

Description automatically generated

Hình 2.10 : Sơ đồ mạch điện khối Relay

Cấu tạo chính :

* Điện trở 1k Ω
* Diode 1N4007
* Relay 5V-10A 5 chân SRD-05VDC-SL-C
* 2 header XH2.54

Chức năng chính của mô-đun relay là chuyển đổi các thiết bị và hệ thống điện lên hoặc tắt. Nó cũng được sử dụng để cách ly mạch điều khiển với thiết bị hoặc hệ thống đang được điều khiển.

Điều này rất quan trọng vì nó cho phép bạn sử dụng vi điều khiển hoặc các thiết bị công suất thấp khác để điều khiển các thiết bị với điện áp và dòng điện cao hơn nhiều.

Mô-đun relay bao gồm một relay, đó là một công tắc điện từ cơ khí, và mạch hỗ trợ. Relay có hai trạng thái chính: trạng thái mở mặc định (NO) và trạng thái đóng mặc định (NC). Khi relay không được cấp nguồn, nó ở trạng thái mặc định của nó, có thể là NO hoặc NC, phụ thuộc vào loại relay.

Để điều khiển relay, một mạch điều khiển được kết nối với mô-đun relay. Mạch điều khiển có thể là vi điều khiển, mạch logic số, hoặc bất kỳ thiết bị điều khiển phù hợp nào khác. Mạch điều khiển cung cấp tín hiệu cần thiết để cấp nguồn hoặc ngắt nguồn cho cuộn relay.

Khi cuộn relay được cấp nguồn, nó tạo ra một trường từ hút các tiếp điểm của relay, làm cho chúng chuyển từ trạng thái mặc định sang trạng thái đối diện. Ví dụ, trong trường hợp của relay mở mặc định (NO), khi cuộn được cấp nguồn, các tiếp điểm đóng, cho phép dòng điện chảy qua mạch được điều khiển. Trong trường hợp của relay đóng mặc định (NC), khi cuộn được cấp nguồn, các tiếp điểm mở, làm gián đoạn dòng điện chảy.

1. **Chốt khóa điện tử**



Hình 2.11 : Chốt khóa điện tử 12V

Thông số :

* Nguồn điện: 12VDC
* Dòng điện: dòng khởi động: 0.8A; dòng làm việc 0.2A
* Yêu cầu nguồn cấp: 12VDC/1-3A
* Kích thước: L20xD3.4xH4.2 cm
* Trọng lượng: 0.46kg
* Vật liệu hộp kim nhôm

## **2.3.6 KHỐI NGUỒN:**

Tạo ra dòng điện ổn định cung cấp cho toàn mạch hoạt động.

Bảng 4 : Điện áp và dòng làm việc của các linh kiện

|  |  |
| --- | --- |
| Tên linh kiện | Điện áp làm việc |
| Arduino Uno | 5V |
| Nguồn 3.3V RFID | 3.3V |
| RFID | 3.3V |
| LCD 16x2 | 5V |
| Ma trận Phím 4x3 | 5V |
| Relay | 5V |

Theo bảng điện áp làm việc, ta chọn nguồn cấp cho mạch là 5V.

Sử dụng máy biến áp nguồn ra là 5 VAC và dòng 3A đủ để cấp dòng và áp cho khối nguồn.

* **IC ỔN ÁP:**

Với những mạch điện không đòi hỏi độ ổn định của điện áp quá cao, sử dụng IC ổn áp thường được người thiết kế sử dụng vì mạch điện khá đơn giản. Các loại ổn áp thường được sử dụng là IC 78xx. Họ 78xx là họ cho ổn định điện áp đầu ra là dương. Còn xx là giá trị điện áp đầu ra như 5V, 6V. Ví dụ 7805 ổn áp 5V, 7808 ổn áp 8V, 7812 ổn áp 12V. Việc dùng các loại IC ổn áp 78xx tương tự nhau….

78xx gồm có 3 chân  
1: V in - Chân nguồn đầu vào.  
2: GND - Chân nối đất.  
3: V out - chân nguồn đầu ra.

* **ỔN ÁP 7805**: *Là IC ổn áp 5V, tương ứng với dòng là 1A.*

Với những mạch điện không đòi hỏi độ ổn định của điện áp quá cao, sử dụng IC ổn áp thường được người thiết kế sử dụng vì mạch điện khá đơn giản. Linh kiện ổn áp 7805 được dùng rất nhiều trong các mạch điện điều khiển dùng để cấp nguồn ổn định cho mạch. Ưu điểm là dễ ghép nối, dễ thiết kế với chi phí thấp, nguồn đầu ra ổn định. Nhược điểm của nó là công suất đầu ra khá thấp (1A) và hoạt động không ổn định khi có nhiễu bên ngoài. Hoạt động được ở giải nhiệt độ khá cao là 0 -125 độ C.

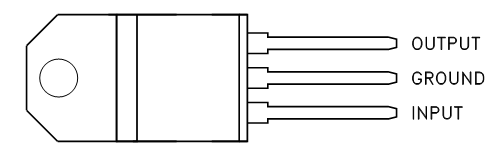
7805 có 3 chân kết nối:

* Chân 1 là chân nguồn đầu vào.
* Chân 2 là chân GND.
* Chân 3 là chân lấy điện áp ra.

Chân 1 - 2 (Chân điện áp đầu vào): Đây là chân cấp nguồn đầu vào cho 7805 hoạt động. Dải điện áp cho phép đầu vào lớn nhất là 40V. Theo datasheet thì giải điện áp đầu ra là 5V ta nên cho điện áp vào là 35V để mạch lúc nào cũng hoạt động ổn định điện áp không bị lên xuống do nguồn đầu vào.

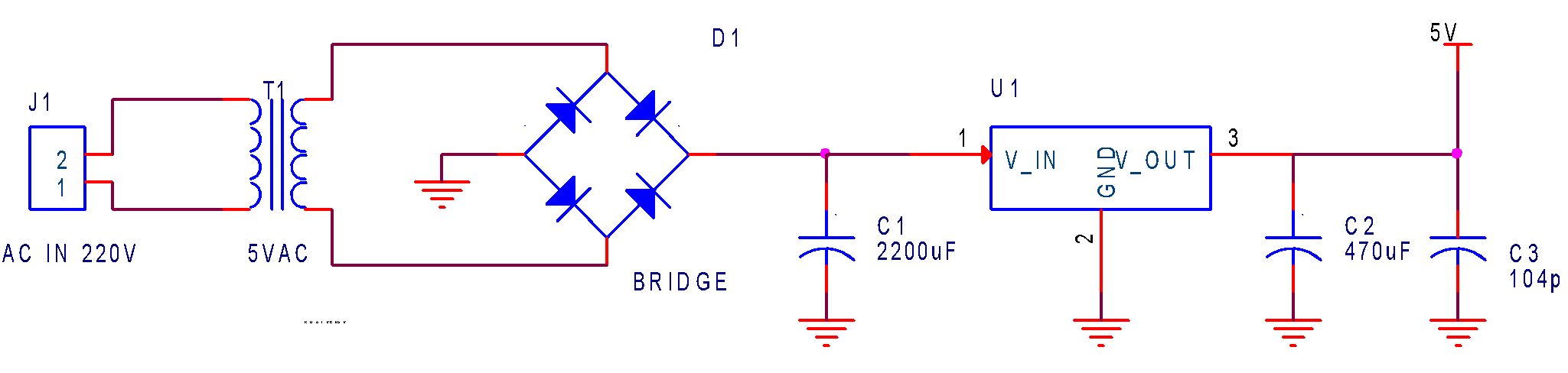
Chân 3 (Chân điện áp đầu ra): Chân này cho chúng ta lấy điện áp đầu ra ổn định 5V.

* Đảm bảo đầu ra ổn định luôn nằm trong giải từ (4.75V đến 5.25V).
* Đảm bảo thông số: V in – V out > 3V. Thông số này phải luôn đảm bảo khi cấp nguồn cho 7805. Tức là điện áp cấp vào cho 7805 phải nằm trong 8V đến 40V. Nếu dưới 8V thì mạch ổn áp không còn tác dụng. Thông thường người ta không bao giờ cấp nguồn 8V vào cả mà người ta phải cấp nguồn lớn hơn ít nhất là gấp đôi nguồn đầu ra để tráng trường hợp sụt áp đầu vào sinh ra nguồn đầu ra không ổn định trong thời gian ngắn.
* Đảm bảo tản nhiệt tốt cho 7805 khi chạy với tải. Khi công suất tăng lên thì do 7805 là linh kiện bán dẫn công suất nên rất nóng khi tải lớn. Để tráng hỏng linh kiện và cho linh kiện hoạt động trong nhiệt độ bình thường thì cần phải tản nhiệt tốt.



Hình 2.12 : Sơ đồ chân 7805*.*

* **LỌC NGUỒN VÀ LỌC NHIỄU:**
* Dùng tụ hóa C1 để lọc điện áp. Vì đây là điện áp 1 chiều nhưng chưa được phẳng vẫn còn các gợn nhấp nhô nên tụ này có tác dụng lọc nguồn cho thành điện áp một chiều phẳng.
* Tụ C1 là lọc nguồn đầu vào cho 7805. Tụ này là tụ hóa phải có điện dung đủ lớn để lọc phẳng điện áp đầu vào và điện áp tụ chịu đựng phải lớn hơn điện áp đầu vào.
* Tụ C2 là lọc nguồn đầu ra cho 7805. Tụ này cũng là tụ hóa dùng để lọc nguồn đầu ra cho bằng phẳng.
* Trong thành phần một chiều còn có các sóng điều hòa bậc 2, 3..., sóng nhấp nhô có tần số cao, nhiễu bên ngoài. Các sóng này ảnh hưởng đến hoạt động của 7805. Nếu trong mạch tồn tại những thành phần sóng này sẽ làm sai sót khó phát hiện trong mạch làm cho mạch hoạt động không ổn định. Tụ lọc nhiễu tần số cao C3. Tụ này phải là tụ không phân cực, tụ Ceramic. Tụ này lọc các thành phần trên cho đầu ra đảm bảo cho mạch hoạt động bình thường.  
  ***SƠ ĐỒ KHỐI NGUỒN:***

******

Hình 2.13 : Sơ đồ khối nguồn 5V.

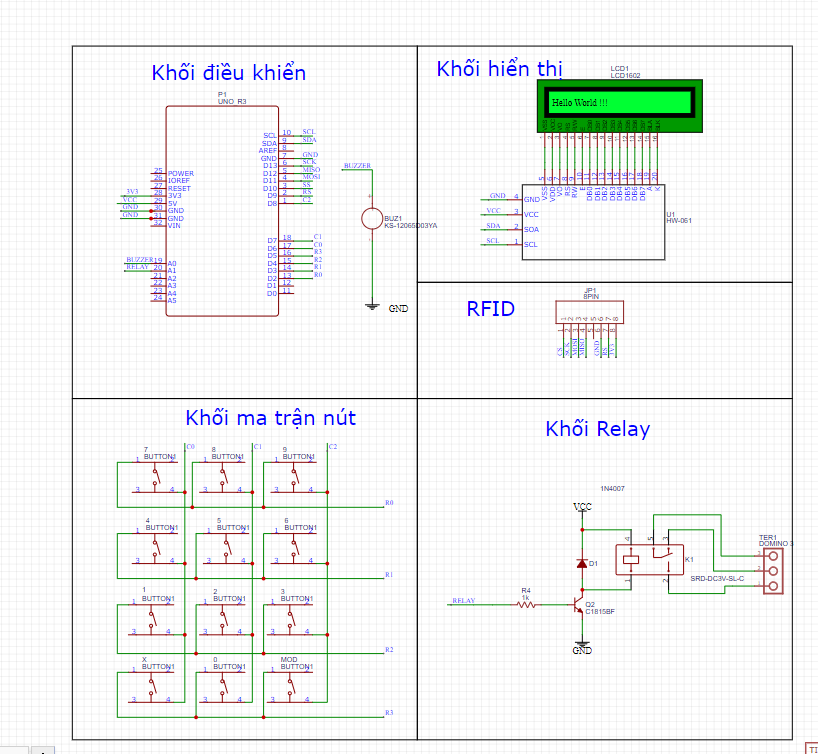
Dựa vào bảng điện áp, dòng làm việc và tính toán thiết kế, ta có thể sử dụng adapter 5V 3A để cấp nguồn cho mạch hoạt động.



Hình 2.14 : Adapter 5V 3A

# SƠ ĐỒ NGUYÊN LÝ TOÀN MẠCH:

Dựa trên các tính toán, lựa chọn linh kiện, tiến hành thiết kế sơ đồ nguyên lý chung cho toàn mạch và được thể hiện ở hình sau:



Hình 2.15: Sơ đồ nguyên lý toàn mạch

**KHỐI RELAY :**

**A diagram of a relay

Description automatically generated with low confidence**

Hình 2.16: Sơ đồ khối relay

**KHỐI HIỂN THỊ :**

A picture containing text, screenshot, display, line

Description automatically generated

Hình 2.17: Sơ đồ LCD và PCF8574 I2C.

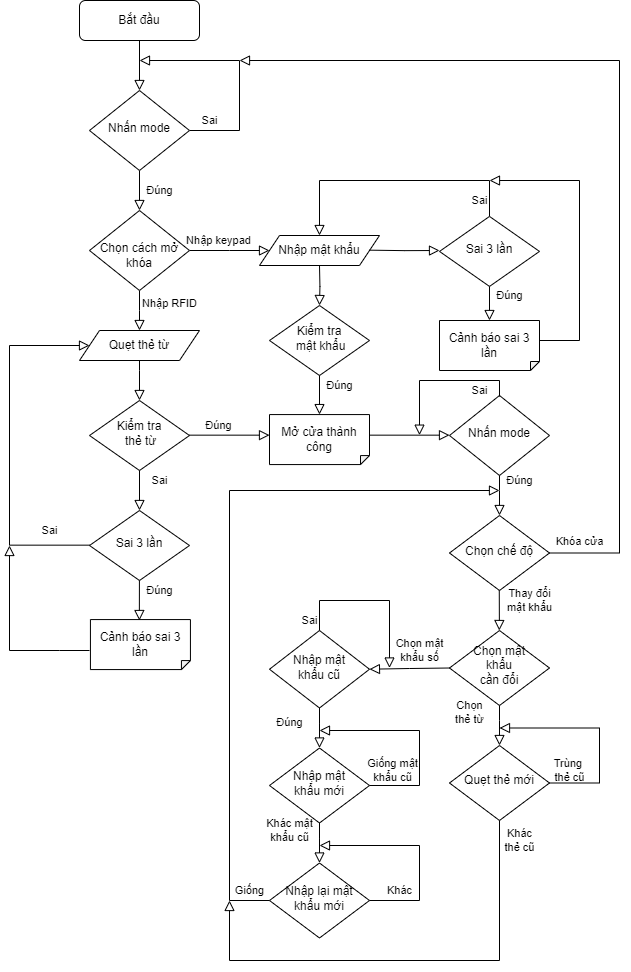
**KHỐI VI ĐIỀU KHIỂN :**

**A picture containing text, diagram, line, screenshot

Description automatically generated**

Hình 2.18: Sơ đồ khối điều khiển .

# 2.5 NGUYÊN LÝ HOẠT ĐỘNG:

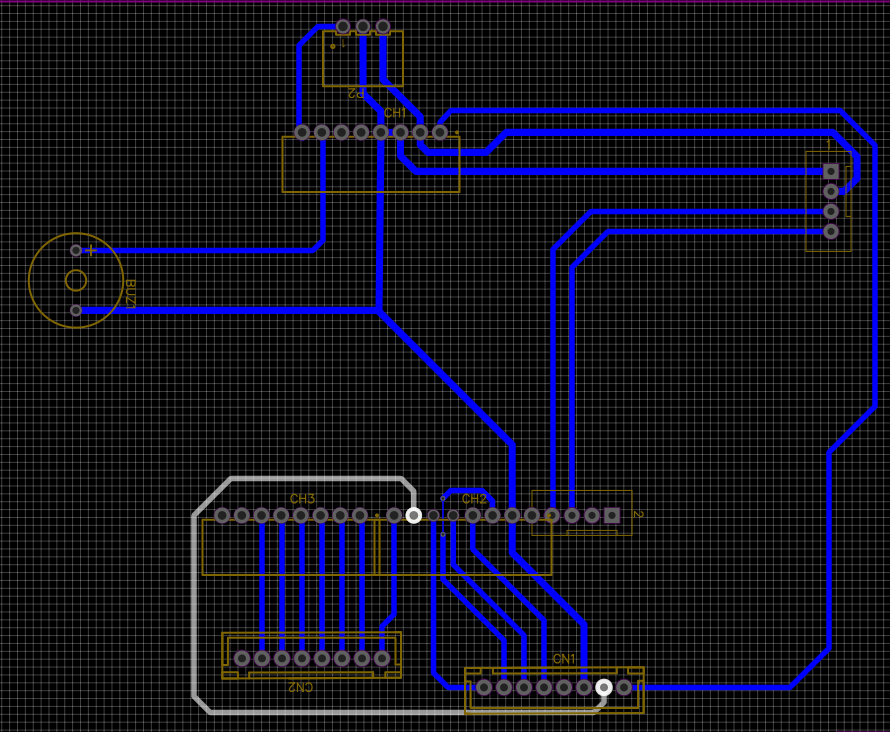


Hình 2.19 : Sơ đồ nguyên lý hoạt động của Khóa điện tử

# CHƯƠNG 3 : KẾT QUẢ THỰC HIỆN

# 3.1 VẼ MẠCH PCB

**a. Board chính**



Hình 3.1: mạch PCB của board chính

1. **Relay :**

A picture containing screenshot, circle, diagram, line

Description automatically generated

Hình 3.2: Mạch PCB của Relay

1. **Ma trận nút**

A picture containing screenshot, line, diagram, pattern

Description automatically generated

Hình 3.3: Mạch PCB của khối ma trận nút

# 3.2 MÔ HÌNH THỰC TẾ



Hình 3.4: Sản phẩm thực tế

# CHƯƠNG 4: KẾT LUẬN VÀ ĐỊNH HƯỚNG ĐỀ TÀI

# 4.1 KẾT QUẢ ĐẠT ĐƯỢC

Đồ án cơ bản hoàn thành được các yêu cầu đề ra , mặc dù còn nhiều thiếu sót và hạn chế tuy nhiên nhờ sự hướng dẫn tận tình của cô hướng dẫn , đồ án đề tài “ Khóa điện tử sử dụng mã bảo mật và RFID “ đã đạt được kết quả tốt.

Áp dụng thành công các lý thuyết đã học , và tự tay thiết kế phần cứng cho đồ án , đã hoàn thành được hệ thống có tính chất bảo mật và hoạt động hiệu quả.

# 4.2 HẠN CHẾ

Vì đồ án hiện tại chỉ mới phát triển ở sơ khai nên còn khá nhiều tính năng chưa được thực hiện như : Kiểm soát thời gian truy cập khóa , hệ thống quản lý người dùng từ xa , thêm thẻ cho người dùng ….

Về phần cứng đồ án được thiết kế còn khá sơ sài , chưa đảm bảo tính chắc chắn của bộ khóa điện tử thông dụng . Chưa có thiết kế Khối pin dự phòng trường hợp bị mất điện. Việc thi công mạch còn làm thủ công không tránh khỏi việc mạch bị nhiễu khiến cho khóa thực hiện sai.

# 4.3 HƯỚNG PHÁT TRIỂN

* Ở đồ án lần này có rất nhiều hướng phát triển rõ ràng :
  + Về phần cứng : có thể tích hợp thêm nhiều các tính năng bảo mật khác như vân tay , nhận diện khuôn mặt ,…. Thiết kế mô hình chắc chắn hơn cho khóa
  + Về phần mềm : đồ án có thể phát triển thêm các chức năng hướng đến người dùng như quản lý , mở khóa trực tiếp trên điện thoại.
* Ngoài ra , đồ án có thể thay thế Arduino bằng các dòng vi điều khiển PIC hay 8051 để tăng hiệu suất và giảm nhiễu khi hoạt động.

**PHỤ LỤC**

**Code :** https://github.com/HoangPhuc02/DoAn1?fbclid=IwAR0j4LWkB3wVaRySY\_wONVMHq0bSh\_4k\_p\_3o-QveFv9vafgJqJMk69Vdgs

# TÀI LIỆU THAM KHẢO

[1] Texas Instruments Incorporated, “ PCF8574 Remote 8-Bit I/O Expander for I2C Bus ” . Texas Instruments, Post Office Box 655303, Dallas, Texas 75265.

[2] SGK , “ What is RFID Module | How RC522 RFID Module works “, Available : https://eazytronic.com/rc522-rfid-module/ ,August 19, 2022.[Accessed : May 10, 2023].

[3] Scott Campbell , “ Basic of the I2C Communication protocol “, Available : https://www.circuitbasics.com/basics-of-the-i2c-communication-protocol/, February 14, 2016[Accessed : May 10 ,2023].

[4] Research design lab, “4x4 matrix keypad V2.0” .

[5] Figa Undala, Dedi Triyanto, Yulrio Brianorman, “Prototype Sistem Keamanan Pintu Menggunakan

Radio Frequency Identification (RFID) “, Dengan Kata Sandi Berbasis Mikrokontroller. Jurnal Coding Sistem Komputer Untan. 2015. Vol. 3 No. 1 : 22-31.

[6] David Dunmur, Harry G. Walton , “ Liquid crystal display “ , Research Scientist in liquid crystal display technology, Sharp Laboratories of Europe.

Available : https://www.britannica.com/technology/liquid-crystal-display , May 26, 2023. [Accessed : May 27,2023].

[7] Swagatam, “Introduction to I2C LCD Adapter Module “ . Available : https://www.homemade-circuits.com/introduction-i2c-lcd-adapter-module/, June 8, 2019 [Accessed : May 20, 2023 ].