**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN HỌC PHẦN: LẬP TRÌNH IoT**

**ĐỀ TÀI: KHÓA CỬA THÔNG MINH**

**Giáo viên hướng dẫn: Nguyễn Tài Tuyên**

**Sinh viên thực hiện: Nguyễn Mậu Lân**

**Ngô Đức Hải**

**Nguyễn Tấn Dũng**

**Phạm Công Chứ**

**Hà Nội, tháng 3 năm 2023**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC ĐẠI NAM**

Logo, company name

Description automatically generated

**BÀI TẬP LỚN**

**TÊN HỌC PHẦN: LẬP TRÌNH IoT**

**ĐỀ TÀI: KHÓA CỬA THÔNG MINH**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Mã Sinh Viên | Họ và Tên | Điểm | |
| Bằng Số | Bằng Chữ |
| 1 | 1571020150 | Nguyễn Mậu Lân |  |  |
| 2 | 1571020081 | Ngô Đức Hải |  |  |
| 3 | 1571020031 | Phạm Công Chứ |  |  |
| 4 | 1571020039 | Nguyễn Tấn Dũng |  |  |

**CÁN BỘ CHẤM THI**

**Hà Nội, tháng 3 năm 2023**

**LỜI CẢM ƠN**

Đầu tiên, em xin gửi lời cảm ơn chân thành đến Trường đại học Đại Nam đã đưa môn học “Lập trình IoT” vào chương trình giảng dạy. Đặc biệt, em xin gửi lời cảm ơn sâu sắc đến giảng viên bộ môn - thầy **Nguyễn Tài Tuyên** đã dạy dỗ, truyền đạt những kiến thức quý báu cho em trong suốt thời gian học tập vừa qua. Trong thời gian tham gia lớp học của thầy, em đã có thêm cho mình nhiều kiến thức bổ ích, tinh thần học tập hiệu quả, nghiêm túc. Đây chắc chắn sẽ là những kiến thức quý báu, là hành trang để em có thể vững bước sau này.

Bộ môn “Lập trình IoT” là môn học thú vị, vô cùng bổ ích và có tính thực tế cao. Đảm bảo cung cấp đủ kiến thức, gắn liền với nhu cầu thực tiễn của sinh viên. Tuy nhiên, do vốn kiến thức còn nhiều hạn chế và khả năng tiếp thu thực tế còn nhiều bỡ ngỡ. Mặc dù em đã cố gắng hết sức nhưng chắc chắn bài tiểu luận khó có thể tránh khỏi những thiếu sót và nhiều chỗ còn chưa chính xác, kính mong thầy xem xét và góp ý để bài này của em được hoàn thiện hơn.

Em xin chân thành cảm ơn!

**MỞ ĐẦU**

Nhà là nơi an toàn và riêng tư, khóa cửa thông minh được sử dụng như một công cụ đắc lực để tăng cường an ninh cho mỗi gia đình. Mỗi một căn hộ thường có nhiều loại cửa trong đó cửa chính là nơi ra vào đễ dàng nhất.

Khóa thông minh cung cấp giải pháp bảo mật cao, giúp kiểm soát việc ra vào được tốt hơn, khóa cửa truyền thống rất dễ bị bẻ gãy bởi kết cấu thô sơ, vì thế không hiếm gặp nhiều vụ trộm xảy ra liên tục. Nhận thấy tầm quan trọng của vấn đề này. Chúng em đã lựa chọn đề tài: “Khóa cửa thông minh sử dụng Node MCU & module RFID RC 522” để làm báo cáo của mình.

Kết quả nghiên cứu từ đề tài này sẽ giúp em có nhiều kinh nghiệm hơn. Và sau này chúng em sẽ có đủ khả năng nghiên cứu chế tạo hoàn chỉnh thiết bị điều khiển hệ thống điện cho ngôi nhà thông minh đáp ứng được nhu cầu của người dùng.

**MỤC LỤC**

[LỜI CẢM ƠN 3](#_Toc512248029)

MỞ ĐẦU [4](#_Toc512248029)

[MỤC LỤC 5](#_Toc512248029)

[DANH MỤC HÌNH MINH HỌA 6](#_Toc512248029)

[CHƯƠNG 1: TỔNG QUAN 7](#_Toc512248029)

[1.1 Lý do chọn đề tài 7](#_Toc512248022)

[1.2 Mục đích nghiên cứu 7](#_Toc512248018)

[1.3 Phạm vi nghiên cứu 7](#_Toc512248022)

[1.4 Nội dung nghiên cứu 7](#_Toc512248018)

[1.5 Phương pháp nghiên cứu 7](#_Toc512248022)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 9](#_Toc512248029)

[2.1 Kit NodeMCU Lua ESP8266 9](#_Toc512248018)

[2.2 Module relay 1 kênh 11](#_Toc512248018)

[2.3 Module RFID RC522 12](#_Toc512248028)

[2.4 Khóa từ 12V 14](#_Toc512248028)

[CHƯƠNG 3: Thực nghiệm 16](#_Toc512248029)

[3.1 Lập trình ESP8266 bằng Arduino IDE 16](#_Toc512248018)

[3.1.1 Giới thiệu 16](#_Toc512248018)

[3.1.2 Cài đặt thư viện cho ESP8266 16](#_Toc512248018)

[3.1.3 Nạp chương trình cho ESP8266 18](#_Toc512248018)

[3.2 Lắp mạch nguyên lý 20](#_Toc512248018)

[3.3 Code chương trình 22](#_Toc512248018)

[3.4 Kết quả 25](#_Toc512248018)

[KẾT LUẬN 28](#_Toc512248038)

[TÀI LIỆU THAM KHẢO 29](#_Toc512248038)

**Danh Mục Hình Minh Họa**

[Hình 2.1: Hình esp8266](#_Toc130398343) 9

[Hình 2.2: Sơ đò chân esp8266 9](#_Toc130398344)

[Hình 2.3: Các bộ phận esp8266 11](#_Toc130398345)

[Hình 2.4: Module Relay 12](#_Toc130398346)

[Hình 2.5: Sơ đồ chân RFID RC522 13](#_Toc130398347)

[Hình 2.6: Khóa chốt từ 12v 15](#_Toc130398348)

[Hình 3.1: Giao diện khi mở Arduino 16](#_Toc130398349)

[Hình 3.2: Chỗ để gắn link 17](#_Toc130398350)

[Hình 3.3: Lựa chọn Boards Manager 17](#_Toc130398351)

[Hình 3.4: Tải Boards esp8266 18](#_Toc130398352)

[Hình 3.5: Lựa chọn Boards 18](#_Toc130398353)

[Hình 3.6: Chọn cổng COM tương ứng 19](#_Toc130398354)

[Hình 3.7: Kết quả test led esp8266 19](#_Toc130398355)

[Hình 3.8: Nối chân esp8266 với RFID 20](#_Toc130398356)

[Hình 3.9: Nối chân esp8266 với module relay 20](#_Toc130398357)

[Hình 3.10: Sơ đồ toàn bộ mạch 21](#_Toc130398358)

[Hình 3.11: Hình ảnh thực tế 21](#_Toc130398359)

[Hình 3.12: Thông báo khi có tín hiệu quẹt thẻ 25](#_Toc130398360)

[Hình 3.13: Hình ảnh còi báo 26](#_Toc130398361)

[Hình 3.14: Giao diện Web 26](#_Toc130398362)

[Hình 3.15: Thông báo khi mở 26](#_Toc130398363)

[Hình 3.16: Hình ảnh button 27](#_Toc130398364)

**CHƯƠNG 1. TỔNG QUAN**

**1.1 Lý do chọn đề tài**

Ngày nay, các loại khóa truyền thống gần như không còn phù hợp với nhịp sống hiện đại hóa ngày này. Trong quá trình sử dụng khóa truyền thống thường làm người dùng đối mặt với một số bất cập thường thấy như thất lạc chìa khóa, phải tạo nhiều chìa khóa cơ cho các thành viên (trong gia đình/ công ty/ …).

Từ lâu bộ chìa khóa cơ đã gắn liền với cửa nhà mỗi gia đình. Khóa có nhiệm vụ bảo vệ không cho người lạ cũng như kẻ gian đột nhập vào nhà với những ý đồ xấu. Chúng ta không thể phủ nhận được rằng bộ khóa cơ thông thường có tác dụng ngăn cản hay kéo dài thời gian hành động của kẻ trộm.

Khóa cơ rất dễ mở, đôi khi những kẻ trộm không cần dùng chìa mà bọn chúng chỉ cần vài thủ thuật thôi là có thể mở được khóa trong ít phút. Đơn giản với 2 thanh sắt dẹp, một thanh để xoay trục ổ khóa, một thanh để đẩy các chốt lò xo về vị trí mở. Thế là ai ai cũng có thể vào nhà bạn chỉ trong vài nháy mắt.

Chính vì thế, khóa cửa thông minh sẽ thay thế các loại khóa truyền thống. Nó đang dần khẳng định vị thế của mình vì: Công nghệ bảo mật cao; tiện ích – cho phép chủ động quản lý ra vào; điều khiển khóa từ xa qua ứng dụng điện thoại; mẫu mã thiết kế sang trọng, hiện đại.

**1.2 Mục đích nghiên cứu**

* Tìm hiểu về hệ thống khóa cửa thông minh
* Tìm hiểu về mạch điều khiển thiết bị khóa cửa thông minh
* Xây dựng mô hình mẫu
* Thiết kế mạch điều khiển thiết bị khóa cửa thông minh
* Kiểm tra, đánh giá ứng dựng của đề tài

**1.3 Phạm vi nghiên cứu**

* Phạm vi đề tài ở mức demo hệ thống và chưa đưa vào thực tế

sử dụng được.

* Cho phép mở khóa cửa bằng thẻ và qua WebServer

**1.4 Nội dung nghiên cứu**

* Mạch esp8266
* Module relay
* Module RFID RC552

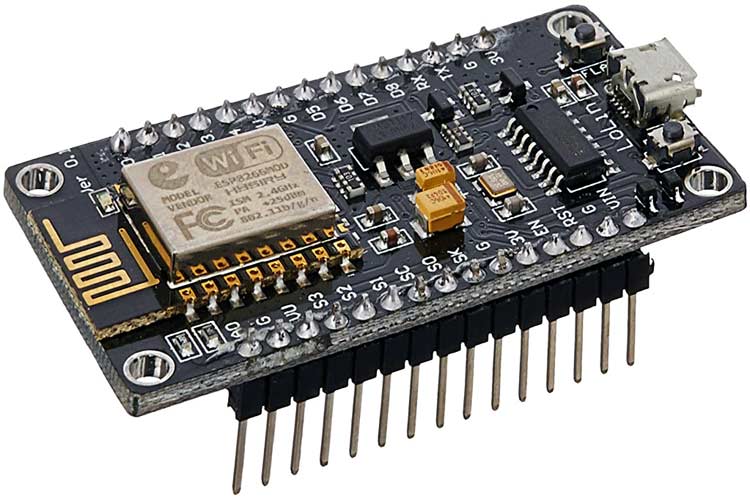
**1.5 Phương pháp nghiên cứu**

* Nghiên cứu tài liệu qua sách báo về lĩnh vực IoT
* TÌm hiểu nguyên lý hoạt động của module Node MCU esp8266 và các mode phụ trợ, thiết bị điện ngoại vi.

**CHƯƠNG 2. CƠ SỞ LÝ THUYẾT**

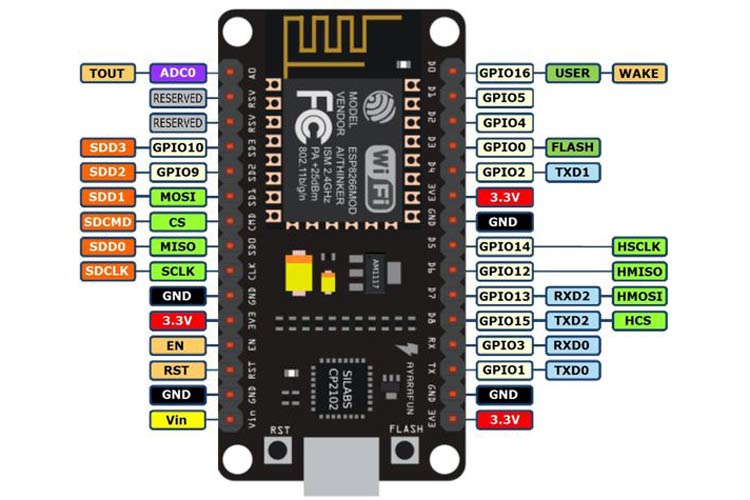
**2.1 Kit NodeMCU Lua ESP8266**

NodeMCU là một phần mềm nguồn mở dựa trên Lua và bảng phát triển được nhắm mục tiêu đặc biệt cho các Ứng dụng dựa trên IoT. Nó bao gồm phần sụn chạy trên ESP8266 Wi-Fi SoC của Espressif Systems và phần cứng dựa trên mô-đun ESP-12.



Hình 2.1. Hình esp8266

Sơ đồ chân NodeMCU ESP8266



Hình 2.2. Sơ đồ chân esp8266

|  |  |  |
| --- | --- | --- |
| Ghim danh mục | Tên | Sự miêu tả |
| Power | Micro-USB, 3.3V, GND, Vin | Micro-USB: NodeMCU có thể được cấp nguồn qua cổng USB  3.3V: 3.3V quy định có thể được cung cấp cho chân này để cấp nguồn cho bo mạch  GND: Chân nối đất  Vin: Nguồn điện bên ngoài |
| Control Pins | EN, RST | Chốt và nút đặt lại bộ vi điều khiển |
| Analog Pin | A0 | Được sử dụng để đo điện áp tương tự trong khoảng 0-3,3V |
| GPIO Pins | GPIO1 đến GPIO16 | NodeMCU có 16 chân đầu vào-đầu ra mục đích chung trên bo mạch của nó |
| SPI Pins | SD1, CMD, SD0, CLK | NodeMCU có sẵn bốn chân để giao tiếp SPI. |
| UART Pins | TXD0,RXD0, TXD2, RXD2 | NodeMCU có hai giao diện UART, UART0 (RXD0 & TXD0) và UART1 (RXD1 & TXD1). UART1 được sử dụng để tải lên phần sụn / chương trình. |
| I2C Pins |  | NodeMCU có hỗ trợ chức năng I2C nhưng do chức năng bên trong của các chân này, bạn phải tìm chân nào là I2C. |

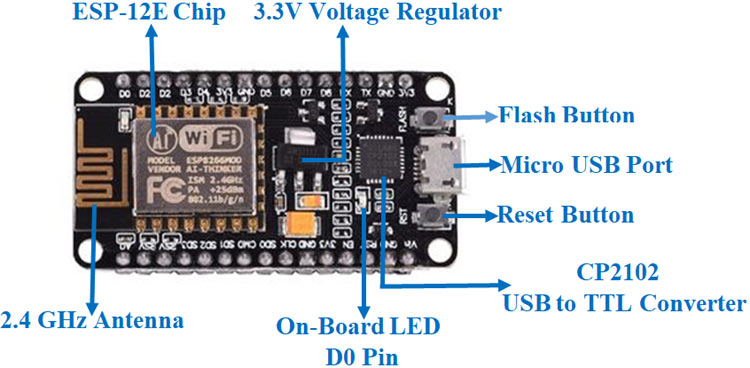
Thông số kỹ thuật và tính năng

* Bộ vi điều khiển: CPU RISC 32-bit Tensilica Xtensa LX106
* Điện áp hoạt động: 3.3V
* Điện áp đầu vào: 7-12V
* Chân I / O kỹ thuật số (DIO): 16
* Chân đầu vào tương tự (ADC): 1
* UARTs: 1
* SPI: 1
* I2Cs: 1
* Bộ nhớ Flash: 4 MB
* SRAM: 64 KB
* Tốc độ đồng hồ: 80 MHz
* USB-TTL dựa trên CP2102 được bao gồm trên bo mạch, cho phép Plug n Play
* Ăng-ten PCB
* Mô-đun có kích thước nhỏ để phù hợp thông minh bên trong các dự án IoT của bạn

Bảng phát triển NodeMCU ESP8266 đi kèm với mô-đun ESP-12E chứa chip ESP8266 có bộ vi xử lý Tensilica Xtensa 32-bit LX106 RISC. Bộ vi xử lý này hỗ trợ RTOS và hoạt động ở tần số xung nhịp có thể điều chỉnh từ 80MHz đến 160 MHz.

NodeMCU có 128 KB RAM và 4MB bộ nhớ Flash để lưu trữ dữ liệu và chương trình. Sức mạnh xử lý cao của nó với Wi-Fi / Bluetooth và các tính năng Điều hành Ngủ sâu tích hợp khiến nó trở nên lý tưởng cho các dự án IoT.

NodeMCU có thể được cấp nguồn bằng giắc cắm Micro USB và chân VIN (Chân nguồn cung cấp bên ngoài). Nó hỗ trợ giao diện UART, SPI và I2C.



Hình 2.3. Các bộ phận trên esp8266

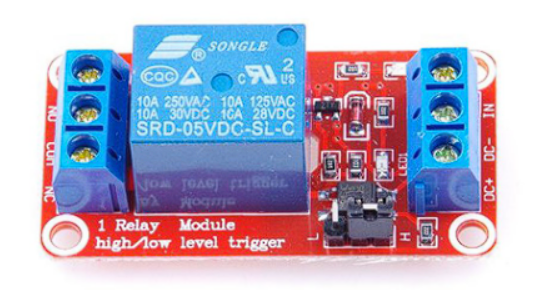
**2.2 Module Relay 1 Kênh 5V**

Module Relay 5V 1 Kênh được dùng như một công tắc điện , dùng để điều khiển các thiết bị công suất lớn ( đèn, động cơ, ...) - Module Relay 5V 1 Kênh gồm 1 rơ le hoạt động tại điện áp 5VDC, 12VDC chịu được hiệu điện thế lên đến 250VAC 10A. Module relay 1 kênh được thiết kế chắc chắn, khả năng cách điện tốt.

Module Relay 1 kênh 5V  gồm 1 rơ le điện áp hoạt động ở mức 5VDC, đầu ra điều khiển hiệu điện tối đa ở mức 250V 10A đối với điện áp xoay chiều AC và 30V với điện áp 1 chiều DC  
 Module relay 1 kênh nhỏ gọn chuyên nghiệp, khả năng chống nhiễu tốt và khả năng cách điện tốt. Trong module đã có sẵn mạch kích relay sử dụng IC cách ly quang và transistor giúp cách ly hoàn toàn mạch vi điều khiển với rơ le bảo đảm vi điều khiển hoạt động ổn định.  
 Có sẵn header rất tiện dụng khi kết nối với vi điều khiển. Có các lỗ bắt vít rất tiện lợi dễ lắp đặt trong hệ thống mạch.

Mạch điều khiển relay 1 kênh này sử dụng chân kích mức Thấp (0V), khi có tín hiệu 0V vào chân IN thì relay sẽ nhảy qua thường Mở của Relay.  
 Ứng dụng với relay module khá nhiều bao gồm cả điện DC hay AC.

Hình ảnh của modelu relay:

****

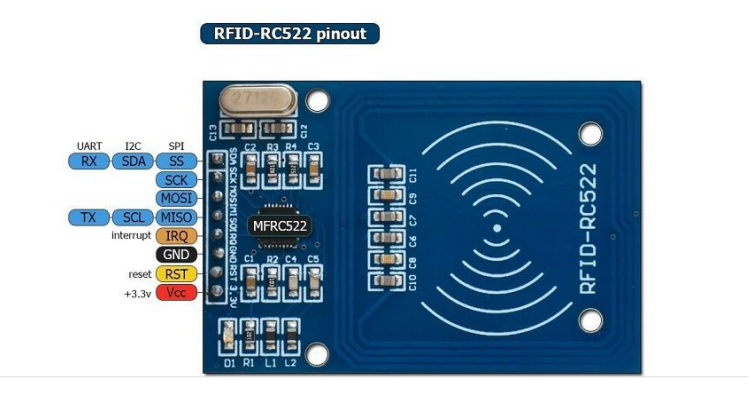
Hình 2.4. Module Relay

Thông số kỹ thuật Module relay:

* Kích thước: 53mm (chiều dài) \* 28.3mm (chiều rộng) \* 19.3mm (H)
* Trọng lượng: 18g
* Bốn lỗ để bắt vít cố định có đường kính 3.1mm, dễ dàng lắp đặt trong hệ thống mạch
* Opto cô lập, tốt chống nhiễu
* Có đèn báo đóng ngắt trên Relay.
* Sử dụng điện áp nuôi DC 5V.
* Đầu ra điện thê đóng ngắt tối đa: DC 30V / 10A, AC 250V / 10A

**2.3 Module RFID RC522**

Module RFID RC522 sử dụng IC MFRC522 của Phillip dùng để đọc và ghi dữ liệu cho thẻ NFC tần số 13.56mhz, với mức giá rẻ thiết kế nhỏ gọn, module này là sự lựa chọn hàng đầu cho các ứng dụng về ghi đọc thẻ RFID. RFID (Radio Frequency Identification) là công nghệ nhận dạng đối tượng bằng sóng vô tuyến. Công nghệ này cho phép nhận biết các đối tượng thông qua hệ thống thu phát sóng radio, từ đó có thể giám sát, quản lý hoặc lưu vết từng đối tượng. Một hệ thống RFID thường bao gồm 2 thành phần chính là thẻ tag (chip RFID chứa thông tin) và đầu đọc (reader) đọc các thông tin trên chip



Hình 2.5. Sơ đồ chân RFID RC522

Thông số kỹ thuật:

|  |  |
| --- | --- |
| Chân | 8 |
| Điện áp hoạt động | 3.3V |
| Dòng khi hoạt động | 1A |
| Dòng ở chế độ chờ | 80Ua |
| Tần số | 13.56MHz |
| Chuẩn truyền | SPI |
| Tốc độ truyền dữ liệu | 10Mbit/s |
| Khoảng cách hoạt động | 0 – 60mm |
| Kích thước | 40mm x 60mm |

Chân cấp nguồn / reset

|  |  |  |
| --- | --- | --- |
| Chân | | Mô tả chi tiết |
| Chân 1 | VCC | Chân nguồn VCC. Trong một số phiên bản của RC522, chân này được ký hiệu là 3V3 thay vì VCC. |
| Chân 2 | RST | Là chân reset được sử dụng để đặt lại giá trị trong trường hợp xảy ra lỗi khi thiết bị không bất kỳ phản hồi. |
| Chân 3 | Ground | Chân nối đất giúp tạo mass chung với các thiết bị bên ngoài (ví dụ bộ nguồn, bi điều khiển hoặc arduino). |

Chân giao tiếp SPI và UART

|  |  |  |  |
| --- | --- | --- | --- |
| Chân | | | Mô tả chi tiết |
| Chân 4 | IRQ | | Linh kiện có thể chuyển sang chế độ ngủ để tiết kiệm điện năng và chân IRQ sẽ khởi động lại nó |
| Chân 5 | MISO | SCL  TX | Chân này kết nối với Arduino / Vi điều khiển để giao tiếp SPI. Truyền dữ liệu từ module sang Arduino.  Chân MISO cũng có thể sử dụng cho các chức năng khác thay vì SPI.  Cũng có thể giao tiếp I2C và UART Serial để giao tiếp dữ liệu với module. |
| Chân 6 | MOSI | | MOSI là chân đầu vào dữ liệu module RFID khi giao tiếp SPI |
| Chân 7 | SCK | | Các chân SCK gửi xung clock khi giao tiếp SPI. |
| Chân 8 | SS | SDA  RX | Chân SS là chân kích hoạt chip giao tiếp SPI. Nhận tín hiệu khi Master (Arduino) giao tiếp SPI.  Chân SS của RFID có thể được sử dụng như một chân thứ hai (SDA) của giao tiếp I2C.  Cũng là chân nhận dữ liệu trong quá trình giao tiếp UART. |

**2.4 Khóa từ 12V**

Khóa chốt điện từ 12V LY-01 có chức năng hoạt động như một ổ khóa cửa sử dụng Solenoid để kích đóng mở bằng điện, được sử dụng nhiều trong nhà thông minh hoặc các loại tủ, cửa phòng, cửa kho,..., khóa sử dụng điện áp 12V, là loại thường đóng (cửa đóng) với chất lượng tốt, độ bền cao.

****

Hình 2.6. Khóa chốt từ 12V

**Thông số kỹ thuật**

* Vật liệu: Thép không gỉ
* Nguồn điện: 12V DC
* Dòng điện làm việc: 0.8A
* Công suất: 9.6W
* Yêu cầu nguồn cấp: 12VDC/1A
* Kích thước: L54 x D38 x H28
* Thời gian cấp nguồn: Nhỏ hơn 10s
* Trọng lượng: 150g

**CHƯƠNG 3: THỰC NGHIỆM**

**3.1 Lập trình ESP 8266 bằng Arduino IDE**

**3.1.1 Giới thiệu**

Để bắt đầu với những dự án Wifi các bạn cần ESP8266 với giá thành rẻ, và dễ dàng sử dụng. Đặc biệt ESP8266 có thể được lập trình bằng Arduino IDE. Trước khi bắt đầu những chuỗi dự án với ESP8266, ở bài viết này, mình sẽ chia sẻ cách cài đặt Arduino IDE để nạp code cho ESP8266.

**3.1.2 Cài đặt thư viện cho ESP8266**

Sau khi tải phần mềm Arduino IDE, chúng ta tiến hành cài đặt như bình thường và mở chương trình lên.

Graphical user interface, text, application

Description automatically generated

Hình 3.1. Giao diện khi mở Arduino

Để tiến hành cài đặt thư viện và chức năng nạp code cho IDE chúng ta làm như sau:

Vào **File→ Preferences**, vào **textbox Additional Board Manager URLs** thêm đường link sau vào

*http://arduino.esp8266.com/stable/package\_esp8266com\_index.json*

Click OK để chấp nhận

Graphical user interface, text, application, email

Description automatically generated

Hình 3.2. Chỗ để gắn link

Tiếp theo vào **Tool→Board→Boards Manager**

Graphical user interface, text, application

Description automatically generated

Hình 3.3. Lựa chọn Boards Manager

Đợi một lát để chương trình tìm kiếm. Ta kéo xuống và click vào **ESP8266 by ESP8266 Community**, click vào Install. Chờ phần mềm tự động download và cài đặt.

Graphical user interface, text, application, email

Description automatically generated

Hình 3.4. Tải Boards esp8266

**3.1.3. Hướng dẫn nạp chương trình cho ESP8266**

Kết nối mudule ESP8266 -12E vào máy tính. Vào **Tool→Board→NodeMCU 1.0**, chọn cổng COM tương ứng với module tương ứng.

Chọn chế độ nạp **Arduino as ISP** và chọn cổng **COM** cho đúng nhé.Vậy là ta đã có môi trường lập trình cho esp8266 rất thân thiện.

Graphical user interface, text, application

Description automatically generated

Hình 3.5. Lựa chọn Boards

Graphical user interface, text, application, email

Description automatically generated

Hình 3.6. Chọn cổng COM tương ứng

Sau khi kết nối ESP8266 với máy tính, chúng ta sẽ test code ESP8266 ở đây: Code ở dưới test led trên board esp8266 12E sáng tắt trong vòng 1 giây.

void setup**()** **{**

pinMode**(**LED\_BUILTIN**,** OUTPUT**);**

**}**

void loop**()** **{**

digitalWrite**(**LED\_BUILTIN**,** HIGH**);**

delay**(**1000**);**

digitalWrite**(**LED\_BUILTIN**,** LOW**);**

delay**(**1000**);**

**}**

A picture containing text, keyboard, indoor, computer

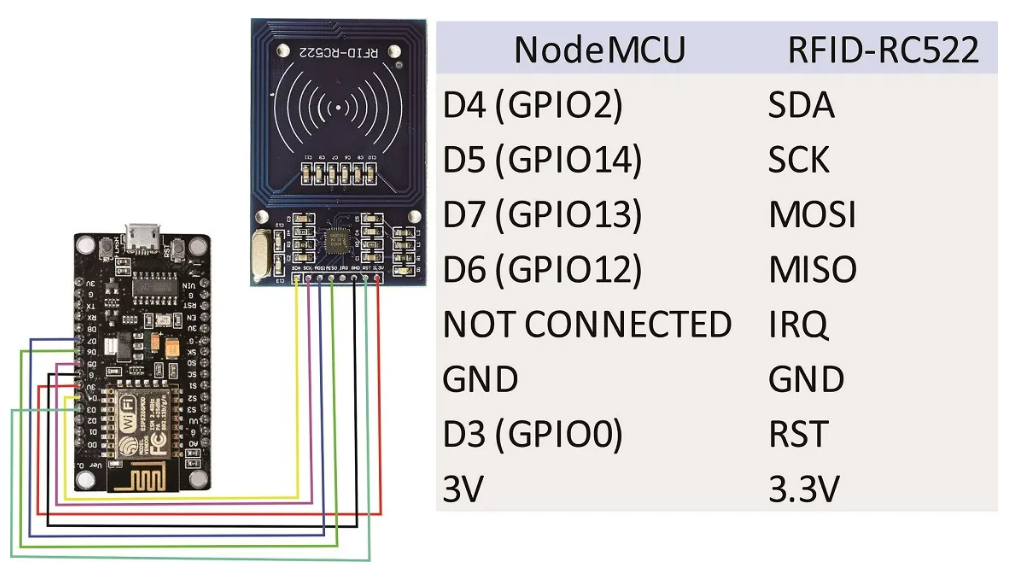
Description automatically generated

Hình 3.7. Kết quả test đèn esp8266

Từ ví dụ đơn giản ở trên ta có thể bắt tay vào lập trình cho NODE MCU để làm những dự dán IOT có ứng dụng cao.

**3.2 Lắp mạch nguyên lý:**

**Kết nối ESP8266 với RFID RC522**

****

Hình 3.8. Nối chân esp8266 với RFID

**Kết nối ESP8266 với module relay**

**A picture containing timeline

Description automatically generated**

Hình 3.9. Nối chân esp8266 với module relay

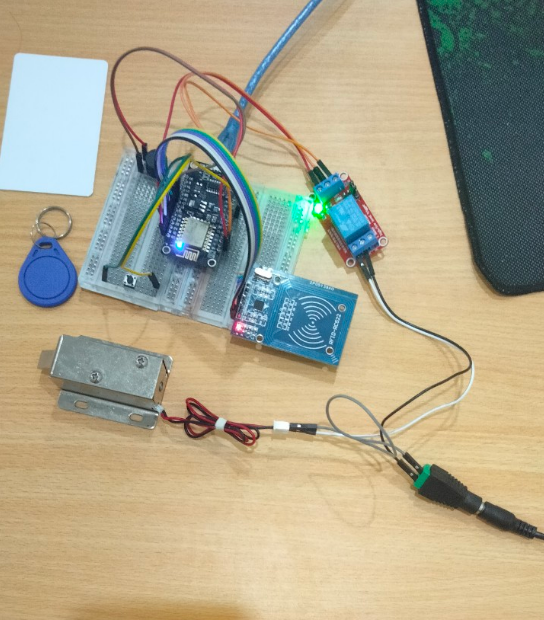
**Toàn bộ mạch**

**Diagram, schematic

Description automatically generated**

Hình 3.10. Sơ đồ toàn bộ mạch

**Hình ảnh mô hình thực tế**

****

Hình 3.11 Hình ảnh thực tế

**3.3 Code chương trình**

#include <SPI.h>

#include <MFRC522.h>

#include <ESP8266WiFi.h>

#include <WiFiClient.h>

#include <ESP8266WebServer.h>

#include <ESP8266mDNS.h>

IPAddress apIP**(**3**,** 3**,** 3**,** 3**);**

ESP8266WebServer server**(**80**);**

constexpr uint8\_t RST\_PIN **=** D3**;**

constexpr uint8\_t SS\_PIN **=** D4**;**

MFRC522 rfid**(**SS\_PIN**,** RST\_PIN**);**

MFRC522**::**MIFARE\_Key key**;**

String tag**;**

String webPage **=**

**{**

"<!DOCTYPE html>"

"<html>"

"<head>"

" <meta http-equiv='Content-Type' content='text/html; charset=utf-8'>"

" <title>CONTROL DOOR</title>"

" <meta name='viewport' content='width=device-width, initial-scale=1'>"

" <style>"

" .b{width: 100px;height: 40px;font-size: 21px;color: #FFF;background-color:#4caf50;border-radius: 10px;}"

" .t{width: 100px;height: 40px;font-size: 21px;color: #FFF;background-color:#f44336;border-radius: 10px;}"

" </style>"

"</head>"

"<body>"

"<div style='width: 330px;height: auto;margin: 0 auto;margin-top: 70px'>"

"<h1 align='center'>CONTROL DOOR</h1>"

" <table align='center'>"

" <tr>"

" <td><a href='/Mo'><button class='b'>MỞ</button></a><td>"

" <tr>"

" </table>"

"</div>"

"</body>"

"</html>"

**};**

void TrangChu**()**

**{**

server**.**send**(**200**,** "text/html"**,** webPage**);**

**}**

void Mo**()**

**{** digitalWrite**(**D1**,** HIGH**);**

Serial**.**println**(**"MỞ"**);**

server**.**send**(**200**,** "text/html"**,** webPage**);**

delay**(**3000**);**

digitalWrite**(**D1**,** LOW**);**

Serial**.**println**(**"KHÓA"**);**

**}**

void setup**()** **{**

Serial**.**begin**(**9600**);**

SPI**.**begin**();**

rfid**.**PCD\_Init**();**

Serial**.**println**();**

Serial**.**println**();**

Serial**.**print**(**"Connecting to "**);**

WiFi**.**softAPConfig**(**apIP**,** apIP**,** IPAddress**(**255**,** 255**,** 255**,** 0**));**

WiFi**.**softAP**(**"Tổ 1"**,**"88888888"**);**

Serial**.**println**(**""**);**

pinMode**(**D8**,**OUTPUT**);**

pinMode**(**D1**,**OUTPUT**);**

digitalWrite**(**D1**,** LOW**);**

pinMode**(**D2**,**INPUT\_PULLUP**);**

server**.**on**(**"/"**,** TrangChu**);**

server**.**on**(**"/Mo"**,** Mo**);**

server**.**begin**();**

**}**

void loop**()** **{**

server**.**handleClient**();**

int buttonState **=** digitalRead**(**D2**);**

**if** **(**buttonState **==** LOW**)** **{**

Serial**.**println**(**"Mở"**);**

digitalWrite**(**D1**,** HIGH**);**

delay**(**3000**);**

**}**

**else**

**{**

digitalWrite**(**D1**,** LOW**);**

**}**

**if** **(** **!** rfid**.**PICC\_IsNewCardPresent**())**

**return;**

**if** **(**rfid**.**PICC\_ReadCardSerial**())** **{**

**for** **(**byte i **=** 0**;** i **<** 4**;** i**++)** **{**

tag **+=** rfid**.**uid**.**uidByte**[**i**];**

**}**

Serial**.**println**(**tag**);**

**if** **(**tag **==** "351013214"**)** **{**

Serial**.**println**(**"Thẻ đúng"**);**

digitalWrite**(**D8**,** HIGH**);**

delay**(**100**);**

digitalWrite**(**D8**,** LOW**);**

delay**(**100**);**

digitalWrite**(**D8**,** HIGH**);**

delay**(**100**);**

digitalWrite**(**D8**,** LOW**);**

delay**(**100**);**

digitalWrite**(**D8**,** HIGH**);**

delay**(**100**);**

digitalWrite**(**D8**,** LOW**);**

delay**(**100**);**

digitalWrite**(**D1**,** HIGH**);**

Serial**.**println**(**"MỞ"**);**

delay**(**3000**);**

digitalWrite**(**D1**,** LOW**);**

Serial**.**println**(**"KHÓA"**);**

**}** **else** **{**

Serial**.**println**(**"Thẻ sai"**);**

digitalWrite**(**D8**,** HIGH**);**

delay**(**1000**);**

digitalWrite**(**D8**,** LOW**);**

**}**

tag **=** ""**;**

**}**

rfid**.**PICC\_HaltA**();**

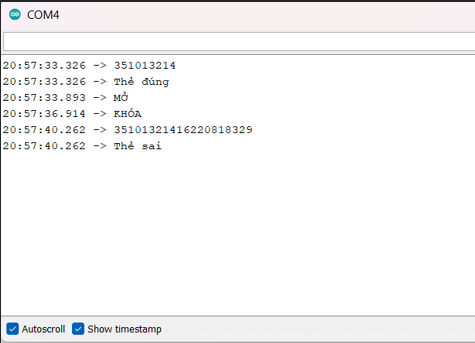
rfid**.**PCD\_StopCrypto1**();**

**}**

**3.4 Kết quả:**

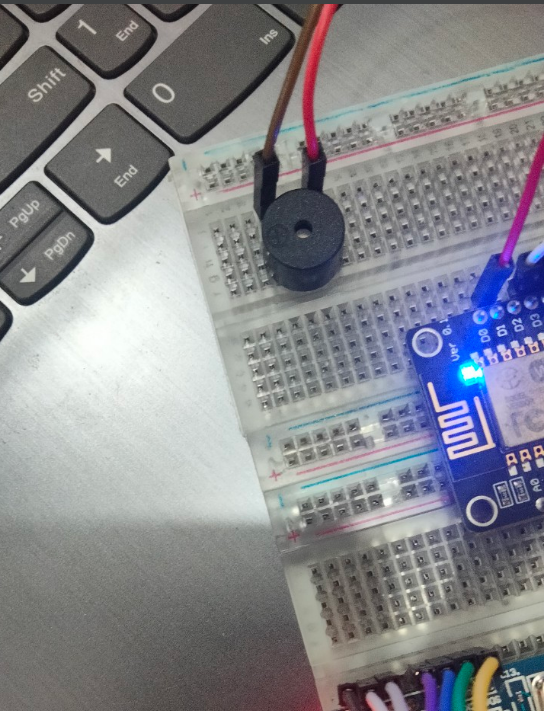
**Mở khóa bằng thẻ:**

Khi sử dụng thẻ để mở khóa cửa, nếu thẻ đúng thì trong phần Serial Monitor sẽ hiện thị số seri của thẻ nếu đúng sẽ hiện thị “Thẻ đúng” và mở khóa rồi khóa lại sau 3 giây. Còn khi thẻ sai cũng sẽ hiện thị seri thẻ rồi thông báo “Thẻ sai” và cửa sẽ không được mở.

****

Hình 3.12. Thông báo khi có tín hiệu quẹt thẻ

Khi thẻ được quét thì bộ phận còi chíp sẽ phát ra âm ra thanh kể cả thẻ sai lẫn thẻ đúng. Và âm thanh của thẻ sai, thẻ đúng sẽ khác nhau để người dùng có thể nhận biết.



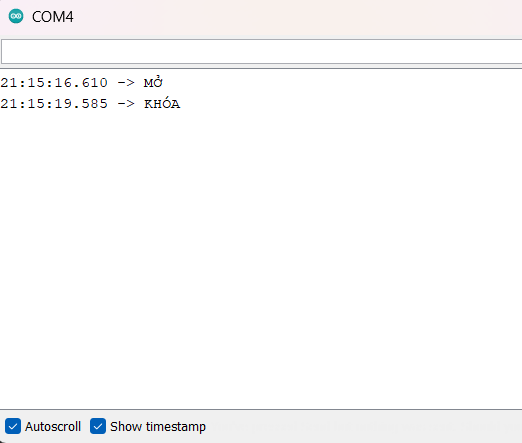
Hình 3.13. Hình ảnh còi báo

**Mở khóa qua WebServer:**

Nếu điều khiển trên WebServer qua điện thoại hoặc máy tính, khi ấn nút mở thì cửa sẽ được mở trong vòng 3 giây và sau đó tự động khóa lại. Và trong Serial Monitor cũng sẽ hiện thị thông báo

Graphical user interface, text, application, chat or text message

Description automatically generated

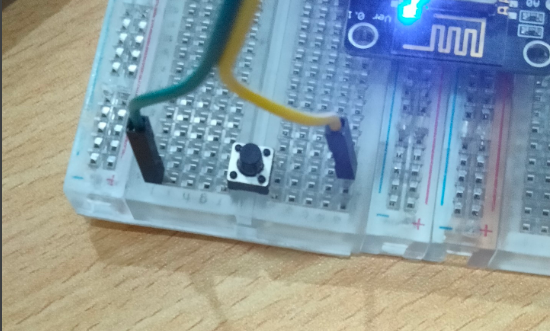


Hình 3.15 . Thông báo khi mở

Hình 3.14. Giao diện Web

**Sử dụng nút bấm**

Khi muốn mở cửa từ bên trong thì chỉ cần một thao tác đơn giản đó là bấm nút bên dưới thì khóa cửa sẽ được mở

****

Hình 3.16. Hình ảnh button

**KẾT LUẬN**

**Những mặt đã làm được**

* Mạch điện với các module nhỏ trên mạch được thiết kế, thi công hoàn chỉnh và đã được thử nghiệm nhiều lần và đã hoạt động ổn định
* Xây dựng được mô hình để điều khiển thiết bị bằng nút nhấn, mở cửa bằng thẻ RFID
* Hệ thống có thể điều khiển được thiết bị điện từ xa thông qua wifi

**Những hạn chế, tồn tại**

* Tốn nhiều dây dẫn để kết nối ngoại vi
* Chưa thể điều khiển thiết bị quá xa (phụ thuộc vào tín hiệu wifi của esp8266)
* Mạch điều khiển vẫn chưa có thể sử dụng được nguồn 220 V trực tiếp mà vẫn phài sử dụng nguồn riêng để cung cấp cho mạch
* Chưa hiển thị được tình trạng cửa trên WebServer

**Hướng phát triển đề tài**

* Khắc phục các hạn chế, tồn tại của hệ thống
* Tự phát triển để có thể ứng dụng thêm phần mềm Blynk để điều khiển thiết bị và xa hơn là phát triển app Android, IOS để điều khiển thiết bị qua wifi
* Tích hợp mạch công tắc cảm ứng với ESP 8266 để tạo thành mạch công tắc mở khóa thông minh có thể đồng bộ vô các hệ sinh thái Smart Home hiện có trên thị trường

**TÀI LIỆU THAM KHẢO**

[1] Ts.Hoàng Xuân Hiển và PGS.TS.Nguyễn Ái Việt (2021), *GIÁO TRÌNH MÔN HỌC PHÁT TRIỂN ỨNG DỤNG IoT*

[2] <https://blog.mecsu.vn/nodemcu-esp8266/>

[3] <https://blog.mecsu.vn/module-doc-rfid-rc522/>

[4] <https://miliohm.com/rc522-rfid-reader-with-nodemcu/>