TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA ĐIỆN – ĐIỆN TỬ**



**DƯƠNG HIỂN HOÀNG LONG**

**HỆ THỐNG CỬA TỰ ĐỘNG BẰNG THẺ TỪ HOẶC MẬT KHẨU**

**ĐỒ ÁN HỆ THỐNG NHÚNG**

**KT ĐIỀU KHIỂN VÀ TỰ ĐỘNG HÓA**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2021**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA ĐIỆN – ĐIỆN TỬ**



**DƯƠNG HIỂN HOÀNG LONG - 41801170**

**HỆ THỐNG CỬA TỰ ĐỘNG BẰNG THẺ TỪ HOẶC MẬT KHẨU**

**ĐỒ ÁN HỆ THỐNG NHÚNG**

**KT ĐIỀU KHIỂN VÀ TỰ ĐỘNG HÓA**

Người hướng dẫn

**TS LÊ ANH TUẤN**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2021**

**LỜI CẢM ƠN**

Em xin chân thành cảm ơn TS. Lê Anh Tuấn, khoa Điện – Điện tử trường Đại học Tôn Đức Thắng đã hướng dẫn nhiệt tình cho em trong suốt quá trình thực hiện đồ án. Em cũng xin cảm ơn các thầy cô, giảng viên Bộ môn Điều khiển và tự động hóa nói riêng và thầy cô trường Tôn Đức Thắng nói chung đã dạy dỗ, cung cấp kiến thức vè các môn đại cương, các kiến thức nền tảng, kiến thức chuyên ngành, thực hành và vận dụng kiến thức vào trong thực tế, tạo điều kiện giúp em có thể hoàn thành tốt đồ án này và trong suốt quá trình học tập.

*TP. Hồ Chí Minh, ngày tháng năm*

*Tác giả*

**CÔNG TRÌNH ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi và được sự hướng dẫn khoa học của **TS Lê Anh Tuấn**. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong luận văn còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.** Trường Đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày tháng năm 20*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

# **MỤC LỤC**

[CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI 1](#_Toc69851672)

[1.1 Giới thiệu đề tài 1](#_Toc69851673)

[1.2 Mục đích nghiên cứu 2](#_Toc69851674)

[1.3 Đối tượng nghiên cứu 2](#_Toc69851675)

[1.4 Phạm vi nghiên cứu 2](#_Toc69851676)

[1.5 Dự kiến kết quả 2](#_Toc69851677)

[CHƯƠNG 2. CƠ SỞ LÍ THUYẾT 3](#_Toc69851678)

[2.1 Mạch hạ áp từ 12V thành 5V 3](#_Toc69851679)

[2.2 Mạch hạ áp tự 5V thành 3.3V 4](#_Toc69851680)

[2.3 Nguyên tắc quét phím 5](#_Toc69851681)

[2.4 LCD 5](#_Toc69851682)

[2.5 PIC 16F877A 6](#_Toc69851683)

[2.6 RFID 12](#_Toc69851684)

[2.7 L298N 12](#_Toc69851685)

[CHƯƠNG 3. GIẢI THUẬT VÀ ĐIỀU KHIỂN 13](#_Toc69851686)

[3.1 Hoạt động của hệ thống 13](#_Toc69851687)

[3.2 Lưu đồ giải thuật trên vi điều khiển 14](#_Toc69851688)

[CHƯƠNG 4. THIẾT KẾ VÀ THI CÔNG 16](#_Toc69851689)

[4.1 Sơ đồ khối của hệ thống 16](#_Toc69851690)

[4.1.1 Khối 1 17](#_Toc69851691)

[4.1.2 Khối 2 18](#_Toc69851692)

[4.1.3 Khối 3 19](#_Toc69851693)

[4.1.4 Khối 4 21](#_Toc69851694)

[4.1.5 Khối 5 23](#_Toc69851695)

[4.1.6 Khối 6 24](#_Toc69851696)

[4.2 Sơ đồ nguyên lí tổng quát: 26](#_Toc69851697)

[4.3 Tiến trình thí nghiệm 26](#_Toc69851698)

[4.4 Kết quả thực nghiệm 27](#_Toc69851699)

[4.3 Kết luận thực nghiệm 27](#_Toc69851700)

[CHƯƠNG 5. KẾT LUẬN 27](#_Toc69851701)

[5.1 Ưu điểm 27](#_Toc69851702)

[5.2 Nhược điểm 27](#_Toc69851703)

[5.3 Hướng phát triển 27](#_Toc69851704)

[Phụ lục 1: Chương trình điều khiển bằng ngôn ngữ C 27](#_Toc69851705)

[Tài liệu tham khảo 46](#_Toc69851706)

# MỤC LỤC HÌNH ẢNH

[Hình 1.1: Cửa tự động 8](#_Toc69176179)

[Hình 2.1: L7805CV 10](#_Toc69176180)

[Hình 2.2: Sơ đồ mạch hạ áp 12V sang 5V 10](#_Toc69176181)

[Hình 2.3: LM1117T-3.3 11](#_Toc69176182)

[Hình 2.4: Sơ đồ mạch hạ áp 5V sang 3.3V 11](#_Toc69176183)

[Hình 2.5: L298N 18](#_Toc69176184)

[Hình 4.1: Sơ đồ nguyên lí bàn phím 22](#_Toc69176185)

[Hình 4.2: Mạch in bàn phím 22](file:///D:\PROJECT\2021_DAHTN_Duong_Hien_Hoang_Long_41801170.docx#_Toc69176186)

[Hình 4.3: Module thẻ từ RFID-RC522 23](#_Toc69176187)

[Hình 4.4: Cấu tạo PIC 16F877A 25](#_Toc69176188)

[Hình 4.5: Sơ đồ chân PIC 16F877A 25](file:///D:\PROJECT\2021_DAHTN_Duong_Hien_Hoang_Long_41801170.docx#_Toc69176189)

[Hình 4.6: Cấu tạo LCD 26](#_Toc69176190)

[Hình 4.7: Sơ đồ chân LCD 27](#_Toc69176191)

[Hình 4.8: Hình ảnh thực tế LCD 27](file:///D:\PROJECT\2021_DAHTN_Duong_Hien_Hoang_Long_41801170.docx#_Toc69176192)

[Hình 4.9: Sơ đồ nguyên lí mạch điều khiển động cơ dùng L298N 28](file:///D:\PROJECT\2021_DAHTN_Duong_Hien_Hoang_Long_41801170.docx#_Toc69176193)

[Hình 4.10: Sơ đồ mạch in mạch điều khiển động cơ dùng L298N 29](#_Toc69176194)

[Hình 4.11: Động cơ DC 30](file:///D:\PROJECT\2021_DAHTN_Duong_Hien_Hoang_Long_41801170.docx#_Toc69176195)

[Hình 4.12: Sơ đồ nguyên lí tổng quát 31](file:///D:\PROJECT\2021_DAHTN_Duong_Hien_Hoang_Long_41801170.docx#_Toc69176196)

# **MỤC LỤC BẢNG BIỂU**

[Bảng 2.1: Ý nghĩa chân LCD 14](#_Toc69176548)

[Bảng 2.2: Ý nghĩa chân PIC 16F877A 18](#_Toc69176549)

# **TỔNG QUAN VỀ ĐỀ TÀI**

## Giới thiệu đề tài

* Ngày nay, khi bước vào thời kì công nghệ 4.0 thì cửa tự động dường như đã là một vật dụng quá quen thuộc với mỗi người chúng ta. Việc ra vào cửa tự động cũng dần được đổi mới thêm nhờ các công cụ bổ sung như vân tay, password, …



Hình .: Cửa tự động

* Đề tài này sẽ tập trung vào nghiên cứu nguyên lí hoạt động và cách lập trình của một hệ thống cửa tự động dùng mật khẩu nhập từ bàn phím hoặc mở bằng thẻ từ. Là một hệ thống quen thuộc chúng ta thường gặp ở các tòa nhà, kí túc xá, công ti, …

## Mục đích nghiên cứu

* Dựa vào kiến thức về lập trình vi điều khiển đã học. Nghiên cứu cách chọn cổng và kết nối linh kiện với vi điều khiển.
* Tìm hiểu thêm kiến thức về các linh kiện khác như: LCD, L298N, modulde .RFID, cách hoạt động và nguyên lí cấu tạo của 1 bàn phím.
* Tìm hiểu thêm các thư viện cho các linh kiện ở trên để lập trình.

## Đối tượng nghiên cứu

* Vi điều khiển PIC 16F877A, cách lập trình, nguyên lí hoạt động.
* Cách sử dụng màn hình LCD.
* Chế tạo bàn phím, lập trình giap tiếp với vi điều khiển.
* Cách lập trình module RFID-RC522 để kiểm tra thẻ từ.
* Lập trình IC L298N để điều khiển động cơ DC.

## Phạm vi nghiên cứu

* Đề tài có thể áp dụng cho nhà máy, chung cư, kí túc xá, …Mật khẩu chỉ gồm có số chưa có các kí tự đặc biệt khác.

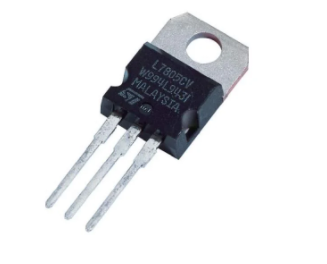
## Dự kiến kết quả

* Sau khi hoàn thành đồ án thì sẽ có mô hình cửa tự động hoạt động theo nguyên tắc: khi nhập dúng mật khẩu hoặc quét được đúng thẻ từ cho phép thì cửa sẽ mở và giữ nguyên trạng thái đó trong vòng 2 phút sau đó cửa sẽ tự động đóng lại cho đến khicó thao tác tiếp theo lặp lại.

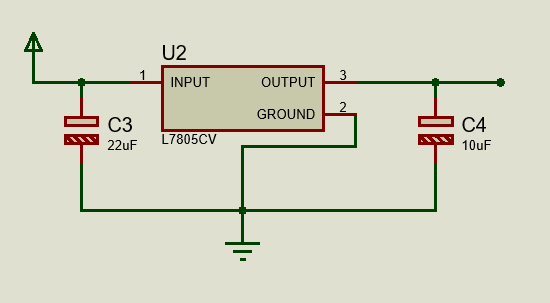
# **CƠ SỞ LÍ THUYẾT**

## Mạch hạ áp từ 12V thành 5V

* Sử dụng IC L7805: Cung cấp áp vào 12V, gắn thêm tụ 22uF vào ngõ vào IN, tụ 10uF vào ngõ ra OUT. Dùng dể cấp nguồn cho PIC 16f877A, IC L298N.



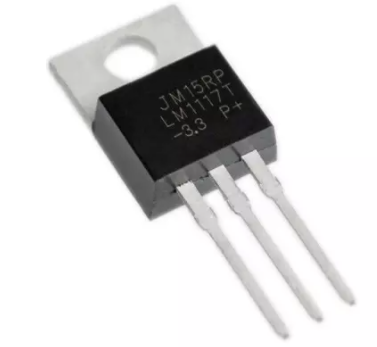
Hình .: L7805CV



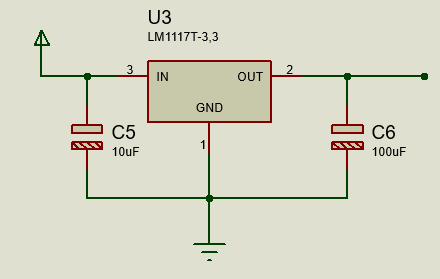
Hình .: Sơ đồ mạch hạ áp 12V sang 5V

## Mạch hạ áp tự 5V thành 3.3V

* Sử dụng IC LM1117-3.3: Cung cấp áp vào 5V, gắn thêm tụ 10uF vào ngõ vào IN, tụ 100uF vào ngõ ra OUT. Dùng cấp nguồn cho module RFID.



Hình .: LM1117T-3.3



Hình .: Sơ đồ mạch hạ áp 5V sang 3.3V

## Nguyên tắc quét phím

* Đầu tiên sẽ đưa tín hiệu HIGH ra tất cả các chân OUTPUT của keypad. Sau đó, lần lượt đưa tín hiệu LOW ra các chân OUTPUT của keypad cứ lặp lại như vậy. Khi có phím được nhấn thì mạch sẽ thông đưa tín hiệu LOW ra các chân INPUT của keypad. Qua việc lập trình và qui định của mỗi người thì khi mà một chân OUTPUT nào đó đang được đưa ra tín hiệu thấp nhưng có chân INPUT được nhấn thì tương ứng vi điều khiển sẽ hiểu là phím gì đang được nhấn.

## LCD

Ý nghĩa các chân LCD:

|  |  |  |  |
| --- | --- | --- | --- |
| STT |  | Tên | Chức năng |
| 1 |  | VSS | Chân nối đất cho LCD |
| 2 |  | VDD | Chân cấp nguồn cho LCD |
| 3 |  | VE | Điều chỉnh độ tương phản cho LCD |
| 4 |  | RS | Chân chọn thanh ghi. Có 2 mức logic “0” và “1”  0: sẽ nối với thanh ghi lệnh  1: sẽ nối với thanh ghi dữ liệu |
| 5 |  | R/W | Chân chọn chế độ đọc/ghi  0: LCD hoạt động ở chế độ ghi  1: LCD hoạt động ở chế độ đọc |
| 6 |  | E | Chân cho phép tín hiệu vào LCD |
| 7 |  | D0 | Data bus line 0 (LSB) |
| 8 |  | D1 | Data bus line 1 |
| 9 |  | D2 | Data bus line 2 |
| 10 |  | D3 | Data bus line 3 |
| 11 |  | D4 | Data bus line 4 |
| 12 |  | D5 | Data bus line 5 |
| 13 |  | D6 | Data bus line 6 |
| 14 |  | D7 | Data bus line 7 (MSB) |
| 15 |  | A | Chân nguồn “+” cho đèn nền |
| 16 |  | K | Chân nguồn “-” cho đèn nền |

Bảng .: Ý nghĩa chân LCD

## PIC 16F877A

* Chức năng các chân:

|  |  |  |
| --- | --- | --- |
| STT | Tên | Chức năng |
| 1 | MCLR/Vpp | MLCR: reset pic khi đầu vào mức thấp  Vpp: ngõ vào áp lập trình |
| 2 | RA0/AN0 | RA0: xuất/nhập số  AN0: ngõ vào tương tự |
| 3 | RA1/AN1 | RA1 : xuất/nhập số AN1 : ngõ vào tương tự |
| 4 | RA2/AN2/VREF-/CVREF | RA2 : xuất/nhập số AN2 : ngõ vào tương tự VREF-: ngõ vào điện áp chuẩn (thấp) của bộ A/D |
| 5 | RA3/AN3/VREF+ | RA3 : xuất/nhập số AN3 : ngõ vào tương tự VREF+ : ngõ vào điện áp chuẩn (cao) của bộ A/D |
| 6 | RA4/TOCKI/C1OUT | RA4 : xuất/nhập số TOCKI : ngõ vào xung clock bên ngoài cho timer0 C1 OUT : Ngõ ra bộ so sánh 1 |
| 7 | RA5/AN4//SS /C2OUT | RA5 : xuất/nhập số AN4 : ngõ vào tương tự 4 SS : ngõ vào chọn lựa SPI phụ C2 OUT : ngõ ra bộ so sánh 2 |
| 8 | RE0//RD/AN5 | RE0 : xuất nhập số RD : điều khiển việc đọc ở port nhánh song song AN5 : ngõ vào tương tự |
| 9 | RE1//WR/AN6 | RE1 : xuất/nhập số WR : điều khiển việc ghi ở port nhánh song song AN6 : ngõ vào tương tự |
| 10 | RE2//CS/AN7 | RE2 : xuất/nhập số CS : Chip lựa chọn sự điều khiển ở port nhánh song song AN7 : ngõ vào tương tự |
| 11 | VDD | Chân nguồn của PIC |
| 12 | VSS | Chân nối đất |
| 13 | OSC1/CLKI | Ngõ vào dao động thạch anh hoặc xung clock bên ngoài. OSC1 : ngõ vào dao động thạch anh hoặc xung clock bên ngoài. Ngõ vào Schmit trigger khi được cấu tạo ở chế độ RC ; một cách khác của CMOS. CLKI : ngõ vào nguồn xung bên ngoài. Luôn được kết hợp với chức năng OSC1. |
| 14 | OSC2/CLKO | Ngõ vào dao động thạch anh hoặc xung clock OSC2 : Ngõ ra dao động thạch anh. Kết nối đến thạch anh hoặc bộ cộng hưởng CLKO : ở chế độ RC, ngõ ra của OSC2, bằng tần số của OSC1 và chỉ ra tốc độ của chu kỳ lệnh |
| 15 | RC0/T1 OCO/T1CKI | RC0 : xuất/nhập số T1OCO : ngõ vào bộ dao động Timer 1 T1CKI : ngõ vào xung clock bên ngoài Timer 1 |
| 16 | RC1/T1OSI/CCP2 | RC1 : xuất/nhập số T1OSI : ngõ vào bộ dao động Timer 1 CCP2 : ngõ vào Capture 2, ngõ ra compare 2, ngõ ra PWM2 |
| 17 | RC2/CCP1 | RC2 : xuất/nhập số CCP1 : ngõ vào Capture 1, ngõ ra compare 1, ngõ ra PWM1 |
| 18 | RC3/SCK/SCL | RC3 : xuất/nhập số SCK : ngõ vào xung clock nối tiếp đồng bộ/ngõ ra của chế độ SPI SCL : ngõ vào xung clock nối tiếp đồng bộ/ ngõ ra của chế độ I2C |
| 19 | RD0/PSP0 | RD0 : xuất/nhập số PSP0 : dữ liệu port nhánh song song |
| 20 | RD1/PSP1 | RD1 : xuất/nhập số PSP1 : dữ liệu port nhánh song song |
| 21 | RD2/PSP2 | RD2 : xuất/nhập số PSP2 : dữ liệu port nhánh song song |
| 22 | RD3/PSP3 | RD3: xuất/nhập số PSP3 : dữ liệu port nhánh song song |
| 23 | RC4/SDI/SDA | RC4 : xuất/nhập số SDI : dữ liệu vào SPI SDA : xuất/nhập dữ liệu vào I2C |
| 24 | RC5/SDO | RC5 : xuất/nhập số SDO : dữ liệu ra SPI |
| 25 | RC6/TX/CK | RC6 : xuất/nhập số TX : truyền bất đồng bộ USART CK : xung đồng bộ USART |
| 26 | RC7/RX/DT | RC7 : xuất/nhập số RX : nhận bất đồng USART DT : dữ liệu đồng bộ USART |
| 27 | RD4/PSP | RD4: xuất/nhập số PSP4 : dữ liệu port nhánh song song |
| 28 | RD5/PSP5 | RD5: xuất/nhập số PSP5 : dữ liệu port nhánh song song |
| 29 | RD6/PSP6 | RD6: xuất/nhập số PSP6 : dữ liệu port nhánh song song |
| 30 | RD7/PSP7 | RD7: xuất/nhập số PSP7 : dữ liệu port nhánh song song |
| 31 | VSS | Chân nối đất |
| 32 | VDD | Chân nguồn của PIC |
| 33 | RB0/INT | RB0 : xuất/nhập số INT : ngắt ngoài |
| 34 | RB1 | xuất/nhập số |
| 35 | RB2 | xuất/nhập số |
| 36 | RB3 | RB3 : xuất/nhập số Chân cho phép lập trình điện áp thấp ICPS |
| 37 | RB4 | xuất/nhập số Ngắt PortB |
| 38 | RB5 | xuất/nhập số Ngắt PortB |
| 39 | RB6/PGC | RB6 : xuất/nhập số PGC : mạch vi sai và xung clock lập trình ICSP Ngắt PortB |
| 40 | RB7/PGD | RB7 : xuất/nhập số PGD : mạch vi sai và dữ liệu lập trình ICSP Ngắt PortB |

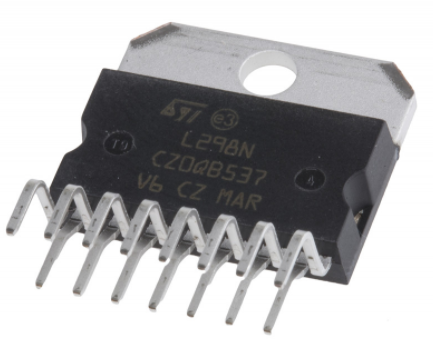
Bảng .: Ý nghĩa chân PIC 16F877A

## RFID

* RFID là chữ viết tắt của từ Radio Frequency Identification. Dịch ra tiếng Việt có nghĩa là công nghệ nhận dạng đối tượng bằng sóng vô tuyến. Hiểu một cách đơn giản thì hai thiết bị phát ra sóng điện từ, có cùng một tần số khi gặp nhau có thể nhận dạng được nhau. Tần số 125Khz và 900Khz là hai tần số thường được sử dụng trong RFID mà chúng ta có thể gặp.
* Thiết bị RFID được cấu tạo từ 2 thành phần chính là: thiết bị phát mã đã được gắn chip và thiết bị để đọc. Thiết bị phát mã sẽ được gắn vào vật cần được nhận dạng còn thiết bị đọc sẽ được gắn anten giúp thu phát sóng điện tử. Thiết bị RFID khác nhau sẽ có mã số khác nhau và không bị trùng lặp. Khi hai thiết bị gặp nhau, tần số trùng khớp thì sẽ nhận dạng được nhau.

## L298N

* Khi cấp mức logic “1” vào chân ENA và ENB thì sẽ cho phép tín hiệu vào các chân IN1, IN2, IN3, IN4. Giá trị logic tại chân IN1 cũng chính là giá trị logic của chân OUT1, tương tự cho các chân IN và OUT còn lại. Chúng ta cũng dựa vào nguyên lí đó mà điều khiển chiều quay của động cơ DC.



Hình .: L298N

# **GIẢI THUẬT VÀ ĐIỀU KHIỂN**

## Hoạt động của hệ thống

* + Password mặc định “123456”.
  + Mở khóa: nhập password hoặc quét thẻ từ 🡪 Nhấn “ENTER” thì cửa sẽ mở sau 2 phút rồi đóng lại.
  + Trong quá trình nhập pass muốn nhập lại thì nhấn “CLEAR”.
  + Để chạy lại chương trình thì nhấn nút “RESET”.

## Lưu đồ giải thuật trên vi điều khiển

Quét thẻ

Quét phím

Nhập pass

So sánh pass, UID

Hiển thị LCD

Động cơ chạy

Reset

* + Khi cấp nguồn cho hệ thống hoạt động thì password được cài đặt sẵn sẽ là “123456”. Khi người dùng nhập password từ bàn phím thì password sẽ được hiển thị lên LCD dưới dạng số sau đó khoảng 300ms sẽ tự động chuyển thành dấu “\*” nhằm giúp cho người dùng có thể thấy được con số mình đã nhập nhưng vẫn đảm bảo tính bảo mật tránh để lộ password. Để hiển thị kí tự lên LCD ta dùng lệnh:
    - lcd\_putc(“nội dung cần hiển thị hi không dấu”).
  + Bàn phím sẽ gồm 12 nút nhấn từ 0 đến 9 và 2 nút “ENTER” và “CLEAR”. Phím “EnTER” mục đích dùng để xác nhận password mà người dùng đã nhập vào có đúng với mật khẩu đã đặt hay không, phím “CLEAR” có tác dụng xóa password đã nhập trước đó khi phát hiện đã nhập sai để có thể tiến hành nhập lại password.
  + Khi người dùng nhập đúng password là “123456” và nhấn “ENTER” thì màn hình LCD sẽ báo là “CORRECT PASS” và sao đó vi điều khiển sẽ gửi lệnh đến module L298 để điều khiển động cơ DC, động cơ sẽ chạy để mở cửa và giữ nguyên trạng thái mở trong 2 phút sau đó sẽ tự động đóng lại cho đến khi password được nhập lại đúng. Nếu password sai thì trên LCD sẽ báo “WRONG PASS” khi đó màn hình sẽ tự trở lại lúc nhập password cho người sử dụng.
  + Cũng tương tự như trên thì nếu dùng thẻ từ thì nếu thể từ có UID đúng với UID cho phép đã được lập trình từ trước thì màn hình LCD sẽ báo là “CORRECT UID” và sao đó vi điều khiển sẽ gửi lệnh đến module L298 để điều khiển động cơ DC, động cơ sẽ chạy để mở cửa và giữ nguyên trạng thái mở trong 2 phút sau đó sẽ tự động đóng lại cho đến khi thẻ từ được quét lên lại. Nếu UID sai thì trên LCD sẽ báo “WRONG UID” .
  + Khi nhấn nút RESET thì chương trình trong vi điều khển sẽ chạy lại từ đầu.

# **THIẾT KẾ VÀ THI CÔNG**

## Sơ đồ khối của hệ thống

Khối 3

Khối 6

Khối 5

Khối 4

Khối 2

Khối 1

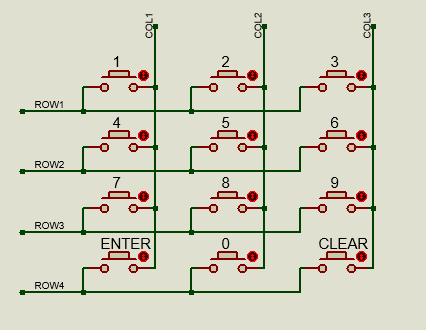
### Khối 1

Chức năng: dùng để nhập mật khẩu

Linh kiện sử dụng: nút nhấn

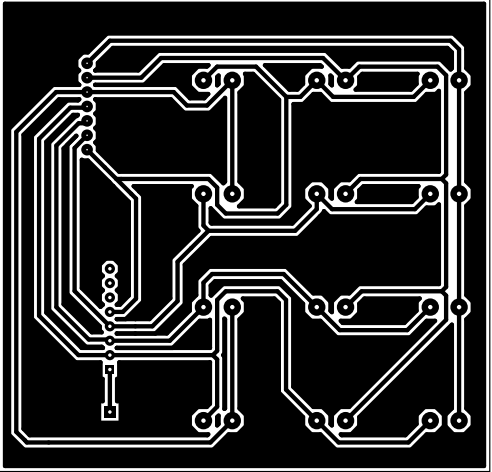
Thông số kĩ thuật: đường kính D=0.5mm, chiều cao H=4mm, chịu dòng 2A

Sơ đồ nguyên lí:



Hình .: Sơ đồ nguyên lí bàn phím

Sơ đồ mạch in



Hình 4.2: Mạch in bàn phím

### Khối 2

Chức năng: dùng để quét thẻ từ

Linh kiện sử dụng: RC-522

Thông số kĩ thuật: Nguồn: 3.3VDC, 13 – 26mA

Dòng ở chế độ chờ: 10-13mA

Dòng ở chế độ nghỉ: <80uA

Tần số sóng mang: 13.56MHz

Khoảng cách hoạt động: 0～60mm (mifare1 card)

Giao tiếp: SPI

Tốc độ truyền dữ liệu: tối đa 10Mbit/s

Nhiệt độ hoạt động: -20 đến 80 ° C

Tốc độ cao SPI: 10Mbit /

Hỗ trợ: ISO / IEC 14443A /MIFAR



Hình .: Module thẻ từ RFID-RC522

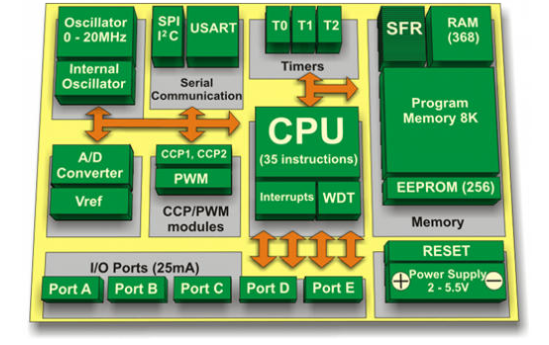
### Khối 3

Chức năng: xử lí dữ liệu để điều khiển chương trình

Linh kiện sử dụng: 16F877A

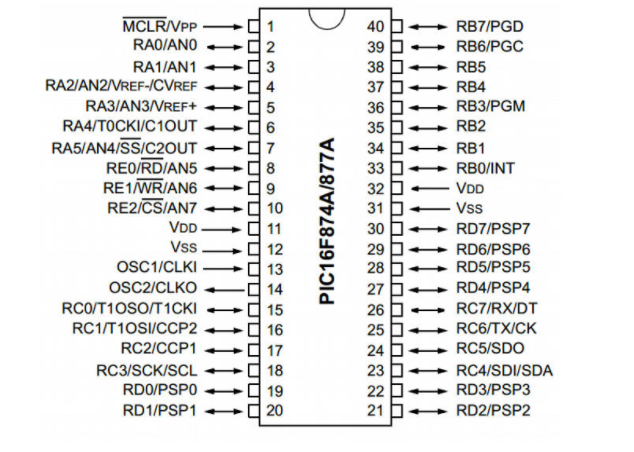
Thông số kĩ thuật:

* Cấu trúc tổng quát:
* 8K x 14 bits/word Flash ROM.
* 368 x 8 Bytes RAM.
* 256 x 8 Bytes EEPROM.
* 5 Port xuất/nhập (A, B, C, D, E) tương ứng 33 chân ra.
* 2 Bộ định thời 8 bit Timer 0 và Timer 2.
* 1 Bộ định thời 16 bit Timer 1, có thể hoạt động ở chế độ tiết kiệm năng lượng (SLEEP MODE) với nguồn xung clock ngoài.
* 2 Bộ Capture/ Compare/ PWM.(Bắt Giữ/ So Sánh/ Điều Biến Xung)
* 1 Bộ biến đổi Analog to Digital 10 bit, 8 ngõ vào.
* 2 Bộ so sánh tương tự (Compartor).
* 1 Bộ định thời giám sát (Watch Dog Timer).
* 1 Cổng giao tiếp song song 8 bit.
* 1 Port nối tiếp.
* 15 Nguồn ngắt (Interrupt).
* Chế độ tiết kiệm năng lượng (Sleep Mode).
* Nạp trương trình bằng cổng nối tiếp ( ICSP™ )(In-Circuit Serial Programming™ -)
* Tập lệnh gồm 35 lệnh có độ dài 14 bit.
* Tần số hoạt động tối đa 20 MHz.



Hình .: Cấu tạo PIC 16F877A

Sơ đồ chân:



Hình 4.5: Sơ đồ chân PIC 16F877A

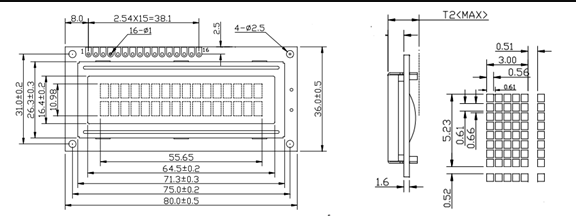
### Khối 4

Chức năng: hiển thị thông tin cho người sử dụng

Linh kiện sử dụng: màn hình LCD

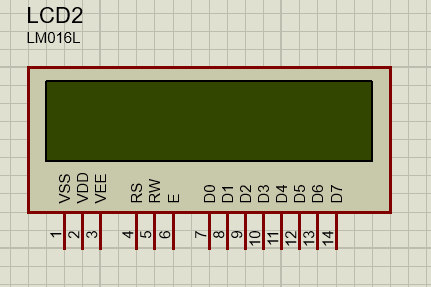
Thông số kĩ thuật:

* Điều kiện hoạt động:
* Điện áp MAX : 7V.
* Điện áp MIN : -0,3V.
* Hoạt động ổn định : 2.7-5.5V.
* Điện áp ra mức cao : > 2.4V.
* Điện áp ra mức thấp : <0.4V.
* Dòng điện cấp nguồn : 350uA - 600uA.
* Nhiệt độ hoạt động : -30 - 75 độ C
* Sơ đồ cấu tạo:

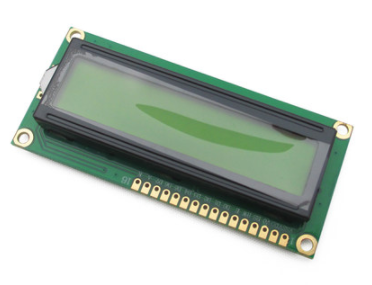


Hình .: Cấu tạo LCD

* Sơ đồ chân:



Hình .: Sơ đồ chân LCD



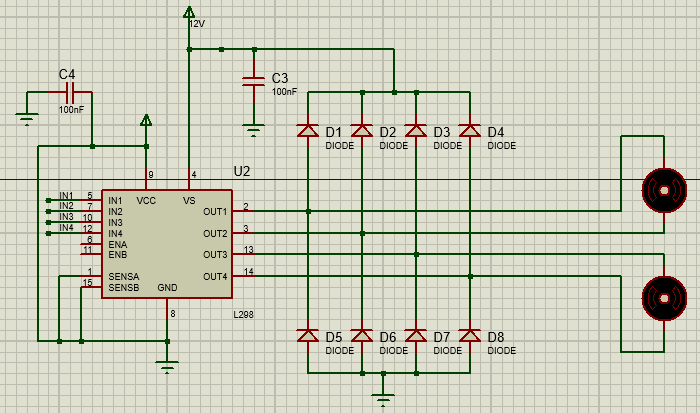
Hình 4.8: Hình ảnh thực tế LCD

### Khối 5

Chức năng: điều khiển động cơ DC

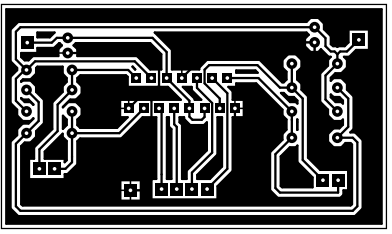
Linh kiện sử dụng: L298N

Thông số kĩ thuật:

* Điều kiện hoạt động:
  + Số chân: 15
  + Số kênh: 2
  + Nhiệt độ hoạt động MAX: 1300C
  + Nhiệt độ hoạt động MIN: -250C
  + Dòng ra MAX: 3A
* Sơ đồ ngyên lí:

Hình 4.9: Sơ đồ nguyên lí mạch điều khiển động cơ dùng L298N

* Sơ đồ mạch in:



Hình .: Sơ đồ mạch in mạch điều khiển động cơ dùng L298N

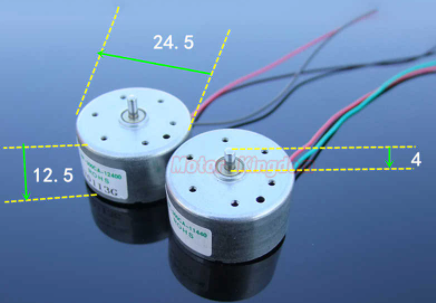
### Khối 6

Chức năng: truyền động đến cửa tự động giúp đóng mở cửa

Linh kiện sử dụng: động cơ DC

Thông số kĩ thuật:

* Điều kiện hoạt động:
* Đường kính ngoài: 24.5mm
* Chiều cao động cơ: 12.5mm
* Chiều dài trục: khoảng 4mm,
* Đường kính trục: 2mm
* Điện áp DC để hoạt động: 4 V 🡪 5 V



Hình 4.11: Động cơ DC

* Tìm hiểu về động cơ DC:
  + Khái niệm: Động cơ điện một chiều là [động cơ điện](http://vi.wikipedia.org/wiki/%C4%90%E1%BB%99ng_c%C6%A1_%C4%91i%E1%BB%87n) hoạt động với dòng điện một chiều.
  + Cấu tạo: [Stator](http://vi.wikipedia.org/wiki/Stator) của động cơ điện 1 chiều thường là 1 hay nhiều cặp [nam châm vĩnh cửu](http://vi.wikipedia.org/wiki/Nam_ch%C3%A2m_v%C4%A9nh_c%E1%BB%ADu), hay [nam châm điện](http://vi.wikipedia.org/wiki/Nam_ch%C3%A2m_%C4%91i%E1%BB%87n), [rotor](http://vi.wikipedia.org/wiki/Rotor) có các cuộn dây quấn và được nối với nguồn điện một chiều, 1 phần quan trọng khác của động cơ điện 1 chiều là bộ phận chỉnh lưu, nó có nhiệm vụ là đổi chiều dòng điện trong khi chuyển động quay của rotor là liên tục. Thông thường bộ phận này gồm có một bộ cổ góp và một bộ chổi than tiếp xúc với cổ góp.

## Sơ đồ nguyên lí tổng quát:

Hình 4.12: Sơ đồ nguyên lí tổng quát

## Tiến trình thí nghiệm

Bước 1: Cấp điện và đo nguồn

Bước 2: Kiểm tra linh kiện, dây dẫn

Bước 3: Kiểm tra xem bàn phím có quét phím đúng hay không

Bước 4: Kiểm tra LCD, RFID, L298N

Bước 5: Kiểm tra xem mạch có chạy đúng như đã lập trình

## Kết quả thực nghiệm

Sau khi cấp nguồn 12V thì đo nguồn ra được 5V

Màn hình LCD hiển thị đúng nút được bấm

RFID có nhận thẻ

L298N điều khiển được động cơ chạy thuận nghịch đúng như lập trình

## 4.3 Kết luận thực nghiệm

Kết quả sau khi chạy mạch đúng với kết quả dự kiến ban đầu ở chương 1

# **KẾT LUẬN**

## 5.1 Ưu điểm

Hệ thống đơn giản dễ sử dụng, ai cũng có thể sử dụng.

Linh kiện đơn giản, dễ tìm.

## 5.2 Nhược điểm

Password chỉ mới có kí tự số, chưa có kí tự chữ hay kí tự đặc biệt khác

Mô hình chưa có khả năng áp dụng rộng trong cuộc sống

## 5.3 Hướng phát triển

Có thể xây dựng thêm bàn phím có các kí tự chữ, kí tự đặc biệt để tăng tính bảo mật

Có thể thêm nhiều cách xác thực khác như: vân tay, nhận diện khuôn mặt

Có thể dùng để điều khiển nhiều loại cửa khác

Phát triển thêm chế độ đổi mật khẩu

# **Phụ lục 1: Chương trình điều khiển bằng ngôn ngữ C**

#include <main.h>

#include <lcd.c>

#include <Built\_in.h>

char A[12];

int i=0, j=0;

char THE[6] ={0x43,0xA8,0xF2,0x02};

char MOC[6] ={0xAA,0x61,0xDF,0x80};

//MOTOR

#define IN1 PIN\_C6

#define IN2 PIN\_C5

#define IN3 PIN\_C4

#define IN4 PIN\_D3

//RFID

int1 quet(char Data[],char UID[] ) //Chuong trinh so sanh the, moc khoa co hop le khong

{

for (int i = 0; i < 4; i++)

{

if(UID[i]== Data[i] )

{

j=1;

}

else

{

j=0;

break;

}

}

return j;

}

//LCD

#define LCD\_ENABLE\_PIN PIN\_D0

#define LCD\_RS\_PIN PIN\_D1

#define LCD\_RW\_PIN PIN\_D2

#define LCD\_DATA4 PIN\_D4

#define LCD\_DATA5 PIN\_D5

#define LCD\_DATA6 PIN\_D6

#define LCD\_DATA7 PIN\_D7

void LCD\_Enable(void)

{

output\_high(LCD\_ENABLE\_PIN);

delay\_us(3);

output\_low(LCD\_ENABLE\_PIN);

delay\_us(50);

}

void LCD\_Send4Bit(unsigned char Data) //Chuyen 4 bit len chan d4 -> d6

{

output\_bit(LCD\_DATA4,Data&&0x01);

output\_bit(LCD\_DATA5,(Data>>1)&&1);

output\_bit(LCD\_DATA6,(Data>>2)&&1);

output\_bit(LCD\_DATA7,(Data>>3)&&1);

}

void LCD\_SendCommand(unsigned char command) //chuyen ma lenh len chan data LCD

{

output\_low(LCD\_RS\_PIN);

LCD\_Send4Bit(command>>4);

LCD\_Enable();

LCD\_Send4Bit(command);

LCD\_Enable();

}

void LCD\_Clear() //xoa man hinh LCD

{

LCD\_SendCommand(0x01);

delay\_ms(10);

}

// KEYPAD

#define COL1 pin\_B1

#define COL2 pin\_B2

#define COL3 pin\_B3

#define ROW1 pin\_B4

#define ROW2 pin\_B5

#define ROW3 pin\_B6

#define ROW4 pin\_B7

void key\_init(void) //kich hoat keypad

{

output\_drive(COL1);

output\_drive(COL2);

output\_drive(COL3);

output\_float(ROW1);

output\_float(ROW2);

output\_float(ROW3);

output\_float(ROW4);

}

void quet\_phim() //tra ve gia tri nut

{

output\_b(0b11111101);

if(!input(ROW1)) //nut 1

{

delay\_ms(5);

while(!input(ROW1))

{

i++;

A[i]= 1;

lcd\_putc("1");

delay\_ms(300);

lcd\_gotoxy(6+i,1);

lcd\_putc("\*");

while(!input(ROW1)) {};

}

}

if(!input(ROW2)) //nut 4

{

delay\_ms(5);

while(!input(ROW2))

{

i++;

A[i]= 4;

lcd\_putc("4");

delay\_ms(300);

lcd\_gotoxy(6+i,1);

lcd\_putc("\*");

while(!input(ROW2)) {};

}

}

if(!input(ROW3)) //nut 7

{

delay\_ms(5);

while(!input(ROW3))

{

A[i]= 7;

lcd\_putc("7");

delay\_ms(300);

lcd\_gotoxy(6+i,1);

lcd\_putc("\*");

while(!input(ROW3)) {};

}

}

if(!input(ROW4)) //nut \*

{

delay\_ms(5);

while(!input(ROW4))

{

i++;

A[i]= '\*';

while(!input(ROW4)) {};

}

}

output\_b(0b11111011);

if(!input(ROW1)) //nut 2

{

delay\_ms(5);

while(!input(ROW1))

{

i++;

A[i]= 2;

lcd\_putc("2");

delay\_ms(300);

lcd\_gotoxy(6+i,1);

lcd\_putc("\*");

while(!input(ROW1)) {};

}

}

if(!input(ROW2)) //nut 5

{

delay\_ms(5);

while(!input(ROW2))

{

i++;

A[i]= 5;

lcd\_putc("5");

delay\_ms(300);

lcd\_gotoxy(6+i,1);

lcd\_putc("\*");

while(!input(ROW2)) {};

}

}

if(!input(ROW3)) //nut 8

{

delay\_ms(5);

while(!input(ROW3))

{

i++;

A[i]= 8;

lcd\_putc("8");

delay\_ms(300);

lcd\_gotoxy(6+i,1);

lcd\_putc("\*");

while(!input(ROW3)) {};

}

}

if(!input(ROW4)) //nut 0

{

delay\_ms(5);

while(!input(ROW4))

{

i++;

A[i]= 0;

lcd\_putc("0");

delay\_ms(300);

lcd\_gotoxy(6+i,1);

lcd\_putc("\*");

while(!input(ROW4)) {};

}

}

output\_b(0b11110111);

if(!input(ROW1)) //nut 3

{

delay\_ms(5);

while(!input(ROW1))

{

i++;

A[i]= 3;

lcd\_putc("3");

delay\_ms(300);

lcd\_gotoxy(6+i,1);

lcd\_putc("\*");

while(!input(ROW1)) {};

}

}

if(!input(ROW2)) //nut 6

{

delay\_ms(5);

while(!input(ROW2))

{

i++;

A[i]= 6;

lcd\_putc("6");

delay\_ms(300);

lcd\_gotoxy(6+i,1);

lcd\_putc("\*");

while(!input(ROW2)) {};

}

}

if(!input(ROW3)) //nut 9

{

delay\_ms(5);

while(!input(ROW3))

{

i++;

A[i]= 9;

lcd\_putc("9");

delay\_ms(300);

lcd\_gotoxy(6+i,1);

lcd\_putc("\*");

while(!input(ROW3)) {};

}

}

if(!input(ROW4)) //nut #

{

delay\_ms(5);

while(!input(ROW4))

{

i++;

A[i]= '#';

while(!input(ROW4)) {};

}

}

}

void motor() //Chuong trinh dieu khien dong co

{

output\_high(IN1);

output\_low(IN2);

output\_low(IN4);

output\_high(IN3);

delay\_ms(3000);

output\_low(IN1);

output\_low(IN2);

output\_low(IN3);

output\_low(IN4);

delay\_ms(3000);

output\_low(IN1);

output\_high(IN2);

output\_high(IN4);

output\_low(IN3);

delay\_ms(3000);

output\_low(IN1);

output\_low(IN2);

output\_low(IN3);

output\_low(IN4);

for(i=1;i<8;i++)

{

A[i]=0;

}

}

void main()

{

char UID[6];

unsigned int TagType;

lcd\_init ();

MFRC522\_Init();

set\_tris\_c(0x00);

set\_tris\_d(0x00);

lcd\_putc("PASS:");

lcd\_gotoxy(7,1);

output\_c(0x00);

while(TRUE)

{

quet\_phim();

if(A[1]==1 && A[2]==2 && A[3]==3 && A[4]==4 && A[5]==5 && A[6]==6 && A[7]=='\*')

{

lcd\_gotoxy(1,2);

lcd\_putc("CORRECT PASS");

motor();

goto reset;

}

if(A[1]!=1 && A[i]=='\*')

{

lcd\_gotoxy(1,2);

lcd\_putc("WRONG PASS");

delay\_ms(2000);

reset\_cpu();

}

if(A[2]!=2 && A[i]=='\*')

{

lcd\_gotoxy(1,2);

lcd\_putc("WRONG PASS");

delay\_ms(2000);

reset\_cpu();

}

if(A[3]!=3 && A[i]=='\*')

{

lcd\_gotoxy(1,2);

lcd\_putc("WRONG PASS");

delay\_ms(2000);

reset\_cpu();

}

if(A[4]!=4 && A[i]=='\*')

{

lcd\_gotoxy(1,2);

lcd\_putc("WRONG PASS");

delay\_ms(2000);

reset\_cpu();

}

if(A[5]!=5 && A[i]=='\*')

{

lcd\_gotoxy(1,2);

lcd\_putc("WRONG PASS");

delay\_ms(2000);

reset\_cpu();

}

if(A[6]!=6 && A[i]=='\*')

{

lcd\_gotoxy(1,2);

lcd\_putc("WRONG PASS");

delay\_ms(2000);

reset\_cpu();

}

if(A[i]=='#') reset\_cpu();

if(MFRC522\_isCard (&TagType))

{

if (MFRC522\_ReadCardSerial (&UID))

{

if(quet(THE,UID)||quet(MOC,UID))

{

LCD\_Clear();

lcd\_gotoxy(1,2);

lcd\_putc("CORRECT UID");

motor();

goto reset;

}

else

{

lcd\_gotoxy(2,1);

lcd\_putc("WRONG");

}

}

}

}

reset:

{

LCD\_Clear();

lcd\_putc("PASS:");

i=0;

lcd\_gotoxy(7,1);

while(TRUE)

{

quet\_phim();

if(A[1]==1 && A[2]==2 && A[3]==3 && A[4]==4 && A[5]==5 && A[6]==6 && A[7]=='\*')

{

LCD\_Clear();

lcd\_gotoxy(1,2);

lcd\_putc("CORRECT PASS");

motor();

goto reset;

}

if(A[1]!=1 && A[i]=='\*')

{

lcd\_gotoxy(1,2);

lcd\_putc("WRONG PASS");

delay\_ms(2000);

reset\_cpu();

}

if(A[2]!=2 && A[i]=='\*')

{

lcd\_gotoxy(1,2);

lcd\_putc("WRONG PASS");

delay\_ms(2000);

reset\_cpu();

}

if(A[3]!=3 && A[i]=='\*')

{

lcd\_gotoxy(1,2);

lcd\_putc("WRONG PASS");

delay\_ms(2000);

reset\_cpu();

}

if(A[4]!=4 && A[i]=='\*')

{

lcd\_gotoxy(1,2);

lcd\_putc("WRONG PASS");

delay\_ms(2000);

reset\_cpu();

}

if(A[5]!=5 && A[i]=='\*')

{

lcd\_gotoxy(1,2);

lcd\_putc("WRONG PASS");

delay\_ms(2000);

reset\_cpu();

}

if(A[6]!=6 && A[i]=='\*')

{

lcd\_gotoxy(1,2);

lcd\_putc("WRONG PASS");

delay\_ms(2000);

reset\_cpu();

}

if(A[i]=='#') reset\_cpu();

if(MFRC522\_isCard (&TagType))

{

if (MFRC522\_ReadCardSerial (&UID))

{

if(quet(THE,UID)||quet(MOC,UID))

{

lcd\_gotoxy(1,2);

lcd\_putc("CORRECT UID");

motor();

goto reset;

}

else

{

lcd\_gotoxy(1,2);

lcd\_putc("WRONG");

}

}

}

}

}

}

# **Tài liệu tham khảo**

**Tiếng Việt:**

[1] Trinh Cong Tran, Kĩ thuật vi điều khiển 1: Slide bài giảng

[2] RFID là gì?, <https://azcomvn.com/tin-tuc/tin-tuc-ky-thuat/rfid-nguyen-ly-hoat-dong-va-ung-dung/>

[3] Nguyễn Trường Thịnh, Nguyễn Tấn Nỏ; [Giáo trình vi điều khiển PIC16F và ngôn ngữ lập trình HI-TECH C](http://idiscovery.tdtu.edu.vn/primo_library/libweb/action/display.do?tabs=detailsTab&ct=display&fn=search&doc=dedupmrg95702070&indx=7&recIds=dedupmrg95702070&recIdxs=6&elementId=6&renderMode=poppedOut&displayMode=full&frbrVersion=&frbg=&&vl(12231591UI0)=any&dscnt=0&vl(1UIStartWith0)=contains&scp.scps=scope%3A%28TDTS%29%2Cscope%3A%28SFX%29%2Cscope%3A%28TDT%29%2Cscope%3A%28SEN%29%2Cprimo_central_multiple_fe&tb=t&vid=tdt&mode=Basic&srt=rank&tab=default_tab&vl(119760700UI1)=all_items&dum=true&vl(freeText0)=402065&dstmp=1618261634918)

[4] Giao tiếp với vi điều khiển PIC16F877A, <https://vidieukhien.xyz/2014/10/05/ung-dung-rfid-trong-dieu-khien-voi-pic16f877a-ccs-c/>

[5] Giao tiếp LCD 16x2, <https://vidieukhien.xyz/2014/04/01/tutgiao-tiep-voi-lcd-16x2-huong-da%CC%83n-la%CC%A3p-trinh-pic-su%CC%89-du%CC%A3ng-ccs-bai-4/>

[6] **L298 gồm các chân,** <http://arduino.vn/bai-viet/893-cach-dung-module-dieu-khien-dong-co-l298n-cau-h-de-dieu-khien-dong-co-dc>

**Tiếng Anh:**

[7] Martin Bates, [Interfacing PIC microcontrollers : embedded design by interactive simulation](http://idiscovery.tdtu.edu.vn/primo_library/libweb/action/display.do?tabs=detailsTab&ct=display&fn=search&doc=dedupmrg303464885&indx=5&recIds=dedupmrg303464885&recIdxs=4&elementId=4&renderMode=poppedOut&displayMode=full&frbrVersion=&frbg=&&vl(12231591UI0)=any&dscnt=0&vl(1UIStartWith0)=contains&scp.scps=scope%3A%28TDTS%29%2Cscope%3A%28SFX%29%2Cscope%3A%28TDT%29%2Cscope%3A%28SEN%29%2Cprimo_central_multiple_fe&tb=t&vid=tdt&mode=Basic&srt=rank&tab=default_tab&vl(119760700UI1)=all_items&dum=true&vl(freeText0)=402065&dstmp=1618287326698), Amsterdam : Elsevier/Newnes, 2006