KHOA KỸ THUẬT VÀ CÔNG NGHỆ

**BỘ MÔN CÔNG NGHỆ THÔNG TIN**

****

**THỰC TẬP ĐỒ ÁN CƠ SỞ NGÀNH**

**HỌC KỲ I, NĂM HỌC 2023 - 2024**

**XÂY DỰNG ỨNG DỤNG NHẬN DẠNG VÂN TAY**

*Sinh viên thực hiện:*

Họ tên: Nguyễn Hữu Luân

MSSV: 117521003

Lớp: DA21TTC

*Giáo viên hướng dẫn:*

Nguyễn Bảo Ân

***Trà Vinh, tháng 1 năm 2024***

**NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN**

*Trà Vinh, ngày ….. tháng …… năm ……*

**Giáo viên hướng dẫn**

*(Ký tên và ghi rõ họ tên)*

*Trà Vinh, ngày ….. tháng …… năm ……*

**Giáo viên hướng dẫn**

*(Ký tên và ghi rõ họ tên)*

**LỜI CẢM ƠN**

**NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG**

*Trà Vinh, ngày ….. tháng …… năm ……*

**Thành viên hội đồng**

*(Ký tên và ghi rõ họ tên)*

Em xin chân thành cảm ơn quý thầy cô khoa Kỹ thuật và Công nghệ, Trường Đại học Trà Vinh và đặc biệt là thầy Nguyễn Bảo Ân đã nhiệt tình hướng dẫn và tạo điều kiện để cho em có cơ hội học hỏi, thực hành và giải đáp những thắc mắc trong quá trình học.

Với những kiến thức đã được học tập và tìm hiểu, nhận được sự hướng dẫn nhiệt tình của thầy Nguyễn Bảo Ân em đã hoàn thành bài báo cáo đồ án cơ sở ngành. Trong quá trình thực hiện đề tài còn thiếu sót nên em mong nhận được đóng góp ý của quý thầy cô để em có thể học tập thêm được nhiều kinh nghiệm và sẽ hoàn thành tốt hơn trong những bài báo cáo sắp tới.

Em xin chân thành cảm ơn!

**MỤC LỤC**

[ĐỀ CƯƠNG CHI TIẾT 6](#_Toc155557952)

[TÓM TẮT ĐỒ ÁN CƠ SỞ NGÀNH 8](#_Toc155557953)

[MỞ ĐẦU 9](#_Toc155557954)

[1. Lý do chọn đề tài: 9](#_Toc155557955)

[2. Mục đích: 9](#_Toc155557956)

[3. Đối tượng nghiên cứu: 9](#_Toc155557957)

[4. Phạm vi nghiên cứu: 9](#_Toc155557958)

[CHƯƠNG 1: TỔNG QUAN 10](#_Toc155557959)

[1.1 Đặt vấn đề: 10](#_Toc155557960)

[1.2 Đặc tả đề tài: 10](#_Toc155557961)

[1.3 Mục tiêu và tính cấp thiết của đề tài: 10](#_Toc155557962)

[CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT 12](#_Toc155557963)

[2.1 Giới thiệu về Arduino 12](#_Toc155557964)

[2.1.1 Khái niệm: 12](#_Toc155557965)

[2.1.2 Lịch sử ra đời: 12](#_Toc155557966)

[2.1.3 Kiến trúc phần cứng: 13](#_Toc155557967)

[2.2 Nguồn Arduino: 13](#_Toc155557968)

[2.2.1 Mạch dao động: 14](#_Toc155557969)

[2.2.2 Mạch Reset: 15](#_Toc155557970)

[2.2.3 Mạch nạp giao tiếp với máy tính: 16](#_Toc155557971)

[2.2.4 Sơ đồ chân của Arduino: 17](#_Toc155557972)

[2.3 Giới thiệu về module cảm biến Vân tay As608 18](#_Toc155557973)

[2.3.1 Giới thiệu: 18](#_Toc155557974)

[2.3.2 Thông số kỹ thuật: 20](#_Toc155557975)

[2.3.3 Chân và giao tiếp với các chân: 20](#_Toc155557976)

[2.4 Giới thiệu module Relay: 21](#_Toc155557977)

[2.4.1 Giới thiệu: 21](#_Toc155557978)

[2.4.2 Các thông số của một module relay: 22](#_Toc155557979)

[2.4.3 Cách sử dụng module relay: 22](#_Toc155557980)

[2.5 Môi trường phát triển phần mềm Arduino: 23](#_Toc155557981)

[CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU 24](#_Toc155557982)

[3.1 Sơ đồ khối của hệ thống: 24](#_Toc155557983)

[3.2 Sơ đồ thuật toán: 24](#_Toc155557984)

[3.3 Cách kết nối các thiết bị trong đồ án: 25](#_Toc155557985)

[3.3.1 Kết nối Module As608 với Arduino: 25](#_Toc155557986)

[3.3.2 Kết nối Module Relay với Arduino: 25](#_Toc155557987)

[3.3.3 Kết nối Module Relay với bóng đèn: 25](#_Toc155557988)

[3.4 Nạp code cho Module As608 và Module Relay bằng phần mềm Arduino IDE: 25](#_Toc155557989)

[3.4.1 Nạp code cho Module As608: 25](#_Toc155557990)

[3.4.2 Nạp code cho Module Relay: 28](#_Toc155557991)

[CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU 29](#_Toc155557992)

[4.1 Nhận dạng đúng vân tay đã nạp: 29](#_Toc155557993)

[4.2 Module Relay hoạt động: 29](#_Toc155557994)

[4.3 Mô hình thử nghiệm: 29](#_Toc155557995)

[CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 31](#_Toc155557996)

[5.1 Về kiến thức: 31](#_Toc155557997)

[5.2 Về thực hành: 31](#_Toc155557998)

[5.3 Tự nhận xét: 31](#_Toc155557999)

[5.3.1 Ưu điểm: 31](#_Toc155558000)

[5.3.2 Hạn chế: 31](#_Toc155558001)

[5.4 Hướng phát triển: 31](#_Toc155558002)

[DANH MỤC TÀI LIỆU THAM KHẢO 33](#_Toc155558003)

[PHỤ LỤC 34](#_Toc155558004)

**DANH MỤC HÌNH ẢNH – BẢNG BIỂU**

[Hình 2.1 Nguồn 14](#_Toc155558100)

[Hình 2.2 Mạch Dao Động 16](#_Toc155558101)

[Hình 2.3 Mạch Reset 16](#_Toc155558102)

[Hình 2.4 Mạch Nạp Và Giao Tiếp Máy Tính 17](#_Toc155558103)

[Hình 2.5 Sơ đồ chân Arduino. 18](#_Toc155558104)

[Hình 2.6 Module cảm biến vân tay As608. 20](#_Toc155558105)

[Hình 2.7 Module Relay 22](#_Toc155558106)

[Hình 2.8 Phần mềm Arduino IDE 24](#_Toc155558107)

[Hình 3.1 Sơ đồ khối của hệ thống 25](#_Toc155558108)

[Hình 3.2 Sơ đồ thuật toán 25](#_Toc155558109)

[Bảng 3.1 Cách kết nối Module As608 với Arduino 26](#_Toc155558110)

[Bảng 3.2 Cách kết nối Module Relay với Arduino 26](#_Toc155558111)

[Bảng 3.3 Cách kết nối Module Relay với bóng đèn 26](#_Toc155558112)

[Hình 3.3 Chọn Board 27](#_Toc155558113)

[Hình 3.4 Chọn cổng Port 27](#_Toc155558114)

[Hình 3.5 Cài đặt thư viện Adafruit Fingerprint Sensor Library 28](#_Toc155558115)

[Hình 3.6 Nạp vân tay vào Module cảm biến vân tay As608 28](#_Toc155558116)

[Hình 3.7 Nạp code cho Module Relay 29](#_Toc155558117)

[Hình 4.1 Nhận dạng đúng vân tay đã nạp 30](#_Toc155558118)

[Hình 4.2 Module Relay hoạt động 30](#_Toc155558119)

[Hình 4.3 Mô hình thực nghiệm 31](#_Toc155558120)

**ĐỀ CƯƠNG CHI TIẾT**

**THỰC TẬP ĐỒ ÁN CƠ SỞ NGÀNH**

|  |  |
| --- | --- |
| **Tên đề tài:** Xây dựng ứng dụng nhận dạng vân tay. | |
| **Giáo viên hướng dẫn:** Nguyễn Bảo Ân. | |
| **Thời gian thực hiện:** từ 06/11/2023 đến 31/12/2023. | |
| **Sinh viên thực hiện:** Nguyễn Hữu Luân - 117521003. | |
| **Nội dung đề tài:**  - Nhận dạng vân tay theo hướng xác minh (verification), sử dụng mạch Arduino và đầu lọc nhận dạng vân tay để khởi động các thiết bị mà người dùng muốn ứng dụng. Chẳng hạn như dùng vân tay để khởi động xe máy, bóng đèn,… Tiện ích khi sử dụng Arduino và đầu lọc để nhận dạng vân tay: Chi phí lắp đặt thấp và ứng dụng được trên nhiều thiết bị.  **Yêu cầu:**  - Có kiến thức về Arduino và các công cụ liên quan.  - Học hỏi các kiến thức liên quan đến ngành điện tử.  - Xử lí được lỗi trong quá trình thực hiện và dễ sử dụng.  **Phương pháp thực hiện:**  - Sử dụng Arduino IDE để lập trình cho mạch Arduino.  - Kết nối Mô Đun As608 với mạch Arduino.  - Truyền dữ liệu vân tay cho Mô Đun As608.  **Kết quả đạt được:**  - Nhận dạng được vân tay.  - Quyển báo cáo Đồ án Cơ sở ngành.  - File Power Point về nội dung đề tài. | |
| **Kế hoạch thực hiện:**   |  |  |  |  | | --- | --- | --- | --- | | Tuần | Thời gian | Nội dung công việc | Người thực hiện | | 1 | Từ ngày 6/11/2023  đến 12/11/2023 | -Nghiên cứu tài liệu Arduino. | Nguyễn Hữu Luân | | 2 | Từ ngày 20/11/2023  đến 26/11/2023 | -Truyền code vào mạch Arduino.  -Đấu dây chạy thử và kiểm lỗi nếu có. | Nguyễn Hữu Luân | | 3 | Từ ngày 4/12/2023  đến 10/12/2023 | -Viết quyển báo cáo.  -Thiết kế file trình chiếu Power Point. | Nguyễn Hữu Luân | | 4 | Từ ngày 18/12/2023  đến 24/12/2023 | -Thử làm với C# hoặc Python để nhận dạng vân tay. | Nguyễn Hữu Luân | | Kết thúc | Từ ngày 25/12/2023  đến 31/12/2023 | - SV hoàn chỉnh quyển báo cáo  - GV chỉnh sửa quyển báo cáo của SV  - GV hướng dẫn SV hoàn chỉnh quyển báo cáo (theo mẫu) và nộp quyển báo cáo | Nguyễn Hữu Luân  Thầy Nguyễn Bảo Ân | | |
| **Xác nhận của GVHD**  **Nguyễn Bảo Ân** | *Ngày 1 tháng 11 năm 2023*  **SV Thực hiện**  **Nguyễn Hữu Luân** |

**TÓM TẮT ĐỒ ÁN CƠ SỞ NGÀNH**

Việc ứng dụng Arduino tạo ra sản phẩm để giải quyết các vấn đề của con người trong cuộc sống là điều vô cùng cần thiết và tiện lợi, tiết kiệm được công sức và chi phí của người dùng. Ngoài ra, mạch Arduino còn có phần mềm hỗ trợ lập trình Arduino IDE với bộ thư viện đa dạng và phong phú. Trong đồ án cơ sở ngành lần này, em sẽ nghiên cứu, tìm hiểu về mạch Arduino và ứng dụng nó để thiết kế một mô hình mở khóa bằng vân tay, việc thiết kế một mô hình mở khóa bằng vân tay sẽ giải quyết được các vấn đề như bảo đảm được tính bảo mật cao so với các loại ổ khóa truyền thống hiện nay.

**MỞ ĐẦU**

**1. Lý do chọn đề tài:**

Mở khóa cửa bằng vân tay Arduino là một trong những ứng dụng của công nghệ sinh trắc học. Công nghệ này giúp người dùng mở khóa cửa một cách nhanh chóng và tiện lợi, đồng thời đảm bảo tính bảo mật cao. Với việc sử dụng cảm biến vân tay, người dùng không cần phải ghi nhớ mật khẩu, thao tác đơn giản và tiếp cận dễ dàng. Bên cạnh đó, đề tài này còn giúp nâng cao kiến thức về Arduino và các ứng dụng của nó. Chúng ta có thể tìm hiểu về cách kết nối cảm biến vân tay với Arduino, cách lập trình và điều khiển khóa cửa.

**2. Mục đích:**

* Nắm được kiến thức cơ bản về Arduino.
* Thiết kế mô hình mở khóa bằng vân tay

**3. Đối tượng nghiên cứu:**

* Nghiên cứu đề tài và tìm hiểu Arduino
* Sử dụng mạch Arduino, phần mềm Arduino IDE, module cảm biến vân tay As608, module Relay và Bóng đèn.

**4. Phạm vi nghiên cứu:**

* Nghiên cứu được thực hiệndựa trên các kiến thức trên Internet, tài liệu về Arduino.
* Đề tài được thực hiện từ 06/11/2023 đến 31/12/2023.

**CHƯƠNG 1: TỔNG QUAN**

**1.1 Đặt vấn đề:**

Với sự phát triển của xã hội hiện nay những thiết bị bảo vệ tài sản bằng khóa cửa trong gia đình hay cơ quan làm việc là một phần không thể thiếu. Hầu hết trên thị trường hiện nay sử dụng các ổ khóa sử dụng bằng chìa để mở khóa, các ổ khóa này gặp vấn đề lớn đó là tính bảo mật không cao, nên dễ dàng bị mở bởi các chìa khoá đa năng.

Việc thiết kế hệ thống mở khóa này phải được tối ưu hóa và đơn giản để tạo sự thuận tiện cho người dùng trong quá trình sử dụng. Tóm lại, việc tạo ra một hệ thống mở khóa cửa bằng vân tay phải thực sự dễ sử dụng và không quá phức tạp để có thể áp dụng rộng rãi, thay thế được các loại ổ khóa truyền thống hiện nay trong tương lai.

**1.2 Đặc tả đề tài:**

Bài toán nhận dạng vân tay mở khóa cửa bằng Arduino được giải quyết bằng cách sử dụng một cảm biến nhận dạng vân tay AS608. Cảm biến này sử dụng giao tiếp USB để giao tiếp với vi điều khiển hoặc kết nối trực tiếp với máy tính thông qua mạch chuyển giao tiếp USB.

Để sử dụng cảm biến vân tay AS608 với Arduino, cần chuẩn bị một số vật liệu như Arduino Uno, module cảm biến vân tay AS608, module Relay và các dây jack cắm để nối cảm biến với Arduino. Sau đó, sử dụng thư viện Fingerprint của Adafruit để lập trình Arduino.

**1.3 Mục tiêu và tính cấp thiết của đề tài:**

Cùng với sự gia tăng phức tạp của các loại tội phạm như hiện nay thì nhu cầu chọn một chiếc khóa an toàn, thông minh và tiện lợi có thể bảo vệ tài sản ở gia đình, cơ quan làm việc,... khỏi các mối nguy hiểm đã làm cho nhu cầu về những chiếc khóa thông minh ngày càng tăng đặc biệt là các khóa cửa vân tay. Mặc dù giá đã giảm nhưng với thu nhập bình quân đầu người trong thời kì kinh tế đang suy thoái như hiện nay thì việc đầu tư một số tiền lớn vào một chiếc khóa bằng vân tay là một lựa chọn rất khó khăn, trong khi đó các loại khóa truyền thống trên thị trường hiện nay tuy rẻ nhưng lại cồng kềnh, khó bảo quản (hay mất chìa khóa chẳng hạn), nếu chọn nhầm hàng chất lượng kém có thể gây ra những hậu quả không đáng có.

**CHƯƠNG 2: NGHIÊN CỨU LÝ THUYẾT**

* 1. **Giới thiệu về Arduino** 
     1. **Khái niệm:**

Arduino là tảng vi mạch thiết kế mở phần cứng (Open-source hardware) và phần mềm (Open-source software). Phần cứng Arduino là những bộ vi điều khiển bo mạch đơn (Single-board microcontroller) được tạo ra tại thị trấn Ivrea ở Ý, nhằm xây dựng các ứng dụng tương tác với nhau hoặc với môi trường được thuận lợi hơn. Phần cứng bao gồm một board mạch nguồn mở được thiết kế trên nền tảng vi xử lý AVR Atmel 8 bit, hoặc ARM Atmel 32-bit. Những model hiện tại được trang bị gồm 1 cổng giao tiếp USB, 6 chân đầu vào analog, 14 chân I/O kỹ thuật số tương thích với nhiều board mở rộng khác nhau.

Arduino bao gồm phần cứng (arduino board) và phần mềm (arduino IDE). Phần mềm để lập trình cho mạch Arduino là Arduino IDE.

* + 1. **Lịch sử ra đời:**

Arduino được khởi động vào năm 2005 như là một dự án dành cho sinh viên trại Interaction Design Institute Ivrea (Viện thiết kế tương tác Ivrea) tại Ivrea, Italy. Vào thời điểm đó các sinh viên sử dụng một "BASIC Stamp" (con tem Cơ Bản) có giá khoảng $100, xem như giá dành cho sinh viên. Massimo Banzi, một trong những người sáng lập, giảng dạy tại Ivrea. Cái tên "Arduino" đến từ một quán bar tại Ivrea, nơi một vài nhà sáng lập của dự án này thường xuyên gặp mặt. Bản thân quán bar này có được lấy tên là Arduino, Bá tước của Ivrea, và là vua của Italy từ năm 1002 đến 1014.

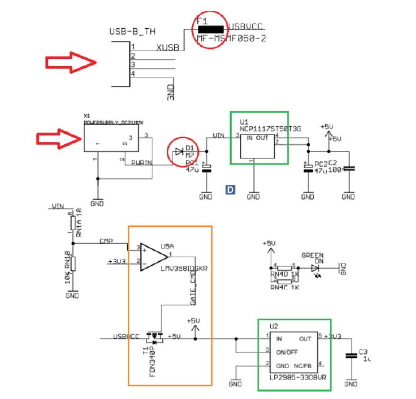
Lý thuyết phần cứng được đóng góp bởi một sinh viên người Colombia tên là Hernando Barragan. Sau khi nền tảng Wiring hoàn thành, các nhà nghiên cứu đã làm việc với nhau để giúp nó nhẹ hơn, rẻ hơn, và khả dụng đối với cộng đồng mã nguồn mở. Trường này cuối cùng bị đóng cửa, vì vậy các nhà nghiên cứu, một trong số đó là David Cuarlielles, đã phổ biến ý tưởng này.

* + 1. **Kiến trúc phần cứng:**

Một mạch Arduino bao gồm một vi điều khiển AVR với nhiều linh kiện bổ sung giúp dễ dàng lập trình và có thể mở rộng với các mạch khác. Một khía cạnh quan trọng của Arduino là các kết nối tiêu chuẩn của nó, cho phép người dùng kết nối với CPU của board với các module thêm vào có thể dễ dàng chuyển đổi, được gọi là shield. Vài shield truyền thông với board Arduino trực tiếp thông qua các chân khác nhau, nhưng nhiều shield được định địa chỉ thông qua serial bus I²C-nhiều shield có thể được xếpchồng và sử dụng dưới dạng song song. Arduino chính thức thường sử dụng các dòng chip megaAVR, đặc biệt là ATmega8, ATmega168, ATmega328, ATmega1280 và ATmega2560.

* 1. **Nguồn Arduino:**

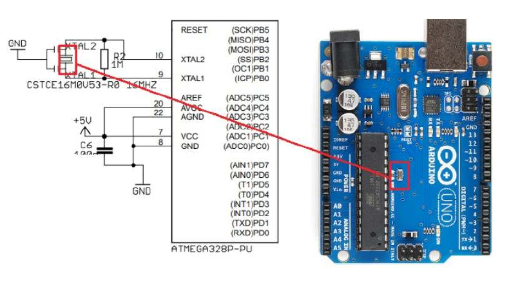
Phần nguồn của Board mạch Arduino được thiết kế để thực hiện các nhiệm vụ sau:



***Hình 2.1 Nguồn***

* Lựa chọn nguồn cung cấp cho board mạch (khối màu cam trong hình dưới). Board mạch Arduino có thể được cung cấp nguồn bởi Adapter thông qua Jack DC hoặc từ cổng USB (2 mũi tên màu đỏ). Trong trường hợp chỉ có 1 trong 2 nguồn cung cấp thì Board Arduino sẽ sử dụng nguồn cung cấp đó. Trong trường hợp có cả 2 nguồn cung cấp thì Arduino sẽ ưu tiên lựa chọn nguồn cung cấp từ Jack DC thay vì từ cổng USB. Việc ưu tiên này được thực hiện bởi OpAmp trong IC LMV358 và MOSFET FDN340P. Điện áp từ Jack DC sau khi qua Diode bảo vệ D1 thì được gọi là điện áp VIN. Điện áp VIN qua cầu phân áp để tạo thành VIN/2 để so sánh với điện áp 3.3V. Vì VIN/2 > 3.3V nên điện áp đầu ra của OpAmp là 5V, điều này làm cho MOSFET không được kích, nguồn cung cấp cho Board Arduino là từ Jack DC sau khi qua ổn áp.
* Tạo ra các điện áp 5v và 3.3v (2 khối màu xanh) để cung cấp cho vi điều khiển và cũng là điểm cấp nguồn cho các thiết bị bên ngoài sử dụng. Mạch Arduino sử dụng IC ổn áp NCP1117 để tạo điện áp 5V từ nguồn cung cấp lớn và IC ổn áp LP2985 để tạo điện áp 3.3V. Đây đều là những IC ổn áp tuyến tính, tuy hiệu suất không cao nhưng ít gợn nhiễu và mạch đơn giản.
* Bảo vệ ngược nguồn, quá tải (vòng tròn màu đỏ). F1 là một cầu chì tự phục hồi, trong trường hợp bạn chỉ sử dụng dây cáp USB để cấp nguồn thì tổng dòng tiêu thụ không được quá 500mA. Nếu không cầu chì sẽ ngăn không cho dòng điện chạy qua. D1 là một Diode, chỉ cho dòng điện 1 chiều chạy qua (từ Jack DC vào mạch), trong trường hợp mạch Arduino của bạn có mắc với các thiết bị khác và có nguồn cung cấp lớn hơn nguồn vào Jack DC, nếu có sai sót chập mạch,… thì sẽ không có trường hợp nguồn các thiết bị bên ngoài chạy ngược vào Adapter.
* Báo nguồn thì đèn nguồn ON sáng lên báo thiết bị đã được cấp nguồn. Nếu các bạn đã cắm nguồn mà đèn nguồn không sáng thì có thể nguồn cung cấp của bạn đã bị hỏng hoặc jack kết nối lỏng, hoặc mạch Arduino kết nối với các linh kiện bên ngoài bị ngắn mạch.
  + 1. **Mạch dao động:**

Mạch giao động tạo ra các xung clock giúp cho vi điều khiển hoạt động, thực thi lệnh… Board mạch Arduino Uno R3 sử dụng thạch anh 16Mhz làm nguồn dao động.

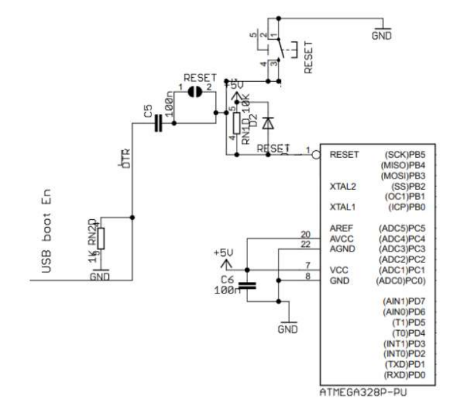


***Hình 2.2 Mạch Dao Động***

* + 1. **Mạch Reset:**

Để vi điều khiển thực hiện khởi động lại thì chân RESET phải ở mức logic LOW (~0V) trong 1 khoản thời gian đủ yêu cầu. Mạch reset của board Arduino UnoR3 phải đảm bảo được 02 việc:

* Reset bằng tay: Khi nhấn nút, chân RESET nối với GND, làm cho MCU RESET. Khi không nhấn nút chân Reset được kéo 5V.
* Reset tự động: Reset tự động được thực hiện ngay khi cấp nguồn cho vi điều khiển nhờ sự phối hợp giữa điện trở nối lên nguồn và tụ điện nối đất. Thời gian tụ điện nạp giúp cho chân RESET ở mức LOW trong 1 khoản thời gian đủ để vi điều khiển thực hiện reset.



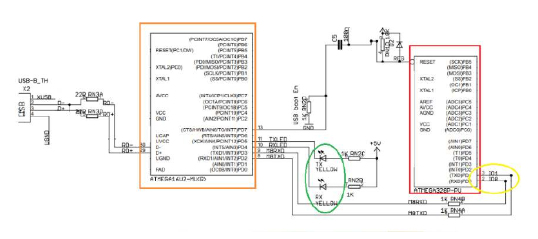
***Hình 2.3 Mạch Reset***

* + 1. **Mạch nạp giao tiếp với máy tính:**

Vi điều khiển Atmega328P trên Board Arduino UnoR3 đã được nạp sẵn 1 bootloader, cho phép nhận chương trình mới thông qua chuẩn giao tiếp UART (chân 0 và 1) ở những giây đầu sau khi vi điều khiển Reset.

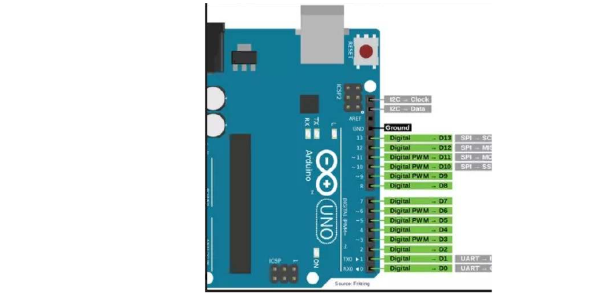
Máy tính giao tiếp với Board mạch Arduino qua chuẩn giao tiếp USB (D+/D-),thông qua một vi điều khiển trung gian là ATMEGA16U2 hoặc một IC trung gian là CH340 (thường thấy trong các mạch sử dụng chip dán). Vi điều khiển hoặc IC này có nhiệm vụ chuyển đổi chuẩn giao tiếp USB thành chuẩn giao tiếp UART để nạp chương trình hoặc giao tiếp truyền nhận dữ liệu với máy tính (Serial).

Phần thiết kế mạch nạp có tích hợp thêm 02 đèn LED,nên khi nạp chương trình sẽ thấy 2 LED này nhấp nháy. Còn khi giao tiếp, nếu có dữ liệu từ máy tính gửi 14 xuống vi điều khiển thì đèn LED Rx sẽ nháy. Còn nếu có dữ liệu từ vi điều khiển gửi lên máy tính thì đèn Tx sẽ nháy.



***Hình 2.4 Mạch Nạp Và Giao Tiếp Máy Tính***

* + 1. **Sơ đồ chân của Arduino:**



***Hình 2.5 Sơ đồ chân Arduino.***

* Chân cấp năng lượng:
* 5V: cấp điện áp 5V đầu ra, dùng để cấp nguồn cho các linh kiện điện tử kết nối với Arduino
* 3.3V: chức năng tương tự như cấp nguồn 5v nhưng đây là cấp điện áp 3.3V đầu ra.
* Ground: hay còn gọi là chân GND, là cực âm của nguồn điện cấp cho Arduino UNO. Khi bạn dùng các thiết bị sử dụng những nguồn điện riêng biệt thì những chân này phải được nối với nhau.
* Vin (Voltage Input): tương tự như chân 5V, nhưng thêm chức năng cấp nguồn ngoài cho Arduino UNO thay vì cắm USB, bạn nối cực dương của nguồn với chân này và cực âm của nguồn với chân GND.
* Các cổng vào/ra (I/O): Arduino cung cấp nhiều các chân I/O ( hay còn gọi là Pin ) để ta giao tiếp hay gửi lệnh điều khiển các thiết bị.
* Các chân Digital (điện tử): Phiên bản Arduino UNO R3 được sở hữu 14 chân digital từ 0 đến 13 dùng để đọc hoặc xuất tín hiệu. Chúng chỉ có 2 mức điện áp có thể điều khiển là 0V và 5V với dòng vào/ra tối đa trên mỗi chân là 40mA. Ngoài ra một số chân digital có chức năng đặc biệt là chân PWM. (Chân PWM: là các chân có dấu '~' đằng trước, các chân này cho phép xuất ra xung PWM với độ phân giải 8bit (giá trị từ 0 đến 255) tương ứng với mức giao động điện áp của chân từ 0V đến 5V, khác với các chân không phải PWM, chỉ có thể chọn giá trị 0V hoặc 5V).
* Các chân Analog: Arduino UNO có 6 chân analog (A0 đến A5) cung cấp độ phân giải tín hiệu 10 bit (0 đến 1023) để đọc giá trị điện áp trong khoảng 0V đến 5V. Đặc biệt, Arduino UNO có 2 chân A4 (SDA) và A5 (SCL) hỗ trợ giao tiếp I2C/TWI với các thiết bị khác.
* Chân TXD và RXD: Đây là các chân Serial dùng để gửi (transmit – TX) và nhận (receive –RX) dữ liệu TTL Serial. Arduino Uno có thể giao tiếp dữ liệu với các thiết bị cần sử dụng thông qua 2 chân này, ngoài ra có thể sử dụng 2 chân này để nạp code cho mạch mà không cần thông qua USB của mạch.
  1. **Giới thiệu về module cảm biến Vân tay As608**
     1. **Giới thiệu:**

Cảm biến nhận dạng vân tay AS608 Fingerprint Sensor sử dụng giao tiếp UART TTL hoặc USB để giao tiếp với Vi điều khiển hoặc kết nối trực tiếp với máy tính (thông qua mạch chuyển USB-UART hoặc giao tiếp USB).

* Nguồn cung cấp điện áp: DC 3.6 đến 6.0V
* Nguồn cung hiện tại: <120mA
* Màu đèn nền: màu xanh lục
* Baudrate: 9600, 19200, 28800, 38400, 57600 (default is 57600)
* Mức độ an toàn: từ thấp đến cao 1,2,3,4,5
* Tỷ lệ chấp nhận sai (FAR): <0,001% (mức độ bảo mật 3)
* Tỷ lệ từ chối giả (FRR): <1.0% (mức độ bảo mật 3)
* Có thể lưu trữ hơn 200 dấu vân tay khác nhau.

Cảm Biến Nhận Dạng Vân Tay AS608 là loại cảm biến nhận dạng vân tay sử dụng giao tiếp UART TTL hoặc USB để giao tiếp với Vi điều khiển hoặc kết nối trực tiếp với máy tính thông qua giao tiếp USB-UART. Cảm biến nhận dạng vân tay AS608 Fingerprint Sensor được tích hợp nhân xử lý nhận dạng vân tay phía trong, tự động gán vân tay với 1 chuỗi data và truyền qua giao tiếp UART ra ngoài nên hoàn toàn không cần các thao tác xử lý hình ảnh, đơn giản chỉ là phát lệnh đọc/ghi và so sánh chuỗi UART nên rất dễ sử dụng và lập trình.

Cảm biến nhận dạng vân tay As608 có thể hoàn thành việc thu thập dấu vân tay, đăng ký dấu vân tay, vân tay tương phản và chức năng tìm kiếm dấu vân tay một cách độc lập.

Khả năng thích ứng mạnh và các thuật toán hiệu suất cao, nó có khả năng mạnh mẽ để loại khác nhau, chẳng hạn như những ngón tay khô, ướt ngón tay, ngón tay và kết cấu ánh sáng đều có tỷ lệ biết chữ cao và điều chỉnh tốt, biểu diễn chịu lỗi.

Mạnh mẽ khả năng kháng tĩnh điện, đó là tốt áp dụng cho các khu vực nơi mà môi trường khô và dễ dàng tĩnh điện.

Phát triển ứng dụng đơn giản, các nhà phát triển không cần phải có chuyên môn vân tay, họ có thể phù hợp với từ lệnh cung cấp, phát triển các sản phẩm ứng dụng vân tay của mình.

Cảm biến nhận dạng vân tay As608 sử dụng giao tiếp UART TTL hoặc USB 1.1 để giao tiếp với Vi điều khiển hoặc kết nối trực tiếp với máy tính thông qua mạch chuyển USB-UART hoặc giao tiếp USB 1.

Cảm biến nhận dạng vân tay As608 được tích hợp nhân xử lý nhận dạng vân tay phía trong, tự động gán vân tay với 1 chuỗi data và truyền qua giao tiếp UART ra ngoài nên hoàn toàn không cần các thao tác xử lý hình ảnh, đơn giản chỉ là phát lệnh đọc/ghi và so sánh chuỗi UART nên rất dễ sử dụng và lập trình.



***Hình 2.6 Module cảm biến vân tay As608.***

* + 1. **Thông số kỹ thuật:**
* Điện áp sử dụng: 3.0~3.6VDC (thường cấp 3.3VDC, lưu ý quan trọng nếu cấp lớn hơn 3.3V DC cảm biến sẽ cháy ngay lập tức).
* Dòng tiêu thụ: 30~60mA, trung bình 40mA
* Communication Interface: USB /UART
* Tốc độ Baudrate UART: 9600 x N (N từ 1~12), mặc định N=6 baudrate =57600,8,1.
* USB communication: 2.0 full speed
* Sensor image size (pixel): 256 x 288 pixels
* Image processing time (s): <0.4s
* Power-on delay (s): <0.1s (the module needs about 0.1S to initialize afterpower on).
* Job search time (s): <0.3s
* FRR (rejection rate) <1%
* FAR (recognition rate) <0.001%
* Fingerprint storage capacity 300 (ID: 0 ~ 299)
  + 1. **Chân và giao tiếp với các chân:**
* V+: chân cấp nguồn chính VCC 3.3VDC cho cảm biến hoạt động.
* Tx: Chân giao tiếp UART TTL TX
* Rx: Chân giao tiếp UART TTL RX
* GND: Chân cấp nguồn GND (Mass / 0VDC)
* TCH: Chân Output của cảm biến chạm Touch, khi chạm tay vào cảm biến chân này sẽ xuất ra mức cao High, để sử dụng tính năng này cần cấp nguồn 3.3 VDC cho chân Va
* VA: Chân cấp nguồn 3.3 VDC cho Touch Sensor.
* U+: Chân tín hiệu USB D+
* U-: Chân tín hiệu USB D-
* Để giao tiếp UART ta cần sử dụng các chân:
* V+: Cấp nguồn 3.3VDC
* Tx: nối với RX của Vi điều kiển (mức TTL từ 3.3~5VDC)
* Rx: nối với TX của Vi điều kiển (mức TTL từ 3.3~5VDC)
* GND: Cấp nguồn GND (Mass chung)

- Để giao tiếp USB ta cần sử dụng các chân:

* V+: Cấp nguồn 3.3VDC
* U+: Chân tín hiệu USB D+
* U-: Chân tín hiệu USB D-
* GND: Cấp nguồn GND (Mass chung).
  1. **Giới thiệu module Relay:**
     1. **Giới thiệu:**

Module relay là một loại linh kiện điện tử thụ động rất hay gặp trong các ứng dụng thực tế. Khi gặp các vấn đề liên quan đến công suất và cần sự ổn định cao. ngoài ra có thể dễ dàng bảo trì.

* **Module**: Module tập hợp các linh kiện điện tử (transistor, tụ điện, điot, cuộn cảm) được gắn kết trên một bo mạch và thực hiện một chức năng nhất định.
* **Relay:** Relay hay còn gọi rơ-le là một công tắc (khóa K). Nhưng khác với công tắc ở một chỗ cơ bản, rơ-le được kích hoạt bằng điện thay vì dùng tay người. Chính vì lẽ đó, rơ-le được dùng làm công tắc điện tử. Vì rơ-le là một công tắc nên nó có 2 trạng thái: đóng và mở.



***Hình 2.7 Module Relay***

* + 1. **Các thông số của một module relay:**

Một module relay được tạo nên bởi 2 linh kiện thụ động cơ bản là rơ-le và transistor, nên module rơ-le có những thông số của chúng.

Hiệu điện thế kích tối ưu:

* 10A - 250VAC: Cường độ dòng điện tối đa qua các tiếp điểm của rơ-le với hiệu điện thế <= 250V (AC) là 10A.
* 10A - 30VDC: Cường độ dòng điện tối đa qua các tiếp điểm của rơ-le với hiệu điện thế <= 30V (DC) là 10A.
* 10A - 125VAC: Cường độ dòng điện tối đa qua các tiếp điểm của rơ-le với hiệu điện thế <= 125V (AC) là 10A.
* 10A - 28VDC: Cường độ dòng điện tối đa qua các tiếp điểm của rơ-le với hiệu điện thế <= 28V (DC) là 10A.
* SRD-05VDC-SL-C: Hiện điện thế kích tối ưu là 5V.

Các mức hiệu điện thế tối đa và cường độ dòng điện tối đa của đồ dùng điện khi nối vào module rơ-le.

* + 1. **Cách sử dụng module relay:**

Rơ-le bình thường gồm có 6 chân. Trong đó có 3 chân để kích, 3 chân còn lại nối với thiết bị mà người dùng muốn điều khiển:

* 3 chân dùng để kích:
* DC +: cấp hiệu điện thế kích tối ưu vào chân này.
* DC -: nối với cực âm.
* IN: chân tín hiệu, tùy vào loại module rơ-le mà nó sẽ làm nhiệm vụ kích rơ-le.
* 3 chân còn lại nối với thiết bị muốn điều khiển:
* COM: chân nối với 1 chân bất kỳ với thiết bị cần điều khiển, nhưng nên mắc vào dây chân lửa (nóng) nếu dùng hiệu điện thế xoay chiều và cực dương nếu là hiệu điện một chiều.
* NO: chân này bạn sẽ nối với chân lửa (nóng) nếu dùng điện xoay chiều và cực dương của nguồn nếu dòng điện một chiều.
* NC: chân này bạn sẽ nối chân lạnh (trung hòa) nếu dùng điện xoay chiều và cực âm của nguồn nếu dùng điện một chiều.
  1. **Môi trường phát triển phần mềm Arduino:**

Các mạch Arduino hay các mạch dựa trên nền tảng Arduino được lập trình bằng ngôn riêng. Ngôn ngữ này dựa trên ngôn ngữ Wiring được viết cho phần cứng nói chung và khi ta xem, ta thấy nó rất giống lập trình C đơn giản, do vậy việc tiếp cận sẽ không mấy khó khăn.

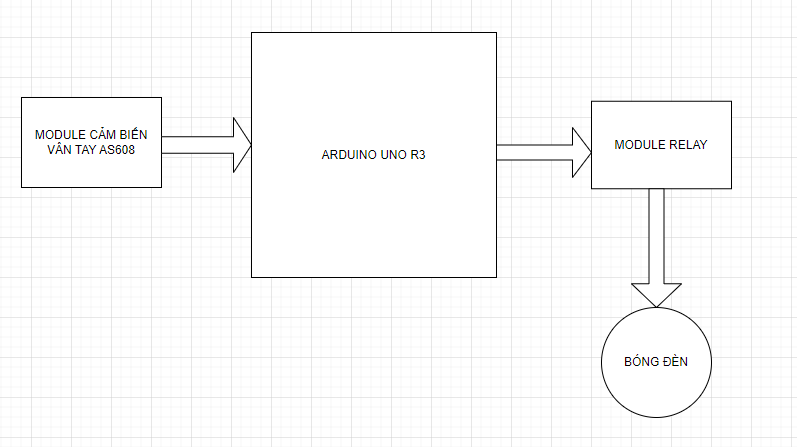
Để lập trình cũng như gửi lệnh và nhận tín hiệu từ mạch Arduino, ta sử dụng mộtmôi trường lập trình Arduino được gọi là Arduino IDE. Khi ta tạo một project mới, ta sẽ có khung code tương ứng.



***Hình 2.8 Phần mềm Arduino IDE***

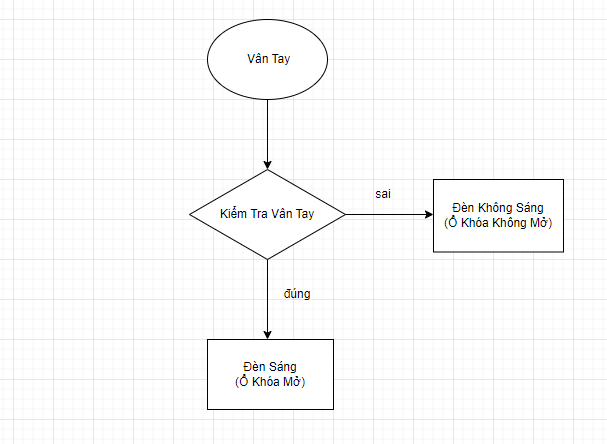
**CHƯƠNG 3: HIỆN THỰC HÓA NGHIÊN CỨU**

**3.1 Sơ đồ khối của hệ thống:**



***Hình 3.1 Sơ đồ khối của hệ thống***

**3.2 Sơ đồ thuật toán:**



***Hình 3.2 Sơ đồ thuật toán***

**3.3 Cách kết nối các thiết bị trong đồ án:**

**3.3.1 Kết nối Module As608 với Arduino:**

Tiến hành kết nối Module cảm biến vân tay As608 với Arduino bằng các jack cắm đã chuẩn bị như sau:

|  |  |  |
| --- | --- | --- |
| Module As608 | Arduino | Ghi chú |
| V+ | 3,3 V | Cấp nguồn dương |
| GND | GND | Cấp nguồn âm |
| TX | Cực số 2 | Truyền tín hiệu |
| RX | Cực Số 3 | Nhận tín hiệu |

***Bảng 3.1 Cách kết nối Module As608 với Arduino***

**3.3.2 Kết nối Module Relay với Arduino:**

Tiến hành kết nối Module Relay với Arduino bằng các jack cắm đã chuẩn bị như sau:

|  |  |  |
| --- | --- | --- |
| Module Relay | Arduino | Ghi chú |
| DC + | 5 V | Cấp nguồn dương |
| DC - | GND | Cấp nguồn âm |
| IN | Cực số 12 | Cực điều khiển |

***Bảng 3.2 Cách kết nối Module Relay với Arduino***

**3.3.3 Kết nối Module Relay với bóng đèn:**

Tiến hành kết nối Module Relay với bóng đèn đã chuẩn bị như sau:

|  |  |  |
| --- | --- | --- |
| Module Relay | Bóng đèn | Ghi chú |
| COM | Cực dương | Cấp nguồn cho bóng đèn |
| NO | Cực dương | Cấp nguồn cho bóng đèn |

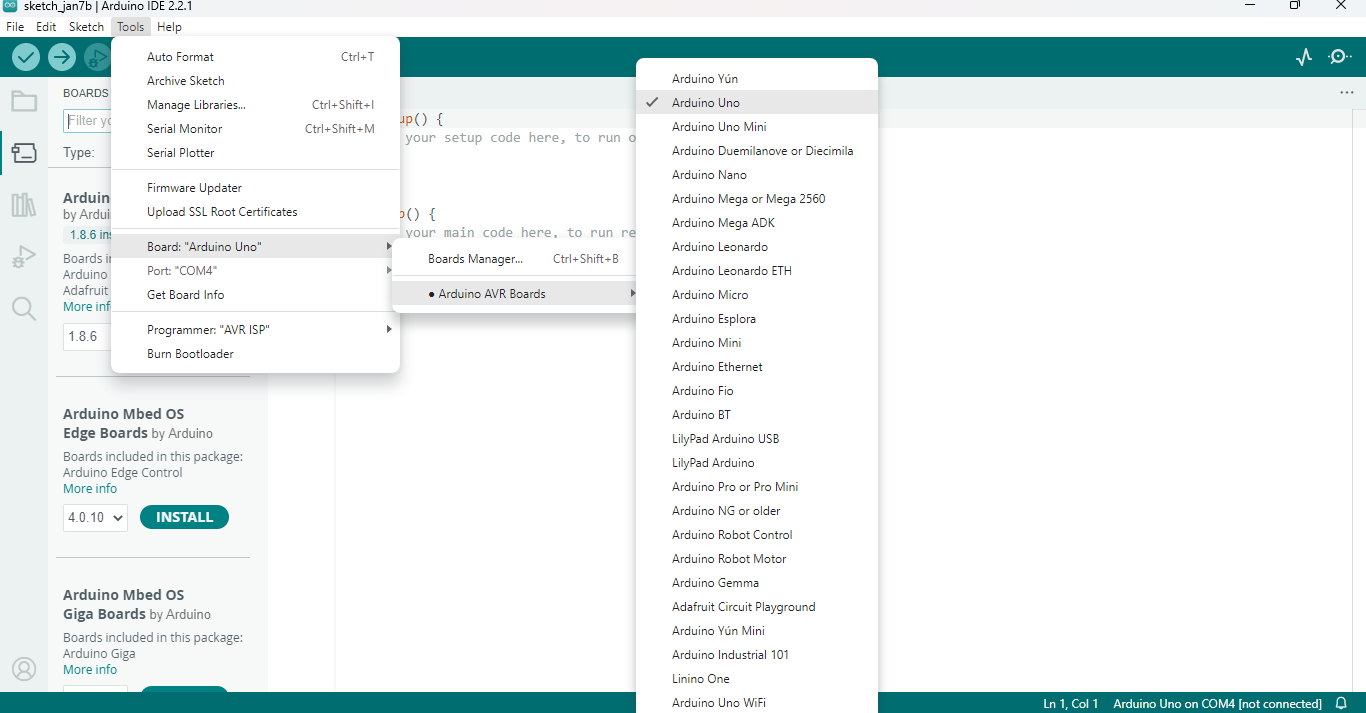
***Bảng 3.3 Cách kết nối Module Relay với bóng đèn***

**3.4 Nạp code cho Module As608 và Module Relay bằng phần mềm Arduino IDE:**

**3.4.1 Nạp code cho Module As608:**

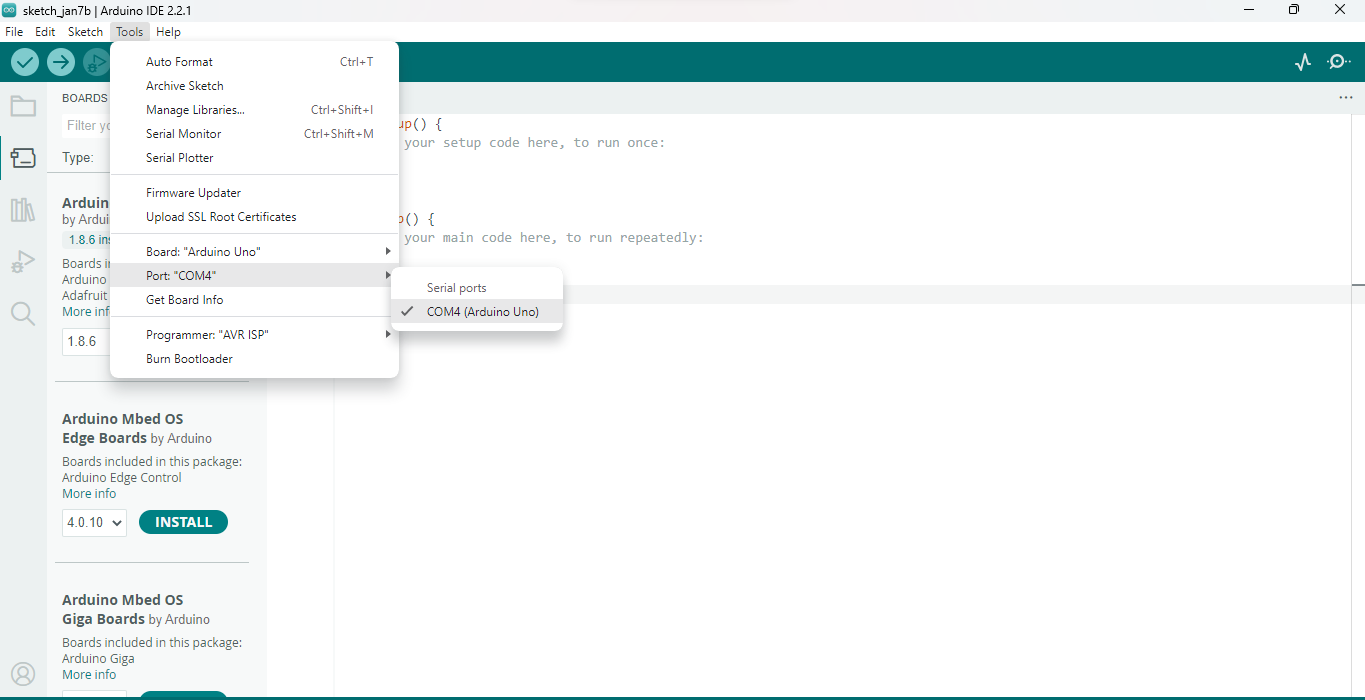
- Bước 1: Chọn Board và cổng Port

+ Vào Tools -> Board -> Arduino AVR Boards -> Arduino Uno.



***Hình 3.3 Chọn Board***

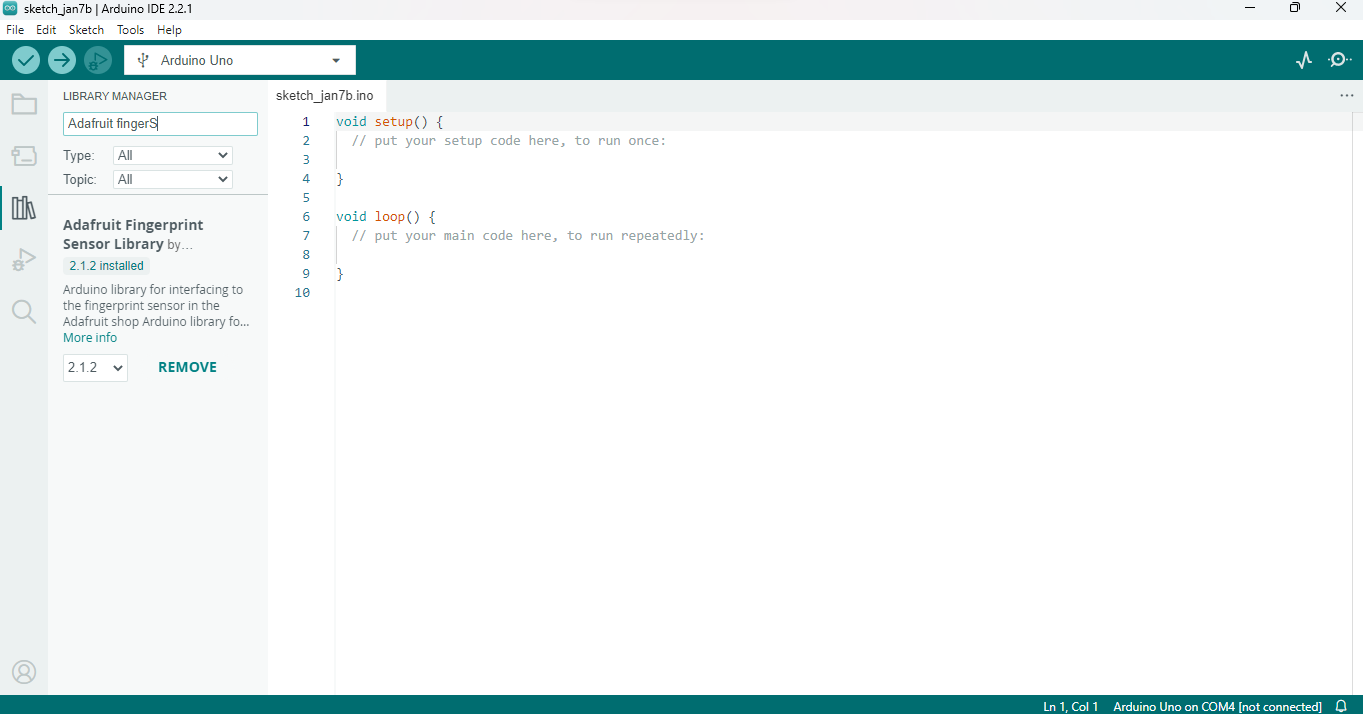
+ Vào Tools -> Port -> Chọn COM tương ứng.



***Hình 3.4 Chọn cổng Port***

- Bước 2: Cài đặt thư viện Adafruit Fingerprint Sensor Library.

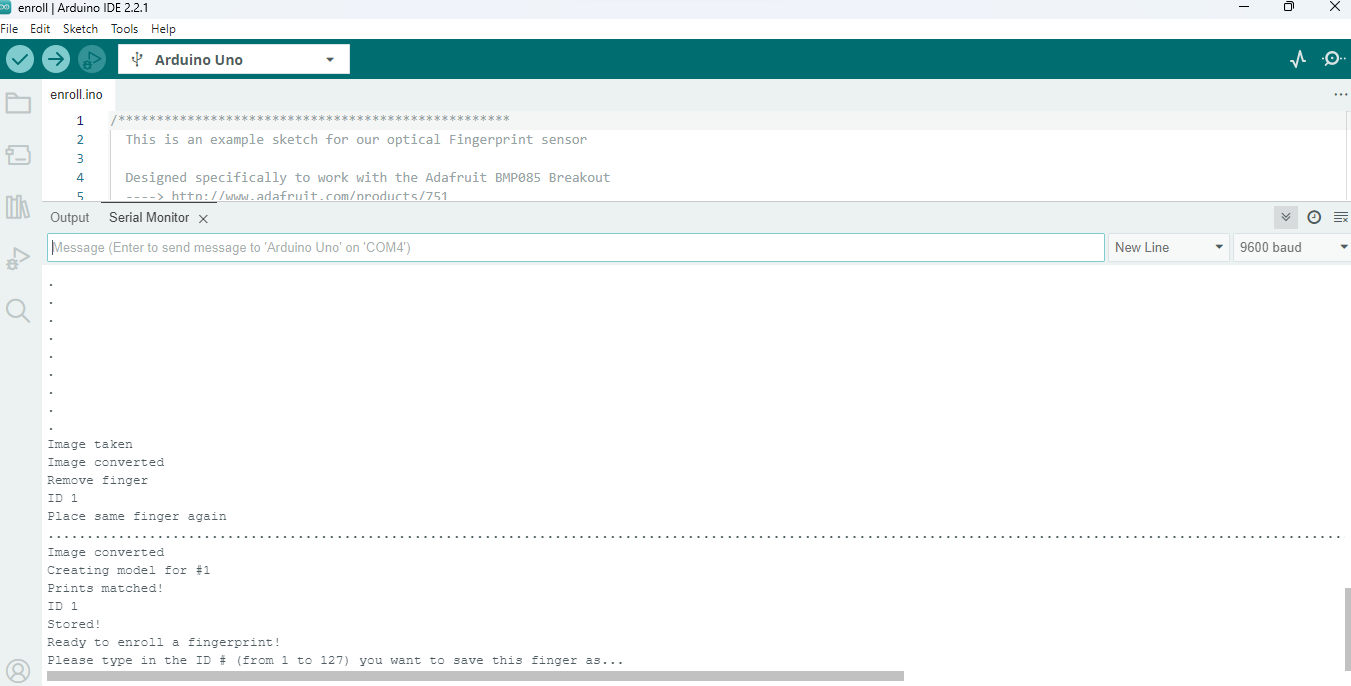
Vào phần mềm Arduino IDE -> Sketch -> Include Library -> Manage libraries -> Tìm thư viện Adafruit Fingerprint Sensor Library trên thanh tìm kiếm và cài đặt.



***Hình 3.5 Cài đặt thư viện Adafruit Fingerprint Sensor Library***

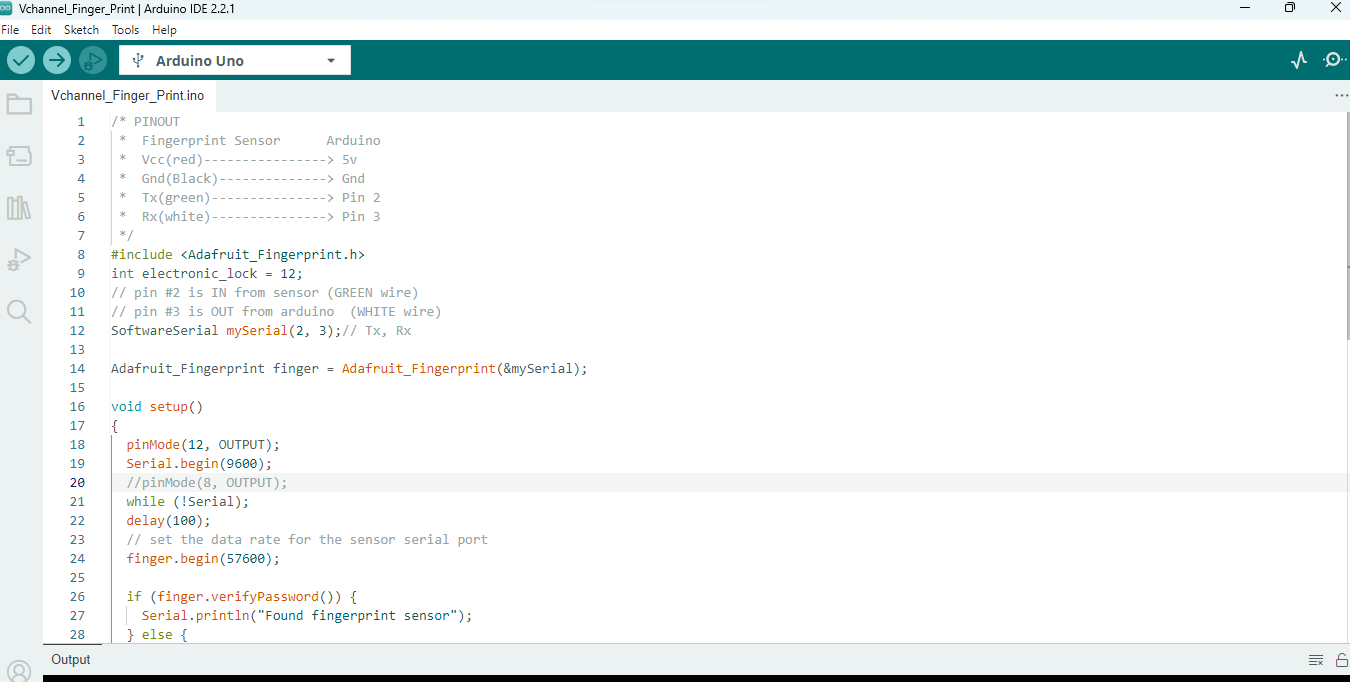
- Bước 3: Nạp vân tay vào Module cảm biến vân tay As608

Vào File -> Examples -> Adafruit Fingerprint Sensor Lib -> enroll -> upload -> nhập số thứ tự vân tay -> đưa vân tay muốn lưu vào đầu lọc Module As608



***Hình 3.6 Nạp vân tay vào Module cảm biến vân tay As608***

**3.4.2 Nạp code cho Module Relay:**



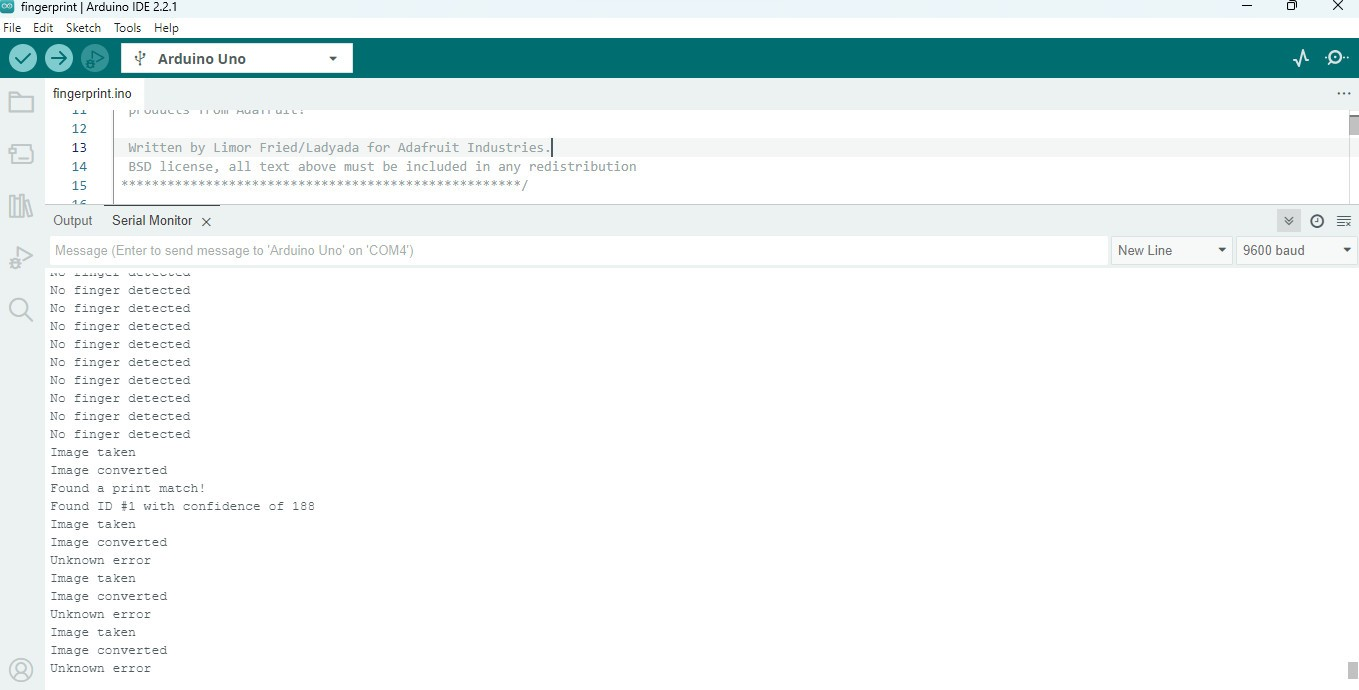
***Hình 3.7 Nạp code cho Module Relay***

**CHƯƠNG 4: KẾT QUẢ NGHIÊN CỨU**

**4.1 Nhận dạng đúng vân tay đã nạp:**

Sau khi đã nạp code cho Module cảm biến vân tay As608 với ID tương ứng với dấu vân tay đã nạp do người dùng đặt, chúng ta tiến hành xác nhận lại xem Arduino đã nhận được tín hiệu vân tay đã truyền qua cực số 2 trên Arduino bằng cách chạy đoạn code với thư viện “fingerprint” đã cài lúc đầu.

Chương trình sẽ chạy một vòng lặp vô hạn khi chúng ta đưa vân tay không đúng vào Module cảm biến vân tay As608, đến khi người dùng đưa đúng vân tay đã nạp thì chương trình sẽ trả về đúng ID ban đầu.

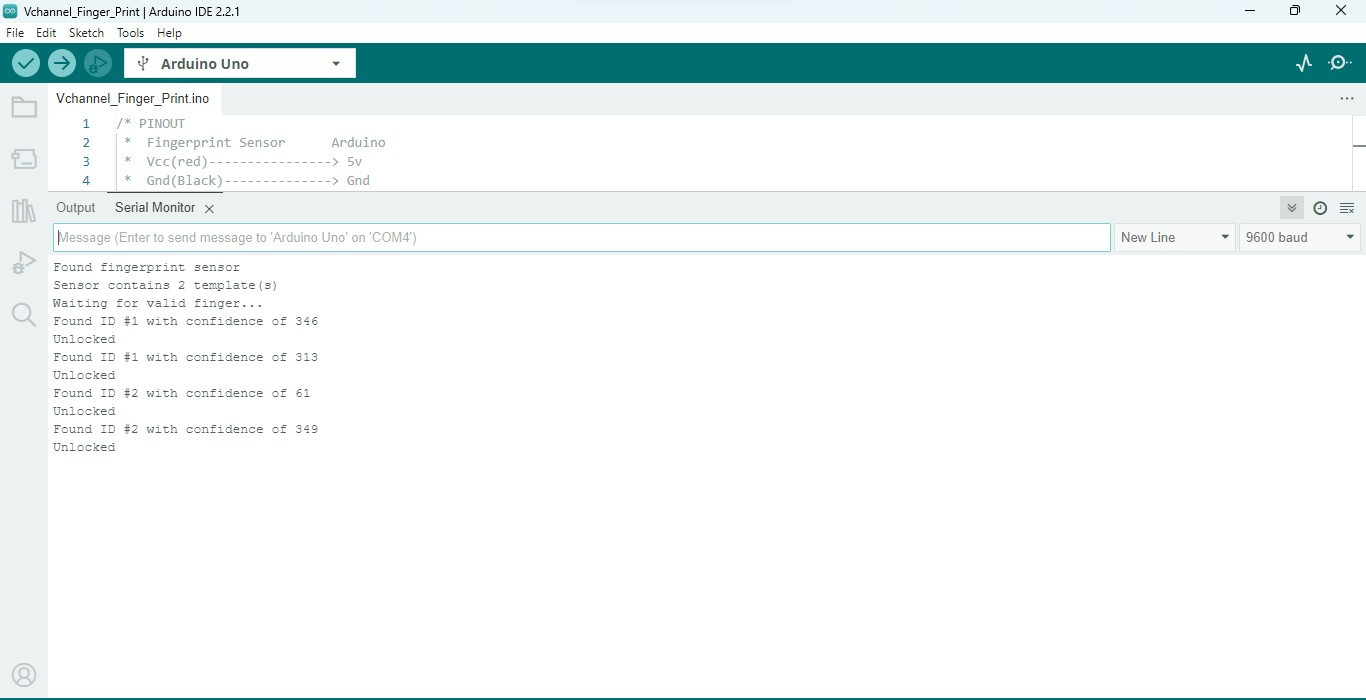


***Hình 4.1 Nhận dạng đúng vân tay đã nạp***

**4.2 Module Relay hoạt động:**

Khi xác nhận vân tay đã được lưu vào Arduino và hoạt động đúng yêu cầu, tiếp theo chúng ta sẽ nạp code cho Module Relay.

Sau khi nạp code chương trình sẽ hiển thị số vân tay đã nạp trước đó và yêu cầu người dùng đưa vân tay đã nạp vào, khi vân tay đúng chương trình sẽ hiện ID tương ứng với vân tay mà chúng ta thiết lập lúc đầu, đồng thời hiển thị mở khóa thành công và Module Realy lúc này sẽ đóng để điện chạy qua.



***Hình 4.2 Module Relay hoạt động***

**4.3 Mô hình thử nghiệm:**

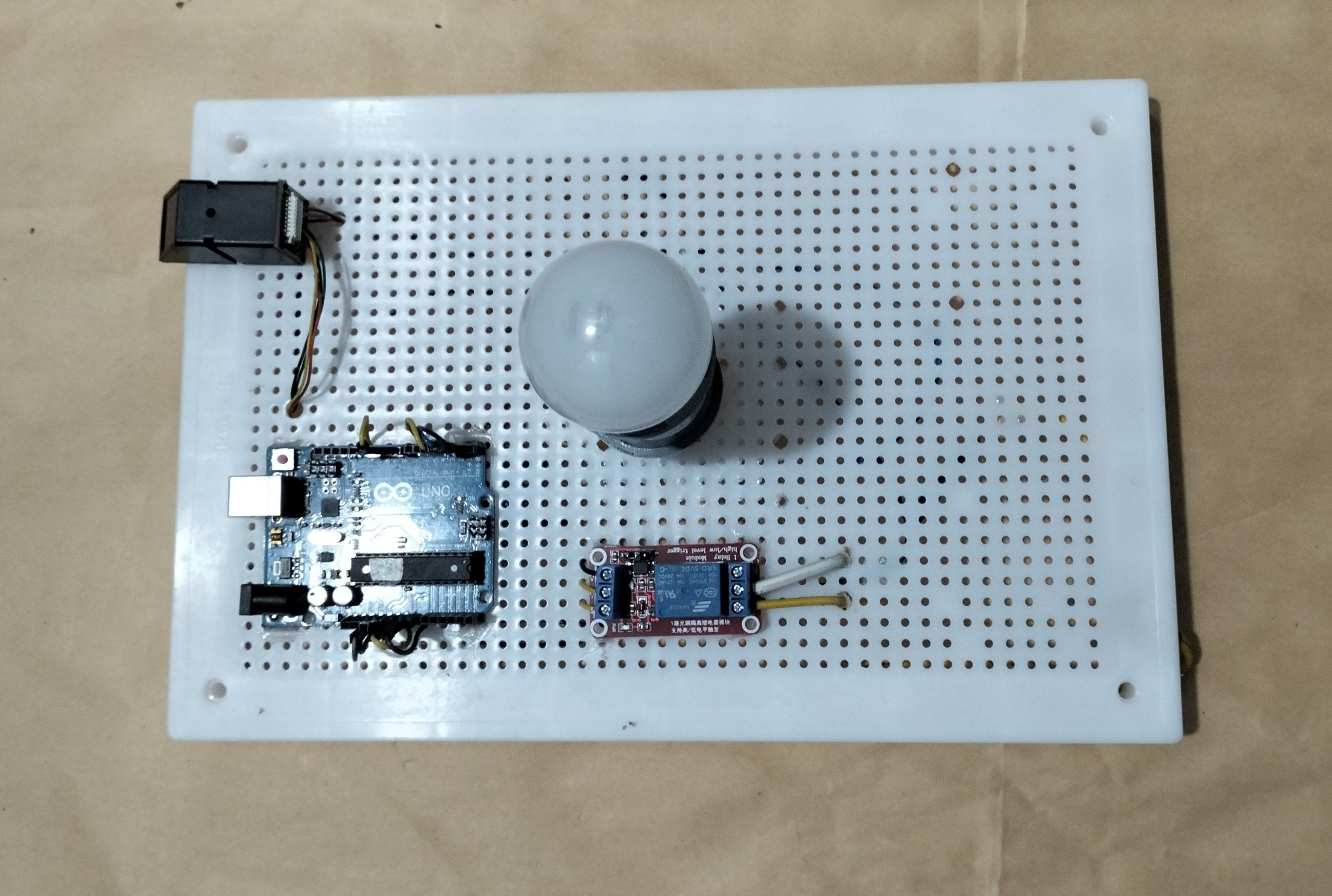
* **Với Module As608:**

Các thiết bị sau khi đã được nạp code đã hoạt động đúng với yêu cầu, với Module cảm biến vân tay chúng ta sẽ cấp nguồn âm và nguồn dương qua chân V+ và GND với chân 3,3V và chân GND trên Arduino, lúc này Module cảm biến vân tay đã sáng đèn.

Tiếp tục truyền dữ liệu và nhận dữ liệu từ Module As608 với chân TX và RX lần lượt vào chân số 2 và chân số 3 của Arduino.

* **Với Module Relay:**

Tương tự như Module As608, chúng ta cũng sẽ cấp nguồn âm và nguồn dương để nuôi Module Relay với DC+ và DC- lần lượt kết nối vào chân 5V và chân GND của Arduino. Tiếp đến để Module Relay nhận được tính hiệu vân tay đúng từ Arduino ta kết nối cổng IN vào chân số 12 của Arduino, lúc này khi vân tay đúng được nạp vào, Arduino gởi tín hiệu thông qua cổng số 12 vừa kết nối với Relay để đóng mạch và cho điện đi qua, lúc này ta sẽ kết nối Relay với bóng đèn và bóng đèn sẽ sáng.



***Hình 4.3 Mô hình thực nghiệm***

**CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

**5.1 Về kiến thức:**

- Nắm vững các kiến thức cơ bản về Arduino và vai trò của Arduino trong việc thiết kế một hệ thống.

- Nắm được nguyên lí hoạt động Arduino và sử dụng một cách thành thạo.

- Nắm được các kiến thức, thuật ngữ cơ bản của Arduino.

**5.2 Về thực hành:**

- Hiểu được quá trình thiết kế một hệ thống hoàn chỉnh.

- Biết được cách sử dụng mạch Arduino.

- Thiết kế và triển khai thành công hệ thống mở khóa bằng vân tay.

**5.3 Tự nhận xét:**

**5.3.1 Ưu điểm:**

- Thiết kế thành công hệ thống mở khóa bằng vân tay, đáp ứng yêu của đề tài.

- Tìm hiểu và nắm được cơ bản về mạch Arduino, phần mềm lập trình Arduino IDE.

- Hệ thống được thiết kế đơn giản, thân thiện và dễ sử dụng.

**5.3.2 Hạn chế:**

Vì thời gian và kiến thức có hạn nên hệ thống còn khá đơn giản và chưa tìm hiểu rõ và kỹ hơn về đề tài này.

- Phân tích thiết kế hệ thống chưa hoàn chỉnh.

- Chưa tối ưu được độ phức tạp của hệ thống.

**5.4 Hướng phát triển:**

Với những ưu điểm vượt trội của khóa cửa bằng vân tay như độ an toàn và bảo một cao, dễ dàng làm quen và sử dụng, chi phi đầu tư và sử dụng thấp… Đồng thời khắc phục được những loại khóa cửa truyền thống ngày nay khi mà ngày càng lộ rõ những nhược điểm về vấn đề an ninh, sự bất tiện khi bị kẹt khóa, làm mất chìa,…

Hiện nay công nghệ vân tay đã được ứng dụng rất phổ biến trên toàn cầu, nó đã giúp cho việc quản lý, kiểm soát ra vào của một cách dễ dàng hơn, vân tay là sự thay thế một cách thông minh nhất cho các hệ thống khóa cửa thông thường dùng bằng chìa khóa, với sự tiện dụng, nhanh chóng tính thẩm mỹ cao, đặc biệt là sự an toàn tuyệt đối.

**DANH MỤC TÀI LIỆU THAM KHẢO**

|  |  |
| --- | --- |
| [1] | Giao Tiếp Cảm Biến Nhận Dạng Vân Tay As608:  <https://tapit.vn/giao-tiep-cam-bien-nhan-dang-van-tay-as608-phan-1/>, ngày truy cập: 14/12/2023 |
| [2] | Giới thiệu cảm biến vân tay, hướng dẫn sử dụng cảm biến vân tay với Arduino, thực hành làm bộ mở khóa cửa bằng cảm biến vân tay và Arduino:  <https://nshopvn.com/blog/huong-dan-su-dung-cam-bien-van-tay-voi-arduino/> , ngày truy cập: 14/12/2023 |
| [3] | Arduino Uno R3 là gì ?  <http://arduino.vn/bai-viet/42-arduino-uno-r3-la-gi>, ngày truy cập: 14/12/2023 |