

BỘ GIÁO DỤC & ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH
KHOA ĐIỆN – ĐIỆN TỬ



ĐỒ ÁN TỐT NGHIỆP
NGÀNH CÔNG NGHỆ KỸ THUẬT ĐIỆN TỬ TRUYỀN THÔNG

THIẾT KẾ VÀ THI CÔNG HỆ THỐNG
HỖ TRỢ MỞ CỬA ỨNG DỤNG CÔNG NGHỆ RFID
TẠI KHÁCH SẠN DÙNG ARDUINO

GVHD: PGS.TS NGUYỄN THANH HẢI
SVTH: NGUYỄN VĂN HOÀNG
MSSV: 15141166
SVTH: PHẠM LÊ MINH KHÔI
MSSV: 15141187



Tp. Hồ Chí Minh, tháng 01/2021

BỘ GIÁO DỤC & ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH
KHOA ĐIỆN – ĐIỆN TỬ
BỘ MÔN ĐIỆN TỬ CÔNG NGHIỆP – Y SINH



ĐỒ ÁN TỐT NGHIỆP

NGÀNH KỸ THUẬT ĐIỆN TỬ TRUYỀN THÔNG

ĐỀ TÀI:

THIẾT KẾ VÀ THI CÔNG HỆ THỐNG HỖ
TRỢ MỞ CỦA ỨNG DỤNG CÔNG NGHỆ
RFID TẠI KHÁCH SẠN DÙNG ARDUINO

GVHD: PGS.TS Nguyễn Thanh Hải

SVTH: Nguyễn Văn Hoàng 15141166

SVTH: Phạm Lê Minh Khôi 15141187

Tp. Hồ Chí Minh – 1/2021

BỘ GIÁO DỤC & ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH
KHOA ĐIỆN – ĐIỆN TỬ
BỘ MÔN ĐIỆN TỬ CÔNG NGHIỆP – Y SINH

ĐỒ ÁN TỐT NGHIỆP

NGÀNH KỸ THUẬT ĐIỆN TỬ TRUYỀN THÔNG

ĐỀ TÀI:

THIẾT KẾ VÀ THI CÔNG HỆ THỐNG HỖ
TRỢ MỞ CỦA ỨNG DỤNG CÔNG NGHỆ
RFID TẠI KHÁCH SẠN DÙNG ARDUINO

GVHD: PGS.TS Nguyễn Thanh Hải

SVTH: Nguyễn Văn Hoàng 15141166

SVTH: Phạm Lê Minh Khôi 15141187

Tp. Hồ Chí Minh – 1/2021

Tp. HCM, ngày tháng năm 2021

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên:	Nguyễn Văn Hoàng	MSSV: 15141166
	Phạm Lê Minh Khôi	MSSV: 15141187
Chuyên ngành:	Kỹ thuật Điện tử - Truyền thông	Mã ngành: 141
Hệ đào tạo:	Đại học chính quy	Mã hệ: 1
Khóa:	2015	Lớp: 15141DT1

I. TÊN ĐỀ TÀI:

THIẾT KẾ VÀ THI CÔNG HỆ THỐNG HỖ TRỢ MỞ CỦA ỦNG DỤNG CÔNG NGHỆ RFID TẠI KHÁCH SẠN DÙNG ARDUINO

II. NHIỆM VỤ

1. Các số liệu ban đầu:

- Nguyễn Văn Hiệp, Giáo trình “Công nghệ nhận dạng bằng sóng vô tuyến”, trường ĐHSPKT, TP.HCM, 2014.
- Phạm Quang Huy, Lê Cảnh Trung, *Lập trình điều khiển Arduino*, Đại học SPKT Tp.HCM.
- Kiến thức cơ bản về các môn Mạch điện, Điện tử cơ bản, Vi xử lý, C/C#. Đồ án môn học 1; Đồ án môn học 2.

2. Nội dung thực hiện:

- Đọc các tài liệu, đồ án tốt nghiệp, đề tài và lên ý tưởng đồ án.
- Tìm hiểu về linh kiện và cách thức hoạt động của các module sử dụng.
- Thiết kế, xây dựng phần cứng của mô hình.
- Viết chương trình điều khiển cho mô hình.
- Thiết kế giao diện lưu trữ, nhập liệu thông tin trên máy tính.
- Chạy thử nghiệm mô hình, chỉnh sửa và hoàn thiện hệ thống.
- Viết báo cáo luận văn.
- Báo cáo đề tài tốt nghiệp.

III. NGÀY GIAO NHIỆM VỤ: 5/10/2020

IV. NGÀY HOÀN THÀNH NHIỆM VỤ: 20/01/2021

V. HỌ VÀ TÊN CÁN BỘ HƯỚNG DẪN: PGS.TS Nguyễn Thanh Hải

CÁN BỘ HƯỚNG DẪN

BM. ĐIỆN TỬ CÔNG NGHIỆP – Y SINH

Tp. HCM, ngày 20 tháng 1 năm 2021

LỊCH TRÌNH THỰC HIỆN ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên 1: Nguyễn Văn Hoàng

Lớp: 15141DT1A

MSSV: 15141166

Họ tên sinh viên 2: Phạm Lê Minh Khôi

Lớp: 15141DT1A

MSSV: 15141187

Tên đề tài: THIẾT KẾ VÀ THI CÔNG HỆ THỐNG HỖ TRỢ MỞ CỦA ỦNG
DỤNG CÔNG NGHỆ RFID TẠI KHÁCH SẠN DÙNG ARDUINO

<i>Tuần/ngày</i>	<i>Nội dung</i>	<i>Xác nhận GVHD</i>
Tuần 1(5/10 – 11/10)	Gặp gỡ GVHD và trao đổi về đề tài tốt nghiệp.	
Tuần 2(12/10 – 18/10)	Tìm hiểu các đề tài nghiên cứu có liên quan, viết đề cương chi tiết.	
Tuần 3(19/10 – 25/10)	Tìm hiểu các linh kiện sử dụng trong đề tài. Xây dựng sơ đồ khối và chức năng từng khối.	
Tuần 4(26/10 – 1/11)	Tìm hiểu các chuẩn giao tiếp sử dụng trong đề tài.	
Tuần 5(2/11 – 8/11)	Tìm hiểu giao tiếp giữa Arduino Pro mini với Module truyền, nhận dữ liệu RFM69.	
Tuần 6(9/11 – 15/11)	Tìm hiểu kết nối RFID RC522 với Arduino, điều khiển servo SG90. Truyền mã thẻ RFID.	
Tuần 7(16/11 – 22/11)	Chạy thử nghiệm mạch mở cửa. Chính sửa lỗi phần cứng và phần mềm.	
Tuần 8(23/11 – 29/11)	Chỉnh sửa, tối ưu code. Hoàn thiện phần cứng.	
Tuần 9(30/11 – 6/12)	Báo cáo tiến độ cho GVHD.	

Tuần 10(7/12 – 13/12)	Viết báo cáo chương 1,2.	
Tuần 11(14/12 – 20/12)	Tiếp tục hoàn thiện mạch. Tạo giao diện quản lý thông tin.	
Tuần 12(21/12 – 27/12)	Hoàn thiện giao diện quản lý. Thi công mô hình.	
Tuần 13(28/12 – 3/1)	Viết báo cáo.	
Tuần 14(4/1 – 10/1)	Chỉnh sửa báo cáo.	
Tuần 15(11/1 – 17/1)	Hoàn thiện báo cáo.	

GV HUỐNG DÂN
(Ký và ghi rõ họ và tên)

LỜI CAM ĐOAN

Đề tài này là do chúng em tự thực hiện dựa vào một số tài liệu và công trình nghiên cứu, không sao chép từ tài liệu hay công trình đã có trước đó.

Người thực hiện đề tài

Nguyễn Văn Hoàng

Phạm Lê Minh Khôi

LỜI CẢM ƠN

Trong quá trình thực hiện đồ án tốt nghiệp, lời đầu tiên cho phép chúng em gửi lời cảm ơn chân thành, sâu sắc tới giáo viên hướng dẫn Thầy PGS.TS Nguyễn Thanh Hải. Thầy đã tận tình hướng dẫn, góp ý, chia sẻ nhiều kinh nghiệm quý báu cũng như tạo điều kiện tốt nhất để chúng em hoàn thành tốt đề tài này.

Chúng em cũng gửi lời cảm ơn đến các Thầy, Cô trong bộ môn Điện Tử Công Nghiệp – Y Sinh nói riêng và các Thầy, Cô trong Khoa Điện – Điện Tử nói chung nói chung đã giúp đỡ chúng em về các kiến thức liên quan tới lĩnh vực nghiên cứu của đề tài, cũng như các kiến thức mà các Thầy, Cô đã giảng dạy trong suốt quá trình học tập tại trường.

Chúng em cũng gửi lời đồng cảm ơn đến ba mẹ, người thân và bạn bè đã luôn đồng hành động viên và giúp đỡ trong suốt quá trình thực hiện đề tài.

Sau cùng, dù có nhiều cố gắng, với kiến thức và kinh nghiệm còn hạn chế nên chúng em không tránh khỏi những điều thiếu sót. Chúng em rất mong nhận được sự đóng góp và chỉ bảo của quý Thầy, Cô để chúng em hoàn thiện tốt đề tài hơn.

Chúng em xin chân thành cảm ơn!

Người thực hiện đề tài
Nguyễn Văn Hoàng
Phạm Lê Minh Khôi

MỤC LỤC

Nhiệm vụ đồ án tốt nghiệp	i
Lịch trình thực hiện đồ án tốt nghiệp	ii
Lời cam đoan.....	iv
Lời cảm ơn.....	v
Mục lục	vi
Liệt kê hình.....	viii
Liệt kê bảng.....	x
Tóm tắt.....	xi
Chương 1. Tổng quan.....	1
1.1 Đặt vấn đề	1
1.2 Mục tiêu	2
1.3 Nội dung nghiên cứu	3
1.4 Giới hạn.....	3
1.5 Bố cục	3
Chương 2. Cơ sở lý thuyết	5
2.1 Ứng dụng của Arduino	5
2.1.1 Giới thiệu về Arduino	5
2.1.2 Các đề tài liên quan.....	5
2.2 Ứng dụng của RFID	8
2.2.1 Giới thiệu về RFID RC522	8
2.2.2 Các đề tài liên quan.....	9
2.3 Ứng dụng của LCD	10
2.4 Ứng dụng của RFM69HCW	11
2.5 Ứng dụng của động cơ servo SG90.....	13
Chương 3. Tính toán và thiết kế.....	15
3.1 Giới thiệu	15
3.2 Mô tả và thiết kế phần cứng.....	16
3.2.1 Mô tả khối vi điều khiển trung tâm	16
3.2.2 Mô tả khối RFID	17
3.2.3 Mô tả khối truyền nhận	19
3.2.4 Mô tả khối hiển thị	20
3.2.5 Mô tả khối báo động	22

3.2.6	Mô tả khối điều khiển	23
3.2.7	Mô tả khối nguồn	24
3.2.8	Thiết kế mạch nguồn.....	25
3.3	Sơ đồ nguyên lý	27
3.4	Nguyên lý hoạt động	28
Chương 4.	Thi công hệ thống	30
4.1	Thi công hệ thống.....	30
4.1.1	Thi công board mạch	30
4.1.2	Thi công mô hình.....	32
4.1.3	Thi công giao diện	35
4.2	Lập trình hệ thống	38
4.2.1	Lưu đồ giải thuật mạch chính	38
4.2.2	Lưu đồ giải thuật mạch điều khiển.....	43
4.3	Tài liệu hướng dẫn thao tác	47
4.4	Thông tin giá hệ thống.....	49
Chương 5.	Kết quả - Nhận xét – Đánh giá	50
5.1	Mô hình hệ thống	50
5.1.1	Mạch điều khiển phòng 1	50
5.1.2	Kết quả hoạt động mạch chính và giao diện.....	52
5.1.3	Kết quả thống kê	56
5.2	Nhận xét – Đánh giá.....	57
Chương 6.	Kết luận và hướng phát triển	59
6.1	Kết luận.....	59
6.2	Hướng phát triển.....	59
Tài liệu tham khảo	60	
Phụ lục	62	

LIỆT KÊ HÌNH

Hình 2.1: Hình ảnh Arduino Mega 2560	5
Hình 2.2: Hình ảnh mô hình khóa điện tử	6
Hình 2.3: Hình ảnh Arduino Uno R3	7
Hình 2.4: Hình ảnh mô hình bãi giữ xe	7
Hình 2.5: Module RFID RC522	8
Hình 2.6: Hình ảnh mô hình điểm danh nhân viên.....	9
Hình 2.7: Mạch I2C + LCD	10
Hình 2.8: Module RFM69HCW	12
Hình 2.9: Sơ đồ khối hệ thống dựa trên Moteino	13
Hình 2.10: Động cơ servo SG90.....	14
Hình 3.1: Sơ đồ khối hệ thống	15
Hình 3.2: Board Arduino ProMini.....	17
Hình 3.3: Module RFID RC522	18
Hình 3.4: Sơ đồ kết nối khối RFID.....	19
Hình 3.5: Sơ đồ giao tiếp module RFM69HCW với Arduino Pro Mini	19
Hình 3.6: Mặt trước của LCD 16x2.....	20
Hình 3.7: Mạch chuyển đổi giao tiếp I2C cho LCD.....	22
Hình 3.8: Sơ đồ Giao tiếp giữa Arduino ProMini với LCD16x2 I2C	22
Hình 3.9: Module buzzer	23
Hình 3.10: Động cơ Servo SG90	23
Hình 3.11: Sơ đồ nguyên lý mạch nguồn 5V	25
Hình 3.12: Sơ đồ nguyên lý mạch nguồn 3.3V	26
Hình 3.13: Sơ đồ nguyên lý mạch chính	27
Hình 3.14: Sơ đồ nguyên lý mạch điều khiển	28
Hình 4.1: Hình 3D mạch chính.....	30
Hình 4.2: Hình 3D mạch điều khiển.....	31
Hình 4.3: Mô hình mạch chính.....	33
Hình 4.4: Vị trí mạch điều khiển	34
Hình 4.5: Mặt ngoài mô hình mạch điều khiển	34
Hình 4.6: Giao diện đăng nhập	35

Hình 4.7: Giao diện quản lý.....	36
Hình 4.8: Giao diện đăng ký thông tin khách.....	37
Hình 4.9: Giao diện thanh toán.....	37
Hình 4.10: Các chức năng khác	38
Hình 4.11: Giao diện thông tin cá nhân.....	38
Hình 4.12: Lưu đồ giải thuật mạch chính.....	39
Hình 4.13: Lưu đồ giải thuật chương trình con đọc thẻ RFID	40
Hình 4.14: Lưu đồ giải thuật chương trình con xử lý kích hoạt thẻ	41
Hình 4.15: Lưu đồ giải thuật chương trình con xử lý dữ liệu truyền đến	42
Hình 4.16: Cấu trúc gói tin bộ thu phát	43
Hình 4.17: Lưu đồ giải thuật mạch điều khiển	44
Hình 4.18: Lưu đồ giải thuật chương trình kích hoạt phòng	45
Hình 4.19: Lưu đồ giải thuật chương trình đọc mã thẻ	46
Hình 4.20: Lưu đồ giải thuật chương trình mở cửa	46
Hình 4.21: Chọn cổng COM kết nối.....	47
Hình 4.22: Thứ tự thuê phòng cho khách	48
Hình 5.1: Mô hình thực tế.....	50
Hình 5.2: Màn hình LCD phòng 1 và vị trí quét thẻ.....	50
Hình 5.3: Cửa mở và tự động đóng lại	51
Hình 5.4: Cửa mở và tự động đóng lại	51
Hình 5.5: Màn hình LCD khi phòng kích hoạt ở phòng 1.....	52
Hình 5.6: Màn hình LCD khi thẻ không hợp lệ.....	52
Hình 5.7: Màn hình giao diện khi kết nối.....	53
Hình 5.8: Giao diện thuê phòng.....	53
Hình 5.9: Kích hoạt phòng	54
Hình 5.10: Thông tin khách	54
Hình 5.11: Lịch sử thẻ hoạt động	55
Hình 5.12: Lịch sử thẻ hoạt động phòng 3	55

LIỆT KÊ BẢNG

Bảng 3.1: Các chân LCD và chức năng.....	20
Bảng 3.2: Thông số dòng điện và điện áp của các linh kiện trong mạch phụ	24
Bảng 4.1: Bảng giá linh kiện sử dụng trong hệ thống	49
Bảng 5.1: Kết quả thống kê mở cửa khi phòng chưa kích hoạt.....	56
Bảng 5.2: Kết quả thống kê mở cửa khi có 1 phòng được kích hoạt.....	56
Bảng 5.3: Kết quả thống kê mở cửa khi có 2 phòng được kích hoạt.....	57
Bảng 5.4: Kết quả thống kê mở cửa khi có 3 phòng được kích hoạt.....	57

TÓM TẮT

Ngày nay trước sự phát triển không ngừng của khoa học kỹ thuật, đã làm cho đời sống con người ngày càng cải thiện hơn. Khi mức sống con người được nâng cao thì nhu cầu về giải trí, du lịch càng nâng cao hơn. Chính vì lẽ đó các khách sạn mọc lên ngày càng nhiều đòi hỏi việc quản lý được chú trọng.

Với trình độ khoa học kỹ thuật ngày càng phát triển, rất nhiều những vấn đề đã được giải quyết một cách nhanh gọn với công nghệ điện tử hiện đại. Các khách sạn đã dần thay thế ổ khóa cơ sử dụng các chìa khóa bằng tay thành khóa điện tử ứng dụng công nghệ RFID hỗ trợ việc mở cửa và giúp việc quản lý dễ dàng hơn.

Nội dung chính của đề tài bao gồm:

- Sử dụng board Arduino ProMini làm vi điều khiển trung tâm.
- Ứng dụng công nghệ RFID trong việc đóng/ mở cửa.
- Thêm, tra cứu, chỉnh sửa thông tin khách hàng qua phần mềm máy tính.
- Lưu trữ thông tin trên cơ sở dữ liệu.

Chương 1. TỔNG QUAN

1.1 ĐẶT VẤN ĐỀ

Ngày nay, khoa học kỹ thuật không ngừng phát triển đã làm cho cuộc sống của con người được tốt đẹp và hoàn thiện hơn. Nhưng bên cạnh đó thì các vấn đề xã hội xảy ra ngày càng nhiều trong đó có nạn trộm cắp tài sản. Các biện pháp bảo vệ truyền thống như dùng ổ khóa cửa không thể đảm bảo về tính an toàn cho nhà cửa. Vì thế cần có một thiết bị thông minh hơn giúp bảo vệ tài sản được an toàn hơn. Như khóa chống trộm thông minh ứng dụng công nghệ RFID để đóng/ mở cửa, có giám sát từ xa qua GSM để gửi tin nhắn cảnh báo như đề tài đồ án tốt nghiệp “Thiết kế và thi công mô hình khoá điện tử có giám sát từ xa sử dụng GSM và RFID” của Nguyễn Thành Dương [1], góp phần hạn chế tối đa mất mát tài sản. RFID còn có thể ứng dụng trong việc quản lý các đối tượng như hàng hóa, xe cộ, kể cả con người. Từ đó nhóm đưa ra lựa chọn đề tài xây dựng một hệ thống mở cửa sử dụng RFID để quản lý thông tin khách hàng cho khách sạn.

Công nghệ RFID là công nghệ nhận dạng đối tượng bằng sóng vô tuyến. Công nghệ này cho phép nhận biết các đối tượng từ đó có thể giám sát quản lý từng đối tượng. Như vậy, việc bạn phải xếp hàng dài để chờ đợi đến lượt mình tính tiền khi đi siêu thị sẽ không còn là vấn đề. Một thiết bị tự động nhận dạng từng món hàng bạn mua và tự động trừ tiền vào tài khoản thanh toán của bạn. Công việc của bạn đơn giản chỉ là mua hàng và đi qua cổng giám sát có gắn thiết bị nhận dạng đó và đây chỉ là một trong rất nhiều ứng dụng tiện ích của công nghệ RFID.

Độ phủ sóng của công nghệ RFID gần như xuất hiện trong mọi lĩnh vực của đời sống. Đó có thể là thẻ từ bạn cầm mỗi lần gửi xe ra, vào tại các bãi giữ xe thông minh như đề tài của Nguyễn Thanh Lộc “Thiết kế và thi công mô hình điều khiển, giám sát bãi giữ xe ô tô tự động” [2]. Hoặc để tăng tính bảo mật và độ chính xác hơn có thể ứng dụng thêm xử lý ảnh để nhận diện biển số xe như đề tài “Thiết kế, thi công bãi giữ xe ứng dụng công nghệ RFID và xử lý ảnh” của Nguyễn Đăng Việt [3]. Điểm danh, quản lý thời gian ra/vào của nhân viên trong công ty như đề tài “Thiết kế và thi công hệ thống điểm danh nhân viên sử dụng vi điều khiển ARM” của Trần Thị Linh Đa [4]. Hay việc sử dụng RFID để lưu thông tin khách hàng hỗ trợ việc quản lý sử dụng máy giặt và

Chương 1. TỔNG QUAN

than toán trở nên dễ dàng và nhanh chóng được thực hiện bởi Lê Trung Hậu qua đề tài đồ án tốt nghiệp “Thiết kế và thi công mô hình cửa hàng giặt sấy tự phục vụ” [5].

Dù cáp và dây điện vẫn đóng vai trò chính trong truyền và nhận thông tin, nhưng việc sử dụng công nghệ không dây đang trở nên quá quen thuộc và có tác động vô cùng lớn trong đời sống hiện nay. Đa số các công nghệ không dây có khoảng cách truyền không xa, với Bluetooth từ 10-100m hay Wifi là 30-50m với điều kiện không có vật cản là hai công nghệ phổ biến và gần gũi nhất. Hiện nay, khoảng cách của công nghệ không dây đã có sự cải thiện lớn với sự ra đời của Lora. Nó có thể hỗ trợ liên lạc ở khoảng cách lên tới 15-20 km, với hàng triệu node mạng. Nó có thể hoạt động trên băng tần không phải cấp phép, với tốc độ thấp từ 0,3kbps đến khoảng 30kbps. Với đặc tính này, mạng LoRa phù hợp với các thiết bị thông minh trao đổi dữ liệu ở mức thấp nhưng duy trì trong một thời gian dài [6].

Bên cạnh việc tăng khoảng cách truyền thông tin và cải thiện tốc độ truyền thì việc bảo mật thông tin truyền cũng là một vấn đề. Công nghệ Zigbee là một tiêu chuẩn mở toàn cầu cho công nghệ không dây được thiết kế để sử dụng tín hiệu vô tuyến kỹ thuật số năng lượng thấp cho các mạng khu vực cá nhân. Nó sử dụng mã hoá AES-128 mang đến độ bảo mật cao tuy nhiên khoảng cách truyền tối đa khoảng 75m [7]. Hay bộ thu phát RFM69HCW cũng là một bộ thu phát sóng vô tuyến có độ bảo mật sử dụng mã hoá AES-128 nhưng có khoảng cách truyền xa hơn 300-500m [8]. RFM69HCW cũng gần giống như Lora tuy nhiên có khoảng cách truyền ngắn hơn, có khả năng tạo một mạng đa điểm với các địa chỉ mạng khác nhau. RFM69HCW hỗ trợ 256 địa chỉ từ 0 đến 255, nếu chọn 0 là trạm thì có thể tạo tối đa 255 nút riêng lẻ kết nối với trạm.

Từ những tìm hiểu trên cùng với các kiến thức đã học, nhóm đưa ra lựa chọn đề tài “**Thiết kế và thi công hệ thống hỗ trợ mở cửa ứng dụng công nghệ RFID tại khách sạn dùng Arduino**”. Hệ thống sử dụng vi điều khiển là Arduino cùng với module RFID RC522 và servo kết hợp với tín hiệu kích hoạt mã thẻ để đóng mở cửa.

1.2 MỤC TIÊU

Thiết kế và thi công mô hình hệ thống đóng mở cửa khách sạn gồm một mạch điều khiển đóng mở cửa dùng vi điều khiển Arduino để giao tiếp với máy tính và thẻ RFID cho nhận dạng. Trong đó, mã thẻ nhận từ mạch xử lý thông tin thẻ đồng thời một trang

Chương 1. TỔNG QUAN

giao diện cho việc nhập và lưu trữ thông tin khách hàng trên máy tính. Hệ thống sử dụng vi điều khiển Arduino làm bộ xử lý trung tâm.

1.3 NỘI DUNG NGHIÊN CỨU

Đề tài được thực hiện gồm có những nội dung chính sau:

- **Nội dung 1:** Tham khảo tài liệu, đọc và tóm tắt tài liệu đưa ra các hướng đề tài.
- **Nội dung 2:** Thiết kế sơ đồ khái, sơ đồ nguyên lí.
- **Nội dung 3:** Kết nối Arduino ProMini với nguồn, các Module RFID RC522, Module RFM69HCW, LCD 16x2, động cơ servo.
- **Nội dung 4:** Viết chương trình điều khiển cho Arduino.
- **Nội dung 5:** Thiết kế giao diện lưu trữ và nhập liệu thông tin trên máy tính.
- **Nội dung 6:** Chạy thử, kiểm tra, đánh giá, hiệu chỉnh.
- **Nội dung 7:** Viết báo cáo thực hiện.
- **Nội dung 8:** Báo cáo đề tài tốt nghiệp.

1.4 GIỚI HẠN

Các thông số giới hạn của đề tài bao gồm:

- Đây chỉ là mô hình chạy thử nghiệm nhỏ gồm 1 mạch xử lý gửi tín hiệu kích hoạt cho 3 mạch điều khiển đóng/mở cửa đại diện cho 3 phòng khách sạn.
- Mô hình chỉ hoạt động với môi trường khô ráo trong nhà.
- Cảnh báo bằng còi khi mã thẻ không hợp lệ.
- Sử dụng servo để đóng/mở cửa.

1.5 BỐ CỤC

Đề tài được trình bày trong 6 chương:

❖ **Chương 1: Tổng quan**

Chương này trình bày đặt vấn đề dẫn nhập lý do chọn đề tài, mục tiêu, nội dung nghiên cứu, các giới hạn thông số và bố cục đồ án.

❖ **Chương 2: Cơ sở lý thuyết**

Chương này trình bày ứng dụng của các module, linh kiện sử dụng cho thiết kế mô hình (Arduino, module RFID, module RFM69HCW, LCD 16x2, động cơ SG90), kiến thức cơ bản và các phần mềm liên quan để hỗ trợ cho lập trình điều khiển mô hình.

❖ **Chương 3: Tính toán và thiết kế**

Chương 1. TỔNG QUAN

Xây dựng sơ đồ khối, trình bày nguyên lý hoạt động của hệ thống. Thiết kế và lựa chọn linh kiện phù hợp cho từng khối. Xây dựng sơ đồ nguyên lý cho từng khối và toàn bộ hệ thống.

❖ **Chương 4: Thi công hệ thống**

Thi công hàn linh kiện lên mạch, lắp ráp kiểm tra toàn bộ mạch. Xây dựng lưu đồ giải thuật, viết chương trình, thi công mô hình.

❖ **Chương 5: Kết quả, nhận xét và đánh giá**

Trình bày những kết quả đạt được (cấu trúc mô hình, giao diện phần mềm). Sau đó đưa ra nhận xét tổng thể về đề tài cũng như những đánh giá dựa trên các mục tiêu đã đặt ra ban đầu.

❖ **Chương 6: Kết luận và hướng phát triển**

Nêu ra ưu điểm và nhược điểm của hệ thống. Đưa ra những điểm cần cải thiện của đề tài, hướng phát triển và phạm vi sử dụng của đề tài trong tương lai gần.

Chương 2. CƠ SỞ LÝ THUYẾT

2.1 ỨNG DỤNG CỦA ARDUINO

2.1.1 Giới thiệu về Arduino

Arduino là một board mạch vi xử lý dùng để lập trình xây dựng các ứng dụng tương tác với nhau hoặc với môi trường được thuận lợi hơn. Ưu điểm của Arduino là ngôn ngữ cực kì dễ học (giống C/C++), các ngoại vi trên bo mạch đều đã được chuẩn hóa, nên không cần biết nhiều về điện tử chúng ta cũng có thể lập trình được. Phần cứng bao gồm một board mạch nguồn mở được thiết kế trên nền tảng vi xử lý AVR Atmel 8bit, 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.

Hiện nay trên thị trường có rất nhiều phiên bản Arduino như Arduino Uno R3, Arduino Uno R3 CH340, Arduino Mega2560, Arduino Nano, Arduino Pro Mino, Arduino Lenadro, Arduino Industrial....

2.1.2 Các đề tài liên quan

- Sử dụng Arduino Mega 2560



Hình 2.1: Hình ảnh Arduino Mega 2560

Đề tài đồ án tốt nghiệp: “Thiết kế và thi công mô hình khoá điện tử có giám sát từ xa sử dụng GSM và RFID” của Nguyễn Thành Dương [1] sử dụng board arduino mega 2560 làm khôi điều khiển, mạch mở khóa bằng 2 lớp: mở khóa bằng thẻ từ RFID rồi mới mở khóa bằng phím ma trận 4x4. Mật khẩu đúng thì người dùng được quyền quản lý khóa: đổi mật khẩu thẻ, thêm thẻ mới, xóa thẻ, kích hoạt-tắt chế độ SMS trên LCD.

Chương 2. CƠ SỞ LÝ THUYẾT

Có khối giám sát việc đóng, mở cửa dùng module sim900A với mục đích là giám sát và cảnh báo thông qua việc gửi tin nhắn và gọi điện. Có cảm biến PIR phát hiện vật thể lạ khi cửa khóa thì mô hình kích hoạt báo động bằng chuông, đèn led, gửi tin nhắn vào số điện thoại được cài đặt. Khi vào nhà thì có một nút nhấn điều khiển động cơ để khóa hay mở cửa.



Hình 2.2: Hình ảnh mô hình khóa điện tử

Đề tài: “Thi công mô hình hệ thống trồng hoa lan” của Nguyễn Quang Thạnh [9] cũng sử dụng Arduino Mega 2560 làm khôi xử lý trung tâm có chức năng nhận dữ liệu từ các khôi cảm biến mưa, cảm biến nhiệt độ và độ ẩm DHT11, cảm biến độ ẩm đất, cảm biến ánh sáng, Realtime, app điều khiển bằng Bluetooth để điều khiển các thiết bị qua module Relay và L298. Truyền dữ liệu cho khôi hiển thị và Module ESP8266 để hiển thị thông tin và gửi lên Firebase.

Đề tài: “Ứng dụng xử lý ảnh điều khiển đèn giao thông tại nút giao thông” của Nguyễn Hoàn Hảo [10], sử dụng Arduino Mega 2560 làm khôi điều khiển. Sau khi nhận được giá trị từ bộ xử lý ảnh thông qua chuẩn kết nối UART, khôi điều khiển sẽ xử lý tính toán thời gian các đèn xanh, đỏ, vàng trên mỗi tuyến đường và cho hiển thị số giây lên khôi hiển thị là led 7 đoạn. Người tham gia giao thông nhận thấy được tín hiệu đèn và tuân thủ, đồng thời sẽ là dữ liệu tiếp theo cho khôi ảnh đầu vào. Camera ghi lại hình ảnh người tham gia giao thông và đưa qua khôi xử lý ảnh để khôi điều khiển tính toán chu kỳ tiếp theo.

- Sử dụng Arduino UNO R3

Chương 2. CƠ SỞ LÝ THUYẾT



Hình 2.3: Hình ảnh Arduino Uno R3

Đề tài: “Thiết kế, thi công bãi giữ xe ứng dụng công nghệ RFID và xử lý ảnh” của Nguyễn Đăng Việt [3] sử dụng board Arduino Uno R3 làm khói xử lý trung tâm. Khi xe vào, hành khách được nhận thẻ và tự quét thẻ. Khối xử lý trung tâm nhận tín hiệu gửi về từ module RFID. Camera sẽ chụp ảnh và gửi về máy tính để nhận dạng biển số. Servo mở để xe chạy vào. Khi xe ra, thẻ được quét, camera cũng sẽ chụp lại hình ảnh xe và biển số xe, dựa vào mã ID của thẻ mà so sánh, nếu đúng với biển số xe ban đầu thì servo mở cho xe ra. Còn nếu không trùng khớp hệ thống sẽ thông báo trên LCD “Không hợp lệ”. Servo sẽ không mở. Việc quản lý xe ra, vào sẽ được lưu lại trên giao diện của máy tính gồm có hình ảnh camera ngoã ra và ngoã vào hoạt động liên tục, ảnh xe vào, ảnh xe ra, ảnh biển số xe sau khi đã xử lý, bảng thông tin xe của khách, ô hiển thị số lượng xe còn trong bãi, nút xem lịch sử xe trong ngày.



Hình 2.4: Hình ảnh mô hình bãi giữ xe

Chương 2. CƠ SỞ LÝ THUYẾT

Đề tài: “Thiết kế và thi công hệ thống quản lý mô hình HOMESTAY, AIRBNB” của Hoàng Bảo Phúc [11]. Dùng khói xử lý là Arduino Uno R3 nhận các thông tin từ ESP8266 và từ khói ngoại vi là các cảm biến vân tay, bàn phím ma trận; so sánh các thông tin để thực hiện mở cửa, hiển thị thông báo lên LCD đồng thời thu thập các thông tin khác gửi về ESP8266. ESP8266 có nhiệm vụ giao tiếp (nhận dữ liệu và gửi dữ liệu) giữa khói xử lý và server.

Đề tài: “Thiết kế và thi công mô hình xe robot dò tìm kim loại điều khiển bằng điện thoại” của Nguyễn Hữu Phước [12]. Sử dụng Arduino Uno R3 làm khói điều khiển trung tâm thu thập các tín hiệu từ khói cảm biến phát hiện kim loại và khói thu phát Wifi sau đó xử lý điều khiển khói công suất điều khiển động cơ. Khối thu phát wifi làm nhiệm vụ giao tiếp giữa khói điều khiển trung tâm và điện thoại, là cầu nối trung gian để nhận dữ liệu từ điện thoại sau đó gửi tới khói trung tâm để xử lý các hoạt động điều khiển và nhận dữ liệu ngược lại từ khói trung tâm để hiện thị lên trên điện thoại.

2.2 ÚNG DỤNG CỦA RFID

2.2.1 Giới thiệu về RFID RC522



Hình 2.5: Module RFID RC522

• Giới thiệu

Công nghệ 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. Có nhiều ưu điểm vượt trội so với công nghệ mã vạch. Công nghệ mã mạch (line of sight technology) chỉ cho phép nhận dạng đối tượng khi máy đọc cần phải đặt đối tượng ở khoảng cách gần. Trong khi đó, với công nghệ RFID có thể xác định đối tượng ở khoảng cách xa từ vài mét cho tới hàng chục mét trong không gian 3 chiều.

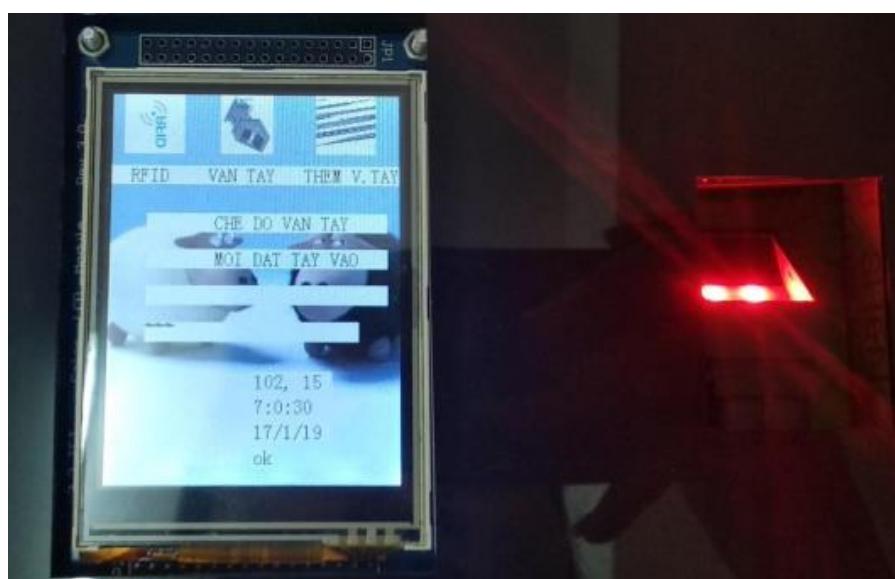
Chương 2. CƠ SỞ LÝ THUYẾT

• Cấu tạo

Một thiết bị hay một hệ thống RFID được cấu tạo bởi hai thành phần chính là thiết bị đọc (reader) và thiết bị phát mã RFID có gắn chip hay còn gọi là tag. Thiết bị đọc được gắn antenna để thu - phát sóng điện từ, thiết bị phát mã RFID tag được gắn với vật cần nhận dạng, mỗi thiết bị RFID tag chứa một mã số nhất định và không trùng lặp nhau.

2.2.2 Các đề tài liên quan

Đề tài: “Thiết kế và thi công hệ thống điểm danh nhân viên sử dụng vi điều khiển ARM” của Trần Thị Linh Đa [4] đã dùng khói điều khiển trung tâm là chip STM32F103VET6 là bộ não của hệ thống. Có màn hình điều khiển để chọn hình thức điểm danh dùng cả hai hình thức quét thẻ RFID và quét vân tay. Dữ liệu sẽ được lưu vào thẻ nhớ SD_card đồng thời đưa lên web server để dễ dàng quản lý. Giao diện đơn giản, dễ sử dụng dành cho nhân viên và quản lý bằng cách đăng nhập vào trang Web. Website của hệ thống chỉ có người quản lý mới có thể đăng nhập, các chức năng trên website: Lưu trữ thời gian điểm danh của từng nhân viên; Thêm, xóa, xem danh sách thông tin nhân viên; Trích xuất dữ liệu điểm danh theo tên, ngày, tháng sang file excel. Nhân viên cũng có thể đăng nhập nhưng chỉ theo dõi được thông tin, giờ vào ra của mình.



Hình 2.6: Hình ảnh mô hình điểm danh nhân viên

Sử dụng board Arduino Uno R3 kết nối với RFID giúp cho việc quản lý trở nên dễ dàng hơn như đề tài: “Quản lý dữ liệu bệnh nhân trong bệnh viện sử dụng công nghệ

Chương 2. CƠ SỞ LÝ THUYẾT

RFID” của Nguyễn Tài Tụ [14]. Khi đăng nhập vào hệ thống và quét thẻ RFID, mã thẻ sẽ được khôi vi điều khiển gửi lên phần mềm máy tính đồng thời mã cũng sẽ hiện ra màn hình LCD, từ phần mềm máy tính sẽ lấy mã thẻ đó truy xuất lên cơ sở dữ liệu để lấy thông tin về hiển thị lên giao diện máy tính. Nếu có thông tin dữ liệu của thẻ trên database, hệ thống cho phép truy xuất những thông tin kèm theo của bệnh nhân như tên, tuổi, giới tính, năm sinh, hình ảnh, ngày vào, ngày ra và bệnh án. Bên cạnh đó, với mỗi lần thêm, sửa hoặc xóa thông tin của bệnh nhân thì hệ thống cũng sẽ cập nhật thông tin trên database.

Ngoài Arduino hay ARM, RFID còn có thể kết nối được với PLC S7 – 1200 CPU 1214C DC/DC/DC để làm khôi xử lý trung tâm như đề tài: “Thiết kế và thi công mô hình điều khiển, giám sát bãi giữ xe ô tô tự động” của Nguyễn Thanh Lộc [2]. Sử dụng laptop tích hợp sẵn những công cụ lập trình có nhiệm vụ thu nhận, xử lý, lưu trữ mã thẻ từ đầu đọc thẻ RFID. Khối đọc thẻ sử dụng Arduino Uno R3 và mạch đọc thẻ RFID RC522 để đọc mã thẻ từ các thẻ Tag, kết nối với máy tính qua cổng USB. Nhận dạng biển số từ hình chụp được bằng Camera sử dụng Webcam Colorvis ND60. PLC nhận tín hiệu từ máy tính và các tín hiệu ngõ vào để điều khiển động cơ giúp nâng và hạ để hoàn tất việc cất xe hoặc trả xe cho khách. Còn có giao diện để tiện cho việc quản lý xe trong bãi.

2.3 ÚNG DỤNG CỦA LCD

LCD thường được dùng làm khôi hiển thị cho các đề tài đơn giản, giúp thông báo trạng thái, các thông số đọc được từ cảm biến hoặc dùng làm đồng hồ thời gian thực, không yêu cầu độ phân giải cao. Để giúp tiết kiệm chân LCD thường dùng chung với mạch giao tiếp I2C.



Hình 2.7: Mạch I2C + LCD

Chương 2. CƠ SỞ LÝ THUYẾT

Như với đề tài: “Thiết kế và thi công mô hình khoá điện tử có giám sát từ xa sử dụng GSM và RFID” của Nguyễn Thành Dương [1] LCD có chức năng thông báo các trạng thái của khóa: khi bắt đầu hoạt động, mô hình ở trạng thái chờ quét thẻ, màn hình LCD sau khi thiết lập xong sẽ hiện “KHOA CUA DIEN TU MOI QUET THE”. Nếu quét thẻ sai màn hình LCD sẽ hiện “VUI LONG THU LAI” và số lần quét sai sẽ tăng lên nếu tiếp tục sử dụng thẻ sai. Khi quét thẻ sai 3 lần trạng thái cảnh báo sẽ được kích hoạt buzzer và led sẽ chớp tắt liên tục. Sau khi quét thẻ thành công và nhập đúng mật khẩu sẽ vào trạng thái quản lý và có quyền thao tác với khóa, màn hình LCD sẽ hiển thị “CUA MO CO QUYEN Q/LY”. Khi ở trong trạng thái quản lý, nhấn A sẽ chọn chế độ thêm thẻ, khi đó màn hình LCD sẽ hiện “MOI QUET THE MOI” để yêu cầu quét thẻ mới, nếu là thẻ mới thì yêu cầu nhập mật khẩu thẻ và nhập xong sẽ thoát ra trạng thái chờ quét thẻ, nếu là thẻ đã có sẽ thoát ra trạng thái quản lý. Ở trạng thái quản lý, khi nhấn C sẽ vào chế độ đổi mật khẩu, màn hình LCD sẽ hiển thị “NHAP M/KHAU CU” và yêu cầu nhập mật khẩu cũ của thẻ. Sau khi nhập đúng mật khẩu sẽ yêu cầu nhập mật khẩu mới và xác nhận. Sau khi xác nhận sẽ thoát ra và ở trạng thái chờ quét thẻ.

2.4 ỨNG DỤNG CỦA RFM69HCW

Mạng cảm biến không dây có thể được phân loại rộng rãi theo các khía cạnh sau: cấu trúc liên kết mạng và hoạt động của giao thức. Module RFM69HCW là một module RF có khả năng hoạt động trên một dải tần số có thể lập trình rộng, bao gồm các băng tần RF 315 MHz, 433 MHz, 868 MHz và 915 MHz. Bộ thu phát hỗ trợ khóa chuyển đổi tần số, khóa dịch chuyển tối thiểu và điều chế khóa bật/tắt [15]. Cấu trúc của gói tin RF RFM69HCW rất đơn giản và linh hoạt. Tất cả các gói được gửi qua bộ thu phát RFM69HCW bao gồm các trường sau: phần mở đầu, từ đồng bộ, tải trọng và các byte CRC tùy chọn. Không giống như BLE, không có các giai đoạn quảng cáo và kết nối giữa hai thiết bị; khi một nút muốn liên lạc với một thiết bị, nó sẽ phát gói dữ liệu của nó và có thể tùy chọn bao gồm một byte địa chỉ để tắt cả các nút ngoại trừ nút nhận mong muốn bỏ qua gói đó.

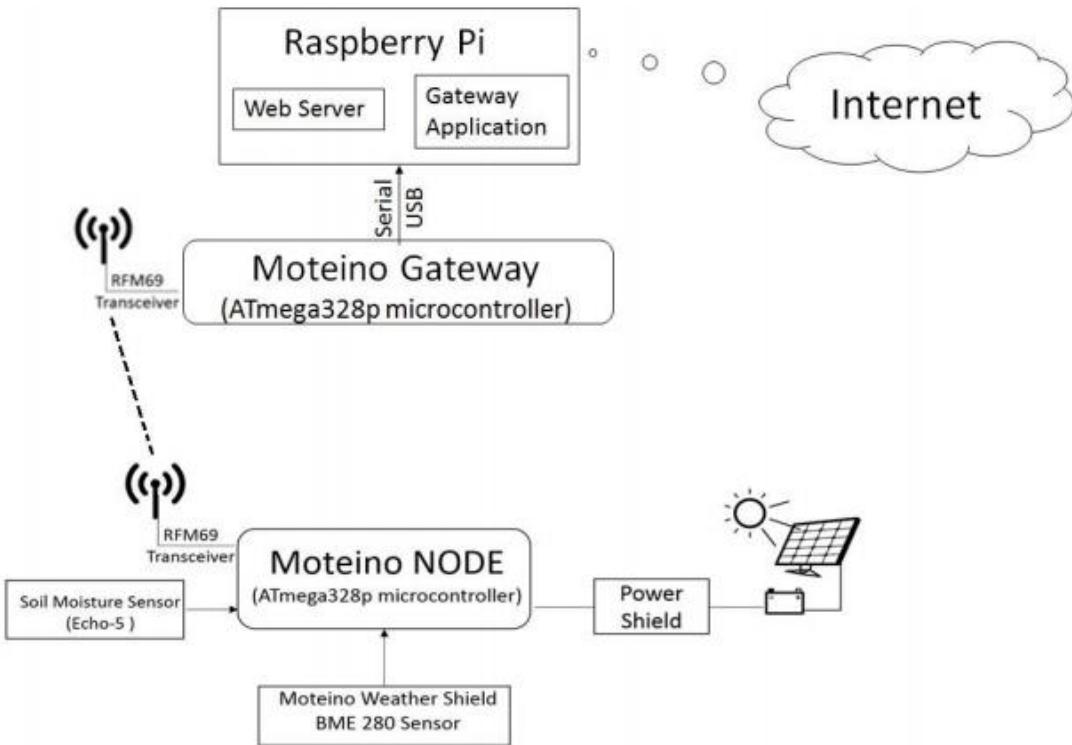


Hình 2.8: Module RFM69HCW

Với đề tài “*Design and Implementation of a Secure Energy-Efficient Hardware Platform for Wireless Sensor Networks*” của Madeleine Waller, các nút cảm biến không dây bao gồm một hoặc nhiều thiết bị cảm biến, bộ thu phát và bộ vi điều khiển để xử lý trạng thái và điều khiển giao tiếp qua RF. Nền tảng nút cảm biến hiện có cơ sở hạ tầng mã hóa để giao tiếp với bộ đàm: Radio Packet RFM69HCW của Sparkfun [8]. Thiết kế nền tảng nút cảm biến bao gồm một bảng mạch in tùy chỉnh với các thành phần chính sau:

- Mạch bảo vệ pin để cho phép cấp nguồn pin hoặc microUSB
- Chip tùy chỉnh với bộ xử lý RISC-V và bộ tăng tốc phần cứng mật mã.
- 2 cảm biến: cảm biến gia tốc LIS3DE và cảm biến nhiệt độ LM95071
- Tiêu đề đột phá SPI để kết nối với nhiều loại radio
- Ổ cắm MicroSD, với thẻ SD để lưu trữ chương trình

Một đề tài khác nói về “*Moteino-based wireless data transfer for environmental monitoring*” của Samuel Iyiola, gồm hai hệ thống con: hệ thống con cảm biến gồm cảm biến đọc nhiệt độ, độ ẩm, áp suất và độ ẩm đất xử lý nó trong bộ điều khiển Atmega328 của TheMoteino và truyền nó đến hệ thống con khác là trạm gốc thông qua module vô tuyến RFM69W cứ 10 phút một lần.



Hình 2.9: Sơ đồ khái niệm hệ thống dựa trên Moteino

Việc truyền dữ liệu không dây đã được thử nghiệm cho cả tình huống trong nhà và ngoài trời. Trong nhà tình huống đã được thử nghiệm trong một môi trường đông dân cư với các văn phòng và các bức tường ngăn cách nút cảm biến từ trạm gốc. Cường độ tín hiệu nhận được là khoảng -88dBm trong 40m với NLOS. Thử nghiệm ngoài trời là một không gian mở với LOS và cường độ tín hiệu nhận được là khoảng -96dBm ở 198m. So với nút cảm biến IRIS, có thời lượng pin là 416 ngày, nút cảm biến dựa trên Moteino có tuổi thọ khoảng chín năm cho cùng Dung lượng pin 2500mAh mà không tính đến tốc độ tự xả của pin. Điều này cực kỳ tiêu thụ dòng điện thấp sẽ giúp giảm kích thước pin hoặc thay thế hoàn toàn pin bằng siêu tụ điện cùng với hệ thống thu năng lượng hiệu quả trong các dự án tương lai [16].

2.5 ÚNG DỤNG CỦA ĐỘNG CƠ SERVO SG90

Servo là một dạng động cơ điện đặc biệt. Không giống như động cơ thông thường là quay liên tục, servo chỉ quay khi được điều khiển (bằng xung PPM), với góc quay nằm trong khoảng bát kì từ 0° - 180° .

Chương 2. CƠ SỞ LÝ THUYẾT



Hình 2.10: Động cơ servo SG90

Động cơ servo SG90 để mô phỏng khóa (chốt) cửa tự động khi có tín hiệu khóa từ khói điều khiển trung tâm Arduino Mega 2560. Sau khi mở được 2 lớp khóa bảo vệ là quét thẻ RFID và nhập mật khẩu thì động cơ servo quay góc 90° để mở cửa. Động cơ cũng được điều khiển góc quay 90° để mở cửa, hoặc 0° để đóng cửa bằng nút nhấn khi có người trong nhà như đề tài: “Thiết kế và thi công mô hình khoá điện tử có giám sát từ xa sử dụng GSM và RFID” của Nguyễn Thành Dương [1].

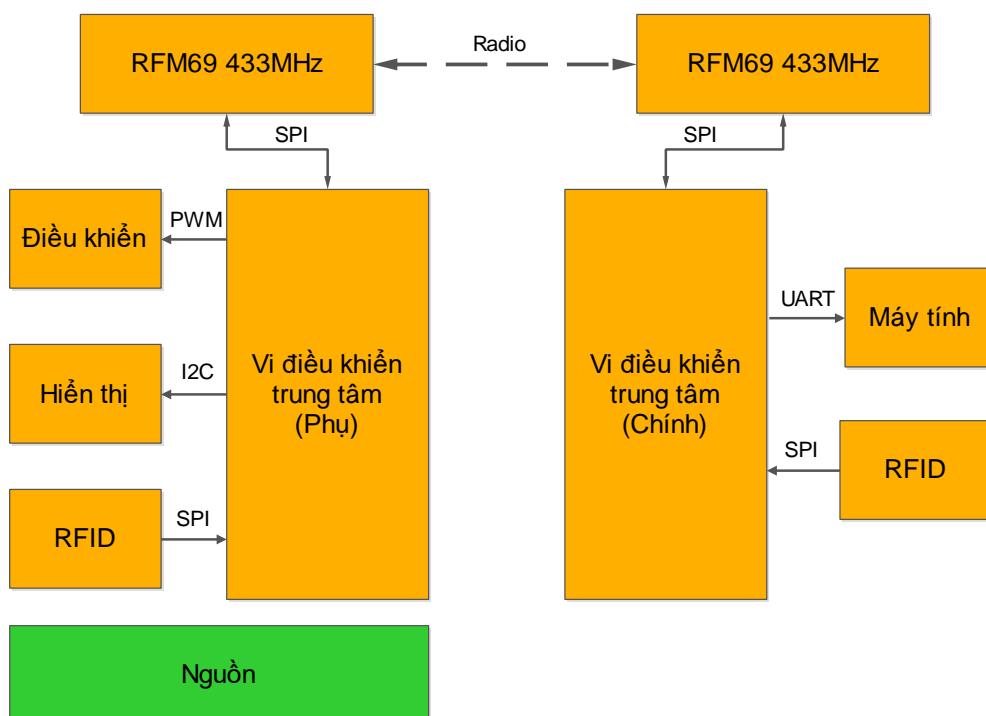
Cũng sử dụng động cơ SG90 để mô phỏng chốt chặn xe. Ban đầu servo ở góc 0° , khi xe được cấp thẻ sau đó tự quét thẻ, được hệ thống nhận dạng qua camera thì servo điều khiển một góc quay 90° có xe vào bãi. Sau khi xe chạy vào thì hạ servor xuống bằng cách quay về lại góc 0° . Tương tự như vậy, dùng thêm một động cơ servo với chốt ra khi xe ra, quét thẻ, camera cũng sẽ chụp lại hình ảnh xe và biển số xe, dựa vào mã ID của thẻ mà so sánh, nếu đúng với biển số xe ban đầu thì cho xe ra là đề tài: “Thiết kế, thi công bãi giữ xe ứng dụng công nghệ RFID và xử lý ảnh” của Nguyễn Đăng Việt [3].

Chương 3. TÍNH TOÁN VÀ THIẾT KẾ

3.1 GIỚI THIỆU

Đề tài yêu cầu thiết kế một mô hình hệ thống mở khóa cửa tự động bằng hình thức quét thẻ RFID và quản lý bằng module RFM69HCW, báo động bằng buzzer khi quét thẻ sai như được mô tả trong Hình 3.1. Dữ liệu đọc được sẽ được gửi lên cơ sở dữ liệu để dễ dàng quản lý. Mô hình gồm 2 phần:

- ✓ Phần thứ nhất là mạch chính gồm khói điều khiển trung tâm có kết nối với máy tính dùng để quản lý thông tin khách cũng như phòng khách sạn trên cơ sở dữ liệu. Khối RFID để kích hoạt thẻ khi có khách thuê phòng. Khối RFM69 truyền nhận dữ liệu qua lại giữa mạch chủ và mạch phụ.
- ✓ Phần thứ hai là mạch phụ (mạch điều khiển) gồm khói điều khiển trung tâm đọc dữ liệu của khói RFID. Nếu thẻ đúng thì khói điều khiển sẽ mở/ đóng cửa. Thẻ sai sẽ có khói báo động cảnh báo. Khối hiển thị báo số phòng, phòng đã được kích hoạt hay chưa, báo thời gian đóng/ mở cửa. Khối RFM69 truyền nhận dữ liệu qua lại giữa mạch chủ và mạch phụ.



Hình 3.1: Sơ đồ khái niệm hệ thống

Chức năng từng khối:

Chương 3. TÍNH TOÁN VÀ THIẾT KẾ

- **Khối vi điều khiển trung tâm (chính):** Đọc và xử lý thông tin đọc từ thẻ RFID truyền thông tin lên máy tính để lưu trữ thông tin, truyền tín hiệu kích hoạt cho mạch phụ bằng sóng vô tuyến thông qua khói RFM69 433MHz.
- **Khối vi điều khiển trung tâm (phụ):** Đọc và xử lý thông tin đọc từ thẻ RFID truyền thông tin lên khói hiển thị kết hợp với tín hiệu kích hoạt từ mạch chính điều khiển đóng/ mở cửa thông qua khói điều khiển.
- **Khối hiển thị:** Hiển thị thông tin của phòng khách sạn.
- **Khối RFID:** Đọc thông tin thẻ RFID.
- **Khối truyền nhận RFM69 433MHz:** truyền nhận thông tin giữa mạch chính và mạch phụ thông qua sóng vô tuyến.
- **Khối điều khiển:** Điều khiển động cơ đóng/ mở cửa.
- **Khối máy tính:** là một máy tính chạy hệ điều hành Microsoft Windows. Đóng vai trò là một máy chủ lưu trữ tất cả các thông tin khách hàng trong cơ sở dữ liệu SQL với giao diện viết bằng ngôn ngữ C#.
- **Khối nguồn:** Cung cấp nguồn điện cho mạch phụ.

3.2 MÔ TẢ VÀ THIẾT KẾ PHẦN CỨNG

3.2.1 Mô tả khói vi điều khiển trung tâm

Khối điều khiển trung tâm sử dụng board Arduino ProMini 3.3V 8Mhz như Hình 3.2, là phiên bản sử dụng điện áp 3.3V và xung nhịp 8Mhz, thích hợp cho các ứng dụng sử dụng pin và tiết kiệm năng lượng. Board có 14 chân digital I/O (trong đó có 6 chân có thể được sử dụng làm đầu ra PWM), 8 chân Analog. Board mạch đi kèm không có mạch USB tích hợp, vì vậy phải sử dụng bộ chuyển đổi nối tiếp USB-to-TTL ngoài bo mạch để nạp chương trình.

Gồm các chân chức năng sau:

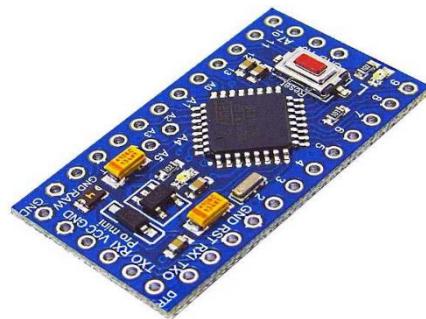
- Chân GND (Ground): cực âm của nguồn điện cấp cho Arduino UNO. Khi dùng các thiết bị sử dụng những nguồn điện riêng biệt thì những chân này phải được nối với nhau.
- Chân VCC: chân cấp điện áp 3.3V. Dòng tối đa cho phép ở chân này là 150mA.
- Chân RAW: để cấp nguồn ngoài cho Arduino trong khoảng 3.35V – 12V, nối cực dương của nguồn với chân này và cực âm của nguồn với chân GND.

Chương 3. TÍNH TOÁN VÀ THIẾT KẾ

- Chân RST: việc nhấn nút Reset trên board để reset vi điều khiển tương đương với việc chân RESET được nối với GND qua 1 điện trở 10KΩ.
- Chân DTR: Dùng cho việc kết nối với việc nạp chương trình cho Arduino ProMini.

Một số chân digital có các chức năng đặc biệt như sau:

- 2 chân Serial (RXI và TXO): dùng để gửi (transmit – TX) và nhận (receive – RX) dữ liệu TTL Serial. Arduino có thể giao tiếp với thiết bị khác thông qua 2 chân này. Đồng thời đây cũng là chân dùng để nạp chương trình cho Arduino Pro MINI
- Chân PWM (3, 5, 6, 9, 10 và 11): các chân này cho phép xuất ra xung PWM với độ phân giải 8bit bằng hàm analogWrite().
- Chân giao tiếp SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). Ngoài các chức năng thông thường, 4 chân này còn dùng để truyền phát dữ liệu bằng giao thức SPI với các thiết bị khác.
- Chân Analog (A0 – A5) cung cấp độ phân giải tín hiệu 10bit để đọc giá trị điện áp trong khoảng 0V – 3.3V. Đặc biệt, chân A4 (SDA) và A5 (SCL) hỗ trợ giao tiếp I2C/TWI với các thiết bị khác.



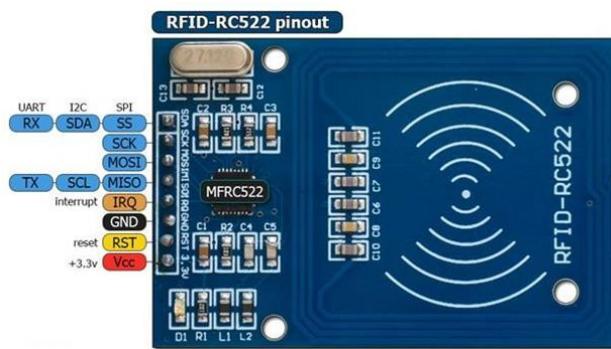
Hình 3.2: Board Arduino ProMini

3.2.2 Mô tả khối RFID

Module RFID RC522 sử dụng IC MFRC522 của Philips được dùng để đọc và ghi dữ liệu cho thẻ NFC tần số 13.56MHz như Hình 3.3, 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.

Chương 3. TÍNH TOÁN VÀ THIẾT KẾ



Hình 3.3: Module RFID RC522

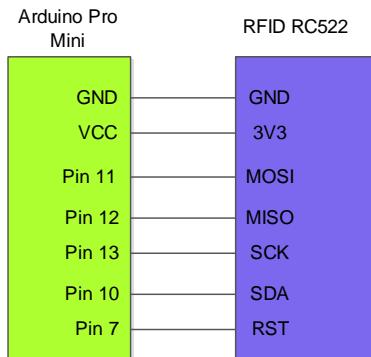
Đặc điểm kỹ thuật:

- Nguồn: 3.3VDC
- Dòng điện hoạt động: từ 13 đến 26mA
- Dòng ở chế độ chờ (Stand by): từ 10 đến 13mA
- Dòng ở chế độ nghỉ (Sleep Mode): nhỏ hơn 80µA
- Dòng tải tối đa: 30mA
- Tần số sóng mang: 13.56MHz
- Khoảng cách đọc tối ưu: từ 0 đến 60mm
- Giao tiếp: SPI
- Tốc độ truyền dữ liệu: tối đa 10Mbit/s
- Nhiệt độ hoạt động: -20 đến 80 °C
- Độ ẩm hoạt động: từ 5 đến 95%
- Các loại thẻ RFID hỗ trợ: mifare1 S50, mifare1 S70, mifare UltraLight, mifare Pro, mifare Desfire

Bộ phát bên trong IC MFRC522 có thể điều khiển anten đầu đọc/ghi được thiết kế để giao tiếp với thẻ và bộ tiếp sóng mà không cần thêm mạch bên ngoài. Ngoài ra các hệ thống lớn còn được kết nối với các máy tính, hạ tầng mạng để truyền nhận thông tin của thẻ tag, thực hiện các tác vụ như liên kết tài khoản, thông tin, tiền phí... Cũng như có các cơ cấu chấp hành để thực thi các yêu cầu đặt ra với hệ thống.

Khối RFID có chức năng đọc thẻ và kích hoạt thẻ để mở cửa cho từng phòng sử dụng module RFID-RC522. Khối RFID bao gồm đầu đọc thẻ RC522 và thẻ MIFARE hoạt động ở mức điện áp 2.5VDC đến 3.3VDC, hỗ trợ chuẩn giao tiếp SPI với khoảng cách đọc từ 0 đến 5cm. Sơ đồ kết nối như Hình 3.4:

Chương 3. TÍNH TOÁN VÀ THIẾT KẾ

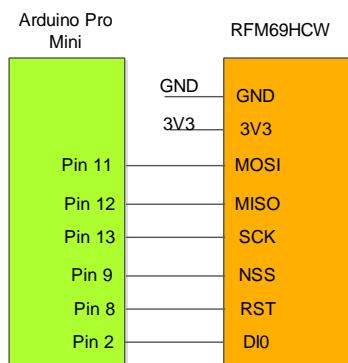


Hình 3.4: Sơ đồ kết nối khối RFID

3.2.3 Mô tả khối truyền nhận

Sử dụng module thu phát RFM69HCW do Sparkfun phân phối, hoạt động ở tần số 433 MHz. Hình 3.6 cho thấy RFM69HCW được kết nối với nền tảng nút điều khiển. Bộ thu phát RFM69HCW hỗ trợ liên kết lên đến vài trăm mét bên ngoài, không có vật cản. Trong nhà, nó có thể hoạt động ở độ sâu khoảng 50 mét, bất chấp các vật cản như nhiều bức tường. Lưu ý rằng phạm vi của đài bị ảnh hưởng bởi tốc độ dữ liệu được chọn và mức công suất đã chọn.

Khối truyền nhận sử dụng Module RFM69HCW là module thu phát hoạt động trên dải tần số 433MHz. Cũng giống RFID, giao tiếp giữa Module RFM69 với Arduino theo chuẩn SPI. Đối với mạch chính, khi thẻ RFID được quét. Mạch chính sẽ kích hoạt thẻ cho phép đóng, mở cửa phòng khi được quét ở mạch phụ điều khiển đóng mở cửa. Thẻ này được cấp cho khách hàng sử dụng.



Hình 3.5: Sơ đồ giao tiếp module RFM69HCW với Arduino Pro Mini

Hình 3.5 là sơ đồ kết nối module RFM69HCW 433MHz với board Arduino Pro Mini. Theo thiết kế khi module RFM69 ở nhận dữ liệu module có mức tiêu tiêu dòng điện lên đến 130mA nhưng dòng ra của chân VCC của Arduino Pro MINI chỉ 150mA.

Chương 3. TÍNH TOÁN VÀ THIẾT KẾ

Chính vì vậy sử dụng nguồn ngoài 3.3V để đảm bảo module hoạt động tốt khi mạch có nhiều ngoại vi được kết nối.

3.2.4 Mô tả khối hiển thị

LCD có rất nhiều dạng phân biệt theo kích thước từ vài kí tự đến hàng chục kí tự, từ 1 hàng đến vài hàng. Ví dụ LCD 16×2 có nghĩa là có 2 hàng, mỗi hàng có 16 kí tự. LCD 20×4 có nghĩa là có 4 hàng, mỗi hàng có 20 kí tự.



Hình 3.6: Mặt trước của LCD 16x2

Hình 3.6 là màn hình LCD 16x2 có 16 chân với các chân tín hiệu như sau:

- Các chân cấp nguồn: Chân số 1 là chân nối mass (0V), chân thứ 2 là Vdd nối với nguồn +5V. Chân thứ 3 dùng để chỉnh contrast thường nối với biến trở, chỉnh cho đến khi thấy được kí tự thì ngừng.
- Các chân điều khiển: Chân số 4 là chân RS dùng để điều khiển lựa chọn thanh ghi. Chân R/W dùng để điều khiển quá trình đọc và ghi. Chân E là chân cho phép dạng xung chốt.
- Các chân dữ liệu D7-D0: Chân số 7 đến chân số 14 là 8 chân dùng để trao đổi dữ liệu giữa thiết bị điều khiển và LCD.
- Các chân LED_A và LED_K: Chân số 15, 16 là 2 chân dùng để cấp nguồn cho đèn nền để có thể nhìn thấy vào ban đêm.

Bảng 3.1: Các chân LCD và chức năng

Chân số	Tên chân	Input/Output	Chức năng tín hiệu
1	VSS	Power	GND
2	VDD	Power	+5V
3	VO	Analog	Contrast Control

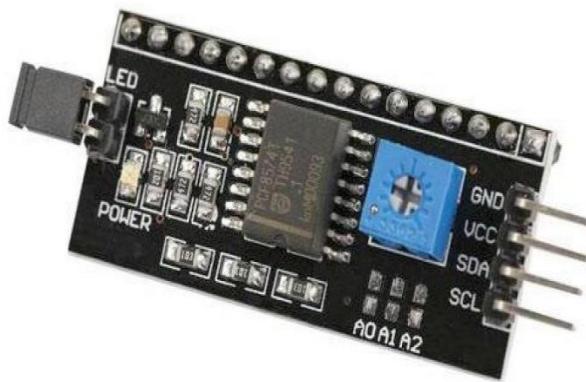
Chương 3. TÍNH TOÁN VÀ THIẾT KẾ

4	RS	Input	Register Select. H: data signal, L: instruction signal
5	R/W	Input	Read/Write. H: read mode, L: write mode
6	E	Input	Enable (strobe)
7	D0	I/O	Data (LSB)
8	D1	I/O	Data
9	D2	I/O	Data
10	D3	I/O	Data
11	D4	I/O	Data
12	D5	I/O	Data
13	D6	I/O	Data
14	D7	I/O	Data (MSB)
15	LED_A	Input	Backlight Anode
16	LED_K	Input	Backlight Cathode

Bảng 3.1 trình bày các ký hiệu và chức năng các chân của LCD. Vì LCD có quá nhiều chân (16 chân) nên gây khó khăn trong quá trình kết nối và chiếm dụng nhiều chân của vi điều khiển. Để khắc phục tình trạng đó thì ta có thể sử dụng module chuyển đổi I2C cho LCD.

Thay vì sử dụng tối thiểu 6 chân của vi điều khiển để kết nối với LCD (RS, EN, D7, D6, D5 và D4) thì với module chuyển đổi, chỉ cần sử dụng 2 chân (SCL, SDA) để kết nối. Module chuyển đổi I2C hỗ trợ các loại LCD sử dụng driver HD44780(LCD1602, LCD 2004, ...), kết nối với vi điều khiển thông qua giao tiếp I2C, tương thích với hầu hết các vi điều khiển hiện nay.

Ưu điểm của phương pháp này là giảm số chân kết nối ngoại vi của vi điều khiển, gọn nhẹ và dễ dàng gắn vào LCD. Bên cạnh đó, nó còn giúp điều chỉnh độ tương phản màn hình, điều khiển đèn nền và thay đổi địa chỉ của LCD mà không cần thay đổi nhiều khi lập trình

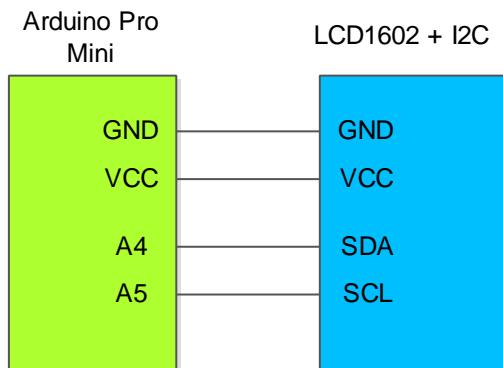


Hình 3.7: Mạch chuyển đổi hỗ trợ giao tiếp I2C cho LCD

Hình 3.7 là mạch chuyển đổi hỗ trợ giao tiếp I2C cho LCD với các đặc điểm kỹ thuật sau:

- Điện áp hoạt động: 2.5 đến 6V DC
- Hỗ trợ màn hình: LCD1602, 1604, 2004 (driver HD44780)
- Giao tiếp: I2C
- Địa chỉ mặc định: 0X27 (có thể điều chỉnh bằng ngắn mạch chân A0/A1/A2)
- Tích hợp Jump chốt để cung cấp đèn cho LCD hoặc ngắt
- Tích hợp biến trơ xoay điều chỉnh độ tương phản cho LCD

Khối sử dụng LCD16x2 để hiển thị các thông tin, thông báo. LCD kết hợp module I2C thông qua chuẩn giao tiếp I2C rút ngắn được số chân còn 2 chân SDA, SCL. Hai chân SDA và SCL kết nối với chân A4 và A5 của Arduino được mô tả như Hình 3.8.



Hình 3.8: Sơ đồ Giao tiếp giữa Arduino ProMini với LCD16x2 I2C

3.2.5 Mô tả khói báo động

Chương 3. TÍNH TOÁN VÀ THIẾT KẾ

Khối báo động dùng chuông Buzzer kết nối với chân 3 của Arduino. Đồng thời cũng thông báo khi quét thẻ.



Hình 3.9: Module buzzer

Hình 3.9 là module buzzer đã có sẵn chân cắm với các đặc điểm kĩ thuật sau:

- Điện áp làm việc: 3-5 VDC
- Tần số âm thanh: 1.5- 2. 5kHz
- Trở kháng: 8Ω

3.2.6 Mô tả khói điều khiển

Động cơ RC Servo 9G là động phô biến dùng trong các mô hình điều khiển nhỏ và đơn giản như cánh tay robot. Động cơ có tốc độ phản ứng nhanh, được tích hợp sẵn Driver điều khiển động cơ, dễ dàng điều khiển góc quay bằng phương pháp điều độ rộng xung PWM.



Hình 3.10: Động cơ Servo SG90

Hình 3.10 là động cơ RC Servo SG90 với các đặc điểm kĩ thuật:

- Điện áp hoạt động: 4.2-6V
- Nhiệt độ: 0 °C - 55 °C
- Tốc độ: 0.3s / 60°.

Chương 3. TÍNH TOÁN VÀ THIẾT KẾ

Khối điều khiển sử dụng 1 động cơ servo SG90 để mở khóa cửa tự động khi có tín hiệu khóa từ khói điều khiển trung tâm Arduino Pro Mini. Động cơ được kết nối với chân số 5 của Arduino làm chân điều khiển, nhận nguồn nuôi 5V từ ngoài.

3.2.7 Mô tả khói nguồn

Các mức điện áp của các linh kiện sử dụng trong mạch là Module RC522 hoạt động ở mức điện áp 3.3VDC, dòng tiêu thụ từ 13-26mA. Module Arduino Pro Mini sử dụng ở mức điện áp 3.3-12VDC dòng định mức 500mA, dòng ra tối đa trên các chân GPIO là 40mA, từ chân VCC là 150mA. Module I2C sử dụng mức điện áp từ 2.7-5.5VDC dòng định mức 60mA. Module RFM69 sử dụng mức điện áp 1.8-3.6VDC dòng định mức 130mA. Module Servo SG90 hoạt động ở mức điện áp 5V, dòng tiêu thụ là 50mA. Dòng tiêu thụ của chuông buzzer ở 3-5V là <25mA.

Bảng 3.2: Thông số dòng điện và điện áp của các linh kiện trong mạch phụ

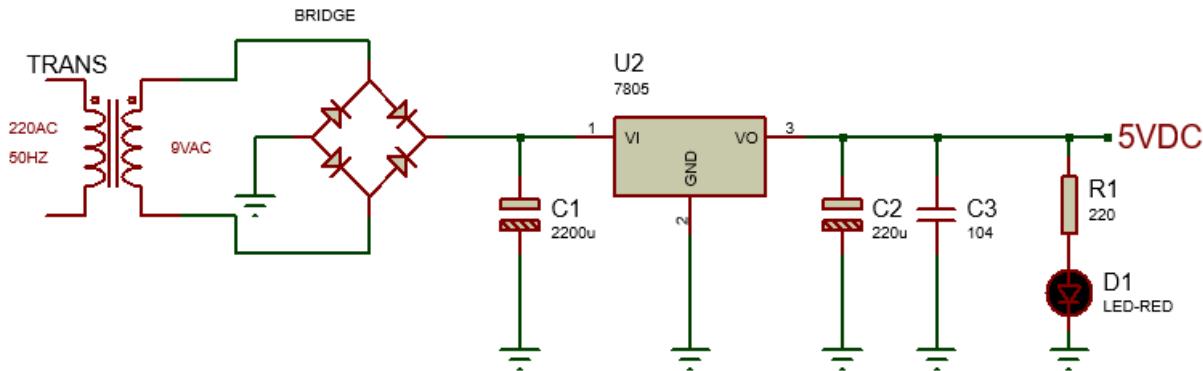
STT	Tên linh kiện	Dòng định mức(mA)	Mức điện áp(V)	Ghi chú
1	Arduino Pro Mini 3.3v-8MHz	500	5V	5V từ mạch nguồn
2	RFID RC522	26	3.3	VCC của Arduino
3	RFM69HCW	130	3.3	3.3V từ mạch nguồn
4	I2C + LCD 16x2	60	5	VCC của Arduino
5	Servo SG90	50	5	5V từ mạch nguồn
6	Buzzer	25	5	PWM của Arduino

Bảng 3.2 trình bày thông số điện áp và dòng điện của các linh kiện sử dụng. Từ các thông tin đã được trình bày như trên dòng tổng của mạch sẽ được tính bằng công thức:

$$\begin{aligned} I_T &= I_{RFID} + I_{I2C} + I_{RFM69} + I_{Arduino} + I_{Servo} + I_{Buzzer} \\ &= 500 + 26 + 130 + 60 + 50 + 25 = 791 \text{ (mA)} \end{aligned} \quad (3.1)$$

3.2.8 Thiết kế mạch nguồn

Sau khi tìm hiểu và tính toán mức điện áp và dòng tiêu thụ của các linh kiện trong mạch thì nhóm quyết định thiết kế nguồn DC 5V-1A để cung cấp cho mạch điều khiển.



Hình 3.11: Sơ đồ nguyên lý mạch nguồn 5V

Hình 3.11 là sơ đồ nguyên lý mạch nguồn 5VDC. Trong mạch có các linh kiện được lựa chọn và tính toán như sau:

- Biến thế 1A dùng để biến đổi điện áp xoay chiều 220VAC cấp bên cuộn dây sơ cấp sang 9VAC cuộn thứ cấp cho phù hợp với tải
- Chọn mạch chỉnh lưu: Ta có thể chọn diode rời hoặc cầu diode tích hợp để làm nhiệm vụ biến đổi điện áp xoay chiều thành điện áp một chiều ở đây chọn 4 diode 1N4007 tạo thành cầu diode. Điện áp một chiều chính là điện áp trung bình sau khi chỉnh lưu được xác định theo công thức:

$$V_d = V_{AC} \times \sqrt{2} = 9 \times \sqrt{2} = 12,73 \text{ V} \quad (3.2)$$

Trong đó: V_d là điện áp trung bình sau khi chỉnh lưu

V_{AC} là điện áp sau khi chỉnh lưu từ máy biến áp.

- Chọn tụ lọc nguồn: Tụ điện có chức năng tích trữ năng lượng điện trong nó lên nó làm mịn điện áp một chiều. Nếu không có tụ điện thì điện áp ở hai đầu ra của mạch chỉnh lưu tuy đã là một chiều nhưng vẫn còn nhấp nhô, không bằng như điện áp một chiều của pin hay ắc quy. Để làm được điện áp một chiều ổn định thì người ta phải mắc thêm tụ điện này. Khi chọn tụ điện cần quan tâm đến hai thông số đó là điện áp định mức và điện dung của nó. Giả sử một biến biến áp có điện áp U_2 là 9V thì điện áp đỉnh trên hai đầu dây của nó sẽ được tính theo công thức (3.2). Vì thế khi chọn tụ lọc nguồn ta phải chọn tụ có điện áp định mức lớn hơn điện áp đỉnh rơi trên hai đầu U_2 của biến áp. Còn trị số điện dung thì

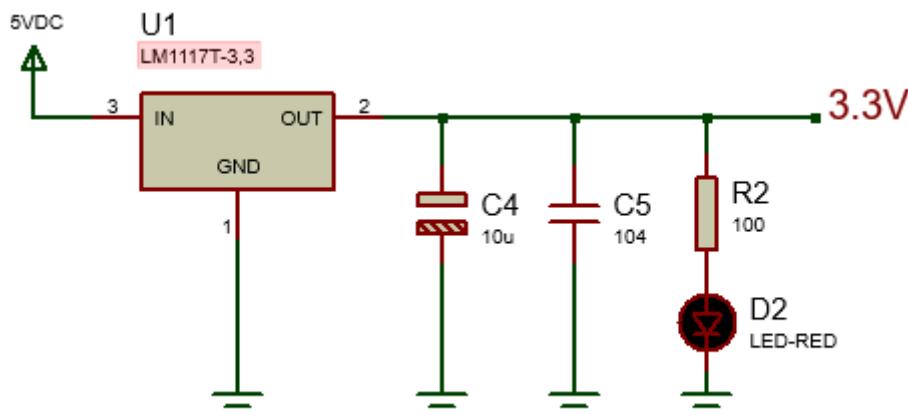
Chương 3. TÍNH TOÁN VÀ THIẾT KẾ

chọn càng lớn càng lọc nguồn tốt nhưng ngược lại sẽ tốn tiền. Với biến áp có đầu ra 9VAC thì chọn tụ lọc nguồn có điện áp định mức lớn hơn điện áp định thì ta chọn tụ C1 có điện áp định mức 2200uF 16V.

- Chọn IC ổn áp: theo nhu cầu sử dụng của mạch thì ta chọn IC ổn áp 7805, IC này cho mức điện áp ngõ ra cố định là 5VDC với dòng điện tối đa là 1A.
- Tụ C2, C3 dùng để lọc điện áp cấp cho tải với C2 cung cấp điện áp tạm thời khi sụt áp và C3 là tụ gồm có trở kháng lớn dùng lọc nhiễu điện áp đầu ra.
- Đèn Led báo nguồn 5VDC: Dùng Led đỏ có điện áp từ 2 - 2.2V với cường độ dòng điện qua Led là 15mA. Điện trở R₁ được tính như công thức (3.3). Chọn R₁ = 220Ω.

$$R_1 = \frac{U_{Nguồn} - U_{Led}}{I_{Led}} = \frac{5 - 2}{0.015} = 200 \Omega \quad (3.3)$$

IC ổn áp LM1117 3.3v với dòng tối đa 800mA được sử dụng để tạo nguồn 3.3V từ mạch nguồn 5V.

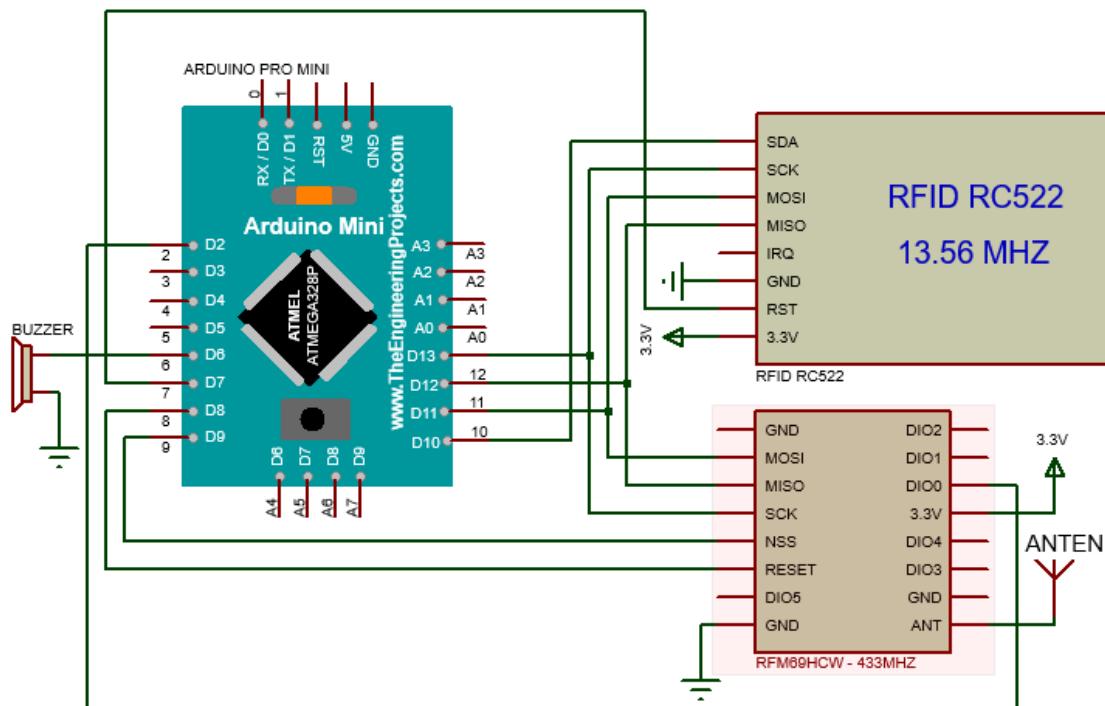


Hình 3.12: Sơ đồ nguyên lý mạch nguồn 3.3V

Hình 3.12 là sơ đồ nguyên lý mạch nguồn 3.3v với tụ C4, C5 dùng để lọc điện áp cấp cho tải với C4 cung cấp điện áp tạm thời khi sụt áp và C5 là tụ gồm có trở kháng lớn dùng lọc nhiễu điện áp đầu ra. Đèn Led báo nguồn 3.3VDC: Dùng Led đỏ có điện áp từ 2 - 2.2V với cường độ dòng điện qua Led là 15mA. Điện trở R₂ được tính như công thức (3.4). Chọn R₂ = 100Ω.

$$R_2 = \frac{U_{Nguồn} - U_{Led}}{I_{Led}} = \frac{3.3 - 2}{0.015} = 86 \Omega \quad (3.4)$$

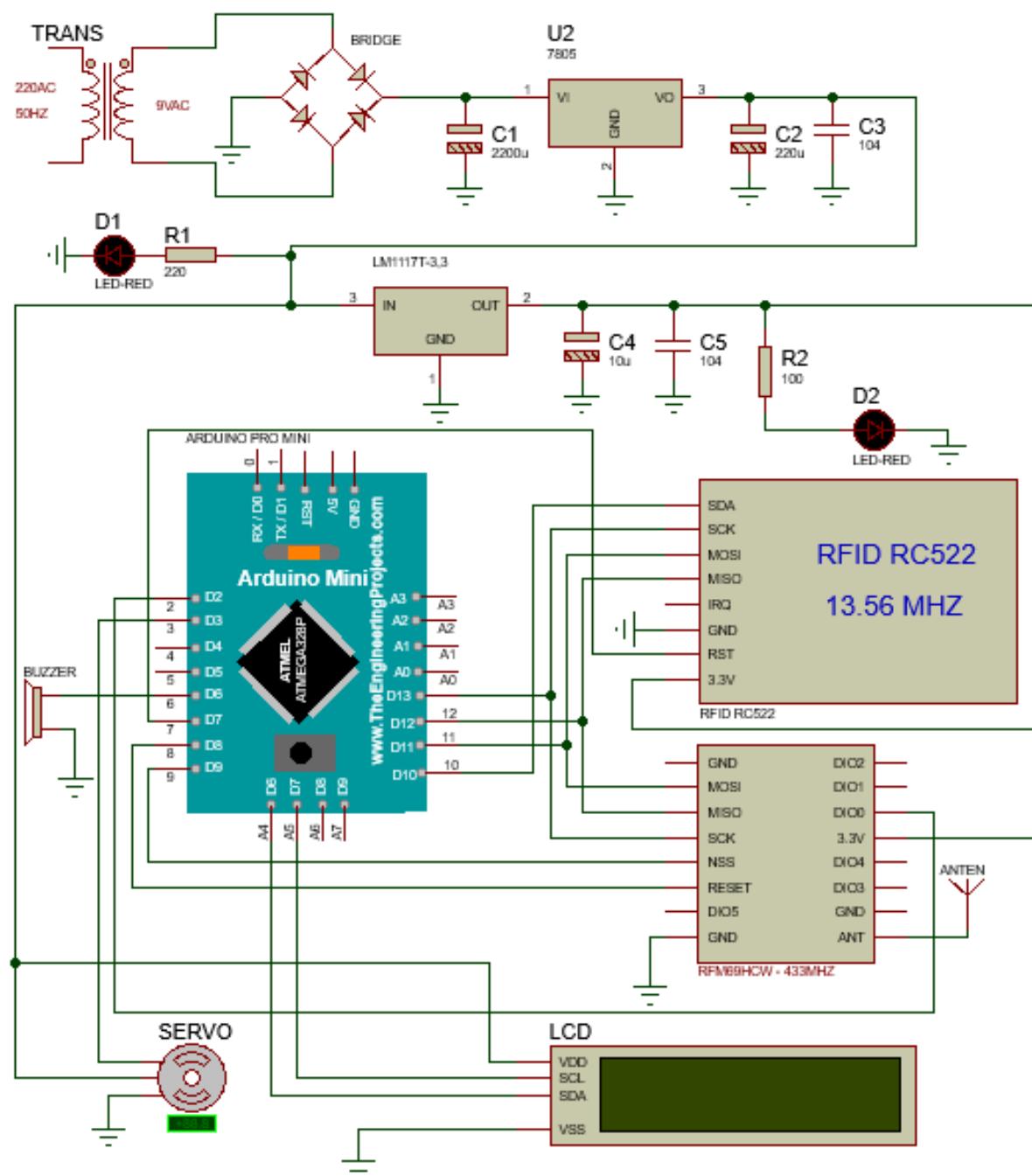
3.3 SƠ ĐỒ NGUYÊN LÝ



Hình 3.13: Sơ đồ nguyên lý mạch chính

Hình 3.13 là sơ đồ nguyên lý mạch chính gồm 1 Arduino Pro Mini kết nối với 3 ngoại vi: buzzer, module RFID RC522 và module RFM69HCW 433MHz. Mạch chính kết nối với máy tính thông qua cáp USB để thực hiện lấy mã ID để lưu trữ thông tin khách và kích hoạt ID phòng cho khách thông qua module truyền phát. Hình 3.15 là sơ đồ nguyên lý của mạch chính.

Hình 3.14 là sơ đồ nguyên lý của mạch điều khiển mở cửa. Mạch gồm module RFID RC522, module RFM69HCW 433MHz, buzzer, servo SG90 thông qua vi điều khiển Arduino Pro Mini điều khiển để mở cửa và truyền nhận thông tin với mạch chính. Mạch nguồn 5V và 3.3V cấp nguồn cho mạch điều khiển hoạt động.



Hình 3.14: Sơ đồ nguyên lý mạch điều khiển hệ thống
đóng – mở cửa phòng khách sạn

3.4 NGUYỄN LÝ HOẠT ĐỘNG

Cấp nguồn cho mạch điều khiển, kết nối mạch chính với máy tính và giao diện quản lý thông tin khách hàng.

Mỗi mạch điều khiển ứng với mỗi phòng có 3 thẻ ID để mở cửa gồm: thẻ Master, thẻ phụ và thẻ khách. Thẻ Master và thẻ phụ có thể mở cửa bất cứ khi nào, thẻ khách chỉ khi kích hoạt mới mở được cửa. Nếu mã thẻ không đúng thì buzzer sẽ kêu báo động.

Chương 3. TÍNH TOÁN VÀ THIẾT KẾ

Mỗi khi có thẻ mở cửa thì mạch sẽ gửi xác nhận thẻ mở đến mạch chính để hiện thị cửa phòng đã được mở cửa thẻ nào để quản lý ra vào.

Với thẻ khách, để có thẻ mở được cửa cần phải thông qua mạch chính kích hoạt mới có thẻ mở được cửa. Dùng thẻ khách quét thông tin thẻ ở mạch chính, lưu thông tin khách hàng sau đó kích hoạt thẻ. Khi đó thẻ ứng với phòng nào sẽ gửi thông tin kích hoạt thẻ khách cho phòng đó thông qua module truyền phát RFM69. Lúc này thẻ khách đã có thẻ mở được cửa ứng với phòng đã được kích hoạt. Khi thẻ này quét 1 lần nữa tại mạch chính để kích hoạt khoá phòng thì thẻ này sẽ không còn mở được cửa phòng được nữa.

Trong quá trình thực hiện, mỗi khi thẻ quét sẽ có tiếng bip thông báo. Lịch sử thẻ mở cửa của 1 phòng sẽ lưu lại khi có thẻ mở cửa cho đến khi có thẻ khách của phòng đó kích hoạt. Lịch sử này sẽ được lưu lại cùng với thời gian mở cửa.

Chương 4. THI CÔNG HỆ THỐNG

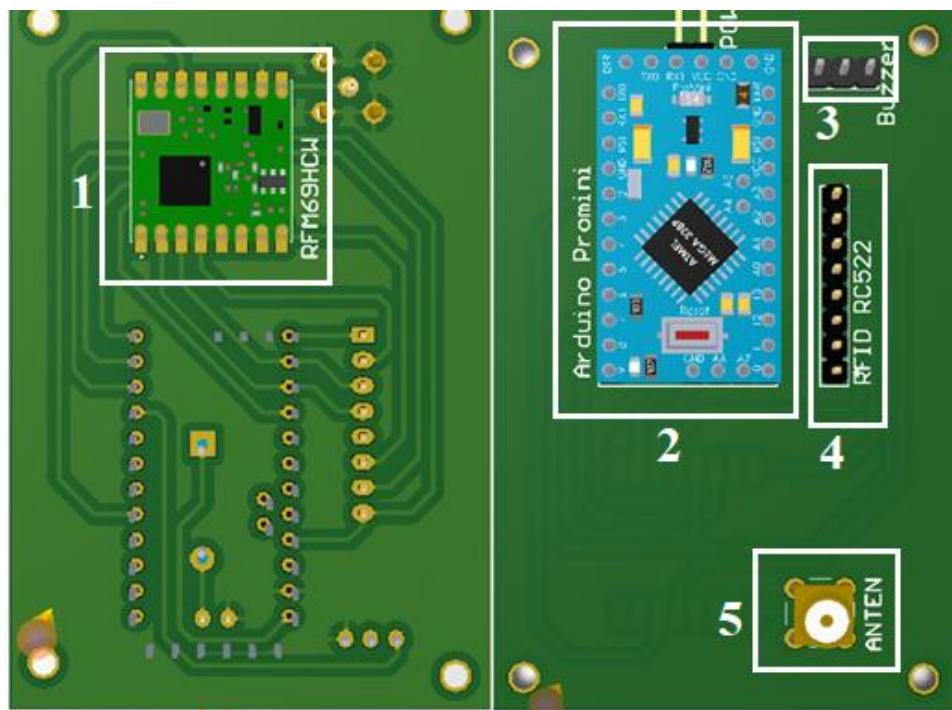
4.1 THI CÔNG HỆ THỐNG

Với đề tài “Thiết kế và thi công hệ thống hỗ trợ mở cửa ứng dụng công nghệ RFID và Arduino tại khách sạn”, nhóm thiết kế bao gồm: Giao diện quản lý thông tin và mô hình hệ thống hỗ trợ mở cửa dùng RFID. Mô hình bao gồm 1 mạch chính gửi tín hiệu kích hoạt phòng đồng thời nhận phản hồi mở cửa từ mạch điều khiển, mạch điều khiển nhận tín hiệu kích hoạt thẻ khách để mở cửa cùng với 2 thẻ đã được kích hoạt trước gồm 3 cửa.

4.1.1 Thi công board mạch

Arduino Pro MINI cắm trên board một lớp và kết nối các module khác bằng cách thông qua các hàng rào đã được sắp xếp và kết nối tương ứng với chân các module.

Với mạch chính nguồn sẽ được cấp thông qua cáp USB để thực hiện giao tiếp với giao diện trên máy tính. Mạch điều khiển sẽ được cấp nguồn bởi một mạch hạ áp tự thiết kế từ 220VAC xuống 5VDC và 3.3VDC cấp cho mạch hoạt động.



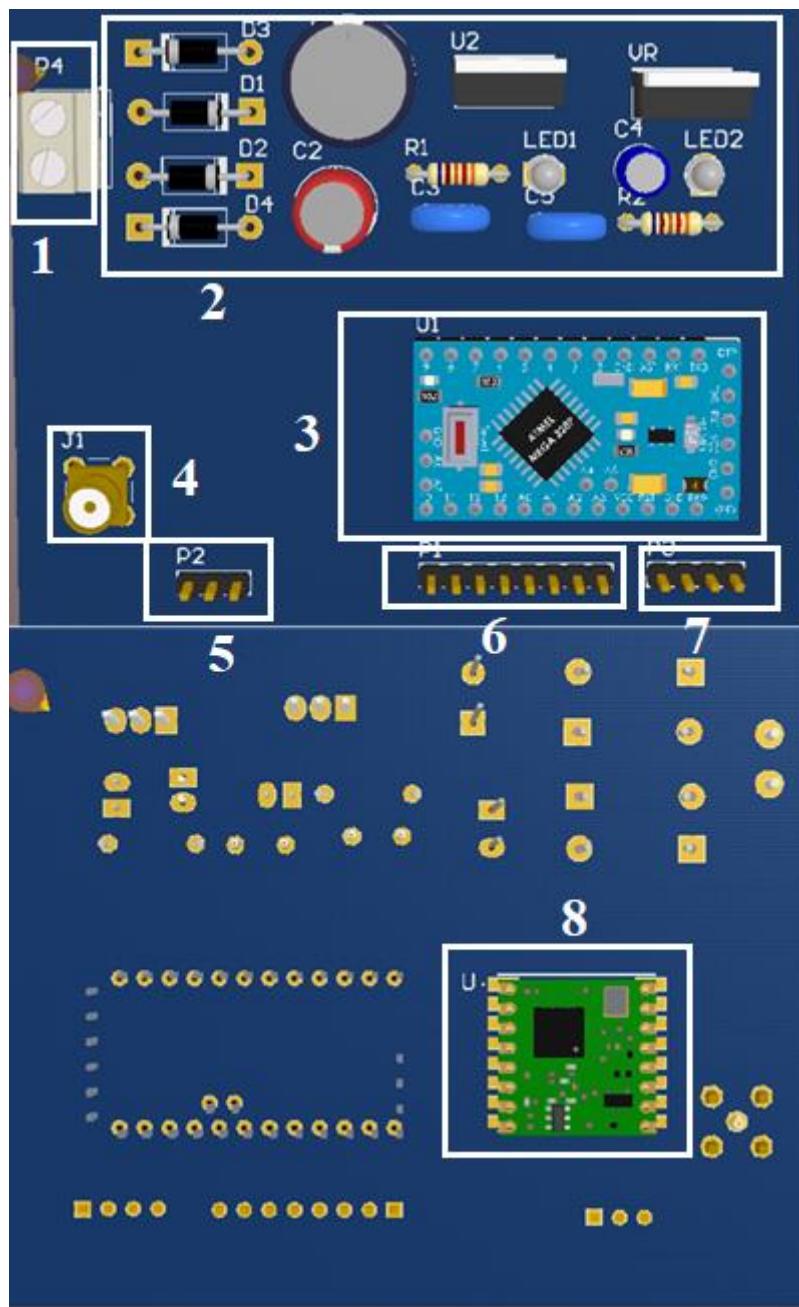
Hình 4.1: Hình 3D mạch chính

Hình 4.1 là hình 3D mặt trước (bên phải) và mặt sau (bên trái) của board mạch chính được vẽ bằng phần mềm Altium 17 để điều khiển kích hoạt thẻ khách và giao tiếp máy tính lưu trữ thông tin trên máy chủ gồm các khối:

- Khối 1: Module truyền nhận RFM69HCW

Chương 4. THI CÔNG HỆ THỐNG

- Khối 2: Board vi điều khiển Arduino ProMini
- Khối 3: Chân kết nối module buzzer
- Khối 4: Chân kết nối module RFID RC522
- Khối 5: Anten



Hình 4.2: Hình 3D mạch điều khiển

Hình 4.2 là mạch 3D của mạch điều khiển gồm các khối:

- Khối 1: Chân cắm nguồn 9VAC từ biến thế cho mạch ổn áp 5VDC
- Khối 2: Mạch ổn áp
- Khối 3: Board vi điều khiển Arduino ProMini

Chương 4. THI CÔNG HỆ THỐNG

- Khối 4: Anten
- Khối 5: Chân kết nối động cơ Servo
- Khối 6: Chân kết nối module RFID RC522
- Khối 7: Chân kết nối module buzzer
- Khối 8: Module truyền nhận RFM69HCW

4.1.2 Thi công mô hình

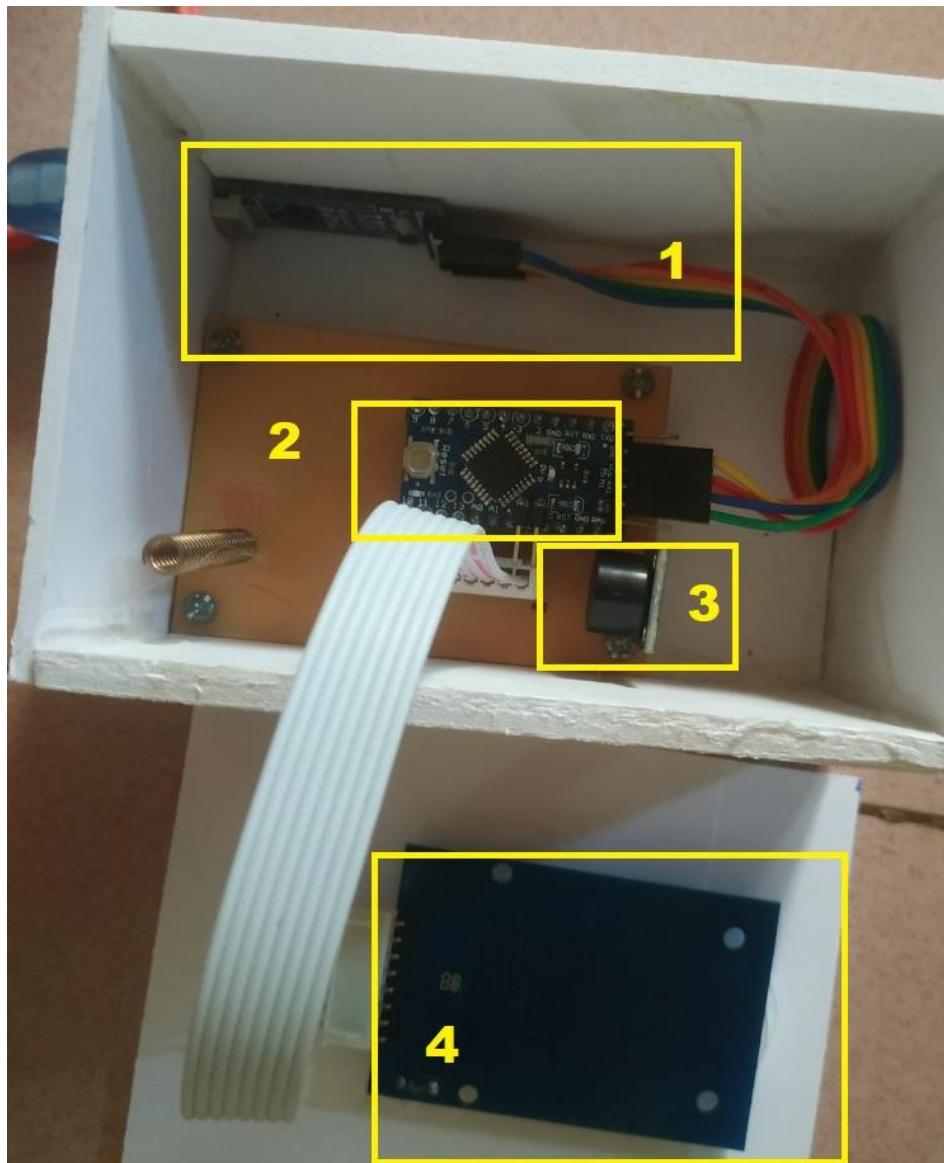
Tổng quan về mô hình:

- Mô hình gồm một mô hình mạch chính để kích hoạt, xử lý gói tin đưa lên Serial và ba mô hình mạch điều khiển cho ba cửa.
- Mô hình mạch chính kết nối với giao diện máy tính thông qua cáp USB.
- LCD 16x2 kết hợp module I2C được sử dụng để hiển thị thông tin với mỗi cửa.
- Buzzer dùng để thông báo đã quét thẻ và cảnh báo khi sai mã thẻ.
- Cửa được đóng/mở sử dụng động cơ Servo SG90 5V.
- Module RFID là module RFID NFC 13.56MHZ RC522 cùng với hai loại thẻ tag được dùng là thẻ nhựa Mifare cho thẻ khách và thẻ master, thẻ móc khoá cho thẻ phụ.
- Board Arduino Pro MINI 8MHz 3.3V cùng module thu phát RFM69HCW được gắn trên board thi công gắn bên trong mô hình. Với mạch điều khiển có thêm một biến áp bên trong mô hình dùng để hạ áp từ 220VAC xuống 9VAC thông qua mạch chỉnh lưu để tạo nguồn 5VDC cấp cho mạch điều khiển.

➤ Mô hình mạch chính

Mô hình mạch chính giao tiếp với máy tính thông qua cáp USB được mô tả như Hình 4.3, gồm các khối:

- Khối 1: Mạch chuyển USB UART CP2102 giao tiếp với máy tính
- Khối 2: Board vi điều khiển Arduino ProMini
- Khối 3: Module buzzer
- Khối 4: Module RFID RC522



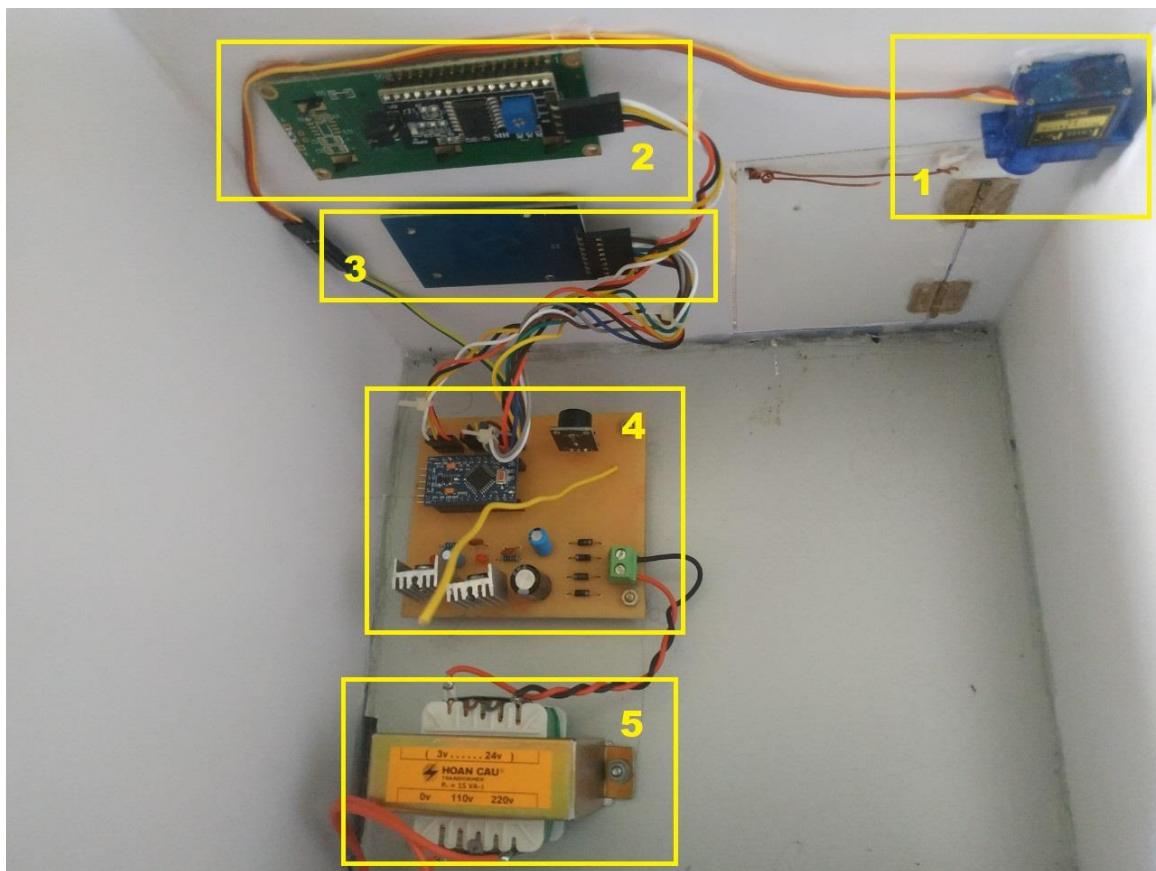
Hình 4.3: Mô hình mạch chính

➤ Mô hình mạch điều khiển

Mô hình mạch điều khiển đóng/mở cửa được mô tả như Hình 4.4 với các khối:

- Khối 1: Động cơ Servo SG90
- Khối 2: Mạch giao tiếp I2C với LCD
- Khối 3: Module RFID RC522
- Khối 4: Board mạch điều khiển
- Khối 5: Biến áp

Chương 4. THI CÔNG HỆ THỐNG



Hình 4.4: Vị trí mạch điều khiển



Hình 4.5: Mặt ngoài mô hình mạch điều khiển

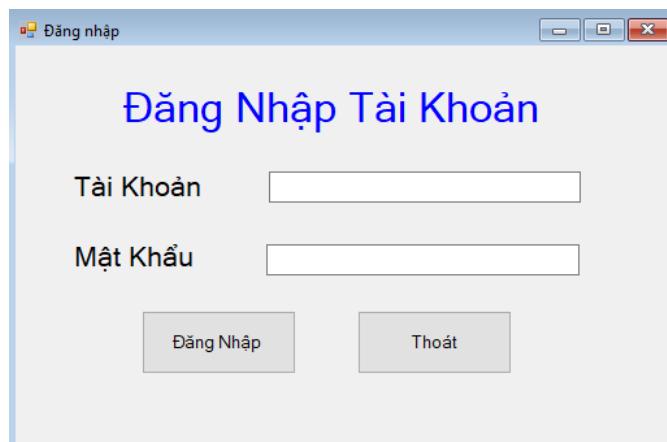
Chương 4. THI CÔNG HỆ THỐNG

Hình 4.5 là mặt ngoài của mô hình mạch điều khiển gồm có màn hình LCD 16x2 ở vị trí 1, vị trí quét thẻ tag ở vị trí 2 và cánh cửa ở vị trí 3. Có ba mô hình mạch điều khiển, các mô hình đều giống nhau chỉ khác địa chỉ cầu hình.

4.1.3 Thi công giao diện

Giao diện của đề tài sẽ chỉ dùng để mô phỏng cho mô hình hỗ trợ mở cửa phòng khách sạn vì vậy giao diện sẽ không đi sâu vào chi tiết giao diện quản lý khách sạn. Giao diện sẽ chỉ thể hiện quá trình lưu trữ thông tin khách đặt phòng, kích hoạt phòng và trả phòng dùng RFID và lưu trữ lịch sử hoạt động đóng/mở cửa của phòng.

Hình 4.6 là giao diện đăng nhập để phân quyền cho các chức năng bên trong giao diện chính. Việc thêm tài khoản sẽ do quản lý thực hiện, sẽ được hướng dẫn ở mục sau.



Hình 4.6: Giao diện đăng nhập

a)

Chương 4. THI CÔNG HỆ THỐNG

Quản lý

Thông tin phòng Thuê phòng Khách thuê phòng Lịch sử thẻ hoạt động

Tìm Phòng Tim

Danh sách phòng

	TÊN PHÒNG	LOAI PHÒNG	GIÁ PHÒNG	TÌNH TRẠNG	KÍCH HOẠT

Đặt phòng Thuê phòng Huỷ phòng

Nhập ID phòng Kích hoạt thẻ

Trả phòng

Thông tin phòng

- Tên Phòng: Null
- Loại Phòng: Null
- Giá Phòng: Null
- Tình Trạng: Null
- ID thẻ: Null

b)

Thông tin Phòng

Họ Tên Giới Tính

Ngày Sinh

Địa Chỉ

CMND Điện Thoại

Ngày Đến

Tìm kiếm

Tìm theo mục

Tìm theo

Sửa thông tin Xoá thông tin

TÊN PHÒNG	HỌ TÊN	GIỚI TÍNH	NGÀY SINH	ĐỊA CHỈ	CMND	ĐIỆN THOẠI	NGÀY ĐẾN	NGÀY ĐỎ

c)

Thông tin phòng Thuê phòng Khách thuê phòng Lịch sử thẻ hoạt động

THỜI GIAN	HOẠT ĐỘNG

d)

Hình 4.7: Giao diện quản lý

- Giao diện thông tin các phòng
- Giao diện đặt phòng
- Giao diện thông tin khách
- Giao diện lịch đóng/mở cửa

Hình 4.7 là giao diện chính bao gồm bốn phần: Hình 4.7a là giao diện hiển thị thông tin các phòng và có thể chỉnh sửa với phân quyền “Quản lý”; Hình 4.7b là giao

Chương 4. THI CÔNG HỆ THỐNG

diện dùng để đặt phòng cho khách và thực hiện kích hoạt phòng, trả phòng cho khách; Hình 4.7c là giao diện hiển thị thông tin khách đã thuê phòng cũng các chức năng tìm kiếm; Hình 4.7d là giao diện hiển thị lịch sử đóng mở cửa, có thể lọc dữ liệu theo phòng. Dữ liệu lịch sử sẽ được làm mới khi có khách thuê phòng đó và sẽ bắt đầu một lịch sử mới.

Đăng ký thông tin

Điền thông tin
Phòng

Họ Tên: _____ Giới Tính: _____

Ngày Sinh: 11/ 1/2021

Địa Chỉ: _____

CMND: _____ Điện Thoại: _____

Ngày Đến: 11/01/2021 10:59

Xác nhận Huỷ

Hình 4.8: Giao diện đăng ký thông tin khách

Thanh toán

Thông tin phòng

Tên phòng | Loại phòng | Giá phòng

Thông tin khách hàng

Họ tên | Giới tính | Ngày sinh | Địa chỉ | CMND | Điện thoại | Ngày đến

Ngày đi: 11/01/2021 10:59 | Số Ngày: _____

Thành Tiền: _____

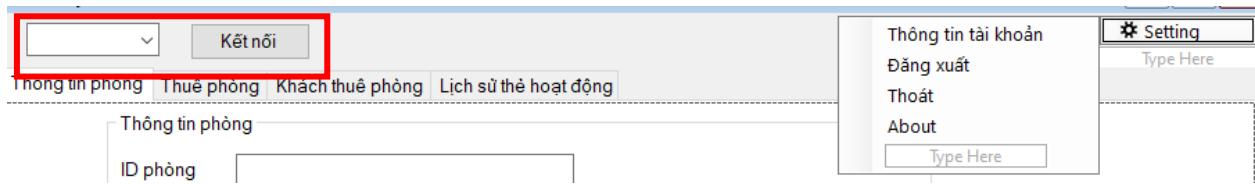
Thanh Toán Huỷ

Hình 4.9: Giao diện thanh toán

Khi có khách thuê phòng sẽ sử dụng giao diện Hình 4.8 để thực hiện đăng ký thông tin cho khách theo các mục trên hình. Khi có khách trả phòng sẽ sử dụng giao diện Hình 4.9 để thực hiện trả phòng. Đây chỉ là giao diện mô phỏng cho việc mở cửa bằng RFID,

Chương 4. THI CÔNG HỆ THỐNG

lưu trữ thông tin khách cũng không thể đi sâu nên chỉ thực hiện thanh toán bằng việc tính số ngày ở và tính tiền cho khách. Giá phòng cũng chỉ mang tính chất tượng trưng.



Hình 4.10: Các chức năng khác

Bên cạnh các phần liên quan khách và phòng thì còn có phần kết nối giao diện phần mềm với mạch chính trong Hình 4.10 phần khung đó. Góc trên bên phải có phần “Setting” gồm “Thông tin tài khoản” dùng để xem thông tin tài khoản đang dùng và chỉnh sửa cũng như thêm tài khoản với phân quyền “Quản lý” như Hình 4.11, “Đăng xuất” thoát tài khoản hiện tại, “Thoát” dùng để thoát khỏi phần mềm, “About” chỉ là hiển thị thông tin liên quan người hướng dẫn và sinh viên thực hiện.

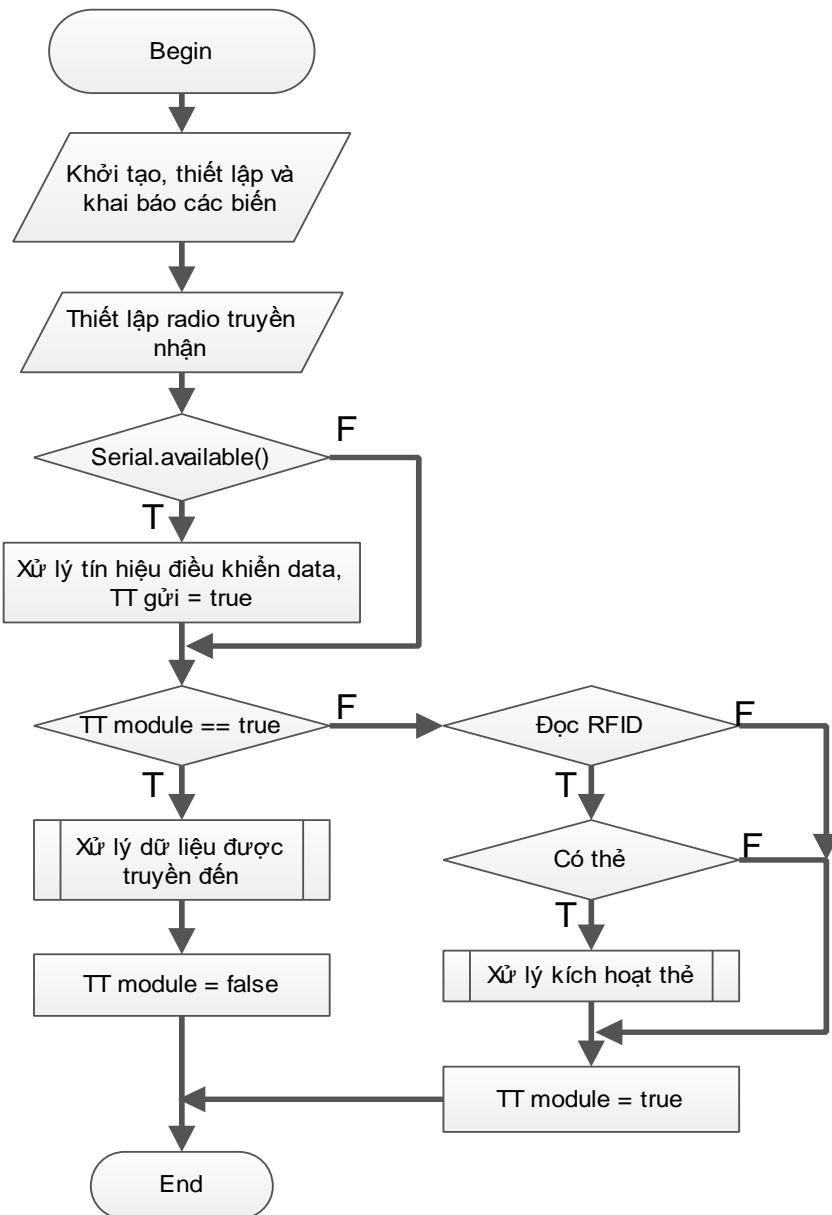
A screenshot of a Windows-style application window titled 'Tài khoản'. The title bar includes standard minimize, maximize, and close buttons. The main area is titled 'Thông tin tài khoản' and contains a table with four columns: STT, Họ Tên, Chức Vụ, and Tài Khoản. The table has three rows. The first row is selected, showing '1' in STT, 'Văn Hoàng' in Họ Tên, 'Quản lý' in Chức Vụ, and 'hoang' in Tài Khoản. The second row shows '3' in STT, 'Khôi' in Họ Tên, 'Nhân viên' in Chức Vụ, and 'khoi' in Tài Khoản. Below the table are four input fields: 'Họ Tên' (Last Name), 'Chức Vụ' (Position), 'Tài Khoản' (Account), and 'Mật Khẩu' (Password). To the right of these fields are four buttons: 'Thêm' (Add), 'Sửa' (Edit), and 'Xoá' (Delete).

Hình 4.11: Giao diện thông tin cá nhân

4.2 LẬP TRÌNH HỆ THỐNG

4.2.1 Lưu đồ giải thuật mạch chính

Chương 4. THI CÔNG HỆ THỐNG



Hình 4.12: Lưu đồ giải thuật mạch chính

Mạch chính có chức năng lấy mã thẻ RFID để thực hiện lưu thông tin khách trên phần mềm lưu trữ thông tin đồng thời lấy mã đó kích hoạt phòng tương ứng mã đó. Khi phòng được mở cửa mạch chính sẽ nhận thông tin thẻ đã mở cửa và truyền đến phần mềm để ghi lại lịch sử hoạt động mở cửa phòng. Hình 4.12 là lưu đồ giải thuật cho mạch chính với cách hoạt động như sau:

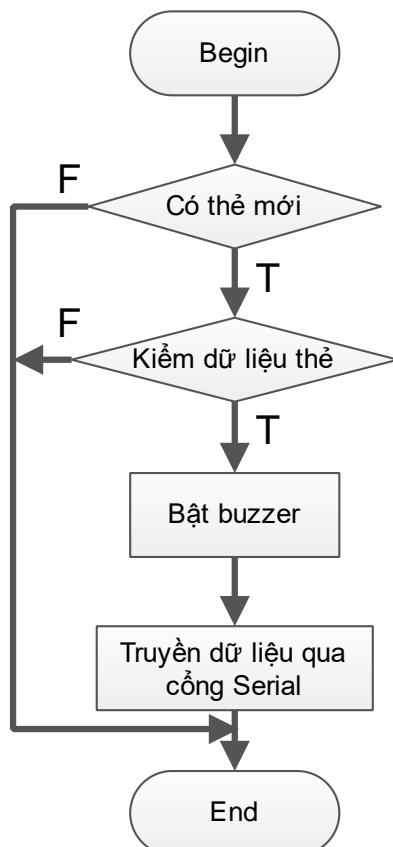
- Chương trình bắt đầu sẽ khởi tạo các biến, khởi tạo các module được sử dụng, cấu hình cho bộ thu phát RFM69HCW.
- Ban đầu sẽ kiểm tra xem có dữ liệu truyền thông qua cổng Serial hay không. Nếu có chương trình sẽ xử lý dữ liệu đó để phục vụ cho quá trình kích hoạt phía

Chương 4. THI CÔNG HỆ THỐNG

sau rồi tiếp tục sang quá trình tiếp theo. Nếu không có dữ liệu trên cổng Serial chương trình cũng sẽ sang quá trình tiếp theo.

- Sau khi kiểm tra dữ liệu trên Serial, chương trình sẽ tiếp tục với một biến cho phép chạy module “ TT module”. Nếu là true thì chương trình sẽ kiểm tra gói tin được bộ thu phát từ các mạch điều khiển gửi về để tạo lịch sử hoạt động cho cửa phòng. Nếu là false chương trình sẽ kiểm tra việc đọc thẻ tag. Biến cho phép module sẽ thay đổi trạng thái sau khi đã kiểm tra module một khoảng thời gian.
- Sau quá trình kiểm tra thẻ tag, nếu thẻ được đọc hợp lệ sẽ tiến hành truyền lên Serial để phần mềm nhận và tiếp hành lưu thông tin, kích hoạt thẻ và phản hồi về vi điều khiển, vi điều khiển thông qua Serial nhận tín hiệu và thông qua bộ thu phát truyền gói tin kích hoạt đến phòng tương ứng.
- Các công việc sẽ được thực hiện tuần tự liên tục từ trên xuống sau đó quay lại từ đầu.

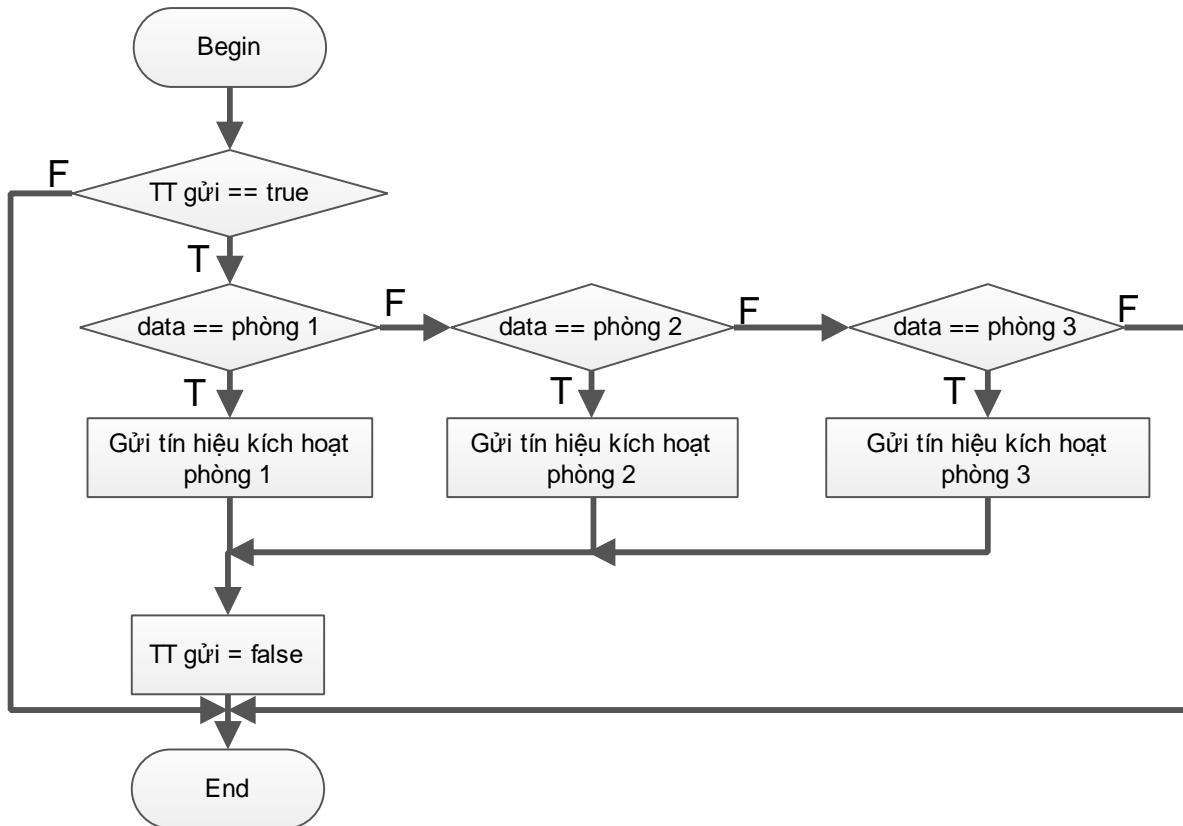
Lưu đồ giải thuật cho đọc mã thẻ tag được thực hiện như Hình 4.13:



Hình 4.13: Lưu đồ giải thuật chương trình con đọc thẻ RFID

Chương 4. THI CÔNG HỆ THỐNG

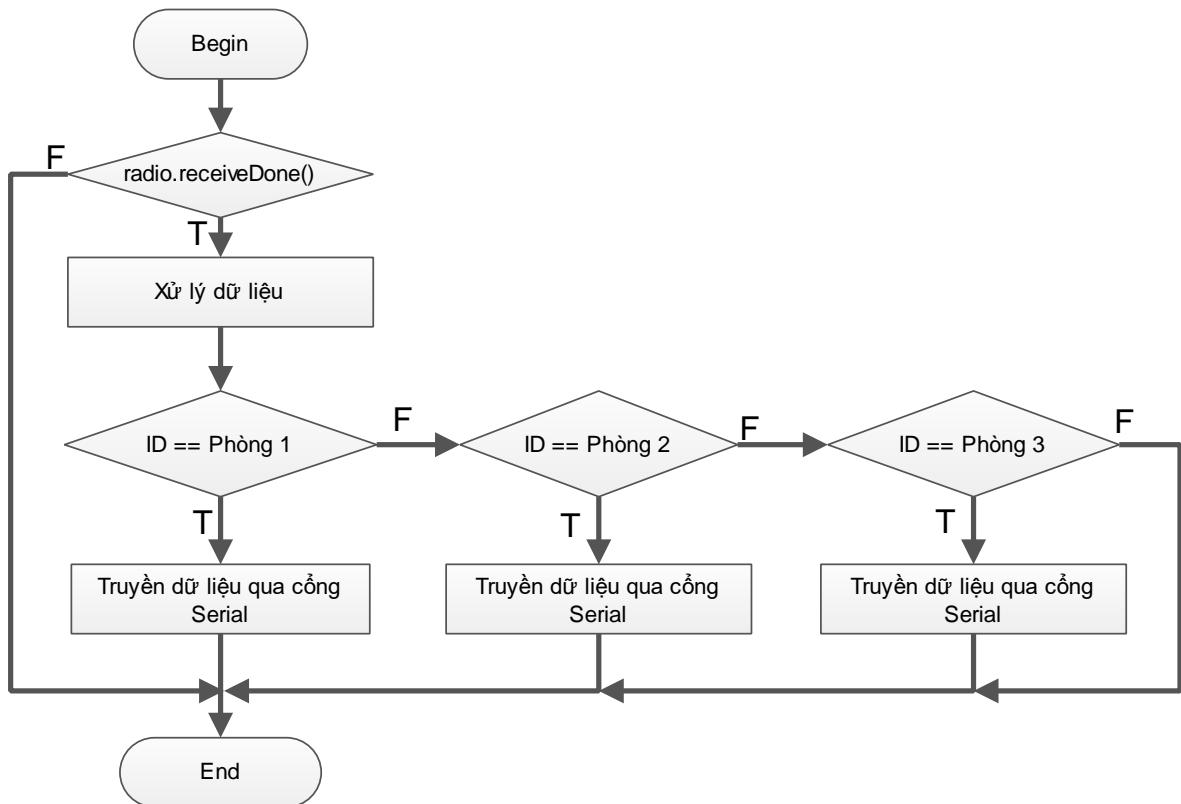
Hoạt động của lưu đồ như sau: Chương trình sẽ kiểm tra xem có thẻ mới được đặt vào hay không. Nếu có chương trình sẽ tiến hành kiểm tra dữ liệu thẻ, bật buzzer sau khi lấy được mã thẻ và truyền lên Serial. Nếu không có thẻ mới hoặc kiểm tra thẻ không hợp lệ chương trình sẽ tiến hành bước tiếp theo.



Hình 4.14: Lưu đồ giải thuật chương trình con xử lý kích hoạt thẻ

Lưu đồ cho việc truyền gói tin kích hoạt thẻ cho phòng tương ứng với mã thẻ tag được trình bày như Hình 4.14 và có hoạt động như sau: Chương trình sẽ kiểm tra “TT gửi”, trạng thái này được cho phép khi có dữ liệu nhận được trên Serial. Nếu trạng thái true, chương trình sẽ tiến hành xử lý dữ liệu nhận được. Nếu trạng thái là false sẽ bỏ qua chương trình này. Dữ liệu nhận được bao gồm 3 mã tương ứng với 3 phòng. Nếu data là dữ liệu phòng 1 thì sẽ tiến hành gửi gói tin bao gồm ID của nơi nhận, dữ liệu gửi là mã tag và chiều dài dữ liệu gửi. Tương tự cho các phòng khác.

Chương 4. THI CÔNG HỆ THỐNG



Hình 4.15: Lưu đồ giải thuật chương trình con xử lý dữ liệu truyền đến

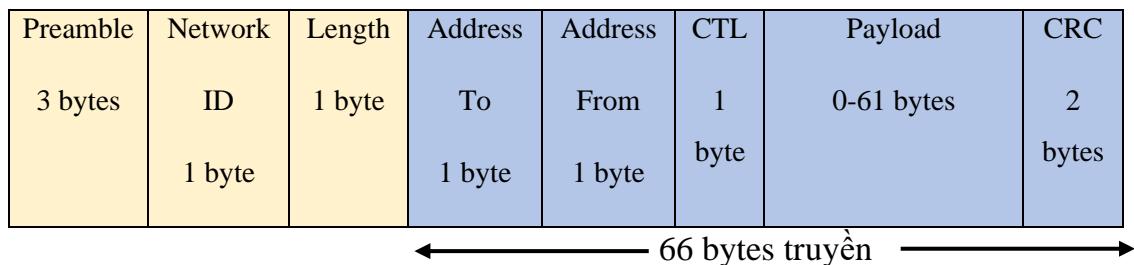
Hình 4.15 là lưu đồ giải thuật cho chương trình xử lý dữ liệu được truyền đến. Chương trình sẽ nhận gói tin từ các phòng gửi về và vì xử lý sẽ xử lý gói tin truyền lên Serial để phần mềm giao diện lưu trữ tạo lịch sử hoạt động cho phòng. Chương trình hoạt động như sau: Bộ thu phát sẽ chờ gói tin được chuyển tới, khi nhận được sẽ tiến hành xử lý xem gói tin được gửi từ phòng nào và tiến hành xử lý dữ liệu tương ứng với phòng đó và truyền lên Serial.

❖ Cấu hình bộ thu phát RFM69HCW

- #define NETWORKID 0: Khai báo ID cho mạng có giá trị từ 0 đến 255.
Các nút giao tiếp với nhau nếu có NETWORKID giống nhau.
- #define MYNODEID 1: Khai báo ID cho nút truyền. MYNODEID này là duy nhất trong mạng, đồng thời đây cũng chính là TONODEID cho các nút khác nếu có dữ liệu gửi lại.
- #define TONODEID 2: khai báo ID sẽ nhận dữ liệu. Nếu truyền dữ liệu cho nhiều nút thì sẽ có nhiều TONODEID. TONODEID chính là các MYNODEID của các nút khác.

Chương 4. THI CÔNG HỆ THỐNG

- `#define FREQUENCY RF69_433MHZ`: Khai báo tần số của module RFM69 sử dụng, module RFM69 có các version tần số phổ biến: 315MHz, 433MHz, 915MHz.
- `#define ENCRYPT true`: Khai báo sử dụng key cho nút. Nút khác muốn nhận được dữ liệu thì phải dùng đúng key. Nếu không muốn key thì có thể thay đổi “true” thành “false”.
- `#define ENCRYPTKEY "DATNBMDTCNYS"`: Khai báo key
- `#define USEACK true`: Khai báo sử dụng phản hồi ACK sau khi gửi dữ liệu đi. Nếu không cần phản hồi từ nút nhận dữ liệu thì chuyển đổi “true” thành “false”.



Hình 4.16: Cấu trúc gói tin bộ thu phát

Hình 4.16 mô tả dữ liệu truyền đi được cấp 66 byte dữ liệu gồm 61 byte cho dữ liệu và năm byte điều khiển: Address to là byte địa chỉ người nhận, Address from byte địa chỉ người gửi và byte điều khiển cho phép mạng đơn giản cấu trúc CTL và hai byte CRC phát hiện lỗi.

4.2.2 Lưu đồ giải thuật mạch điều khiển

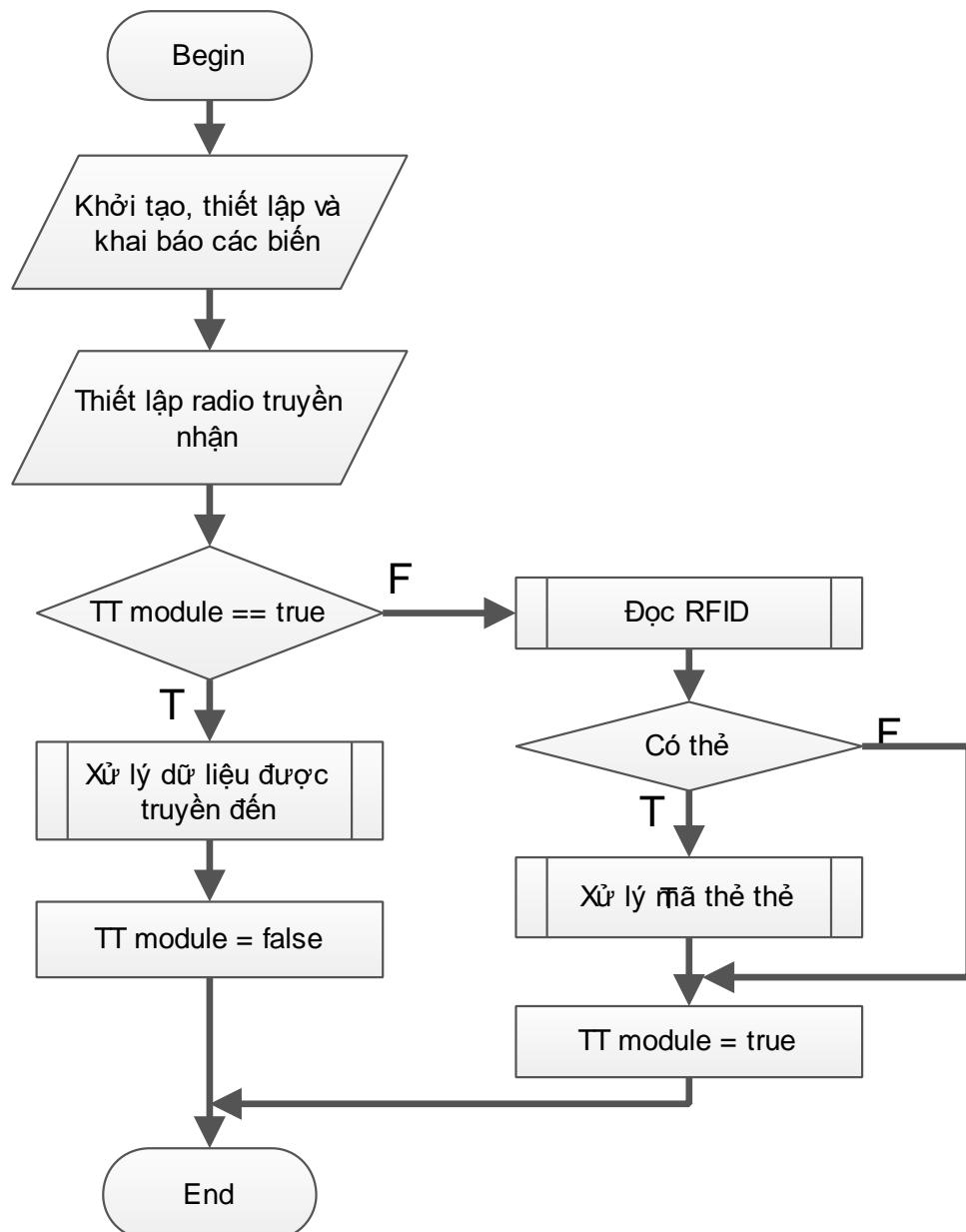
Mô hình đề tài gồm ba mạch điều khiển cho ba cửa phòng hoạt động riêng lẻ. Dưới đây sẽ trình bày lưu đồ chung cho ba mạch này.

Hình 4.17 là lưu đồ chính cho mạch điều khiển gồm các hoạt động sau:

- Chương trình bắt đầu sẽ khởi tạo các biến, khởi tạo các module được sử dụng, cấu hình cho bộ thu phát RFM69HCW.
- Chương trình sẽ kiểm tra biến cho phép chạy module “TT module”. Nếu là true thì chương trình sẽ kiểm tra gói tin được bộ thu phát từ mạch chính gửi đến để kích hoạt thẻ tag khách. Nếu là false chương trình sẽ kiểm tra việc đọc thẻ tag. Biến cho phép module sẽ thay đổi trạng thái sau khi đã kiểm tra module một khoảng thời gian.

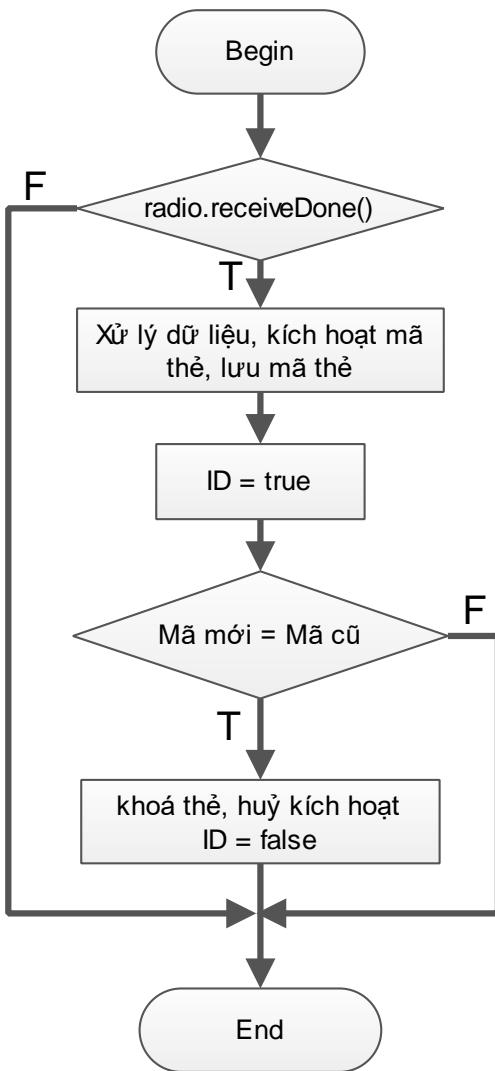
Chương 4. THI CÔNG HỆ THỐNG

- Sau quá trình kiểm tra thẻ tag, nếu thẻ được đọc hợp lệ sẽ tiến hành so sánh với các mã thẻ tag đã được lưu bao gồm mã thẻ khách được kích hoạt khi mạch chính gửi đến nếu không sít không hợp lệ, mã thẻ master là thẻ dùng chung cho cả ba cửa và thẻ phụ ứng với mỗi phòng.
- Các công việc sẽ được thực hiện tuần tự liên tục từ trên xuống sau đó quay lại từ đầu.



Hình 4.17: Lưu đồ giải thuật mạch điều khiển

Chương 4. THI CÔNG HỆ THỐNG

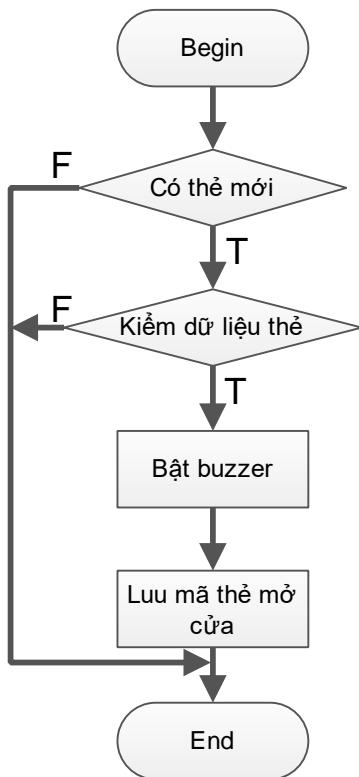


Hình 4.18: Lưu đồ giải thuật chương trình kích hoạt phòng

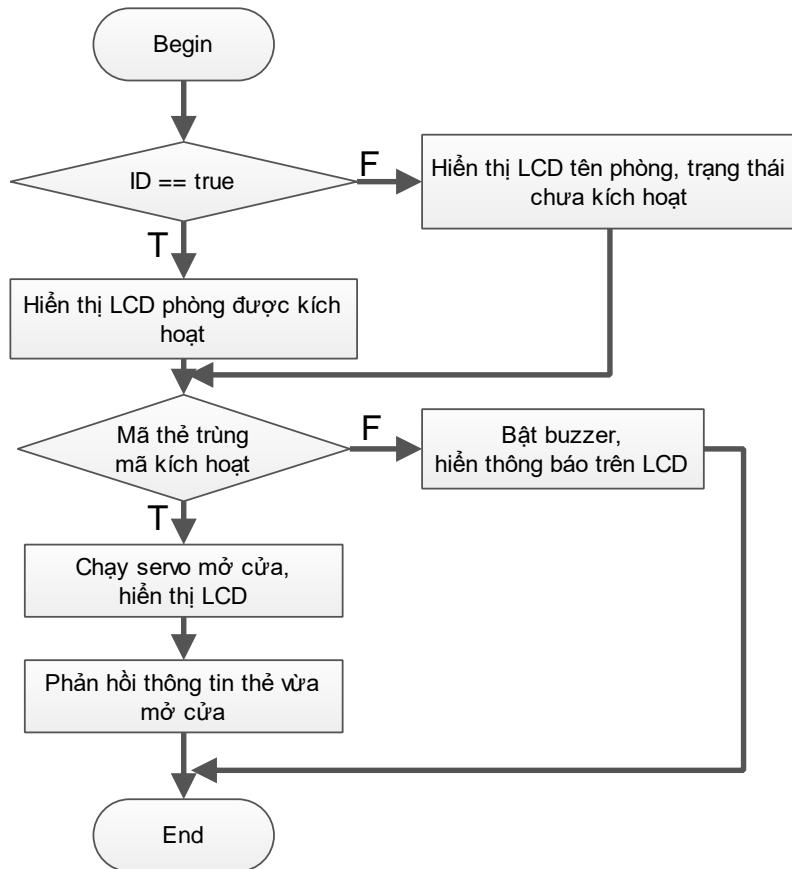
Lưu đồ hình 4.18 là lưu đồ cho chương trình kích hoạt mã thẻ khách với hoạt động như sau: Chương trình sẽ kiểm tra xem có gói tin được gửi đến hay không. Nếu có sẽ tiến hành xử lý lấy mã thẻ khách đồng thời lưu lại để so sánh mở cửa. Bộ thu phát vẫn sẽ tiến hành nhận gói tin, khi có gói tin vi điều khiển sẽ xử lý mã thẻ mới và so sánh với mã khách cũ nếu khớp sẽ tiến hành khoá thẻ này. Đây chính là cơ chế kích hoạt phòng sử dụng trong đề tài.

Hình 4.19 thể hiện lưu đồ cho chương trình đọc mã thẻ để mở cửa phòng với quá trình hoạt động như sau: Chương trình sẽ kiểm tra xem có thẻ mới được đặt vào hay không. Nếu có chương trình sẽ tiến hành kiểm tra dữ liệu thẻ, bật buzzer sau khi lấy được mã thẻ và lưu lại để so sánh. Nếu không có thẻ mới hoặc kiểm tra thẻ không hợp lệ chương trình sẽ tiến hành bước tiếp theo.

Chương 4. THI CÔNG HỆ THỐNG



Hình 4.19: Lưu đồ giải thuật chương trình đọc mã thẻ



Hình 4.20: Lưu đồ giải thuật chương trình mở cửa

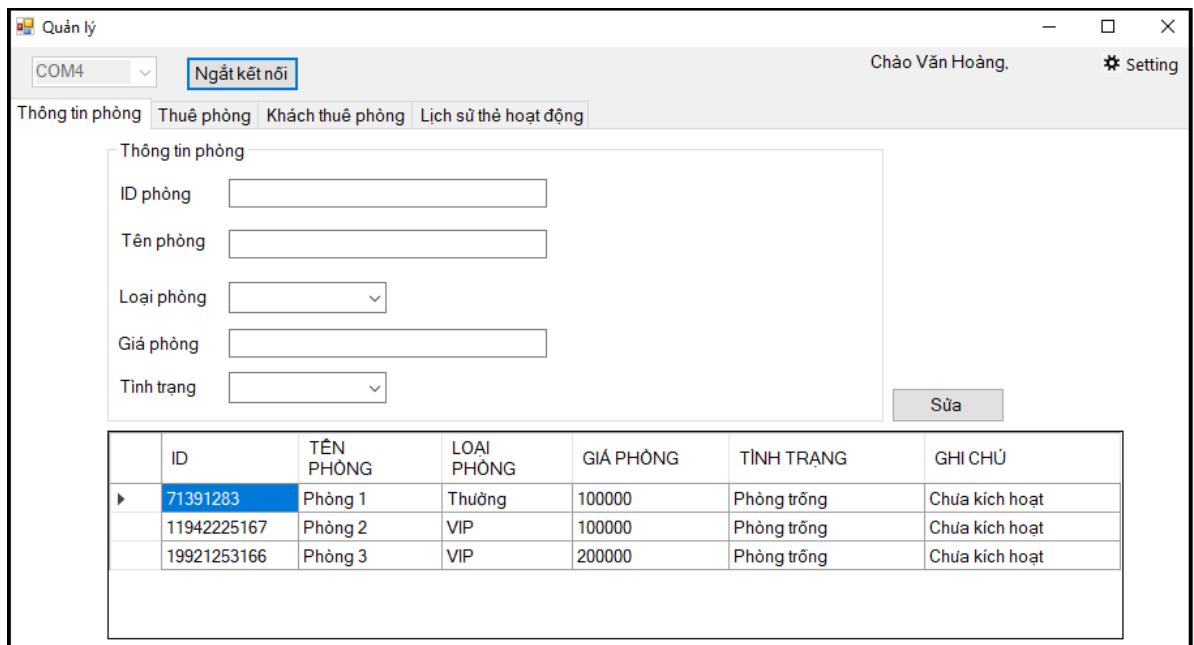
Chương 4. THI CÔNG HỆ THỐNG

Lưu đồ giải thuật cho chương trình mở cửa phòng được thể hiện như Hình 4.20. Màn hình LCD sẽ hiển thị tên phòng và trạng thái tương ứng với trạng thái kích hoạt thẻ khách ID của lưu đồ kích hoạt thẻ: Đã kích hoạt với trạng thái true và false chưa kích hoạt. Mã thẻ nhận được từ quá trình đọc được qua module RFID sẽ được so sánh với các mã được lưu gồm thẻ khách khi được kích hoạt, thẻ master và thẻ phụ. Nếu phù hợp sẽ tiến hành chạy Servo mở cửa và hiển thị thời gian đếm ngược cửa tự động khoá với thời gian 5s sau đó sẽ gửi gói tin thẻ vừa mở cửa thông qua bộ thu phát về mạch chính. Nếu mã thẻ không phù hợp sẽ tiến hành bật buzzer cảnh báo và thông báo trên LCD.

4.3 TÀI LIỆU HƯỚNG DẪN THAO TÁC

Bước 1: Cấp nguồn cho mô hình mạch điều khiển, cắm cáp USB kết nối mạch chính với máy tính có phần mềm giao diện.

Bước 2: Thực hiện đăng nhập vào phần mềm và chọn cổng COM để kết nối với mạch. Kết quả sẽ được như Hình 4.21.



Hình 4.21: Chọn cổng COM kết nối

Bước 3: Các thông tin phòng đã được thêm trước. Tiến hành chuyển sang tab “Thuê phòng”, chọn phòng trống và nhấp chọn “Thuê phòng” để tiến hành đăng ký thông tin. Khi chọn phòng, thông tin phòng sẽ hiển thị ở phần “Thông tin phòng”. Sau đó quét thẻ phòng tương ứng rồi chọn “Kích hoạt thẻ” để kích hoạt thẻ khách cho phòng.

Chương 4. THI CÔNG HỆ THỐNG

	TÊN PHÒNG	LOAI PHÒNG	GIÁ PHÒNG	TÌNH TRẠNG	KÍCH HOẠT
▶	Phòng 1	Thường	100000	Phòng có kh...	Đã kích hoạt
▶	Phòng 2	VIP	100000	Phòng trống	Chưa kích hoạt
	Phòng 3	VIP	200000	Phòng trống	Chưa kích hoạt

Hình 4.22: Thứ tự thuê phòng cho khách

Nếu khách muốn đặt trước thì chọn giống như thuê nhưng thay vì chọn “Thuê phòng” thì chọn “Đặt phòng”. Có thể huỷ phòng đặt trước bằng việc chọn phòng và nhấn “Huỷ phòng”. Với các phòng không thể đặt hay thuê sẽ có thông báo tương ứng.

Bước 4: Với cửa phòng ta có hai thẻ đã được kích hoạt sẵn là thẻ master và thẻ phụ có thẻ mở cửa bất cứ lúc nào. Thẻ khách chỉ mở được khi LCD hiển thị đã kích hoạt. Để mở cửa chỉ cần để thẻ vào đúng vị trí và chờ tiếng buzzer. Sau đó cửa sẽ tự mở và sẽ tự đóng lại sau 5s.

Bước 5: Khi khách muốn trả phòng, chọn tab “Thuê phòng” sau đó quét thẻ phòng đó ở mạch chính, sau đó chọn “Trả phòng” giao diện thanh toán sẽ hiện lên và tự động tính tiền với số ngày được tính với ngày hiện tại. Sau khi nhấn “Thanh toán” phòng sẽ được làm lại, lịch sử vẫn còn cho đến khi có khách mới đăng ký phòng này.

Với giao diện, việc thêm tài khoản, sửa tài khoản, sửa thông tin phòng và xoá thông tin khách chỉ được với phân quyền “Quản lý”, còn lại các chức năng khác đều giống như phân quyền “Nhân viên”. Lần đầu đăng nhập có thể sử dụng tài khoản admin để thực hiện phân quyền (Tài khoản: admin, mật khẩu: admin).

Chương 4. THI CÔNG HỆ THỐNG

4.4 THÔNG TIN GIÁ HỆ THỐNG

Bảng 4.1: Bảng giá linh kiện sử dụng trong hệ thống

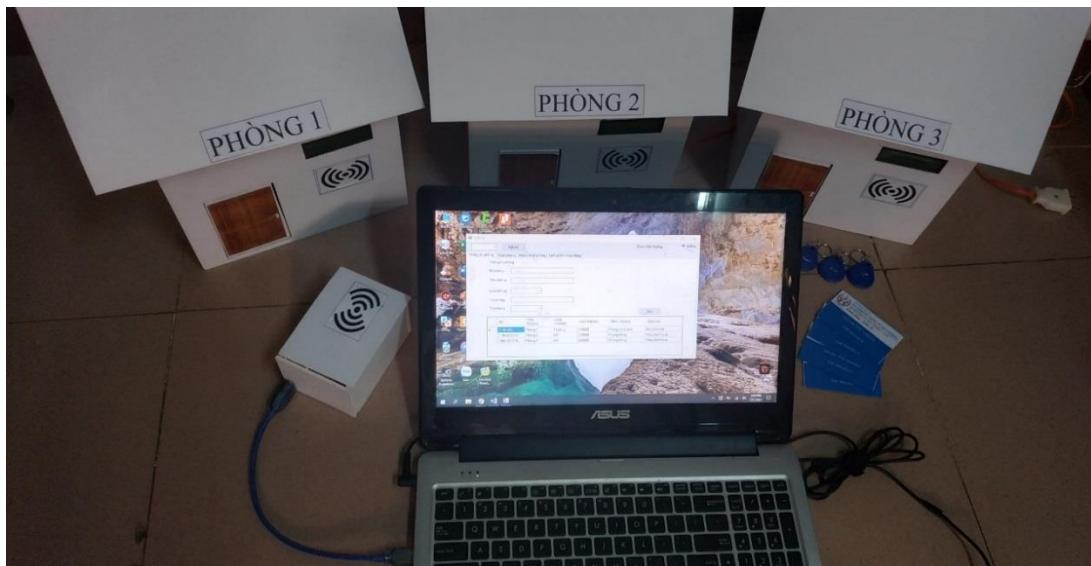
STT	Tên linh kiện	Số lượng (cái)	Đơn giá (VNĐ)	Thành tiền
1	Arduino ProMini 3.3V 8MHZ	4	44,000	176,000
2	RFID RC522	4	28,000	112,000
3	RFM69HCW	4	138,000	552,000
4	LCD 16x2	3	28,000	112,000
5	Mạch giao tiếp I2C cho LCD	3	13,000	52,000
6	Động cơ Servo SG90	3	28,000	112,000
7	Buzzer KY-012	4	10,000	40,000
8	Mạch chuyển đổi USB to UART	1	55,000	55,000
9	Biến áp 1A	3	40,000	120,000
10	IC ổn áp LM7805	3	2,000	6,000
11	IC ổn áp LM1117	3	5,500	16,500
12	Diode 1N4007	12	200	2,400
13	Tụ 2200uF 16V	3	2,000	6,000
14	Tụ 220uF 25V	3	500	1,500
Tổng:				1,363,400

Bảng giá các linh kiện được sử dụng trong hệ thống được mô tả trong Bảng 4.1, trong đó liệt kê tên linh kiện, số lượng và giá thành của linh kiện đó. Tổng giá tiền trong bảng chỉ bao gồm các linh kiện sử dụng trong hệ thống không bao gồm các chi phí thi công mạch và lắp ráp mô hình.

Chương 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ

5.1 MÔ HÌNH HỆ THỐNG

Sau khoảng thời gian nghiên cứu, tìm hiểu và thực hiện đề tài “*Thiết kế và thi công hệ thống hỗ trợ mở cửa ứng dụng công nghệ RFID tại khách sạn dùng Arduino*” nhóm đã học hỏi được thêm rất nhiều kiến thức, biết thêm được những mô hình quản lý hoạt động khách trong thực tế. Hình 5.1 là mô hình thực tế của đề tài đã được nhóm hoàn thành và thu được những kết quả sẽ được trình bày trong chương này.



Hình 5.1: Mô hình thực tế

5.1.1 Mạch điều khiển Phòng 1



Hình 5.2: Màn hình LCD phòng 1 và vị trí quét thẻ

Chương 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ

Khi cấp nguồn cho mô hình mạch điều khiển, màn hình LCD sẽ hiện thị thông tin phòng và tình trạng kích hoạt của phòng lúc này là chưa kích hoạt. Hình 5.2 là màn hình LCD của phòng 1 sau khi cấp nguồn.

- Trường hợp phòng chưa kích hoạt



Hình 5.3: Cửa mở và tự động đóng lại



Hình 5.4: Cửa mở và tự động đóng lại

Khi phòng chưa kích hoạt có hai mã thẻ có thẻ mở được gồm thẻ master và thẻ phụ của mỗi phòng. Khi quét thẻ hợp lệ kết quả sẽ như Hình 5.3 và Hình 5.4. Cửa sẽ từ từ mở ra và sẽ đóng lại sau 5s có đếm ngược thời gian trên màn hình LCD.

- Trường hợp phòng đã được kích hoạt

Chương 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ

Khi phòng đã được kích hoạt, ngoài hai thẻ master và thẻ phụ thì thẻ khách cũng sẽ mở được cửa. Quá trình mở cửa cũng không có gì thay đổi, LCD chỉ thay đổi thông báo tình trạng phòng từ “CHUA KICH HOAT” thành “DA KICH HOAT” như Hình 5.5.



Hình 5.5: Màn hình LCD khi phòng kích hoạt ở phòng 1

- Trường hợp thẻ không hợp lệ

Dù phòng ở trạng thái kích hoạt hay không thì khi quét thẻ tag không hợp lệ, màn hình LCD sẽ thông báo thẻ không hợp lệ như Hình 5.6 và buzzer sẽ kêu tám âm thanh liên tiếp để cảnh báo thẻ không hợp lệ.



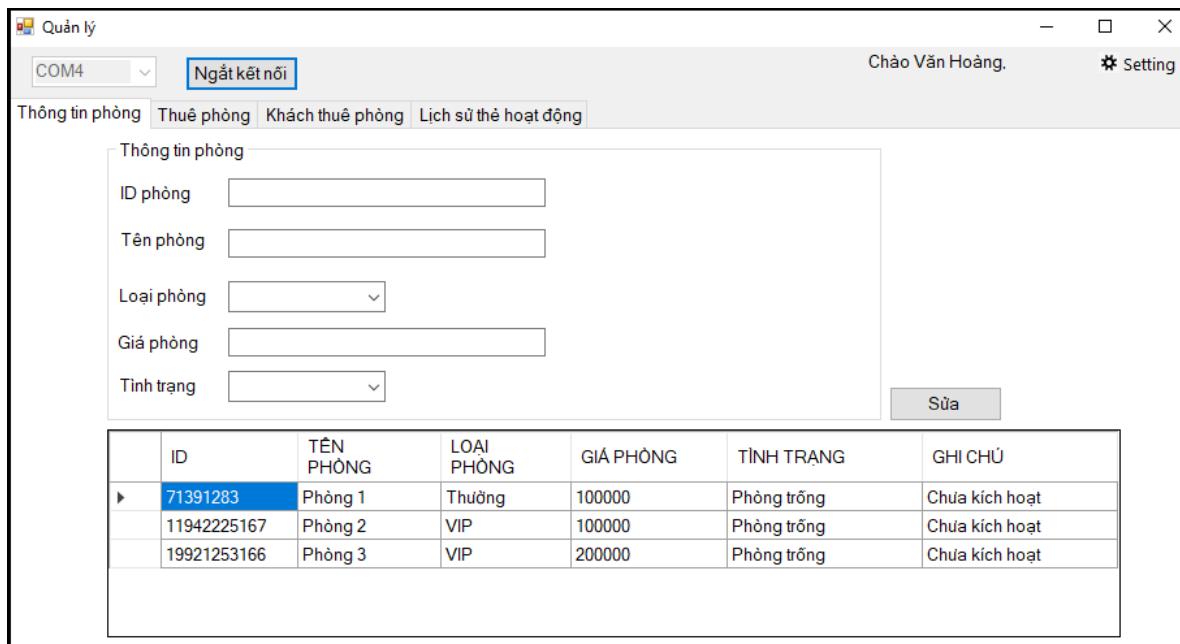
Hình 5.6: Màn hình LCD khi thẻ không hợp lệ

Tương tự với mạch điều khiển phòng 2 và 3, các trường hợp cũng giống như mạch phòng 1. Mỗi phòng sẽ có thẻ phụ và thẻ khách khác nhau và có chung một thẻ master mở cửa được cả ba phòng.

5.1.2 Kết quả hoạt động mạch chính và giao diện

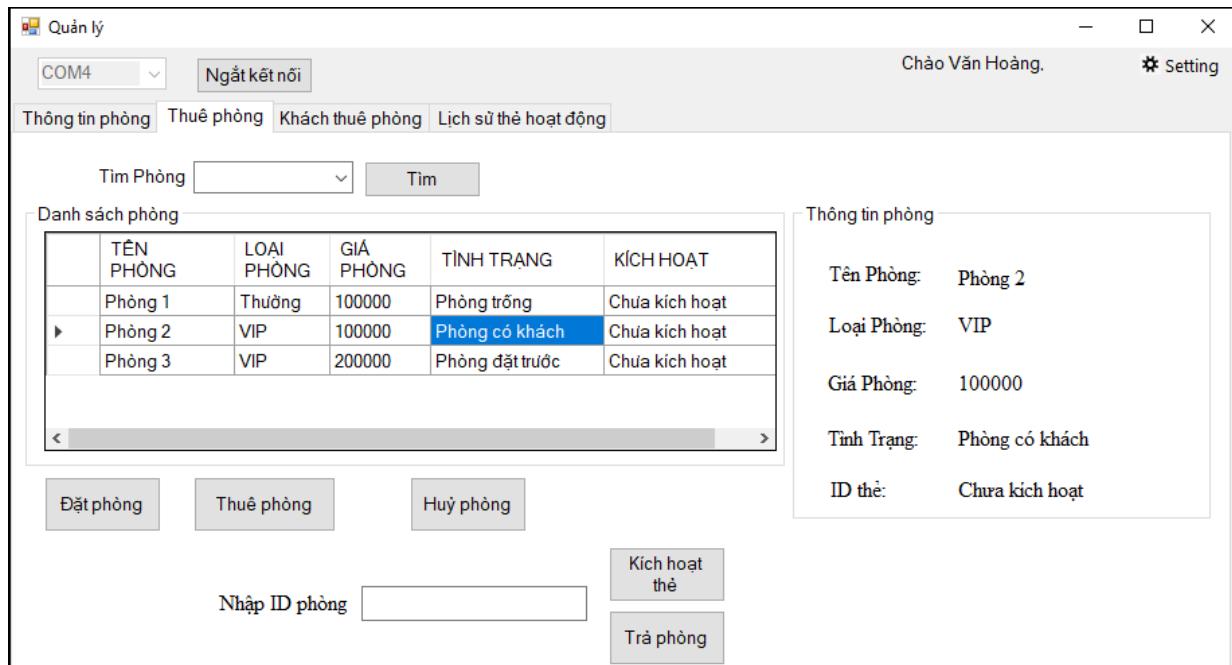
Sau khi kết nối mạch chính với máy tính, sử dụng tài khoản để đăng nhập vào giao diện quản lý và chọn cổng kết nối. Khi sử dụng tài khoản đăng nhập của cá nhân, họ tên sẽ được hiện thị trên giao diện như Hình 5.7.

Chương 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ



Hình 5.7: Màn hình giao diện khi kết nối

Trên màn hình giao diện lúc này sẽ có các thông tin về các phòng đã được lưu sẵn. Các thông tin này có thể được thay đổi nếu phân quyền tài khoản là “Quản lý”. Để xem phân quyền này thực hiện như sau: Chọn *Setting* → *Thông tin tài khoản*. Một form “Tài khoản” hiện lên sẽ có thông tin về tài khoản được đăng nhập bao gồm cả phân quyền.



Hình 5.8: Giao diện thuê phòng

Sau khi chọn phòng và tiến hành đăng ký thông tin khách thì trên giao diện phần “TÌNH TRẠNG” sẽ thay đổi từ “Phòng trống” sang “Phòng có khách” với nhấn “Thuê

Chương 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ

phòng” hoặc sang “Phòng đặt trước” nếu nhấn “Đặt phòng”. Lúc này phòng vẫn chưa được kích hoạt, cần quét mã thẻ của phòng tương ứng sau đó nhấn “Kích hoạt thẻ” phòng mới chuyển trạng thái kích hoạt như Hình 5.9.

	TÊN PHÒNG	LOẠI PHÒNG	GIÁ PHÒNG	TÌNH TRẠNG	KÍCH HOẠT
▶	Phòng 1	Thường	100000	Phòng trống	Chưa kích hoạt
	Phòng 2	VIP	100000	Phòng có khách	Chưa kích hoạt
	Phòng 3	VIP	200000	Phòng đặt trước	Chưa kích hoạt

Đặt phòng Thuê phòng Huỷ phòng

Nhập ID phòng Kích hoạt thẻ
Trả phòng

Hình 5.9: Kích hoạt phòng

Với “Phòng đặt trước” cũng như vậy. Sau khi kích hoạt thẻ, cột “KÍCH HOẠT” sẽ chuyển từ “Chưa kích hoạt” sang “Đã kích hoạt”. Khi khách trả phòng cũng như kích hoạt thẻ chỉ cần quét mã thẻ và nhấn “Trả phòng” thì các trạng thái phòng sẽ trở về như cũ.

TÊN PHÒNG	HỌ TÊN	GIỚI TÍNH	NGÀY SINH	ĐỊA CHỈ	CMND	ĐIỆN THOẠI	NGÀY ĐẾN	NGÀY ĐI	THANH TOÁN
▶ Phòng 2	KH01	Nữ	12/1/2021	hcm	639854217123	065231794	12/01/2021 07...	12/01/2021 07...	Đã thanh toán
Phòng 1	kh1	Nam	12/1/2021	hcm	123456789	0123456789	10/01/2021 08...	12/01/2021 08...	Đã thanh toán
Phòng 2	kh2	Nam	12/1/2021	hn	654789321	0987654321	10/01/2021 08...	12/01/2021 08...	Đã thanh toán

Hình 5.10: Thông tin khách

Chương 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ

Thông tin khách hàng sẽ được lưu trữ lại và có thể chỉnh sửa cũng như tìm kiếm thông tin khách khi cần tại tab “Khách thuê phòng” như Hình 5.10. Lịch sử mở cửa của phòng sẽ được hiển thị tại tab “Lịch sử thẻ hoạt động”. Hình 5.11 là lịch sử mở cửa cửa ba phòng.

THỜI GIAN		HOẠT ĐỘNG
12/01/2021 9:09:50 AM		Phòng 2 được mở bằng ID Khách
12/01/2021 9:09:59 AM		Phòng 3 được mở bằng ID Master
12/01/2021 9:10:06 AM		Phòng 1 được mở bằng ID Khách
12/01/2021 9:10:16 AM		Phòng 2 được mở bằng ID Khách
12/01/2021 9:10:30 AM		Phòng 3 được mở bằng ID Phụ
12/01/2021 9:10:34 AM		Phòng 2 được mở bằng ID Khách
12/01/2021 9:10:40 AM		Phòng 1 được mở bằng ID Khách
12/01/2021 9:10:55 AM		Phòng 3 được mở bằng ID Phụ
12/01/2021 9:11:00 AM		Phòng 2 được mở bằng ID Phụ
12/01/2021 9:11:07 AM		Phòng 1 được mở bằng ID Khách
12/01/2021 9:11:26 AM		Phòng 1 được mở bằng ID Khách
12/01/2021 9:11:33 AM		Phòng 3 được mở bằng ID Phụ
12/01/2021 9:11:40 AM		Phòng 2 được mở bằng ID Khách
12/01/2021 9:11:51 AM		Phòng 3 được mở bằng ID Master
12/01/2021 9:11:57 AM		Phòng 1 được mở bằng ID Khách
12/01/2021 9:12:06 AM		Phòng 2 được mở bằng ID Khách

Hình 5.11: Lịch sử thẻ hoạt động

THỜI GIAN		HOẠT ĐỘNG	Phòng 3
▶	12/01/2021 08:43:01	Phòng 3 được kích hoạt	
	12/01/2021 8:43:52 AM	Phòng 3 được mở bằng ID Khách	
	12/01/2021 8:44:08 AM	Phòng 3 được mở bằng ID Khách	
	12/01/2021 8:45:09 AM	Phòng 3 được mở bằng ID Khách	
	12/01/2021 8:45:39 AM	Phòng 3 được mở bằng ID Khách	
	12/01/2021 8:46:29 AM	Phòng 3 được mở bằng ID Khách	
	12/01/2021 8:46:45 AM	Phòng 3 được mở bằng ID Khách	
	12/01/2021 8:47:27 AM	Phòng 3 được mở bằng ID Phụ	
	12/01/2021 8:48:07 AM	Phòng 3 được mở bằng ID Khách	
	12/01/2021 8:48:34 AM	Phòng 3 được mở bằng ID Khách	
	12/01/2021 8:49:16 AM	Phòng 3 được mở bằng ID Khách	
	12/01/2021 8:50:07 AM	Phòng 3 được mở bằng ID Khách	
	12/01/2021 8:50:36 AM	Phòng 3 được mở bằng ID Khách	
	12/01/2021 8:51 AM	Phòng 3 đã thanh toán	
	12/01/2021 8:52:06 AM	Phòng 3 được mở bằng ID Phụ	
	12/01/2021 8:52:20 AM	Phòng 3 được mở bằng ID Master	

Hình 5.12: Lịch sử thẻ hoạt động phòng 3

Chương 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ

Trong phần lịch sử này có thể lọc ra lịch sử riêng của từng phòng như Hình 5.12 là lịch sử hoạt động của phòng 3.

5.1.3 Kết quả thống kê

Trong quá trình hoạt động, có một số lỗi xảy ra như sau: mạch điều khiển bị treo, mạch chính bị treo và khi thẻ mở cửa nhưng mạch chính không nhận được phản hồi về thẻ vừa mở cửa. Nguyên nhân gồm: Quét thẻ quá nhanh dẫn đến mạch chưa kịp nhận được tín hiệu, xử lý phân luồng chưa tốt làm cho tín hiệu truyền gói tin bị xung đột dẫn đến treo vi điều khiển. Kết quả thống kê các trường hợp hoạt động của mô hình được mô tả ở Bảng 5.1, Bảng 5.2, Bảng 5.3, Bảng 5.4.

Bảng 5.1: Kết quả thống kê mở cửa khi phòng chưa kích hoạt

Phòng	Thẻ Master		Thẻ phụ		Tỉ lệ phản hồi hoạt động thẻ (hai thẻ được quét luân phiên)
	Số lần quét	Số lần thành công	Số lần quét	Số lần thành công	
1	10	10	10	10	19/20
2	10	10	10	9	19/20
3	10	10	10	10	20/20

Bảng 5.2: Kết quả thống kê mở cửa khi có 1 phòng được kích hoạt

Phòng	Thẻ master		Thẻ phụ		Thẻ khách		Tỉ lệ phản hồi hoạt động thẻ (ba thẻ được quét luân phiên)
	Số lần quét	Số lần thành công	Số lần quét	Số lần thành công	Số lần quét	Số lần thành công	
1	10	10	10	10	Chưa kích hoạt		20/20
2	10	10	10	8	Chưa kích hoạt		18/20
3	5	5	5	5	10	10	20/20

Chương 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ

Bảng 5.3: Kết quả thống kê mở cửa khi có 2 phòng được kích hoạt

Phòng	Thẻ master		Thẻ phụ		Thẻ khách		Tỉ lệ phản hồi hoạt động thẻ (ba thẻ được quét luân phiên)
	Số lần quét	Số lần thành công	Số lần quét	Số lần thành công	Số lần quét	Số lần thành công	
1	5	5	5	5	8	7	17/18
2	5	5	5	5	10	10	18/20
3	10	10	10	10	Chưa kích hoạt		20/20

Bảng 5.4: Kết quả thống kê mở cửa khi có 3 phòng được kích hoạt

Phòng	Thẻ master		Thẻ phụ		Thẻ khách		Tỉ lệ phản hồi hoạt động thẻ (ba thẻ được quét luân phiên)
	Số lần quét	Số lần thành công	Số lần quét	Số lần thành công	Số lần quét	Số lần thành công	
1	5	5	5	5	9	8	18/19
2	2	2	4	4	10	10	16/16
3	5	4	4	4	8	8	16/17

5.2 NHẬN XÉT – ĐÁNH GIÁ

Sau thời gian tìm hiểu và nghiên cứu về đề tài, nhóm đã hoàn thành hệ thống thi công gồm có:

- Mô hình mạch chính giao tiếp với máy tính thông qua mạch giao tiếp USB UART.
- Ba mô hình mạch điều khiển đóng/mở cửa bằng thẻ tag RFID.
- Giao diện lưu trữ thông tin được viết trên Microsoft Visual Studio 2019 và sử dụng cơ sở dữ liệu SQL Server Management Studio 2019.

Hệ thống sau khi hoàn thành sẽ đáp ứng các chức năng sau: Đóng/mở cửa bằng hai thẻ Master và thẻ phụ trong cả hai trường hợp phòng chưa kích hoạt và phòng đã kích hoạt; Sử dụng giao diện để kích hoạt thẻ khách cho phòng; Các thông báo về tình

Chương 5. KẾT QUẢ - NHẬN XÉT – ĐÁNH GIÁ

trang của phòng được hiện thị trên LCD 16x2; Khi thẻ tag được sử dụng để đóng/mở cửa, thông tin về thẻ tag sẽ được lưu trữ trên cơ sở dữ liệu; Các thông tin được lưu trữ trên cơ sở dữ liệu có thể tìm kiếm, sửa đổi hoặc xoá ứng với các phân quyền tài khoản. Sau khi đã tiến hành chạy thử nghiệm và nhóm đã thu được những kết quả sau:

- Khi phòng chưa kích hoạt, thẻ Master và thẻ phụ có thể đóng/mở cửa, lịch sử được truyền về mạch chính và lưu trữ cơ sở dữ liệu.
- Khi phòng đã kích hoạt thẻ khách, phòng sẽ có ba thẻ có thể mở cửa và lịch sử thẻ sẽ được lưu lại. Mỗi khi phòng được kích hoạt lại, lịch sử hoạt động thẻ trước đó sẽ bị xoá đi và bắt đầu lịch sử mới.
- Khi có thẻ không hợp lệ, LCD sẽ có thông báo không hợp lệ và buzzer sẽ kêu tám âm thanh liên tiếp.
- Giao diện quản lý thông tin được thiết kế gồm có giao diện thông tin phòng, giao diện thuê phòng, giao diện thông tin khách, giao diện lịch sử hoạt động của thẻ. Kích hoạt thẻ được thực hiện tại giao diện thuê phòng sau khi đã chọn phòng trống và nhập thông tin khách.
- Về thời gian: Cửa sẽ được mở ngay khi quét thẻ, sau 1 – 2s giao diện sẽ nhận được phản hồi thẻ mở cửa phòng để lưu vào lịch sử. Với quá trình kích hoạt, phòng sẽ chuyển trạng thái từ chưa hoạt sang kích hoạt sau khi quét thẻ khách tại mạch chính sau 2 – 3s.

Chương 6. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

6.1 KẾT LUẬN

Đề tài: “*Thiết kế và thi công hệ thống hỗ trợ mở cửa ứng dụng công nghệ RFID tại khách sạn dùng Arduino*” đã được hoàn thành, hệ thống có thể đóng/mở cửa tự động sử dụng thẻ tag RFID. Hệ thống dùng để hỗ trợ việc đóng/mở cửa trong thực tế với tính tự động cao và việc truy xuất thông tin dễ dàng giúp nâng cao hiệu quả công việc và hiệu suất quản lý. Đề tài được thiết kế và thi công dùng vi điều khiển Arduino để điều khiển mở cửa phòng khách sạn, trong đó RFID được dùng để nhận dạng thẻ tag hỗ trợ quá trình điều khiển của Arduino. Hơn nữa, hệ thống thiết kế có một PC để nhập, hiển thị và lưu trữ thông tin khách hàng, lịch sử mở cửa của từng phòng.

Hệ thống hoạt động tương đối ổn định, chính xác với mô hình đã thi công. Tuy nhiên, hệ thống chưa có tính áp dụng thực tế cao do mới chỉ hoạt động với mức mô hình ba phòng, vẫn có lỗi xảy ra khi làm việc do thuật toán chưa được tối ưu. Mô hình vẫn ở mức mô phỏng, hệ thống thực tế sẽ phức tạp với nhiều khâu xử lý hơn.

6.2 HƯỚNG PHÁT TRIỂN

Sau khi nghiên cứu và tìm hiểu thực tế, nhóm đã đưa ra các hướng mở rộng như sau:

- Thêm các chức năng khác như mật khẩu hay vân tay để phòng khi quên thẻ hay mất thẻ.
- Sử dụng cảm biến để nhận biết chốt khoá cửa đã đúng vị trí chưa để đảm bảo cửa đã được khoá.

TÀI LIỆU THAM KHẢO

[1] Nguyễn Thành Dương, Trần Vinh Sơn, “*Thiết kế và thi công mô hình khoá điện tử có giám sát từ xa sử dụng GSM và RFID*”, Đồ án tốt nghiệp, trường ĐHSPKT, Tp.HCM, 2018-2019, HK2.

[2] Nguyễn Thanh Lộc, Mai Văn Thành, “*Thiết kế và thi công mô hình điều khiển, giám sát bãi giữ xe ô tô tự động*”, Đồ án tốt nghiệp, trường ĐHSPKT, Tp.HCM, 2018-2019, HK1.

[3] Nguyễn Đăng Việt, Trần Trí Đạt, “*Thiết kế, thi công bãi giữ xe ứng dụng công nghệ RFID và xử lý ảnh*”, Đồ án tốt nghiệp, trường ĐHSPKT, Tp.HCM, 2018-2019, HK2.

[4] Trần Thị Linh Đa, Nguyễn Châu Ngân, “*Thiết kế và thi công hệ thống điểm danh nhân viên sử dụng vi điều khiển ARM*”, Đồ án tốt nghiệp, trường ĐHSPKT, Tp.HCM, 2018-2019, HK1.

[5] Lê Trung Hậu, Phạm Thanh Tuân, “*Thiết kế và thi công mô hình của hàng giặt sấy tự phục vụ*”, Đồ án tốt nghiệp, trường ĐHSPKT, Tp.HCM, 2018-2019, HK2.

[6] Trần Quốc Tiến, Nguyễn Thanh Phong, “*Thiết kế và thi công hệ thống IOT phục vụ cho nông nghiệp ứng dụng Gateway*”, Đồ án tốt nghiệp, trường ĐHSPKT, Tp.HCM, 2018-2019, HK2.

[7] Smart Home OnSky, “*Zigbee là gì, vì sao nhà thông minh sử dụng công nghệ này?*”, <http://www.onskyinc.com>, 3/2019.

[8] MikeGrusin, “*RFM69HCW Hookup Guide*”, <http://www.learn.sparkfun.com>, 4/2016.

[9] Nguyễn Quang Thạnh, Phan Thanh Triều, “*Thi công mô hình hệ thống trồng hoa lan*”, Đồ án tốt nghiệp, trường ĐHSPKT, Tp.HCM, 2018-2019, HK2.

[10] Nguyễn Hoàn Hảo, Huỳnh Ngọc Giang, “*Ứng dụng xử lý ảnh điều khiển đèn giao thông tại nút giao thông*”, Đồ án tốt nghiệp, trường ĐHSPKT, Tp.HCM, 2019-2020, HK1.

[11] Hoàng Bảo Phúc, Nguyễn Chí Thương, “*Thiết kế và thi công hệ thống quản lý mô hình HOMESTAY, AIRBNB*”, Đồ án tốt nghiệp, trường ĐHSPKT, Tp.HCM, 2019-2020, HK1.

TÀI LIỆU THAM KHẢO

[12] Nguyễn Hữu Phước, Võ Thanh Phong, “*Thiết kế và thi công mô hình xe robot dò tìm kim loại điều khiển bằng điện thoại*”, Đồ án tốt nghiệp, trường ĐHSPKT, Tp.HCM, 2018-2019, HK2.

[13] Nguyễn Văn Hiệp, Giáo trình “*Công nghệ nhận dạng bằng sóng vô tuyến*”, trường ĐHSPKT, TP.HCM, 2014.

[14] Nguyễn Tài Tụ, “*Quản lý dữ liệu bệnh nhân trong bệnh viện sử dụng công nghệ RFID*”, Đồ án tốt nghiệp, trường ĐHSPKT, Tp.HCM, 2018-2019, HK1.

[15] Madeleine Waller, ”*Design and Implementation of a Secure Energy-Efficient Hardware Platform for Wireless Sensor Networks*”, Master of science, Massachusetts Institute of Technology, 9/2018

[16] Samuel Iyiola, “*Moteino-based wireless data transfer for environmental monitoring*”, Master of science, University of North Texas, 5/2017

PHỤ LỤC

- ❖ Code Arduino của mạch chính

```
#include <RFM69.h>
#include <SPI.h>
#include <MFRC522.h>
#define SS_RFID 10 // pin SDA tren RFID RC522
#define RST_PIN 7
MFRC522 mfrc522(SS_RFID, RST_PIN);
#define NETWORKID    0
#define MYNODEID     1
#define NODE_ID1      2
#define NODE_ID2      4
#define NODE_ID3      6
#define FREQUENCY   RF69_433MHZ
#define ENCRYPT      true
#define ENCRYPTKEY   "DATNBMDTCNYS2021"
#define USEACK       false
RFM69 radio;
#define buzzer 6
String str = "";
String strID = "";
String strID_Send = "";
String strID_Act = "";
int TONODEID;
boolean act_send = false;
boolean act = false;
boolean ack = false;
void readRFID();
void sendID();
void receiveID();
```

PHỤ LỤC

```
void checkID();  
  
void setup(){  
    Serial.begin(9600);  
    SPI.begin();  
    pinMode(buzzer, OUTPUT);  
    analogWrite(buzzer, 0);  
    pinMode(8,OUTPUT);  
    digitalWrite(8, HIGH);  
    delay(100);  
    digitalWrite(8, LOW);  
    delay(100);  
    Serial.print("Node ");  
    Serial.print(MYNODEID,DEC);  
    Serial.println(" ready");  
    radio.initialize(FREQUENCY, MYNODEID, NETWORKID);  
    radio.setHighPower();  
    if (ENCRYPT)  
        radio.encrypt(ENCRYPTKEY);  
        mfrc522.PCD_Init();  
    }  
void loop(){  
    if(Serial.available())  
    {  
        act_send=true;  
        char s = (char)Serial.read();  
        str += s;  
    }  
    if(act){  
        receiveID();  
        delay(1000);  
    }  
}
```

```
act = !act;  
}  
else{  
    radio.setHighPower(false);  
    readRFID();  
    if(strID != ""){  
        checkID();  
        if(ack && act_send)  
        {  
            sendID();  
            str="";  
            act_send= false;  
            ack=false;  
        }  
        delay(1);  
    }  
    act = !act;  
}  
}  
  
void readRFID(){  
    if (mfrc522.PICC_IsNewCardPresent()){ // Look for new cards  
        if(mfrc522.PICC_ReadCardSerial()){ //Select one of the cards  
            strID="";  
            strID_Send="";  
            for (byte i = 0; i < mfrc522.uid.size; i++) {  
                strID += String(mfrc522.uid.uidByte[i],DEC); //Helper routine to dump a byte  
                array as dec values to Serial  
            }  
            Serial.println(strID);  
            analogWrite(buzzer, 200);  
            delay(200);  
            analogWrite(buzzer, 0);  
        }  
    }  
}
```

```
}

}

mfrc522.PICC_HaltA(); // Halt PICC
mfrc522.PCD_StopCrypto1();

}

void sendID(){
    radio.setHighPower();
    byte len = strID_Send.length() + 1;
    char buf[20];
    for (byte i = 1; i <= len; len++){
        strID_Send.toCharArray(buf,len);
    }
    if (USEACK)
    {
        if (radio.sendWithRetry(TONODEID, buf, strlen(buf)))
            Serial.println("ACK received!");
        else
            Serial.println("no ACK received :(");
    }
    else // don't use ACK
    {
        radio.send(TONODEID, buf, strlen(buf));
    }
}

void receiveID(){
    radio.setHighPower();
    char instring[20]={0};
    byte len=0;
    if (radio.receiveDone())
    {
```

PHỤ LỤC

```
for (byte i = 0; i < radio.DATALEN; i++){
    instring[i] = (char)radio.DATA[i];
}

String str(instring);
strID_Act = str;

if(radio.SENDERID == 2)
{
    if(strID_Act == "ID Guest") Serial.println("Phong 1 ID Khach");
    if(strID_Act == "ID Master") Serial.println("Phong 1 ID Master");
    if(strID_Act == "ID Labor") Serial.println("Phong 1 ID Phu");
}

if(radio.SENDERID == 4)
{
    if(strID_Act == "ID Guest") Serial.println("Phong 2 ID Khach");
    if(strID_Act == "ID Master") Serial.println("Phong 2 ID Master");
    if(strID_Act == "ID Labor") Serial.println("Phong 2 ID Phu");
}

if(radio.SENDERID == 6)
{
    if(strID_Act == "ID Guest") Serial.println("Phong 3 ID Khach");
    if(strID_Act == "ID Master") Serial.println("Phong 3 ID Master");
    if(strID_Act == "ID Labor") Serial.println("Phong 3 ID Phu");
}
```

```
void checkID(){
    if(str == "1"){
        ack=true;
        strID_Send = strID;
        TONODEID = NODE_ID1;
```

```
}
```

```
if(str == "2"){

    ack=true;

    strID_Send = strID;

    TONODEID = NODE_ID2;

}
```

```
if(str == "3"){

    ack=true;

    strID_Send = strID;

    TONODEID = NODE_ID3;

}

}
```

❖ Code Arduino của mạch điều khiển đóng-mở cửa phòng 1

```
#include <RFM69.h>

#include <SPI.h>

#define NETWORKID    0

#define MYNODEID     2

#define TONODEID      1

#define FREQUENCY   RF69_433MHZ

#define ENCRYPT      true

#define ENCRYPTKEY   "DATNBMDTCNYS2021"

#define USEACK       false

RFM69 radio;

#include <MFRC522.h>

#define SS_RFID 10 // pin SDA trên RFID RC522

#define RST_PIN 7

MFRC522 mfrc522(SS_RFID, RST_PIN); // Create MFRC522 instance.

#include <Servo.h>

Servo myservo;

#define servo 5
```

```
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,16,2);
#include "pitches.h"
// notes in the melody:
int melody[] = {NOTE_B7, NOTE_G6, NOTE_FS7, NOTE_F4, NOTE_DS5,
NOTE_B7, NOTE_B6, NOTE_A5};
// note durations: 4 = quarter note, 8 = eighth note, etc.:
int noteDurations[] = {4, 4, 4, 4, 4, 4, 4, 4, 4};
#define buzzer 3
String strID = "";
String strID_Act = "";
String strID_Key = "";
String strID_Master="202315026";
String strID_Labor="90170214128";
String strID_Send = "";
boolean act = false;
boolean openID = false;
void readRFID();
void receiveID();
void sendID();
void openDoor();
void openError();
void setup(){
    Serial.begin(9600);
    SPI.begin();
    pinMode(buzzer, OUTPUT);
    analogWrite(buzzer, 0);
    myservo.attach(servo);
    lcd.init();
```

PHỤ LỤC

```
lcd.backlight();
lcd.clear();
pinMode(8,OUTPUT);
digitalWrite(8, HIGH);
delay(100);
digitalWrite(8, LOW);
delay(100);
Serial.print("Node ");
Serial.print(MYNODEID,DEC);
Serial.println(" ready");
radio.initialize(FREQUENCY, MYNODEID, NETWORKID);
radio.setHighPower();
if (ENCRYPT)
    radio.encrypt(ENCRYPTKEY);
mfrc522.PCD_Init();
myservo.write(180);
}

void loop(){
if(act){
    receiveID();
    delay(1000);
    act = !act;
} else{
    radio.setHighPower(false);
    if(openID){
        lcd.setCursor(0, 0);
        lcd.print("DAY LA PHONG 1 ");
        lcd.setCursor(0, 1);
        lcd.print("DA KICH HOAT ");
    } else{
        lcd.setCursor(0, 0);
    }
}
}
```

PHỤ LỤC

```
lcd.print("DAY LA PHONG 1 ");
lcd.setCursor(0, 1);
lcd.print("CHUA KICH HOAT ");
}

readRFID();

if(strID == strID_Act)&&(strID_Act != ""){
    strID_Send = "ID Guest";
    sendID();
    openDoor();
}

if(strID == strID_Master){
    strID_Send = "ID Master";
    sendID();
    openDoor();
}

if(strID == strID_Labor){
    strID_Send = "ID Labor";
    sendID();
    openDoor();
}

if((strID != "")&&(strID != strID_Labor)&&(strID != strID_Master)&&(strID != strID_Act)){
    openError();
}

act = !act;

}

strID = "";

}

void readRFID(){
if (mfrc522.PICC_IsNewCardPresent()) { // Look for new cards
    if(mfrc522.PICC_ReadCardSerial()){ //Select one of the cards
```

```
for (byte i = 0; i < mfrc522.uid.size; i++) {
    strID += String(mfrc522.uid.uidByte[i],DEC); //Helper routine to dump a byte
array as dec values to Serial
}
Serial.print("ID: ");
Serial.println(strID);
analogWrite(buzzer, 200);
delay(200);
analogWrite(buzzer, 0);
}
}

mfrc522.PICC_HaltA(); // Halt PICC
mfrc522.PCD_StopCrypto1();
}

void receiveID(){
radio.setHighPower();
char instring[20]={0};
byte len=0;
if (radio.receiveDone())
{
    Serial.print("received from node ");
    Serial.print(radio.SENDERID, DEC);
    Serial.print(": ");
    for (byte i = 0; i < radio.DATALEN; i++){
        instring[i] = (char)radio.DATA[i];
    }
    String str(instring);
    strID_Act = str;
    Serial.println(strID_Act);
    if(strID_Act == strID_Key){
        strID_Act = "";
    }
}
```

```
strID_Key ="";  
openID = false;  
else{  
    strID_Key = strID_Act;  
    openID = true;  
}  
Serial.print("RSSI: ");  
Serial.println(radio.RSSI);  
}  
}  
void sendID(){  
    radio.setHighPower();  
    byte len = strID_Send.length() + 1;  
    char buf[20];  
    for (byte i = 1; i <= len; len++){  
        strID_Send.toCharArray(buf,len);  
    }  
    Serial.print("sending to node ");  
    Serial.print(TONODEID, DEC);  
    Serial.print(": ");  
    Serial.println(strID_Send);  
    if (USEACK)  
    {  
        if (radio.sendWithRetry(TONODEID, buf, strlen(buf)))  
            Serial.println("ACK received!");  
        else  
            Serial.println("no ACK received :(");  
    }  
    else // don't use ACK  
    {  
        radio.send(TONODEID, buf, strlen(buf));  
    }  
}
```

```
}

}

void openDoor(){
    int pos = 180;
    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("CUA DANG MO      ");
    lcd.setCursor(0, 1);
    lcd.print("      ");
    for (pos = 180; pos >= 20; pos -= 1) {
        myservo.write(pos);
        delay(15);
    }
    for(int i = 4 ; i >= 0; i--){
        lcd.setCursor(0, 0);
        lcd.print("CUA SE DONG LAI ");
        lcd.setCursor(0, 1);
        lcd.print("SAU ");
        lcd.print(i);
        lcd.print("s      ");
        delay(1000);
    }
    lcd.setCursor(0, 0);
    lcd.print("CUA DANG DONG      ");
    lcd.setCursor(0, 1);
    lcd.print("      ");
    for (pos = 20; pos <= 180; pos += 1) {
        myservo.write(pos);
        delay(15);
    }
}
```

PHỤ LỤC

```
void openError(){
    lcd.setCursor(0, 0);
    lcd.print("THE KHONG HOP LE");
    lcd.setCursor(0, 1);
    lcd.print("          ");
    for (int thisNote = 0; thisNote < 8; thisNote++) {
        int noteDuration = 1000 / noteDurations[thisNote];
        tone(buzzer, melody[thisNote], noteDuration);
        int pauseBetweenNotes = noteDuration * 1.30;
        delay(pauseBetweenNotes);
        noTone(buzzer);
    }
    delay(500);
}
```

❖ Code Arduino của mạch điều khiển đóng-mở cửa phòng 2

```
#include <RFM69.h>
#include <SPI.h>
#define NETWORKID    0
#define MYNODEID     4
#define TONODEID     1
#define FREQUENCY   RF69_433MHZ
#define ENCRYPT      true
#define ENCRYPTKEY   "DATNBMDTCNYS2021"
#define USEACK       false
RFM69 radio;
#include <MFRC522.h>
#define SS_RFID 10 // pin SDA trên RFID RC522
#define RST_PIN 7
MFRC522 mfrc522(SS_RFID, RST_PIN); // Create MFRC522 instance.
```

PHỤ LỤC

```
#include <Servo.h>
Servo myservo;
#define servo 5
#include <Wire.h>;
#include <LiquidCrystal_I2C.h>;
LiquidCrystal_I2C lcd(0x27,16,2);
#include "pitches.h"
// notes in the melody:
int melody[] = {NOTE_B7, NOTE_G6, NOTE_FS7, NOTE_F4, NOTE_DS5,
NOTE_B7, NOTE_B6, NOTE_A5};
// note durations: 4 = quarter note, 8 = eighth note, etc.:
int noteDurations[] = {4, 4, 4, 4, 4, 4, 4, 4};
#define buzzer 3
String strID = "";
String strID_Act = "";
String strID_Key = "";
String strID_Master="202315026";
String strID_Labor="2342211026";
String strID_Send = "";
boolean act = false;
boolean openID = false;
void readRFID();
void receiveID();
void sendID();
void openDoor();
void openError();
void setup(){
    Serial.begin(9600);
    SPI.begin();
    pinMode(buzzer, OUTPUT);
    analogWrite(buzzer, 0);
```

PHỤ LỤC

```
myservo.attach(servo);

lcd.init();
lcd.backlight();
lcd.clear();

pinMode(8,OUTPUT);
digitalWrite(8, HIGH);

delay(100);

digitalWrite(8, LOW);

delay(100);

Serial.print("Node ");

Serial.print(MYNODEID,DEC);

Serial.println(" ready");

radio.initialize(FREQUENCY, MYNODEID, NETWORKID);

radio.setHighPower();

if (ENCRYPT)

    radio.encrypt(ENCRYPTKEY);

    mfrc522.PCD_Init();

    myservo.write(180);

}

void loop(){

if(act){

    receiveID();

    delay(1000);

    act = !act;

} else

    radio.setHighPower(false);

    if(openID){

        lcd.setCursor(0, 0);

        lcd.print("DAY LA PHONG 2 ");

        lcd.setCursor(0, 1);

        lcd.print("DA KICH HOAT ");

    }
```

```
 }else{
    lcd.setCursor(0, 0);
    lcd.print("DAY LA PHONG 2 ");
    lcd.setCursor(0, 1);
    lcd.print("CHUA KICH HOAT ");
}
readRFID();
if((strID == strID_Act)&&(strID_Act != "")){
    strID_Send = "ID Guest";
    sendID();
    openDoor();
}
if(strID == strID_Master){
    strID_Send = "ID Master";
    sendID();
    openDoor();
}
if(strID == strID_Labor){
    strID_Send = "ID Labor";
    sendID();
    openDoor();
}
if((strID != "")&&(strID != strID_Labor)&&(strID != strID_Master)&&(strID != strID_Act)){
    openError();
}
act = !act;
}
strID = "";
```

```
void readRFID(){
    if (mfrc522.PICC_IsNewCardPresent()) { // Look for new cards
        if(mfrc522.PICC_ReadCardSerial()) { //Select one of the cards
            for (byte i = 0; i < mfrc522.uid.size; i++) {
                strID += String(mfrc522.uid.uidByte[i],DEC); //Helper routine to dump a byte
                array as dec values to Serial
            }
            Serial.print("ID: ");
            Serial.println(strID);
            analogWrite(buzzer, 200);
            delay(200);
            analogWrite(buzzer, 0);
        }
    }
    mfrc522.PICC_HaltA(); // Halt PICC
    mfrc522.PCD_StopCrypto1();
}

void receiveID(){
    radio.setHighPower();
    char instring[20]={0};
    byte len=0;
    if (radio.receiveDone())
    {
        Serial.print("received from node ");
        Serial.print(radio.SENDERID, DEC);
        Serial.print(": ");
        for (byte i = 0; i < radio.DATALEN; i++){
            instring[i] = (char)radio.DATA[i];
        }
        String str(instring);
        strID_Act = str;
    }
}
```

```
Serial.println(strID_Act);
if(strID_Act == strID_Key){
    strID_Act = "";
    strID_Key = "";
    openID = false;
} else{
    strID_Key = strID_Act;
    openID = true;
}
Serial.print("RSSI: ");
Serial.println(radio.RSSI);
}

void sendID(){
radio.setHighPower();
byte len = strID_Send.length() + 1;
char buf[20];
for (byte i = 1; i <= len; len++){
    strID_Send.toCharArray(buf,len);
}
Serial.print("sending to node ");
Serial.print(TONODEID, DEC);
Serial.print(": ");
Serial.println(strID_Send);
if (USEACK)
{
    if (radio.sendWithRetry(TONODEID, buf, strlen(buf)))
        Serial.println("ACK received!");
    else
        Serial.println("no ACK received :(");
}
```

```
}

else // don't use ACK

{

    radio.send(TONODEID, buf, strlen(buf));

}

}

void openDoor(){

    int pos = 180;

    lcd.clear();

    lcd.setCursor(0, 0);

    lcd.print("CUA DANG MO    ");

    lcd.setCursor(0, 1);

    lcd.print("          ");

    for (pos = 180; pos >= 20; pos -= 1) {

        myservo.write(pos);

        delay(15);

    }

    for(int i = 4 ; i >= 0; i--){

        lcd.setCursor(0, 0);

        lcd.print("CUA SE DONG LAI ");

        lcd.setCursor(0, 1);

        lcd.print("SAU ");

        lcd.print(i);

        lcd.print("s      ");

        delay(1000);

    }

    lcd.setCursor(0, 0);

    lcd.print("CUA DANG DONG   ");

    lcd.setCursor(0, 1);

    lcd.print("          ");

    for (pos = 20; pos <= 180; pos += 1) {
```

PHỤ LỤC

```
// in steps of 1 degree
myservo.write(pos);
delay(15);
}

}

void openError(){
lcd.setCursor(0, 0);
lcd.print("THE KHONG HOP LE");
lcd.setCursor(0, 1);
lcd.print("      ");
for (int thisNote = 0; thisNote < 8; thisNote++) {
    int noteDuration = 1000 / noteDurations[thisNote];
    tone(buzzer, melody[thisNote], noteDuration);
    int pauseBetweenNotes = noteDuration * 1.30;
    delay(pauseBetweenNotes);
    noTone(buzzer);
}
delay(500);
}
```

❖ Code Arduino của mạch điều khiển đóng-mở cửa phòng 3

```
#include <RFM69.h>
#include <SPI.h>
#define NETWORKID    0
#define MYNODEID     6
#define TONODEID     1
#define FREQUENCY   RF69_433MHZ
#define ENCRYPT      true
#define ENCRYPTKEY   "DATNBMDTCNYS2021"
#define USEACK       false
```

PHỤ LỤC

```
RFM69 radio;  
#include <MFRC522.h>  
#define SS_RFID 10 // pin SDA tren RFID RC522  
#define RST_PIN 7  
MFRC522 mfrc522(SS_RFID, RST_PIN); // Create MFRC522 instance.  
#include <Servo.h>  
Servo myservo;  
#define servo 5  
#include <Wire.h>;  
#include <LiquidCrystal_I2C.h>;  
LiquidCrystal_I2C lcd(0x27,16,2);  
#include "pitches.h"  
// notes in the melody:  
int melody[] = {NOTE_B7, NOTE_G6, NOTE_FS7, NOTE_F4, NOTE_DS5,  
NOTE_B7, NOTE_B6, NOTE_A5};  
// note durations: 4 = quarter note, 8 = eighth note, etc.:  
int noteDurations[] = {4, 4, 4, 4, 4, 4, 4, 4};  
#define buzzer 3  
String strID = "";  
String strID_Act = "";  
String strID_Key = "";  
String strID_Master="202315026";  
String strID_Labor="15442164";  
String strID_Send = "";  
boolean act = false;  
boolean openID = false;  
void readRFID();  
void receiveID();  
void sendID();  
void openDoor();  
void openError();
```

```
void setup(){
    Serial.begin(9600);
    SPI.begin();
    pinMode(buzzer, OUTPUT);
    analogWrite(buzzer, 0);
    myservo.attach(servo);
    lcd.init();
    lcd.backlight();
    lcd.clear();
    pinMode(8,OUTPUT);
    digitalWrite(8, HIGH);
    delay(100);
    digitalWrite(8, LOW);
    delay(100);
    Serial.print("Node ");
    Serial.print(MYNODEID,DEC);
    Serial.println(" ready");
    radio.initialize(FREQUENCY, MYNODEID, NETWORKID);
    radio.setHighPower();
    if (ENCRYPT)
        radio.encrypt(ENCRYPTKEY);
    mfrc522.PCD_Init();
    myservo.write(180);
}

void loop(){
    if(act){
        receiveID();
        delay(1000);
        act = !act;
    }else{
        radio.setHighPower(false);
    }
}
```

PHỤ LỤC

```
if(openID){  
    lcd.setCursor(0, 0);  
    lcd.print("DAY LA PHONG 3 ");  
    lcd.setCursor(0, 1);  
    lcd.print("DA KICH HOAT ");  
}  
else{  
    lcd.setCursor(0, 0);  
    lcd.print("DAY LA PHONG 3 ");  
    lcd.setCursor(0, 1);  
    lcd.print("CHUA KICH HOAT ");  
}  
readRFID();  
if((strID == strID_Act)&&(strID_Act != "") ){  
    strID_Send = "ID Guest";  
    sendID();  
    openDoor();  
}  
if(strID == strID_Master){  
    strID_Send = "ID Master";  
    sendID();  
    openDoor();  
}  
if(strID == strID_Labor){  
    strID_Send = "ID Labor";  
    sendID();  
    openDoor();  
}  
if((strID != "")&&(strID != strID_Labor)&&(strID != strID_Master)&&(strID != strID_Act)){  
    openError();  
}
```

```
act = !act;
}
strID = "";
}

void readRFID(){
    if (mfrc522.PICC_IsNewCardPresent()) { // Look for new cards
        if(mfrc522.PICC_ReadCardSerial()){ //Select one of the cards
            for (byte i = 0; i < mfrc522.uid.size; i++) {
                strID += String(mfrc522.uid.uidByte[i],DEC); //Helper routine to dump a byte
array as dec values to Serial
            }
            Serial.print("ID: ");
            Serial.println(strID);
            analogWrite(buzzer, 200);
            delay(200);
            analogWrite(buzzer, 0);
        }
    }
    mfrc522.PICC_HaltA(); // Halt PICC
    mfrc522.PCD_StopCrypto1();
}

void receiveID(){
    radio.setHighPower();
    char instring[20]={0};
    byte len=0;
    if (radio.receiveDone())
    {
        Serial.print("received from node ");
        Serial.print(radio.SENDERID, DEC);
        Serial.print(": ");
        for (byte i = 0; i < radio.DATALEN; i++){

```

```
instring[i] = (char)radio.DATA[i];  
}  
String str(instring);  
strID_Act = str;  
Serial.println(strID_Act);  
if(strID_Act == strID_Key){  
    strID_Act = "";  
    strID_Key = "";  
    openID = false;  
}else{  
    strID_Key = strID_Act;  
    openID = true;  
}  
Serial.print("RSSI: ");  
Serial.println(radio.RSSI);  
}  
}  
void sendID(){  
    radio.setHighPower();  
    byte len = strID_Send.length() + 1;  
    char buf[20];  
for (byte i = 1; i <= len; len++){  
        strID_Send.toCharArray(buf,len);  
    }  
    Serial.print("sending to node ");  
    Serial.print(TONODEID, DEC);  
    Serial.print(": ");  
    Serial.println(strID_Send);  
if (USEACK)  
{  
    if (radio.sendWithRetry(TONODEID, buf, strlen(buf)))
```

```
Serial.println("ACK received!");  
else  
    Serial.println("no ACK received :(");  
}  
else // don't use ACK  
{  
    radio.send(TONODEID, buf, strlen(buf));  
}  
}  
  
void openDoor(){  
    int pos = 180;  
    lcd.clear();  
    lcd.setCursor(0, 0);  
    lcd.print("CUA DANG MO      ");  
    lcd.setCursor(0, 1);  
    lcd.print("      ");  
    for (pos = 180; pos >= 20; pos -= 1) {  
        myservo.write(pos);  
        delay(15);  
    }  
    for(int i = 4 ; i >= 0; i--){  
        lcd.setCursor(0, 0);  
        lcd.print("CUA SE DONG LAI ");  
        lcd.setCursor(0, 1);  
        lcd.print("SAU ");  
        lcd.print(i);  
        lcd.print("s      ");  
        delay(1000);  
    }  
    lcd.setCursor(0, 0);
```

```
lcd.print("CUA DANG DONG      ");
lcd.setCursor(0, 1);
lcd.print("          ");
for (pos = 20; pos <= 180; pos += 1) {
    // in steps of 1 degree
    myservo.write(pos);
    delay(15);
}
}

void openError(){
lcd.setCursor(0, 0);
lcd.print("THE KHONG HOP LE");
lcd.setCursor(0, 1);
lcd.print("          ");
for (int thisNote = 0; thisNote < 8; thisNote++) {
    int noteDuration = 1000 / noteDurations[thisNote];
    tone(buzzer, melody[thisNote], noteDuration);
    int pauseBetweenNotes = noteDuration * 1.30;
    delay(pauseBetweenNotes);
    noTone(buzzer);
}
delay(500);
}
```

