

TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM

KHOA ĐIỆN ĐIỆN TỬ

BỘ MÔN KỸ THUẬT MÁY TÍNH – VIỄN THÔNG



HCMUTE

ĐỒ ÁN MÔN HỌC 1

**THIẾT KẾ VÀ THI CÔNG HỆ THỐNG GIÁM SÁT
NHÀ THÔNG MINH**

NGÀNH CÔNG NGHỆ KỸ THUẬT ĐIỆN TỬ - TRUYỀN THÔNG

Sinh viên: **NGUYỄN TÀI ANH TUẤN**

MSSV: 22161203

GVHD: **ThS. TRƯƠNG QUANG PHÚC**

TP. HỒ CHÍ MINH – 05/2025

LỜI CẢM ƠN

Trước tiên, em xin gửi lời cảm ơn chân thành đến Ban giám hiệu Trường Đại học Sư Phạm Kỹ Thuật TP.HCM, quý thầy cô trong Khoa Điện – Điện Tử, đặc biệt là Bộ môn Kỹ thuật Máy tính – Viễn thông, đã tạo điều kiện thuận lợi cho em trong quá trình học tập và thực hiện đồ án môn học.

Em xin bày tỏ lòng biết ơn sâu sắc đến ThS. Trương Quang Phúc – Giảng viên hướng dẫn của em. Người đã tận tình hướng dẫn, hỗ trợ và định hướng em trong suốt quá trình thực hiện đồ án. Những góp ý quý báu của thầy đã giúp em hoàn thiện đồ án một cách tốt nhất.

TP.Hồ Chí Minh, tháng 05 năm 2025

Sinh viên thực hiện

Nguyễn Tài Anh Tuấn

TÓM TẮT

Tại các hộ gia đình hiện đại, việc triển khai hệ thống giám sát và điều khiển thông minh thường gặp khó khăn do chi phí đầu tư cao và yêu cầu kết nối ổn định. Để giải quyết vấn đề này, đồ án đã thiết kế và phát triển một hệ thống nhà thông minh nhằm thực hiện các chức năng chính như giám sát môi trường, xác thực người dùng, điều khiển thiết bị và cài đặt an ninh, được tổ chức theo mô hình phân tán dựa trên hai vi điều khiển chính. Hệ thống tận dụng STM32 và ESP32, với kết nối thông qua giao thức UART để truyền dữ liệu nội bộ và liên kết với nền tảng đám mây qua kết nối không dây, trong đó STM32 chịu trách nhiệm đo các thông số môi trường như ánh sáng, nhiệt độ, độ ẩm và khí gas, hiển thị dữ liệu trên màn hình LCD qua giao tiếp I2C và truyền dữ liệu đến khói giao tiếp, trong khi khói giao tiếp sử dụng ESP32 để tiếp nhận dữ liệu, gửi lên Firebase để lưu trữ và xây dựng Web Dashboard hiển thị dữ liệu thời gian thực, hỗ trợ giám sát từ xa và khói nhận diện thực hiện xác thực người dùng qua các phương thức mật khẩu, thẻ RFID, vân tay, hiển thị thông tin trên màn hình và điều khiển servo để mở cửa khi xác thực thành công. Hệ thống này góp phần nâng cao chất lượng sống, đảm bảo an ninh và tăng tiện nghi cho ngôi nhà thông qua việc áp dụng các công nghệ giám sát, xác thực hiện đại, kết nối linh hoạt và tiết kiệm chi phí triển khai.

MỤC LỤC

LỜI CẢM ƠN	i
TÓM TẮT	ii
MỤC LỤC.....	iii
DANH MỤC CÁC TỪ VIẾT TẮT	vi
DANH MỤC BẢNG BIỂU	viii
DANH MỤC HÌNH ẢNH.....	ix
CHƯƠNG 1: TỔNG QUAN.....	1
1.1. GIỚI THIỆU CHUNG.....	1
1.2. MỤC TIÊU ĐỀ TÀI	1
1.3. TÌNH HÌNH NGHIÊN CỨU TRONG VÀ NGOÀI NƯỚC.....	1
1.3.1. Tình hình nghiên cứu ở nước ngoài	1
1.3.2. Tình hình nghiên cứu ở trong nước	2
1.4. PHƯƠNG PHÁP NGHIÊN CỨU	3
1.5. BỘ CỤC BÀI BÁO CÁO	4
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT	6
2.1. TỔNG QUAN VỀ HỆ THỐNG GIÁM NHÀ THÔNG MINH.....	6
2.2. VI ĐIỀU KHIỂN TRONG HỆ THỐNG NHÀ THÔNG MINH	6
2.2.1. Vi điều khiển STM32	6
2.2.2. Vi điều khiển ESP32	9
2.3. CÁC MODULE CẢM BIẾN, ĐIỀU KHIỂN, HIỂN THỊ	11
2.3.1. Cảm biến nhiệt độ, độ ẩm.....	11
2.3.2. Cảm biến cường độ ánh sáng	13
2.3.3. Cảm biến khí GAS	15

2.3.4. Module RFID RC522	16
2.3.5. Cảm biến vân tay AS608.....	18
2.3.6. Keypad 4x4.....	20
2.3.7. Module Relay 4 kênh	22
2.3.8. Động cơ Servo.....	24
2.3.9. Màn hình LCD 20X4 giao tiếp I2C.....	26
2.4. CÁC CHUẨN TRUYỀN THÔNG SỬ DỤNG.....	28
2.4.1. Giao tiếp UART	28
2.4.2. Giao tiếp I2C	29
2.4.3. Giao tiếp SPI	30
2.5. LUU TRỮ DỮ LIỆU VÀ GIAO DIỆN NGƯỜI DÙNG	32
2.5.1. Giới thiệu về Firebase Realtime Database	32
2.5.2. Giới thiệu Web Dashboard giám sát và điều khiển.....	32
CHƯƠNG 3: THIẾT KẾ HỆ THỐNG	34
3.1. YÊU CẦU THIẾT KẾ HỆ THỐNG	34
3.2. ĐẶC TẢ HỆ THỐNG	34
3.2.1. Chức năng hệ thống.....	34
3.2.2. Sơ đồ khái niệm.....	36
3.3. THIẾT KẾ PHẦN CỨNG	38
3.3.1. Sơ đồ nguyên lý toàn bộ hệ thống.....	38
3.3.2. Khối vi điều khiển	39
3.3.3. Khối cảm biến	40
3.3.4. Khối điều khiển thiết bị	41
3.3.5. Khối hiển thị	43
3.3.6. Khối nhận diện	44

3.3.7. Khối nguồn	45
3.4. THIẾT KẾ PHẦN MỀM	48
3.4.1. Lưu đồ giải thuật cho toàn hệ thống.....	48
3.4.2. Lưu đồ giải thuật cho các chế độ.....	49
CHƯƠNG 4: KẾT QUẢ VÀ ĐÁNH GIÁ.....	57
4.1. KẾT QUẢ PHẦN CỨNG	57
4.1.1. Mô hình phần cứng thực tế.....	57
4.1.2. Kiểm tra chế độ xác thực.....	59
4.1.3. Kiểm tra chế độ cảm biến và điều khiển thiết bị.....	64
4.1.4. Kiểm tra chế độ cài đặt.....	67
4.2. GIAO DIỆN HIỂN THỊ WEB.....	70
4.2.1. Giao diện Dashboard	70
4.2.2. Giao diện lịch sử lưu trữ.....	73
4.3. ĐÁNH GIÁ VÀ NHẬN XÉT HỆ THỐNG	75
4.3.1. Đánh giá hiệu suất	75
4.3.2. Ưu điểm và những hạn chế của hệ thống	76
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	77
5.1. KẾT LUẬN	77
5.2. HƯỚNG PHÁT TRIỂN	77
TÀI LIỆU THAM KHẢO	80

DANH MỤC CÁC TỪ VIẾT TẮT

Từ viết tắt	Từ viết đầy đủ	Nghĩa tiếng việt
ADC	Analog-to-Digital Converter	Bộ chuyển đổi tương tự sang số
AI	Artificial Intelligence	Trí tuệ nhân tạo
BLE	Bluetooth Low Energy	Bluetooth năng lượng thấp
DAC	Digital-to-Analog Converter	Bộ chuyển đổi số sang tương tự
DSP	Digital Signal Processing	Xử lý tín hiệu số
FPU	Floating Point Unit	Đơn vị xử lý số thực
FreeRTOS	Free Real-Time Operating System	Hệ điều hành thời gian thực
GPIO	General Purpose Input/Output	Ngõ vào/ra
HTTPS	HyperText Transfer Protocol Secure	Giao thức truyền tải siêu văn bản bảo mật
I2C	Inter-Integrated Circuit	Giao tiếp giữa các mạch tích hợp
IoT	Internet of Things	Internet kết nối vạn vật
LCD	Liquid Crystal Display	Màn hình tinh thể lỏng
MQTT	Message Queuing Telemetry Transport	Giao thức truyền thông điệp
PCB	Printed Circuit Board	Bảng mạch in
PWM	Pulse Width Modulation	Điều chế độ rộng xung
RFID	Radio Frequency Identification	Nhận diện tần số vô tuyến

SPI	Serial Peripheral Interface	Giao diện ngoại vi nối tiếp
SRAM	Static Random-Access Memory	Bộ nhớ truy cập ngẫu nhiên tĩnh
UART	Universal Asynchronous Receiver - Transmitter	Bộ thu phát không đồng bộ tổng hợp
Wi-Fi	Wireless Fidelity	Tín hiệu không dây

DANH MỤC BẢNG BIỂU

Bảng 2.1: Bảng thông số kỹ thuật chính của STM32F407	7
Bảng 2.2: Bảng thông số kỹ thuật chính của ESP32-S	10
Bảng 2.3: Thông số kỹ thuật DHT22	12
Bảng 2.4: Tên và chức năng từng chân DHT22.....	12
Bảng 2.5: Thông số kỹ thuật BH1750.....	13
Bảng 2.6: Tên và chức năng từng chân BH1750	14
Bảng 2.7: Thông số kỹ thuật MQ-2	15
Bảng 2.8: Tên và chức năng từng chân MQ-2	16
Bảng 2.9: Thông số kỹ thuật RFID	17
Bảng 2.10: Tên và chức năng từng chân RFID.....	17
Bảng 2.11: Thông số kỹ thuật AS608	19
Bảng 2.12: Tên và chức năng từng chân AS608.....	19
Bảng 2.13: Thông số kỹ thuật Keypad 4x4.....	20
Bảng 2.14: Tên và chức năng từng chân Keypad 4x4	21
Bảng 2.15: Thông số kỹ thuật Module Relay	22
Bảng 2.16: Tên và chức năng từng chân Module Relay 4 kênh	23
Bảng 2.17: Thông số kỹ thuật Servo.....	24
Bảng 2.18: Tên và chức năng từng chân Servo	25
Bảng 2.19: Thông số kỹ thuật LCD20x4	26
Bảng 2.20: Tên và chức năng từng chân LCD20x4 I2C.....	27
Bảng 3.1: Bảng kết nối chân giữa STM32 và ESP32	39
Bảng 3.2: Bảng kết nối chân khói cảm biến với STM32	40
Bảng 3.3: Bảng kết nối chân khói điều khiển thiết bị với STM32	42
Bảng 3.4: Bảng kết nối chân khói hiển thị với STM32	43
Bảng 3.5: Bảng kết nối chân khói nhận diện với ESP32	44
Bảng 3.6: Bảng thông số điện áp, dòng và công suất tiêu thụ của hệ thống.....	46

DANH MỤC HÌNH ẢNH

Hình 2.1 Vi điều khiển STM32F407	6
Hình 2.2: Các port GPIO của STM32F407.....	9
Hình 2.3: Vi điều khiển ESP32	9
Hình 2.4: ESP32 và chức năng cơ bản của từng chân	11
Hình 2.5: DHT22 và vị trí của từng chân	12
Hình 2.6: BH1750 và vị trí từng chân.....	14
Hình 2.7: MQ-2 và vị trí từng chân.....	16
Hình 2.8: RFID và vị trí từng chân	17
Hình 2.9: AS608 và vị trí từng chân	19
Hình 2.10: Keypad 4x4 và vị trí từng chân.....	21
Hình 2.11: Module Relay 4 kênh và vị trí từng chân.....	23
Hình 2.12: Servo MG90S và vị trí từng chân	25
Hình 2.13: Module I2C PC8574	27
Hình 2.14: LCD 20x4.....	27
Hình 2.15: Giao tiếp UART giữa hai thiết bị.....	28
Hình 2.16: Giao tiếp I2C giữa hai thiết bị.....	29
Hình 2.17: Giao tiếp SPI giữa hai thiết bị.....	31
Hình 2.18: Hình ảnh minh họa giao diện Realtime Database.....	32
Hình 3.1: Sơ đồ khối hệ thống	36
Hình 3.2: Sơ đồ nguyên lý toàn bộ hệ thống.....	38
Hình 3.3: Sơ đồ nguyên lý khối vi điều khiển	39
Hình 3.4: Sơ đồ kết nối khối cảm biến với STM32	40
Hình 3.5: Sơ đồ kết nối khối điều khiển với STM32.....	41
Hình 3.6: Sơ đồ kết nối khối hiển thị với STM32.....	43
Hình 3.7: Sơ đồ kết nối khối nhận diện với ESP32	44
Hình 3.8: Sơ đồ nguyên lý khối nguồn	46
Hình 3.9: Lưu đồ giải thuật toàn hệ thống	48
Hình 3.10: Lưu đồ giải thuật cho chọn chế độ	50
Hình 3.11: Lưu đồ giải thuật chế độ xác thực	51
Hình 3.12: Lưu đồ giải thuật cập nhập menuPosition.....	52
Hình 3.13: Lưu đồ giải thuật chế độ cảm biến và điều khiển thiết bị	53
Hình 3.14: Lưu đồ giải thuật chế độ cài đặt (setup).....	54
Hình 3.15: Lưu đồ giải thuật cập nhập setup menuPosition	55
Hình 4.1: Mạch PCB thực tế của hệ thống	57
Hình 4.2: Khi cấp nguồn LCD sẽ hiển thị thông tin	58

Hình 4.3: Nhấn ‘A’ vào chế độ xác thực	59
Hình 4.4: Trạng thái ban đầu là đóng cửa.....	60
Hình 4.5: Khi mật khẩu đúng thì cửa sẽ mở	60
Hình 4.6: Khi nhập mật khẩu sai.....	61
Hình 4.7: Trạng thái ban đầu là cửa đóng và yêu cầu quét thẻ.....	61
Hình 4.8: Khi quét đúng thẻ đã lưu cửa sẽ mở	62
Hình 4.9: Khi quét thẻ sai	62
Hình 4.10: Ban đầu đóng và yêu cầu đặt vân tay.....	63
Hình 4.11: Vân tay đúng cửa sẽ mở.....	63
Hình 4.12: Khi vân tay sai.....	64
Hình 4.13: LCD hiển thị dữ liệu cảm biến.....	64
Hình 4.14: Hiển thị dữ liệu cảm biến lên Firebase	65
Hình 4.15: dashboard hiển thị dữ liệu cảm biến	65
Hình 4.16: Biểu đồ cảm biến thay đổi theo dữ liệu	66
Hình 4.17: Hiện tại bốn thiết bị đang bật ‘1’.....	66
Hình 4.18: Hiện tại bốn thiết bị đang bật.....	67
Hình 4.19: Bốn LED trên Relay đang bật.....	67
Hình 4.20: Xác thực mật khẩu để vào chế độ cài đặt.....	68
Hình 4.21: Chế độ cài đặt.....	68
Hình 4.22: Nhập mật khẩu mới.....	69
Hình 4.23: Khi thay đổi mật khẩu thành công	69
Hình 4.24: Thông tin thẻ RFID	70
Hình 4.25: Thông tin vân tay	70
Hình 4.26: Thông tin mật khẩu	70
Hình 4.27: Giao diện chính	71
Hình 4.28: Khu vực hiển thị dữ liệu cảm biến.....	71
Hình 4.29: Khu vực hiển thị biểu đồ.....	72
Hình 4.30: Khu vực điều khiển thiết bị.....	72
Hình 4.31: Giao diện History	73
Hình 4.32: Bảng dữ liệu cảm biến	73
Hình 4.33: Lịch sử điều khiển thiết bị	73
Hình 4.34: Lịch sử cảnh báo	74
Hình 4.35: Các tính năng khác	74
Hình 4.36: Xuất Excel toàn bộ dữ liệu.....	75

CHƯƠNG 1: TỔNG QUAN

1.1. GIỚI THIỆU CHUNG

Trong thời đại công nghệ phát triển mạnh mẽ, sự bùng nổ của Internet of Things (IoT) và trí tuệ nhân tạo (AI) đã mở ra một kỷ nguyên mới, nơi khái niệm Smart Home trở thành xu hướng tất yếu, định hình lại cách con người sống và tương tác với không gian sống. Hệ thống nhà thông minh không chỉ mang lại sự tiện nghi vượt trội thông qua tự động hóa các công việc hàng ngày mà còn đóng vai trò quan trọng trong việc nâng cao an ninh, tối ưu hóa quản lý năng lượng và cải thiện chất lượng cuộc sống trong các hộ gia đình hiện đại. Tuy nhiên, khi nhu cầu về một môi trường sống an toàn, tiện nghi và tiết kiệm ngày càng gia tăng, các giải pháp giám sát và điều khiển truyền thống dần bộc lộ những hạn chế nghiêm trọng, bao gồm khả năng truy cập từ xa hạn chế, thiếu tính linh hoạt trong điều khiển tự động và khó khăn trong việc tích hợp đa dạng các chức năng. Những bất cập này đặt ra yêu cầu cấp thiết về việc phát triển một hệ thống nhà thông minh toàn diện, có khả năng khắc phục các rào cản hiện tại để đáp ứng nhu cầu thực tế.

1.2. MỤC TIÊU ĐỀ TÀI

Đề tài xây dựng một hệ thống nhà thông minh, với mục tiêu thu thập xử lý dữ liệu nhiệt độ, độ ẩm, nồng độ khí gas và ánh sáng trong không gian sống. Hệ thống cũng hướng đến việc nâng cao bảo mật và hỗ trợ điều khiển từ xa thông qua tích hợp các phương thức xác thực bảo mật an toàn, đồng thời sử dụng giao thức Wi-Fi để truyền dữ liệu, cho phép người dùng giám sát và quản lý thiết bị từ xa thông qua giao diện web. Hệ thống sẽ được thử nghiệm để đánh giá hiệu suất, bao gồm độ tin cậy của cảm biến, tốc độ truyền dữ liệu qua Wi-Fi và hiệu quả của phương thức xác thực, từ đó sẽ đưa ra các giải pháp cải tiến hoặc mở rộng trong tương lai.

1.3. TÌNH HÌNH NGHIÊN CỨU TRONG VÀ NGOÀI NƯỚC

1.3.1. Tình hình nghiên cứu ở nước ngoài

Lĩnh vực nhà thông minh đã và đang phát triển mạnh mẽ trên phạm vi toàn cầu, trở thành một trong những ứng dụng trọng tâm của IoT. Các nghiên cứu quốc

té không ngừng khám phá ra các phương pháp mới để nâng cao tính tiện nghi, hiệu quả năng lượng, bảo mật và khả năng tương tác của hệ thống giám sát và điều khiển nhà ở.

Trong bài báo khảo sát [1], Bài báo này tập trung vào các giải pháp công nghệ tiên tiến cho phép kiểm soát và tự động hóa các thiết bị trong nhà thông qua một trung tâm điều khiển trung tâm, điện thoại thông minh, lệnh thoại và IoT. Nghiên cứu này đã phân tích các công nghệ giao tiếp đa dạng như Wi-Fi, Bluetooth, Zigbee cùng với các giao diện điều khiển như ứng dụng di động, trợ lý giọng nói. Ngoài ra, còn đề cập đến các tính năng bảo mật quan trọng nhưng lại có thách thức chính về khả năng tương tác và chi phí cao.

Trong nghiên cứu [2], cung cấp một cái nhìn tổng quan về các hoạt động và chức năng cơ bản của nhà thông minh từ giai đoạn đầu phát triển. Bài báo này đã hệ thống hóa các khái niệm, ứng dụng và công nghệ nền tảng được sử dụng trong các hệ thống nhà thông minh tại thời điểm đó, bao gồm các phương pháp quản lý năng lượng và điều khiển thiết bị. Mặc dù là một công trình mang tính nền tảng, các công nghệ và xu hướng mới nhất như trí tuệ nhân tạo, học máy hay giao thức IoT chuyên biệt sau này vẫn chưa được đề cập tới.

Các nghiên cứu trên quốc tế đã đạt được nhiều thành tựu trong việc phát triển các hệ thống nhà thông minh tiên tiến với nhiều tính năng đa dạng. Tuy nhiên, cũng có nhiều thách thức về mặt chi phí tối ưu, khả năng tích hợp giữa các nền tảng khác nhau và việc xây dựng hệ thống đủ mạnh để đáp ứng mọi nhu cầu phức tạp vẫn là thách thức lớn.

1.3.2. Tình hình nghiên cứu ở trong nước

Tại Việt Nam, lĩnh vực nhà thông minh cũng đang có những bước phát triển tích cực, đặc biệt trong bối cảnh cuộc cách mạng công nghiệp 4.0 và chuyển đổi số được đẩy mạnh. Các nghiên cứu trong nước thường tập trung vào việc ứng dụng các công nghệ IoT hiện có để xây dựng các giải pháp thiết thực, phù hợp với điều kiện và nguồn lực trong nước.

Trong nghiên cứu [3], thiết kế và triển khai một hệ thống nhà thông minh cho phép điều khiển và giám sát các thiết bị từ xa thông qua chuẩn không dây Zigbee. Hệ thống cung cấp giao diện trực quan trên ứng dụng di động, cho phép người dùng điều khiển thủ công và giám sát trạng thái của thiết bị. Mặc dù đã đạt được khả năng điều khiển và giám sát từ xa, nhưng dự án này vẫn còn một số hạn chế như có độ trễ cao khi điều khiển qua ứng dụng, chỉ hỗ trợ trên Android và chưa đạt được mức tự động hóa hoàn toàn.

Trong nghiên cứu [4], đóng góp đáng kể vào lĩnh vực này bằng cách phát triển một mô hình nhà thông minh tích hợp phần cứng và phần mềm dựa trên công nghệ IoT. Hệ thống này tập trung vào việc mô phỏng các chức năng cơ bản như thu thập dữ liệu môi trường, điều khiển thiết bị qua giao diện Web Dashboard, và cung cấp khả năng giám sát thời gian thực. Điểm nổi bật là việc sử dụng các cảm biến môi trường và giao thức truyền thông để tạo ra một mô hình thử nghiệm, minh chứng cho tính khả thi của việc áp dụng IoT trong điều kiện nội địa. Tuy nhiên, nghiên cứu này cũng gặp phải những hạn chế tương tự, bao gồm việc kết nối còn mang tính thủ công và thiếu các kịch bản điều khiển phức tạp, đồng thời mô hình chỉ dừng ở giai đoạn mô phỏng mà chưa được triển khai rộng rãi trong thực tế.

Nhìn chung, các nghiên cứu trong nước đã cho thấy sự phát triển tích cực của các hệ thống nhà thông minh dựa trên IoT với nhiều giải pháp thiết thực đã được triển khai. Tuy nhiên, vẫn còn những thách thức đáng kể về khả năng tự động hóa hoàn toàn, sự đa dạng và linh hoạt trong kết nối không dây, cũng như việc tối ưu hóa các kịch bản điều khiển phức tạp để đáp ứng toàn diện nhu cầu của người dùng.

1.4. PHƯƠNG PHÁP NGHIÊN CỨU

Để thực hiện đề tài thiết kế hệ thống nhà thông minh, quá trình nghiên cứu được tiến hành một cách có hệ thống, bắt đầu bằng việc thiết kế kiến trúc hệ thống theo mô hình phân tán với các khối chính, xác định luồng dữ liệu và giao thức giao tiếp giữa chúng, đồng thời xây dựng sơ đồ khối phần cứng để thể hiện kết nối tổng thể. Tiếp theo, lập trình được thực hiện để phát triển các chức năng cho từng khối trên nền tảng lập trình phù hợp, thiết lập giao tiếp hai chiều giữa các khối, kết nối

với nền tảng lưu trữ đám mây để hiển thị dữ liệu thời gian thực và lập trình cơ chế xác thực đa phương thức. Về giao diện, hệ thống được tích hợp với bảng điều khiển trực tuyến để theo dõi và điều khiển từ xa, đồng thời hiển thị thông tin tại chỗ trên màn hình để người dùng tương tác. Cuối cùng, quá trình thử nghiệm bao gồm kiểm tra từng thành phần riêng lẻ, tích hợp toàn bộ hệ thống để kiểm tra đồng bộ, và đo lường các thông số như độ ổn định, độ trễ truyền dữ liệu, cùng khả năng hoạt động trong môi trường thực tế. Các phương pháp này được thực hiện tuần tự và kết hợp chặt chẽ, đảm bảo hệ thống đạt hiệu quả và đáp ứng yêu cầu đề ra.

1.5. BỘ CỤC BÀI BÁO CÁO

Bộ cục của bài báo cáo gồm những thành phần sau:

Chương 1: Tổng quan

Trình bày lý do chọn đề tài, mục tiêu nghiên cứu, phạm vi nghiên cứu, phương pháp nghiên cứu và bộ cục tổng thể của báo cáo.

Chương 2: Cơ sở lý thuyết

Trình bày các kiến thức nền tảng liên quan đến đề tài, bao gồm khái niệm về hệ thống nhà thông minh, giao tiếp vi điều khiển, công nghệ kết nối không dây, nền tảng lưu trữ đám mây, cùng các thiết bị cảm biến và xác thực được sử dụng trong hệ thống.

Chương 3: Thiết kế hệ thống

Mô tả chức năng tổng thể của hệ thống, thiết kế phần cứng, phần mềm và giao diện giám sát, bao gồm sơ đồ khối, lưu đồ thuật toán và cách các thành phần tương tác với nhau để đảm bảo vận hành hiệu quả.

Chương 4: Kết quả thực nghiệm và đánh giá

Trình bày kết quả triển khai thực tế, bao gồm mô hình phần cứng, phần mềm vận hành, kết quả hiển thị dữ liệu trên giao diện và tại chỗ, cùng với đánh giá hiệu suất và độ tin cậy của hệ thống.

Chương 5: Kết luận và hướng phát triển

Dựa vào kết quả đạt được từ chương 4, đưa ra kết luận về những việc đã làm được và chưa làm được của đề tài. Từ đó đưa ra các hướng để cải thiện đề tài và nâng cao tính ứng dụng để đưa dự án vào thực tiễn.

Với bộ cục gồm 5 chương, đề tài được triển khai một cách tuần tự và logic, đi từ cơ sở lý thuyết, thiết kế, thực nghiệm, đến đánh giá và định hướng phát triển. Cách trình bày chặt chẽ này đảm bảo tính rõ ràng, khoa học và thuận tiện cho quá trình nghiên cứu, triển khai, cũng như tham khảo sau này.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. TỔNG QUAN VỀ HỆ THỐNG GIÁM NHÀ THÔNG MINH

Trong những năm gần đây, công nghệ IoT đã tạo nên những bước tiến vượt bật trong việc hiện đại hóa không gian sống, đặc biệt là thông qua hệ thống giám sát nhà thông minh – đây là một giải pháp ngày càng phổ biến trong các hộ gia đình Việt Nam [5]. Việc áp dụng IoT vào nhà thông minh đã mang lại những lợi ích lớn trong công tác giám sát và quản lý không gian sống một cách hiệu quả [6]. Hệ thống giám sát nhà thông minh dựa trên IoT cho phép người dùng theo dõi từ xa theo thời gian thực các thông số môi trường như nhiệt độ, độ ẩm, ánh sáng và nồng độ khí gây hại trong nhà. Dữ liệu được thu thập liên tục thông qua các cảm biến môi trường, sau đó sẽ gửi về bộ điều khiển trung tâm để xử lý, lưu trữ và đồng bộ với nền tảng điện toán đám mây, phục vụ cho việc phân tích và điều khiển từ xa.

Ngoài ra, hệ thống còn tích hợp các phương thức xác thực thông minh như nhận diện RFID, vân tay và mật khẩu, đảm bảo quản lý an ninh một cách an toàn và chính xác. Nhờ vậy, mà người dùng có thể dễ dàng kiểm soát hoạt động an ninh và vận hành các thiết bị trong nhà, tạo nên không gian sống tiện nghi và an toàn.

2.2. VI ĐIỀU KHIỂN TRONG HỆ THỐNG NHÀ THÔNG MINH

2.2.1. Vi điều khiển STM32



Hình 2.1 Vi điều khiển STM32F407

Hình 2.1, chính là vi điều khiển STM32 thuộc dòng vi điều khiển 32-bit hiệu suất cao, được thiết kế để đáp ứng các ứng dụng nhúng yêu cầu xử lý tín hiệu nhanh chóng và điều khiển chính xác [7]. Trong hệ thống giám sát nhà thông minh, STM32 đóng vai trò xử lý dữ liệu từ các cảm biến môi trường như nhiệt độ, độ ẩm, ánh sáng và chất lượng không khí, đồng thời cũng điều khiển các thiết bị. Vi điều khiển này cũng được trang bị lõi xử lý ARM Cortex-M4 hoạt động ở tần số tối đa là 168MHz, cung cấp khả năng thực hiện các phép toán phức tạp và xử lý tín hiệu số hiệu quả. Bộ nhớ flash lên đến 1MB cho phép lưu trữ chương trình, trong khi SRAM 192KB hỗ trợ lưu trữ dữ liệu tạm thời, đảm bảo hoạt động ổn định trong các điều kiện giám sát thời gian thực. STM32 cũng hỗ trợ nhiều giao thức giao tiếp như UART, I2C và SPI, giúp kết nối dễ dàng với các module khác.

Để có cái nhìn tổng quan hơn về các thông số kỹ thuật của STM32, thì dưới đây là Bảng 2.1 tóm tắt thông số kỹ thuật của vi điều khiển này.

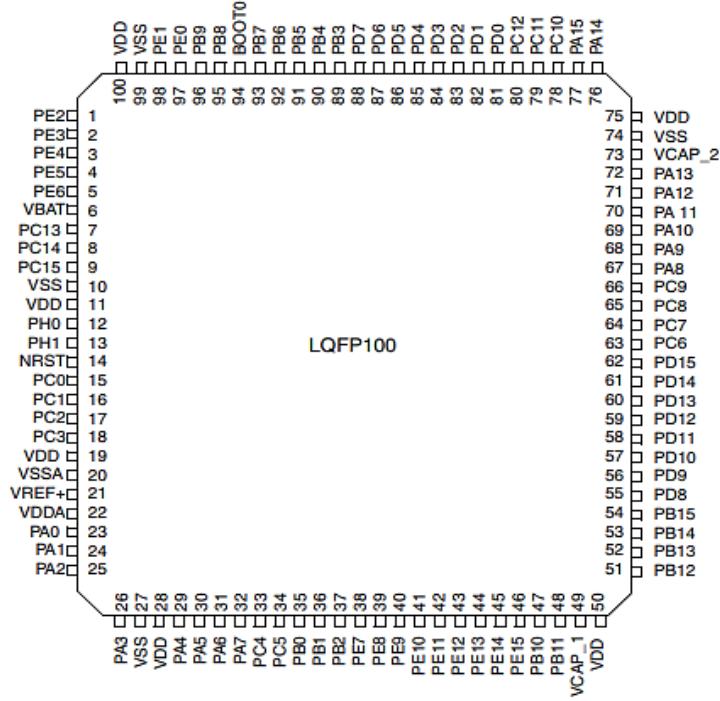
Bảng 2.1: Bảng thông số kỹ thuật chính của STM32F407

Thông số	Mô tả
Kiến trúc	ARM Cortex-M4
Tần số hoạt động	Tối đa 168MHz
Bộ nhớ flash	1MB
Bộ nhớ SRAM	192KB
Giao thức hỗ trợ	UART, I2C, SPI, CAN, USB, Ethernet MAC, SDIO, ADC, DAC, PWM
Số chân GPIO	Có 100 GPIO
Nguồn hoạt động	3V đến 5V

Hình 2.2 mô tả vi điều khiển này có sáu nhóm port GPIO, bao gồm PA, PB, PC, PD, PE và PH, ứng với mỗi nhóm sẽ chứa 16 chân được tổ chức để hỗ trợ các

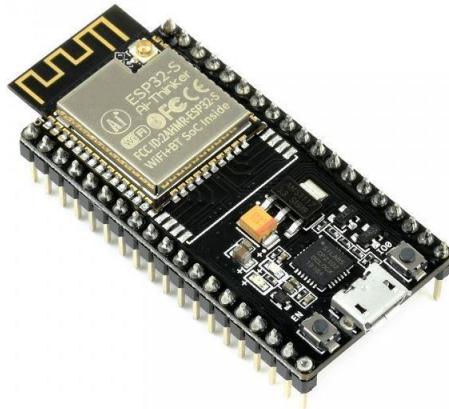
chức năng đa dạng. Các port này có thể cấu hình cho nhiều chế độ khác nhau như sau:

- Ngõ vào (Input): Đọc tín hiệu số mức HIGH và LOW từ các nút nhấn, cảm biến. Có thể cấu hình thêm điện trở kéo lên (pull-up), kéo xuống (pull-down) hoặc không (floating).
- Ngõ ra (Output): Xuất tín hiệu để điều khiển các thiết bị. Có thể cấu hình chế độ đẩy kéo (Push-pull) hoặc cực máng hở (Open-drain). Chế độ open-drain thường được dùng trong giao tiếp bus đa điểm như I2C.
- Tốc độ ngõ ra (Output speed): Cho phép người dùng chọn tốc độ chuyển đổi trạng thái của chân ngõ ra là low, medium, high hoặc very high speed. Các trạng thái này rất hữu ích trong việc kiểm soát nhiều hoặc đáp ứng yêu cầu tốc độ của giao tiếp.
- UART/USART: Giao tiếp nối tiếp không đồng bộ với các module khác.
- I2C: Giao tiếp hai dây tốc độ thấp/trung bình, thường dùng cho các cảm biến hoặc màn hình LCD.
- SPI: Giao tiếp nối tiếp tốc độ cao, dùng cho bộ nhớ Flash ngoài, thường dùng cho màn hình OLED/TFT hoặc các module khác.
- ADC: Chuyển đổi tín hiệu analog thành tín hiệu số.
- DAC: Chuyển đổi tín hiệu số thành tín hiệu tương tự.
- PWM: Tạo tín hiệu xing điều chỉnh rộng, dùng để điều khiển động cơ.
- CAN: Giao tiếp mạng trọng ứng dụng ô tô hoặc trong công nghiệp.
- USB: Giao tiếp với các thiết bị USB ở chế độ host hoặc device.
- Ethernet MAC: Hỗ trợ giao tiếp mạng có dây.
- SDIO: Giao tiếp với thẻ nhớ SD:
- Ngắt ngoài (External Interrupts): Hầu hết các chân GPIO có thể được cấu hình để tạo ra một ngắt khi có sự thay đổi trạng thái (cạnh lên, cạnh xuống) của tín hiệu đầu vào.



Hình 2.2: Các port GPIO của STM32F407

2.2.2. Vi điều khiển ESP32



Hình 2.3: Vi điều khiển ESP32

Hình 2.3 là vi điều khiển ESP32 có tích hợp khả năng giao tiếp không dây, được dùng để kết nối mạng và truyền dữ liệu thời gian thực trong hệ thống giám sát nhà thông minh [8]. Với lõi xử lý ESP32-D0WDQ6 với hai nhân Xtensa 32-bit LX6 hoạt động ở tần số tối đa là 240MHz. ESP32 hỗ trợ xử lý đa nhiệm hiệu quả, cho phép thực hiện đồng thời các tác vụ như thu thập dữ liệu và gửi yêu cầu mạng mà không gây chậm trễ. Bộ nhớ flash 4MB và SRAM 520KB đáp ứng tốt nhu cầu lưu trữ chương trình và dữ liệu tạm thời, đảm bảo cho hệ thống hoạt động mượt

mà. Điểm nổi bật khác đó là khả năng kết nối Wi-Fi theo chuẩn 802.11 b/g/n, cho phép truyền dữ liệu an toàn qua giao thức HTTPS đến nền tảng đám mây như Firebase, hỗ trợ giám sát và điều khiển từ xa. Ngoài ra, ESP32 cũng hỗ trợ các giao thức như UART, I2C và SPI để kết nối với các module khác.

Để có cái nhìn tổng quan hơn về các thông số kỹ thuật của ESP32, thì dưới đây là Bảng 2.2 tóm tắt thông số kỹ thuật của vi điều khiển này.

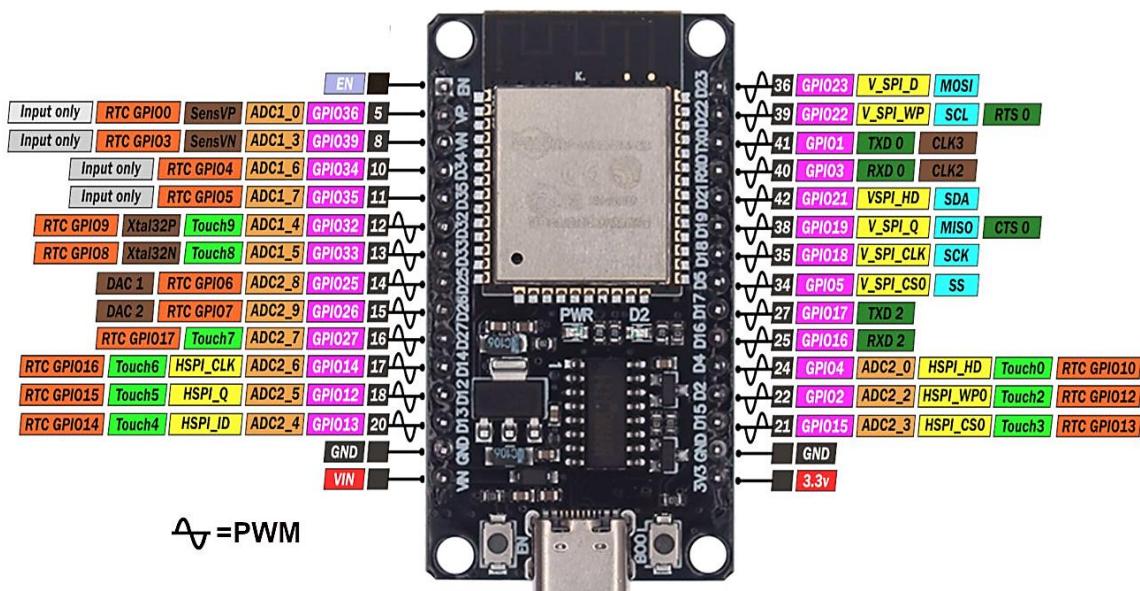
Bảng 2.2: Bảng thông số kỹ thuật chính của ESP32-S

Thông số	Mô tả
Kiến trúc	Dual-core Tensilica Xtensa LX6
Tần số hoạt động	Tối đa 240MHz
Bộ nhớ flash	4MB
Bộ nhớ SRAM	520KB
Giao thức hỗ trợ	UART, I2C, SPI, ADC, DAC, PWM
Kết nối không dây	Wi-Fi 802.11 b/g/n (2.4GHz), Bluetooth v4.2 BR/EDR &BLE.
Số chân GPIO	Có 38 GPIO
Nguồn hoạt động	3V

Hình 2.4, mô tả sơ đồ trực quan của vi điều khiển này, mỗi chân GPIO trên ESP32 có thể hoạt động đa chức năng không chỉ là hoạt động như ngõ vào và ngõ ra như thông thường mà còn có thể cấu hình để đảm nhiệm nhiều vai trò chức năng khác nhau như:

- Ngõ vào (Input) và ngõ ra (Output): Cấu hình làm ngõ vào và ngõ ra để đọc trạng thái mức HIGH và LOW hoặc điều khiển thiết bị.
- ADC: Một số chân có khả năng đọc tín hiệu analog từ cảm biến và chuyển đổi sang tín hiệu số.
- DAC: Một số chân có thể tạo ra tín hiệu analog sau đó chuyển qua tín hiệu số.
- PWM: Hầu hết các chân GPIO đều có thể dùng để tạo tín hiệu PWM, rất hiệu quả trong việc điều khiển động cơ.

- UART: Giao tiếp truyền thông nối tiếp không đồng bộ với các thiết bị khác.
- I2C: Giao tiếp hai dây, thường dùng để kết nối với màn hình LCD.
- SPI: Giao tiếp nối tiếp tốc độ cao, thường dùng cho RFID, màn hình TFT, thẻ nhớ hoặc các module khác.
- Cảm biến chạm (Touch Pins): Một số chân có hỗ trợ tính năng cảm ứng, cho phép phát hiện sự thay đổi khi có sự tiếp xúc, hữu ích cho việc tạo các nút cảm ứng.
- GPIO của RTC (Real-Time Clock): Một số chân có thể được sử dụng khi ESP32 ở chế độ tiết kiệm năng lượng (Deep Sleep), giúp duy trì trạng thái hoặc kích hoạt chip khi từ trạng thái ngủ.



Hình 2.4: ESP32 và chức năng cơ bản của từng chân

2.3. CÁC MODULE CẢM BIẾN, ĐIỀU KHIỂN, HIỂN THỊ

2.3.1. Cảm biến nhiệt độ, độ ẩm

DTH22 là một cảm biến kỹ thuật số được thiết kế để đo nhiệt độ và độ ẩm, phổ biến trong các dự án IoT và hệ thống giám sát môi trường và được sản xuất bởi Aosong Electronics [9]. Cảm biến này sử dụng một thermistor để đo nhiệt độ và một cảm biến độ ẩm điện dung để đo độ ẩm, kết hợp với mạch xử lý tích hợp để chuyển đổi tín hiệu analog thành tín hiệu số. Dữ liệu được truyền qua giao thức

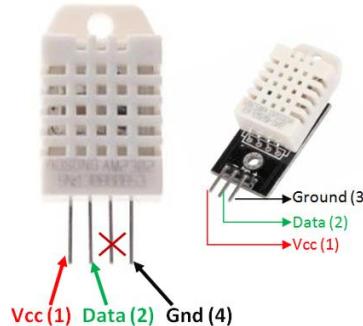
giao tiếp một dây (single-wire protocol), cho phép kết nối đơn giản với các vi điều khiển hoặc hệ thống xử lý tín hiệu.

Các thông số kỹ thuật chi tiết được trình bày trong Bảng 2.3 sau đây, cung cấp cái nhìn tổng quan về đặc điểm kỹ thuật của DHT22:

Bảng 2.3: Thông số kỹ thuật DHT22

Thông số	Chi tiết
Phạm vi đo nhiệt độ	-40°C đến 80°C
Độ chính xác nhiệt độ	$\pm 0.5^\circ\text{C}$
Phạm vi đo độ ẩm	0% đến 100% RH (Relative Humidity)
Độ chính xác độ ẩm	$\pm 2\% \text{RH}$
Dòng tiêu thụ	Trung bình 1mA, tối đa 2.5mA
Thời gian phản hồi	2 giây
Giao tiếp	Giao thức một dây (single-bus)
Nguồn	3.3V đến 5V

DHT22 được thiết kế với 4 chân, trong đó 1 chân là không sử dụng (NC-No Connection). Sơ đồ chân của cảm biến được minh họa như hình sau:



Hình 2.5: DHT22 và vị trí của từng chân

Hình 2.5, giúp người dùng dễ dàng nhận biết được vị trí và chức năng của từng chân để kết nối với vi điều khiển. Chi tiết về tên và chức năng của các chân được trình bày trong bảng sau:

Bảng 2.4: Tên và chức năng từng chân DHT22

Tên	Chức năng
Vcc (1)	Nguồn điện

VCC	3.3V đến 5V
GND	Nối với mass (đất)
DATA	Chân dữ liệu
NC	Không kết nối

Dựa trên Bảng 2.4, có thể thấy mỗi chân của DHT22 đảm nhiệm một vai trò cụ thể trong việc vận hành và kết nối cảm biến. Chân VCC cung cấp nguồn điện cho cảm biến, hoạt động ổn định với điện áp từ 3.3V đến 5V, thường được khuyến nghị sử dụng 5V để đảm bảo hiệu suất tối ưu. Chân DATA chịu trách nhiệm truyền dữ liệu số (nhiệt độ và độ ẩm) dưới dạng tín hiệu xung theo giao thức một dây, cần kết nối thêm một điện trở kéo lên (pull-up resistor) từ $4.7\text{k}\Omega$ đến $10\text{k}\Omega$ với nguồn VCC để đảm bảo tín hiệu ổn định. Chân GND đóng vai trò tham chiếu điện áp, được nối với điểm chung của hệ thống. Chân NC không được sử dụng trong các kết nối thông thường.

2.3.2. Cảm biến cường độ ánh sáng

Cảm biến cường độ ánh sáng là một linh kiện được thiết kế để đo mức độ ánh sáng trong môi trường, thường được dùng trong các ứng dụng cần điều chỉnh ánh sáng hoặc giám sát điều kiện chiếu sáng. Diễn hình là cảm biến BH1750, một cảm biến kỹ thuật số được phát triển bởi ROHM Semiconductor [10], nổi bật nhờ khả năng đo ánh sáng chính xác và tích hợp dễ dàng với các vi điều khiển.

BH1750 sử dụng một cảm biến quang nhạy với ánh sáng khả kiến, có khả năng đo cường độ ánh sáng trong dải từ 0 đến 65.535 lux, tương đương với áng sáng từ ánh trăng mờ đến ánh sáng mặt trời trực tiếp. Cảm biến này chuyển đổi tín hiệu ánh sáng thành tín hiệu số thông qua bộ chuyển đổi analog sang số và truyền dữ liệu qua giao thức I2C.

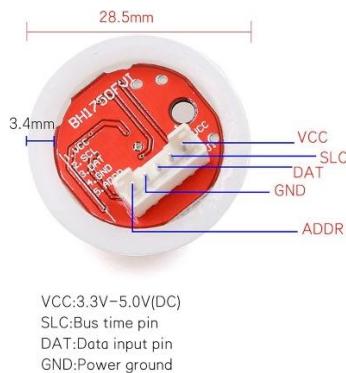
Các thông số kỹ thuật chi tiết được trình bày trong Bảng 2.5 sau đây, cung cấp cái nhìn tổng quan về đặc điểm kỹ thuật của BH1750:

Bảng 2.5: Thông số kỹ thuật BH1750

Thông số	Chi tiết
----------	----------

Phạm vi đo	0 đến 65535 lux
Độ phân giải	1 lux
Độ chính xác	± 3% trong điều kiện ánh sáng ổn định.
Nguồn	3.3V
Dòng tiêu thụ	Trung bình $120 \mu A$, chế độ ngủ $0.1 \mu A$
Thời gian	24ms đến 400ms
Giao tiếp	I2C, địa chỉ mặc định 0x23.

BH1750 được thiết kế với 5 chân, trong đó một số chân đảm nhiệm các vai trò cụ thể để vận hành và kết nối cảm biến. Sơ đồ chân của cảm biến được minh họa như hình sau:



Hình 2.6: BH1750 và vị trí từng chân

Hình 2.6, giúp người dùng dễ dàng nhận biết được vị trí và chức năng của từng chân để kết nối với vi điều khiển. Chi tiết về tên và chức năng của các chân được trình bày trong bảng sau:

Bảng 2.6: Tên và chức năng từng chân BH1750

Tên	Chức năng
VCC	3.3V
GND	Nối với mass (đất)
SCL	Chân xung Clock I2C
SDA	Chân dữ liệu I2C
ADDR	Chân chọn địa chỉ I2C, nối với GND (0x23) hoặc VCC (0x5C).

Dựa trên Bảng 2.6, có thể thấy mỗi chân của BH1750 đảm nhiệm một vai trò cụ thể trong việc vận hành và kết nối cảm biến. Chân VCC cung cấp nguồn 3.3V để cảm biến hoạt động ổn định.. Chân GND đóng vai trò là tham chiếu điện áp, được nối với điểm chung của hệ thống. Chân SCL và SDA là các chân giao tiếp I2C, lần lượt đảm nhiệm vai trò truyền xung Clock và dữ liệu số, cần kết nối thêm điện trở kéo lên (pull-up) với nguồn VCC để đảm bảo tín hiệu ổn định, Chân ADDR cho phép thay đổi địa chỉ I2C của cảm biến.

2.3.3. Cảm biến khí GAS

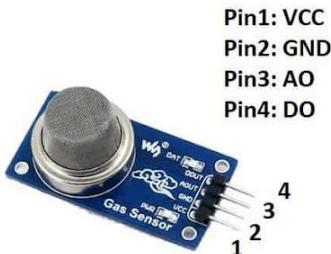
Cảm biến khí gas là một loại linh kiện được thiết kế để phát hiện và đo các loại khí khác nhau trong môi trường, thường được sử dụng trong các ứng dụng giám sát chất lượng không khí hoặc phát hiện rò rỉ khí. Diễn hình là cảm biến MQ-2, được sản xuất bởi Hanwei Electronics [11], nổi bật với khả năng phát hiện nhiều loại khí như metan (CH_4), khí butan (C_4H_{10}), LPG (gas hóa lỏng) và khói.

Các thông số kỹ thuật chi tiết được trình bày trong Bảng 2.7 sau đây, cung cấp cái nhìn tổng quan về đặc điểm kỹ thuật của MQ-2:

Bảng 2.7: Thông số kỹ thuật MQ-2

Thông số	Chi tiết
Phạm vi đo	200 đến 10000 ppm
Độ nhạy	$\pm 5\%$
Nguồn	5V
Dòng tiêu thụ	150 mA.
Thời gian phản hồi	Dưới 10 giây
Giao tiếp	Analog (qua ADC), không cần giao thức số (Digital).

MQ-2 được thiết kế với 4 chân, trong đó một số chân đảm nhiệm các vai trò cụ thể để vận hành và kết nối cảm biến. Sơ đồ chân của cảm biến được minh họa như hình sau:



Hình 2.7: MQ-2 và vị trí từng chân

Hình 2.7, giúp người dùng dễ dàng nhận biết được vị trí và chức năng của từng chân để kết nối với vi điều khiển. Chi tiết về tên và chức năng của các chân được trình bày trong bảng sau:

Bảng 2.8: Tên và chức năng từng chân MQ-2

Tên	Chức năng
VCC	5V
GND	Nối với mass
AO	Chân tín hiệu Analog
DO	Chân tín hiệu Digital

Dựa trên Bảng 2.8, có thể thấy mỗi chân của MQ-2 đảm nhiệm một vai trò cụ thể trong việc vận hành và kết nối cảm biến. Chân VCC cung cấp nguồn 5V để hoạt động ổn định. Chân GND đóng vai trò là tham chiếu điện áp, được nối với điểm chung của hệ thống. Chân AO cung cấp tín hiệu analog tương ứng với nồng độ khí, có thể được kết nối với chân ADC của vi điều khiển để đọc giá trị. Chân DO là đầu ra kỹ thuật số, thường được điều chỉnh bởi một biến trở để thiết lập ngưỡng khí.

2.3.4. Module RFID RC522

Module RFID RC522 là một linh kiện được thiết kế để thực hiện giao tiếp không dây dựa trên công nghệ RFID, thường được sử dụng để đọc và ghi dữ liệu trên thẻ hoặc thẻ từ thông qua sóng radio. Module này sử dụng chip MFRC522 do NXP Semiconductors sản xuất [12], phù hợp cho các ứng dụng như kiểm soát truy

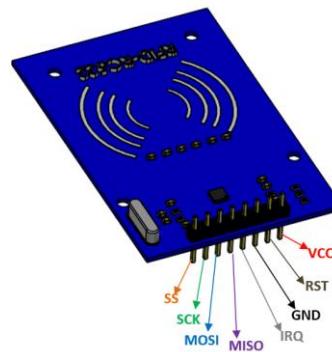
cập hoặc quản lý tài sản. MFRC522 hoạt động ở tần số 13.56MHz, sử dụng giao tiếp SPI để kết nối với các vi điều khiển.

Các thông số kỹ thuật chi tiết được trình bày trong Bảng 2.9 sau đây, cung cấp cái nhìn tổng quan về đặc điểm kỹ thuật của RFID:

Bảng 2.9: Thông số kỹ thuật RFID

Thông số	Chi tiết
Tần số hoạt động	13.56 MHz
Khoảng cách đọc	Tối đa 5 cm.
Nguồn	3.3V
Dòng tiêu thụ	13 - 26 mA.
Hỗ trợ thẻ	Thẻ MIFARE 1K, 4K và thẻ không tiếp xúc
Giao tiếp	SPI

RFID RC522 được thiết kế với 8 chân, trong đó một số chân đảm nhiệm các vai trò cụ thể để vận hành và kết nối cảm biến. Sơ đồ chân của cảm biến được minh họa như hình sau:



Hình 2.8: RFID và vị trí từng chân

Hình 2.8, giúp người dùng dễ dàng nhận biết được vị trí và chức năng của từng chân để kết nối với vi điều khiển. Chi tiết về tên và chức năng của các chân được trình bày trong bảng sau:

Bảng 2.10: Tên và chức năng từng chân RFID

Tên	Chức năng

VCC	3.3V
GND	Nối với mass (đất)
RST	Chân reset
IRQ	Chân ngắt
MISO	Chân dữ liệu vào
MOSI	Chân dữ liệu ra
SCK	Chân xung Clock
SS	Chân chọn chip

Dựa trên Bảng 2.10, có thể thấy mỗi chân của RFID đảm nhiệm một vai trò cụ thể trong việc vận hành và kết nối cảm biến. Chân VCC cung cấp nguồn 3.3V cho module, đảm bảo hoạt động ổn định. Chân GND đóng vai trò tham chiếu điện áp, được nối với điểm chung của hệ thống. Chân SCK, MISO, MOSI và SS là các chân giao tiếp SPI, lần lượt chịu trách nhiệm truyền xung nhịp, dữ liệu ra, dữ liệu vào và chọn chip. Chân RST được dùng để khởi động lại module khi cần, trong khi chân IRQ hỗ trợ ngắt để thông báo khi đọc thẻ, tuy nhiên chân này thường không được sử dụng nếu không cần thiết.

2.3.5. Cảm biến vân tay AS608

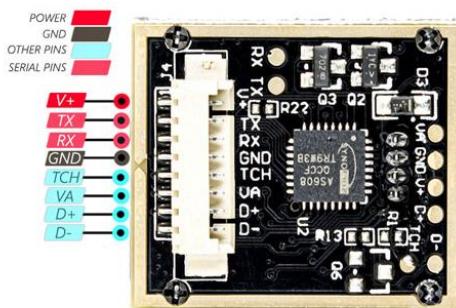
Cảm biến vân tay AS608 là một linh kiện được thiết kế để nhận diện và xác thực vân tay, thường được sử dụng trong các ứng dụng bảo mật như kiểm soát truy cập hoặc xác minh danh tính. Module này sử dụng chip AS608 do Alientek phát triển [13], tích hợp cảm biến quang học (optical sensor) để quét hình ảnh vân tay với độ phân giải cao. AS608 có khả năng lưu trữ và so sánh dấu vân tay thông qua thuật toán xử lý hình ảnh và nhận dạng mẫu tích hợp. Module này giao tiếp với vi điều khiển qua giao thức UART, cho phép truyền dữ liệu nhanh chóng và dễ dàng tích hợp vào các hệ thống nhúng. AS608 cũng đi kèm với bộ nhớ trong, có thể lưu trữ đến 300 dấu vân tay, phù hợp cho các ứng dụng quy mô nhỏ đến trung bình.

Các thông số kỹ thuật chi tiết được trình bày trong Bảng 2.11 sau đây, cung cấp cái nhìn tổng quan về đặc điểm kỹ thuật của AS608:

Bảng 2.11: Thông số kỹ thuật AS608

Thông số	Chi tiết
Dung lượng lưu trữ	13.56 MHz
Tỷ lệ nhận diện sai (FAR)	Dưới 0.001%
Tỷ lệ không nhận diện (FRR)	Dưới 1%
Nguồn	3.3V
Dòng tiêu thụ	80 mA (khi quét), 20 mA (khi nghỉ)
Giao tiếp	UART, baud rate 57600 bps
Thời gian phản hồi	Dưới 1 giây

AS608 được thiết kế với 6 chân, trong đó một số chân đảm nhiệm các vai trò cụ thể để vận hành và kết nối cảm biến. Sơ đồ chân của cảm biến được minh họa như hình sau:



Hình 2.9: AS608 và vị trí từng chân

Hình 2.9, giúp người dùng dễ dàng nhận biết được vị trí và chức năng của từng chân để kết nối với vi điều khiển. Chi tiết về tên và chức năng của các chân được trình bày trong bảng sau:

Bảng 2.12: Tên và chức năng từng chân AS608

Tên	Chức năng
VCC	3.3V
GND	Nối với mass (đất)
TX	Chân giao tiếp UART
RX	Chân giao tiếp UART

TCH	Chân output của cảm biến chạm Touch
VA	Chân cấp nguồn 3.3V cho Touch Sensor
D+	Chân tín hiệu USB+
D-	Chân tín hiệu USB-

Dựa trên Bảng 2.12, có thể thấy mỗi chân của AS608 đảm nhiệm một vai trò cụ thể trong việc vận hành và kết nối cảm biến. Chân VCC cung cấp nguồn điện 3.3V cho module, đảm bảo hoạt động ổn định. Chân GND đóng vai trò tham chiếu điện áp, được nối với điểm chung của hệ thống. Chân TX và RX là các chân giao tiếp UART, lần lượt chịu trách nhiệm truyền và nhận dữ liệu giữa cảm biến và vi điều khiển, cần được kết nối với các chân UART tương ứng (RX và TX) trên vi điều khiển, đảm bảo tốc độ baud rate phù hợp (mặc định là 57600 bps).

2.3.6. Keypad 4x4

Keypad 4x4 là một module đầu vào dạng bàn phím ma trận, được thiết kế để cung cấp giao diện nhập dữ liệu thủ công cho các hệ thống điện tử nhúng. Module này bao gồm 16 phím được sắp xếp trong một ma trận 4 hàng và 4 cột, thường được đánh số từ 0 đến 9 cùng với các ký hiệu như *, #, A, B, C, D, phù hợp cho các ứng dụng cần nhập mã hoặc lệnh. Keypad 4x4 hoạt động dựa trên nguyên lý đóng/mở công tắc khi phím được nhấn, tạo ra kết nối giữa các hàng và cột tương ứng, từ đó vi điều khiển có thể quét và xác định phím nào được nhấn thông qua giao tiếp kỹ thuật số. Module này thường được làm từ vật liệu bền như nhựa hoặc silicon, đảm bảo độ tin cậy trong các điều kiện sử dụng thông thường.

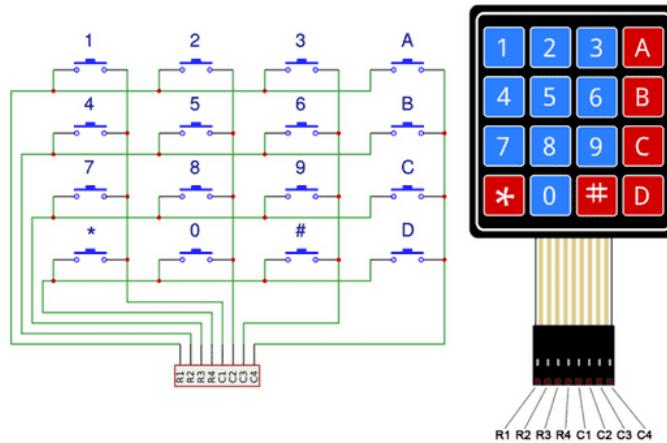
Các thông số kỹ thuật chi tiết được trình bày trong Bảng 2.13 sau đây, cung cấp cái nhìn tổng quan về đặc điểm kỹ thuật của Keypad 4x4:

Bảng 2.13: Thông số kỹ thuật Keypad 4x4

Thông số	Chi tiết
Số phím	16 phím (0-9, A-D, *, #)
Dòng tiêu thụ	Dưới 1 mA
Giao tiếp	GPIO

Kích thước	70 mm x 77 mm
------------	---------------

Keypad 4x4 được thiết kế với 8 chân, trong đó một số chân đảm nhiệm các vai trò cụ thể để vận hành và kết nối. Sơ đồ chân của bàn phím được minh họa như hình sau:



Hình 2.10: Keypad 4x4 và vị trí từng chân

Hình 2.10, giúp người dùng dễ dàng nhận biết được vị trí và chức năng của từng chân để kết nối với vi điều khiển. Chi tiết về tên và chức năng của các chân được trình bày trong bảng sau:

Bảng 2.14: Tên và chức năng từng chân Keypad 4x4

Chân	Ký hiệu	Chức năng chân
Chân hàng (Rows: R1 – R4)	R1	Hàng 1 được kết nối với các phím 1, 2, 3, A.
	R2	Hàng 2 được kết nối với các phím 4, 5, 6, B.
	R3	Hàng 3 được kết nối với các phím 7, 8, 9, C.
	R4	Hàng 4 được kết nối với các phím *, 0, #, D.
	C1	Hàng 5 được kết nối với các phím 1, 4, 7, *.
	C2	Hàng 6 được kết nối với các phím 2,

Chân cột (Columns: C1 – C4)		5, 8, 0.
	C3	Hàng 7 được kết nối với các phím 3, 6, 9, #.
	C4	Hàng 8 được kết nối với các phím A, B, C, D.

Dựa trên Bảng 2.14, có thể thấy các chân của Keypad 4x4 được chia thành hai nhóm chính: R1 đến R4 đại diện cho 4 hàng của ma trận, và C1 đến C4 đại diện cho 4 cột. Khi một phím được nhấn, nó tạo ra kết nối giữa một hàng và một cột tương ứng, cho phép vi điều khiển quét tuần tự các hàng và cột để xác định phím được kích hoạt. Các chân này cần được kết nối với các chân GPIO của vi điều khiển, thường kèm theo điện trở kéo lên (pull-up resistor) hoặc được cấu hình với điện trở nội tại của vi điều khiển để đảm bảo tín hiệu ổn định.

2.3.7. Module Relay 4 kênh

Module Relay 4 kênh là một linh kiện được thiết kế để điều khiển các thiết bị điện thông qua tín hiệu số [14], thường được sử dụng trong các ứng dụng tự động hóa hoặc điều khiển từ xa. Module này bao gồm 4 relay (rơ-le) độc lập, mỗi relay hoạt động như một công tắc điện tử, cho phép bật/tắt các thiết bị điện áp cao (như động cơ, đèn, hoặc quạt) bằng tín hiệu điện áp thấp từ vi điều khiển. Relay trên module thường là loại relay cơ điện (electromechanical relay), hoạt động dựa trên nguyên lý đóng/mở tiếp điểm nhờ từ trường tạo ra bởi cuộn dây bên trong khi có dòng điện chạy qua. Module Relay 4 kênh thường được tích hợp thêm các opto-coupler để cách ly tín hiệu điều khiển và mạch tải, đảm bảo an toàn và giảm nhiễu. Module này tương thích với các vi điều khiển thông qua tín hiệu đầu vào kỹ thuật số (thường là mức logic 3.3V hoặc 5V).

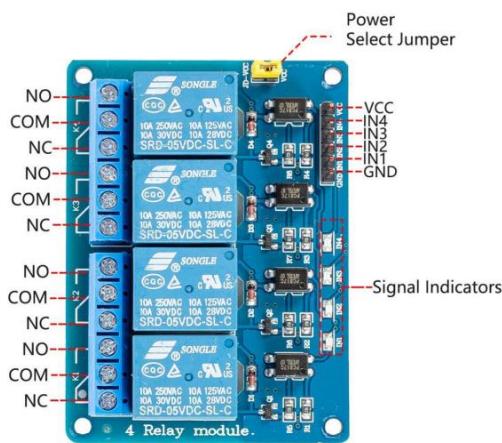
Các thông số kỹ thuật chi tiết được trình bày trong Bảng 2.15 sau đây, cung cấp cái nhìn tổng quan về đặc điểm kỹ thuật của Module Relay 4 kênh:

Bảng 2.15: Thông số kỹ thuật Module Relay

Thông số	Chi tiết
Số kênh	4 kênh

Dòng tiêu thụ	20 mA
Giao tiếp	GPIO
Nguồn	5V
Thời gian phản hồi	Dưới 10 ms

Module Relay 4 kênh được thiết kế với các chân đầu vào điều khiển và các đầu ra tải riêng biệt. Phía đầu vào bao gồm 6 chân để cấp nguồn và điều khiển, trong khi phía đầu ra có 12 chân (3 chân cho mỗi kênh) để kết nối với tải. Sơ đồ chân của module được minh họa như hình sau:



Hình 2.11: Module Relay 4 kênh và vị trí từng chân

Hình 2.11, giúp người dùng dễ dàng nhận biết vị trí và chức năng của từng chân để kết nối với vi điều khiển và thiết bị tải. Chi tiết về tên và chức năng của các chân được trình bày trong bảng sau:

Bảng 2.16: Tên và chức năng từng chân Module Relay 4 kênh

Tên	Chức năng
VCC	Chân cấp nguồn 5V
GND	Chân nối mass
IN1	Điều khiển Relay 1 (Tín hiệu logic High/Low)
IN2	Điều khiển Relay 2 (Tín hiệu logic High/Low)
IN3	Điều khiển Relay 3 (Tín hiệu logic High/Low)
IN4	Điều khiển Relay 4 (Tín hiệu logic High/Low)

COM1-4	Chân chung mỗi kênh (Kết nối tải)
NO1-4	Chân thường mở mỗi kênh
NC1-4	Chân thường đóng mỗi kênh

Dựa trên Bảng 2.16, có thể thấy các chân của Module Relay 4 kênh được chia thành hai nhóm chính: nhóm điều khiển và nhóm đầu ra tải. Chân VCC cung cấp nguồn 5V để vận hành module, trong khi chân GND đóng vai trò tham chiếu điện áp, được nối với điểm chung của hệ thống. Các chân IN1 đến IN4 là các chân điều khiển, nhận tín hiệu kỹ thuật số (mức logic 0 hoặc 1) từ vi điều khiển để kích hoạt hoặc tắt từng relay tương ứng; tín hiệu mức thấp (LOW) thường kích hoạt relay, và mức cao (HIGH) sẽ tắt relay. Phía đầu ra, mỗi kênh có 3 chân: COM (chân chung), NO (thường mở - Normally Open), và NC (thường đóng - Normally Closed), cho phép kết nối linh hoạt với tải tùy theo yêu cầu mạch (thường sử dụng cặp COM-NO để bật/tắt thiết bị).

2.3.8. Động cơ Servo

Động cơ Servo là một loại linh kiện cơ điện được thiết kế để điều khiển góc quay chính xác, thường được sử dụng trong các ứng dụng cần điều khiển vị trí như robot, mô hình điều khiển từ xa, hoặc cơ cấu tự động. Diễn hình là động cơ SG90, một servo nhỏ gọn do TowerPro sản xuất [15], được ưa chuộng nhờ kích thước nhỏ, trọng lượng nhẹ, và khả năng điều khiển đơn giản. SG90 hoạt động dựa trên tín hiệu PWM (Pulse Width Modulation), trong đó độ rộng xung điều khiển góc quay của trực động cơ, thường trong khoảng từ 0° đến 180° . Động cơ này bao gồm một động cơ DC, bộ truyền động bánh răng, và mạch điều khiển tích hợp, cho phép quay chính xác và giữ vị trí ổn định khi không có tín hiệu thay đổi.

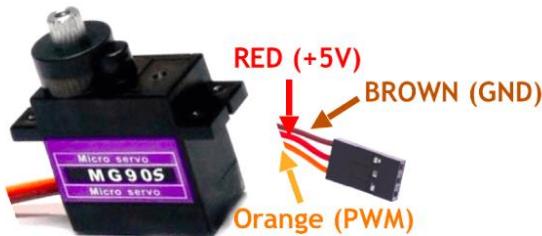
Các thông số kỹ thuật chi tiết được trình bày trong Bảng 2.17 sau đây, cung cấp cái nhìn tổng quan về đặc điểm kỹ thuật của SG90:

Bảng 2.17: Thông số kỹ thuật Servo

Thông số	Chi tiết
Góc quay	0° đến 180°

Tốc độ	0.1 giây/60°
Mô-men xoắn	2.5 kg.cm
Nguồn	5V
Dòng tiêu thụ	10 mA (trạng thái nghỉ), 150 mA (trạng thái hoạt động)
Giao tiếp	PWM

Động cơ Servo SG90 được thiết kế với 3 dây kết nối, đảm nhiệm vai trò cấp nguồn và nhận tín hiệu điều khiển. Sơ đồ dây của động cơ được minh họa như hình sau:



Hình 2.12: Servo MG90S và vị trí từng chân

Hình 2.12, giúp người dùng dễ dàng nhận biết vị trí và chức năng của từng dây để kết nối với các vi điều khiển. Chi tiết về tên và chức năng của các dây được trình bày trong bảng sau:

Bảng 2.18: Tên và chức năng từng chân Servo

Tên	Chức năng
Dây màu đỏ	Dây cấp nguồn 5V.
Dây màu nâu	Dây nối mass
Dây màu cam	Tín hiệu điều khiển PWM

Dựa trên Bảng 2.18, có thể thấy mỗi dây của SG90 đảm nhiệm một vai trò cụ thể trong việc vận hành động cơ. Dây nâu là dây nối đất (GND), được kết nối với điểm chung của hệ thống. Dây đỏ cung cấp nguồn điện cho động cơ, hoạt động ổn định trong khoảng từ 4.8V đến 6V, thường được kết nối với nguồn 5V từ vi điều khiển hoặc nguồn ngoài. Dây cam là dây tín hiệu, nhận tín hiệu PWM từ vi

điều khiển để điều khiển góc quay; tín hiệu PWM có chu kỳ 20ms (tần số 50Hz) và độ rộng xung từ 1ms đến 2ms tương ứng với góc quay từ 0° đến 180°.

2.3.9. Màn hình LCD 20X4 giao tiếp I2C

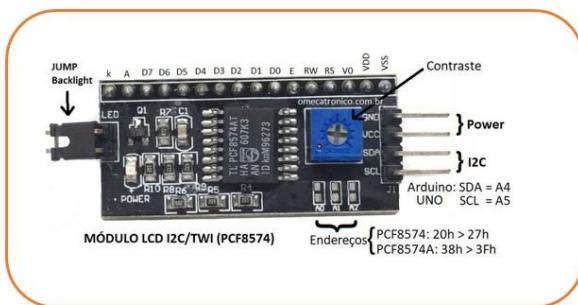
Màn hình LCD 20x4 giao tiếp I2C là một linh kiện hiển thị được thiết kế để trình bày thông tin dưới dạng văn bản, thường được sử dụng trong các ứng dụng nhúng để hiển thị dữ liệu hoặc trạng thái hệ thống. Màn hình này có kích thước 20 ký tự x 4 dòng, cho phép hiển thị tối đa 80 ký tự cùng lúc, phù hợp cho các ứng dụng cần hiển thị nhiều thông tin. Module giao tiếp I2C (thường sử dụng chip PCF8574) được tích hợp để giảm số lượng chân kết nối, cho phép giao tiếp với vi điều khiển qua giao thức I2C với chỉ 2 dây tín hiệu SDA và SCL [16]. Màn hình LCD 20x4 hoạt động dựa trên công nghệ tinh thể lỏng, với đèn nền LED tích hợp để hỗ trợ hiển thị trong điều kiện thiếu sáng, và một điện trở biến (potentiometer) để điều chỉnh độ tương phản.

Để hiểu rõ hơn về khả năng và hiệu suất của linh kiện này, các thông số kỹ thuật chính được trình bày trong Bảng 2.19 sau đây, cung cấp cái nhìn tổng quan về đặc điểm kỹ thuật của màn hình LCD 20x4 giao tiếp I2C:

Bảng 2.19: Thông số kỹ thuật LCD20x4

Thông số	Chi tiết
Kích thước hiển thị	20 ký tự và 4 dòng
Nguồn cấp	5V
Dòng tiêu thụ	Khoảng 40 mA (với đèn nền)
Giao tiếp	I2C, địa chỉ mặc định 0x27

Màn hình LCD 20x4 giao tiếp I2C được thiết kế với 4 chân trên module I2C, đảm nhiệm vai trò cấp nguồn và giao tiếp. Sơ đồ chân của module được minh họa như hình sau:



Hình 2.13: Module I2C PC8574



Hình 2.14: LCD 20x4

Hình 2.13 và 2.14, giúp người dùng dễ dàng nhận biết vị trí và chức năng của từng chân để kết nối với các vi điều khiển. Chi tiết về tên và chức năng của các chân được trình bày trong bảng sau:

Bảng 2.20: Tên và chức năng từng chân LCD20x4 I2C

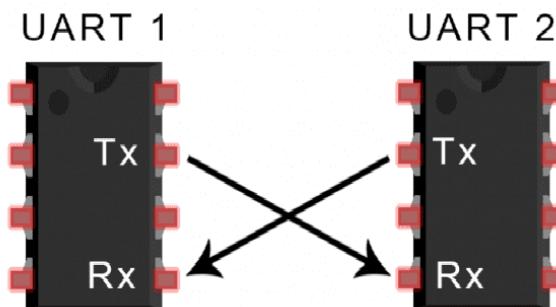
Tên	Chức năng
VCC	Chân cấp nguồn 5V
GND	Chân nối mass
SDA	Chân dữ liệu I2C
SCL	Chân tín hiệu xung Clock

Dựa trên Bảng 2.20, có thể thấy mỗi chân của module LCD 20x4 giao tiếp I2C đảm nhiệm một vai trò cụ thể trong việc vận hành và kết nối. Chân VCC cung cấp nguồn điện 5V để vận hành màn hình và module I2C. Chân GND đóng vai trò tham chiếu điện áp, được nối với điểm chung của hệ thống. Chân SDA và SCL là các chân giao tiếp I2C, lần lượt chịu trách nhiệm truyền dữ liệu và xung nhịp, cần kết nối với các chân I2C tương ứng trên vi điều khiển (thường yêu cầu điện trở kéo lên từ $2.2k\Omega$ đến $4.7k\Omega$ với nguồn VCC để đảm bảo tín hiệu ổn định). Địa chỉ I2C của module (mặc định là 0x27) có thể được điều chỉnh thông qua các chân địa chỉ trên module I2C nếu cần.

2.4. CÁC CHUẨN TRUYỀN THÔNG SỬ DỤNG

2.4.1. Giao tiếp UART

Giao tiếp UART (Universal Asynchronous Receiver-Transmitter) là một chuẩn truyền thông nối tiếp không đồng bộ, được sử dụng rộng rãi để trao đổi dữ liệu giữa vi điều khiển và các thiết bị ngoại vi. Giao thức này không yêu cầu tín hiệu đồng bộ hóa như clock bên ngoài, thay vào đó sử dụng tốc độ baud rate được cấu hình trước để đảm bảo đồng bộ giữa thiết bị gửi và nhận. UART truyền dữ liệu theo từng byte, với mỗi byte bao gồm một bit bắt đầu (start bit), các bit dữ liệu (thường 8 bit) và một hoặc hai bit dừng (stop bit), đôi khi kèm theo bit kiểm tra chẵn/lẻ (parity bit) để phát hiện lỗi. Giao thức này nổi bật với tính đơn giản và hiệu quả trong các ứng dụng giao tiếp tầm ngắn, như kết nối với cảm biến, module GPS, hoặc thiết bị ngoại vi như máy tính qua cổng COM.



Hình 2.15: Giao tiếp UART giữa hai thiết bị

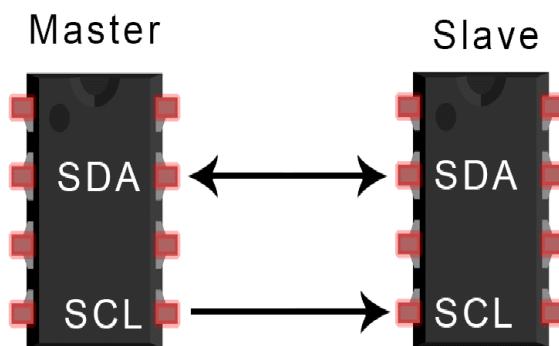
Tầng vật lý của giao tiếp UART xác định các đặc điểm phần cứng cần thiết để truyền dữ liệu. Giao thức này sử dụng hai dây chính: TX (transmit) để gửi dữ liệu và RX (receive) để nhận dữ liệu, cùng với các dây nguồn và đất để hoàn tất kết nối. Tín hiệu trên các dây được biểu diễn dưới dạng mức điện áp logic, thường là 0V cho mức thấp (logic 0) và 3.3V hoặc 5V cho mức cao (logic 1), tùy thuộc vào vi điều khiển hoặc thiết bị sử dụng. UART không hỗ trợ truyền dữ liệu đa điểm (multi-drop), mà chỉ hoạt động theo mô hình điểm-điểm (point-to-point), nghĩa là mỗi kết nối UART chỉ liên kết hai thiết bị. Khoảng cách truyền dữ liệu thường bị giới hạn dưới 15m, tùy thuộc vào tốc độ baud rate và chất lượng dây dẫn, do không có cơ chế đồng bộ hóa phức tạp hoặc xử lý nhiễu.

Giao tiếp UART có nhiều ưu điểm như cấu hình đơn giản, chi phí thấp, và hỗ trợ truyền dữ liệu full-duplex, làm cho giao thức này trở thành lựa chọn phổ biến trong các ứng dụng cơ bản. Tuy nhiên, UART cũng có nhược điểm như khoảng cách truyền bị giới hạn, không phù hợp cho truyền dữ liệu đa điểm, và dễ bị nhiễu ở tốc độ cao hoặc trên dây cáp dài nếu không có biện pháp bảo vệ tín hiệu.

UART được sử dụng rộng rãi trong các ứng dụng lý thuyết như giao tiếp giữa vi điều khiển và máy tính qua cổng COM, kết nối với các cảm biến có đầu ra nối tiếp như cảm biến nhiệt độ, hoặc tích hợp với module GPS để nhận dữ liệu vị trí. Giao thức này cũng phổ biến trong việc lập trình và gỡ lỗi hệ thống nhưng thông qua các cổng nối tiếp.

2.4.2. Giao tiếp I2C

Giao tiếp I2C (Inter-Integrated Circuit) là một chuẩn truyền thông nối tiếp đồng bộ, để kết nối các thiết bị điện tử với nhau, thường trong khoảng cách ngắn. Giao thức này được thiết kế để truyền dữ liệu giữa một thiết bị chủ (master) và nhiều thiết bị phụ (slave) thông qua hai dây tín hiệu chính, giúp giảm số lượng chân kết nối trên vi điều khiển. I2C sử dụng cơ chế địa chỉ hóa để xác định thiết bị cần giao tiếp, với mỗi thiết bị phụ có một địa chỉ duy nhất. Giao thức này hỗ trợ tốc độ truyền dữ liệu từ 100 kbps (chế độ chuẩn) đến 400 kbps (chế độ nhanh) và thậm chí cao hơn trong các phiên bản mới.



Hình 2.16: Giao tiếp I2C giữa hai thiết bị

Tầng vật lý của giao tiếp I2C bao gồm hai dây chính: SDA (Serial Data) để truyền dữ liệu và SCL (Serial Clock) để cung cấp tín hiệu đồng bộ. Cả hai dây đều

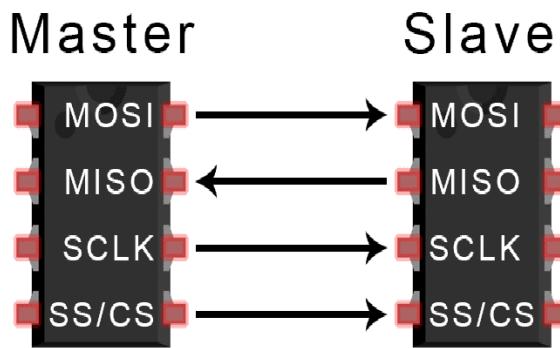
được kết nối với nguồn điện áp 3.3V hoặc 5V thông qua các điện trở kéo lên có giá trị từ $2.2k\Omega$ đến $4.7k\Omega$, đảm bảo tín hiệu ở mức cao khi không có thiết bị nào kéo xuống mức thấp. I2C hoạt động theo mô hình đa điểm (multi-drop), cho phép một thiết bị chủ kết nối với nhiều thiết bị phụ trên cùng một bus, miễn là mỗi thiết bị có địa chỉ duy nhất. Tín hiệu trên bus được biểu diễn dưới dạng mức logic. Khoảng cách truyền dữ liệu thường bị giới hạn dưới 1m do điện dung của dây dẫn và nhiễu tín hiệu, đặc biệt ở tốc độ cao.

Giao tiếp I2C có ưu điểm nổi bật như chỉ sử dụng hai dây tín hiệu, hỗ trợ kết nối đa thiết bị (multi-slave) trên cùng một bus, và dễ dàng mở rộng hệ thống nhờ cơ chế địa chỉ hóa. Tuy nhiên, nhược điểm của giao thức này bao gồm tốc độ truyền dữ liệu thấp hơn so với SPI, dễ bị ảnh hưởng bởi nhiễu ở khoảng cách xa, và cần quản lý địa chỉ cẩn thận để tránh xung đột giữa các thiết bị phụ.

I2C được sử dụng rộng rãi trong các ứng dụng lý thuyết như kết nối vi điều khiển với cảm biến môi trường như cảm biến nhiệt độ, độ ẩm, màn hình LCD có giao tiếp I2C, hoặc bộ nhớ EEPROM để lưu trữ dữ liệu. Giao thức này cũng phổ biến trong các hệ thống nhúng cần giao tiếp với nhiều thiết bị ngoại vi mà không yêu cầu tốc độ truyền dữ liệu cao, chẳng hạn như trong các thiết bị đo lường hoặc điều khiển cơ bản.

2.4.3. Giao tiếp SPI

Giao tiếp SPI (Serial Peripheral Interface) là một chuẩn truyền thông nối tiếp đồng bộ, truyền dữ liệu nhanh giữa vi điều khiển và các thiết bị ngoại vi. Giao thức này sử dụng một thiết bị chủ (master) để điều khiển một hoặc nhiều thiết bị phụ (slave) thông qua tín hiệu clock đồng bộ, cho phép truyền dữ liệu với tốc độ cao hơn so với nhiều giao thức khác. SPI hoạt động dựa trên bốn dây chính, bao gồm tín hiệu dữ liệu, xung nhịp, và chọn thiết bị, giúp tối ưu hóa hiệu suất trong các ứng dụng yêu cầu băng thông lớn. Giao thức này nổi bật với khả năng truyền dữ liệu toàn song công (full-duplex), cho phép gửi và nhận dữ liệu đồng thời.



Hình 2.17: Giao tiếp SPI giữa hai thiết bị

Tầng vật lý của giao tiếp SPI bao gồm bốn dây chính: SCLK (Serial Clock) để cung cấp tín hiệu đồng bộ, MOSI (Master Out Slave In) để gửi dữ liệu từ thiết bị chủ đến thiết bị phụ, MISO (Master In Slave Out) để nhận dữ liệu từ thiết bị phụ về thiết bị chủ, và SS/CS (Slave Select/Chip Select) để chọn thiết bị phụ cụ thể. Tín hiệu trên các dây được biểu diễn dưới dạng mức logic 0V cho mức thấp và 3.3V hoặc 5V cho mức cao, tùy thuộc vào điện áp hoạt động của hệ thống. SPI hỗ trợ kết nối đa điểm thông qua nhiều chân SS/CS, nhưng mỗi thiết bị phụ cần một chân SS riêng biệt trên thiết bị chủ. Khoảng cách truyền dữ liệu thường giới hạn dưới 1m do nhiễu và điện dung dây dẫn, và hiệu suất có thể giảm nếu không sử dụng cáp chất lượng cao.

Giao tiếp SPI có ưu điểm như tốc độ truyền dữ liệu cao, hỗ trợ full-duplex, và dễ dàng tích hợp với các thiết bị ngoại vi hiệu suất cao. Tuy nhiên, nhược điểm của giao thức này là yêu cầu nhiều dây kết nối tối thiểu 4 dây, không hỗ trợ địa chỉ hóa như I2C và cần một chân SS riêng cho mỗi thiết bị phụ, làm tăng độ phức tạp khi kết nối nhiều thiết bị.

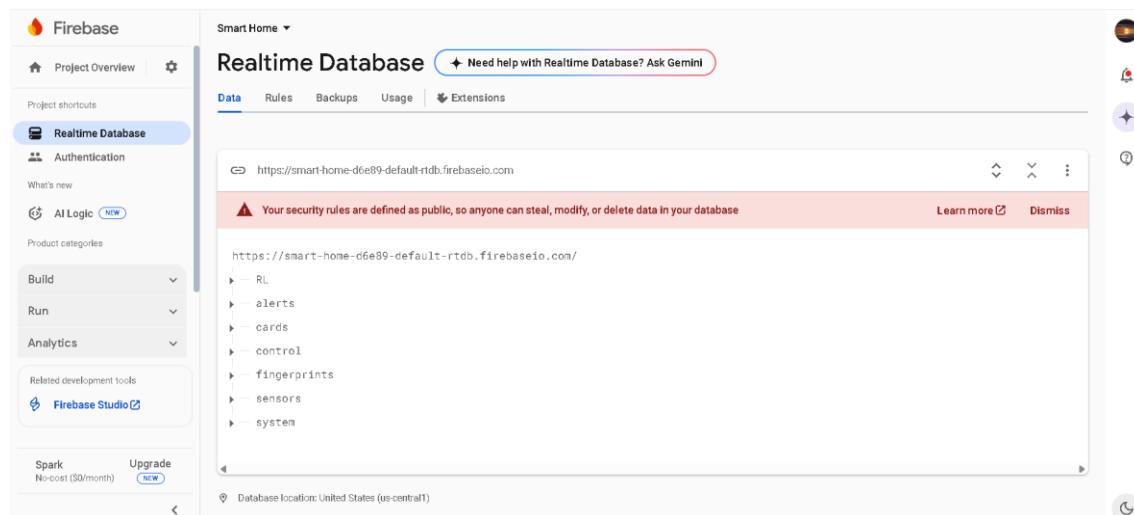
SPI được sử dụng rộng rãi trong các ứng dụng lý thuyết như kết nối vi điều khiển với bộ nhớ flash, màn hình OLED hoặc TFT, cảm biến áp suất, hoặc các module giao tiếp RF như RFM95. Giao thức này cũng phổ biến trong các hệ thống nhưng yêu cầu truyền dữ liệu nhanh và đáng tin cậy, chẳng hạn như trong các thiết bị đo lường chính xác hoặc giao tiếp với các thiết bị ngoại vi tốc độ cao.

2.5. LUU TRU DUC LIEU VÀ GIAO DIEN NGUOI DUNG

2.5.1. Giới thiệu về Firebase Realtime Database

Firebase là một nền tảng phát triển ứng dụng do Google phát triển, cung cấp nhiều dịch vụ trên đám mây để hỗ trợ việc xây dựng và quản lý các ứng dụng hiện đại. Nền tảng này bao gồm các công cụ như xác thực người dùng, lưu trữ tệp và đặc biệt là các quản lý dữ liệu, trong đó nổi bật là cơ sở dữ liệu thời gian thực.

Firebase Realtime Database, một thành phần cốt lõi của nền tảng này, đây là một cơ sở dữ liệu NoSQL được thiết kế để lưu trữ dữ liệu dưới dạng cấu trúc JSON linh hoạt. Cơ sở dữ liệu này cho phép đồng bộ hóa tức thời giữa các thiết bị kết nối, giúp dữ liệu như giá trị cảm biến, trạng thái thiết bị hoặc thông số hệ thống được cập nhật ngay lập tức trên toàn hệ thống. Với mô hình sự kiện, Firebase Realtime Database hỗ trợ theo dõi liên tục các thay đổi dữ liệu, chẳng hạn như khi một cảm biến gửi giá trị mới hoặc trạng thái thiết bị thay đổi, làm cho nó trở thành công cụ quan trọng trong các hệ thống nhúng cần quản lý dữ liệu động và phản hồi nhanh chóng.



Hình 2.18: Hình ảnh minh họa giao diện Realtime Database

2.5.2. Giới thiệu Web Dashboard giám sát và điều khiển

Web Dashboard là một giao diện người dùng dựa trên trình duyệt web, được tạo ra để hỗ trợ giám sát dữ liệu thời gian thực và điều khiển thiết bị từ xa một cách

trực quan và hiệu quả. Giao diện này thường được xây dựng dựa trên ba công nghệ cốt lõi của phát triển web là HTML, CSS, và JavaScript.

- HTML (HyperText Markup Language) là ngôn ngữ đánh dấu dùng để xây dựng cấu trúc cơ bản của trang web, định nghĩa các thành phần như tiêu đề, bảng hoặc nút bấm.
- CSS (Cascading Style Sheets) là công nghệ định dạng, giúp tạo phong cách cho giao diện như điều chỉnh màu sắc, kích thước chữ hoặc bố cục để trang web trở nên trực quan và thân thiện hơn.
- JavaScript là ngôn ngữ lập trình phía trình duyệt, cho phép xử lý các tương tác động như là cập nhật dữ liệu mà không cần tải lại trang hoặc phản hồi khi người dùng nhấn nút.

Nhờ sự kết hợp của HTML, CSS, và JavaScript mà Web Dashboard có thể hiển thị dữ liệu thời gian thực, chẳng hạn như giá trị cảm biến hoặc trạng thái thiết bị, đồng thời cho phép người dùng gửi các lệnh điều khiển, như bật/tắt thiết bị hoặc thay đổi cài đặt, thông qua các thao tác đơn giản trên trình duyệt.

Để kết nối với Firebase, Web Dashboard thường sử dụng API hoặc SDK của Firebase, cho phép giao diện truy cập dữ liệu từ Realtime Database và hiển thị chúng dưới dạng văn bản, biểu đồ hoặc trạng thái thiết bị. Đồng thời, người dùng có thể gửi lệnh điều khiển thông qua giao diện, với các lệnh này được truyền ngược lại qua Firebase đến thiết bị. Với khả năng truy cập từ bất kỳ đâu có kết nối mạng, Web Dashboard trở thành một công cụ quan trọng trong việc quản lý và vận hành các hệ thống nhúng từ xa.

CHƯƠNG 3: THIẾT KẾ HỆ THỐNG

3.1. YÊU CẦU THIẾT KẾ HỆ THỐNG

Hệ thống nhà thông minh được thiết kế để đáp ứng các chức năng cốt lõi nhằm đảm bảo quản lý ngôi nhà hiệu quả, tiện nghi và an toàn. Đầu tiên, hệ thống thực hiện việc thu thập dữ liệu môi trường trong nhà, bao gồm ánh sáng, nhiệt độ và độ ẩm, chất lượng không khí, nhằm đảm bảo điều kiện sống tối ưu cho người dùng. Dữ liệu môi trường được hiển thị thời gian thực trên màn hình LCD tại chỗ, giúp người dùng dễ dàng theo dõi tình trạng môi trường ngay tại nhà.

Để hỗ trợ giám sát và điều khiển từ xa, hệ thống truyền dữ liệu từ STM32F407 sang ESP32 qua giao tiếp UART hai chiều, đảm bảo kết nối ổn định trong phạm vi ngôi nhà. ESP32 sau đó gửi dữ liệu lên Firebase Realtime Database qua kết nối Wi-Fi, cho phép lưu trữ và truy cập từ xa để phân tích. Toàn bộ dữ liệu môi trường, cùng với trạng thái thiết bị trong nhà, được hiển thị trên Web Dashboard, giúp người dùng quản lý ngôi nhà mọi lúc, mọi nơi một cách tiện lợi. Đồng thời, tại cửa ra vào, hệ thống tích hợp xử lý để xác thực người dùng qua mật khẩu, vân tay hoặc thẻ UID. Khi xác thực thành công, hệ thống sẽ điều khiển servo để mở cửa, trong khi thông tin xác thực và cài đặt được hiển thị trên LCD. Các chức năng này kết hợp với nhau để tạo ra một giải pháp nhà thông minh toàn diện, hiện đại.

3.2. ĐẶC TẢ HỆ THỐNG

3.2.1. Chức năng hệ thống

Đầu tiên, hệ thống có chức năng giám sát môi trường, thu thập dữ liệu thời gian thực từ các cảm biến ánh sáng, nhiệt độ và độ ẩm, khí gas. Các cảm biến này kết nối với STM32 để đo lường các thông số, sau đó dữ liệu được truyền qua giao tiếp UART hai chiều đến ESP32 để xử lý và gửi lên Firebase Realtime Database. Chức năng này cho phép người dùng theo dõi trạng thái môi trường của ngôi nhà, phát hiện các bất thường như rò rỉ khí gas để đưa ra cảnh báo kịp thời.

Thứ hai, hệ thống hỗ trợ xác thực người dùng để kiểm soát truy cập vào ngôi nhà. ESP32 xử lý dữ liệu từ khói nhận diện, bao gồm Keypad4x4 để nhập mật khẩu, cảm biến vân tay AS608 để quét vân tay và RFID để đọc thẻ UID. Dữ liệu xác thực được so sánh với thông tin lưu trữ trên Firebase. Nếu xác thực thành công, ESP32 gửi lệnh đến STM32 qua UART để điều khiển servo mở cửa. Chức năng này đảm bảo an ninh, chỉ cho phép người dùng được ủy quyền truy cập vào nhà.

Thứ ba, hệ thống cho phép điều khiển thiết bị trong nhà từ xa thông qua Web Dashboard. Người dùng có thể bật/tắt thiết bị như đèn thông qua module relay 4 kênh. Lệnh điều khiển được gửi từ Web Dashboard qua Firebase đến ESP32, sau đó truyền qua UART đến STM32 để thực thi. Chức năng này tăng cường sự tiện lợi, giúp người dùng quản lý ngôi nhà mọi lúc, mọi nơi.

Thứ tư, hệ thống tích hợp giao tiếp cloud với Firebase Realtime Database để lưu trữ dữ liệu cảm biến, thông tin xác thực và lịch sử điều khiển. ESP32 đảm nhiệm vai trò kết nối Wi-Fi, gửi dữ liệu lên Firebase theo thời gian thực và nhận lệnh từ Web Dashboard. Chức năng này đảm bảo dữ liệu được lưu trữ an toàn và có thể truy cập từ xa, hỗ trợ khả năng mở rộng trong tương lai.

Thứ năm, hệ thống hiển thị thông tin trên màn hình LCD, bao gồm dữ liệu cảm biến, trạng thái xác thực và thông báo hệ thống. STM32 điều khiển LCD để cung cấp giao diện trực quan tại chỗ, giúp người dùng dễ dàng theo dõi tình trạng hệ thống mà không cần truy cập Web Dashboard.

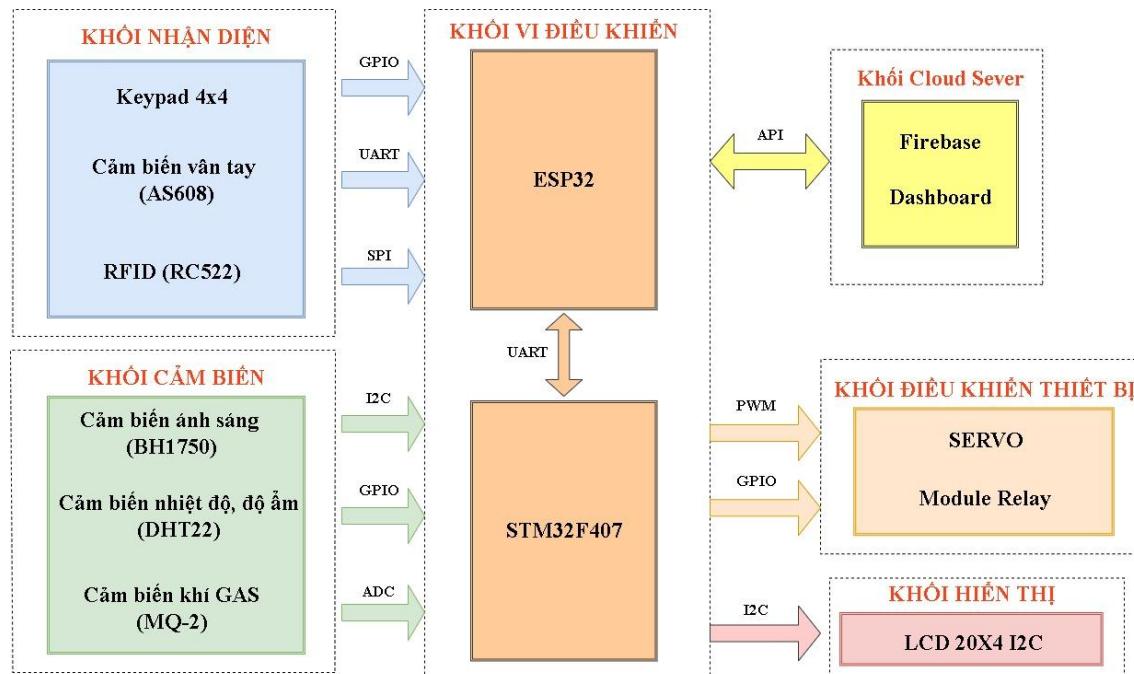
Cuối cùng, hệ thống hỗ trợ chế độ cài đặt để người dùng tùy chỉnh thông tin. Người dùng truy cập chế độ này bằng phím 'D' trên Keypad, sau khi xác thực quyền truy cập. Menu cài đặt hiển thị trên LCD, cho phép người dùng thay đổi mật khẩu, đăng ký vân tay mới qua AS608 hoặc thêm thẻ RFID. Dữ liệu cài đặt được ESP32 gửi lên Firebase để lưu trữ, đảm bảo thông tin được đồng bộ hóa và sử dụng trong các lần xác thực sau.

Hệ thống vận hành theo ba chế độ chính:

- **Chế độ giám sát cảm biến và điều khiển từ xa:** Kích hoạt bằng phím 'C', cho phép người dùng theo dõi dữ liệu môi trường và điều khiển thiết bị qua Web Dashboard.
- **Chế độ xác thực:** Kích hoạt bằng phím 'A', yêu cầu xác thực qua mật khẩu, vân tay hoặc thẻ UID trước khi mở cửa hoặc truy cập các chế độ khác.
- **Chế độ cài đặt:** Kích hoạt bằng phím 'D', cho phép tùy chỉnh thông tin xác thực như đã mô tả.

Các chức năng này kết hợp với nhau để mang lại một giải pháp nhà thông minh toàn diện, hiện đại, đáp ứng nhu cầu quản lý ngôi nhà một cách hiệu quả.

3.2.2. Sơ đồ khái niệm hệ thống



Hình 3.1: Sơ đồ khái niệm hệ thống

Hình 3.1 là sơ đồ khái niệm hệ thống nhà thông minh được thiết kế để thể hiện mối quan hệ và kết nối giữa các khái niệm, đảm bảo hoạt động đồng bộ và hiệu quả. Hệ thống bao gồm các khái niệm chính sau:

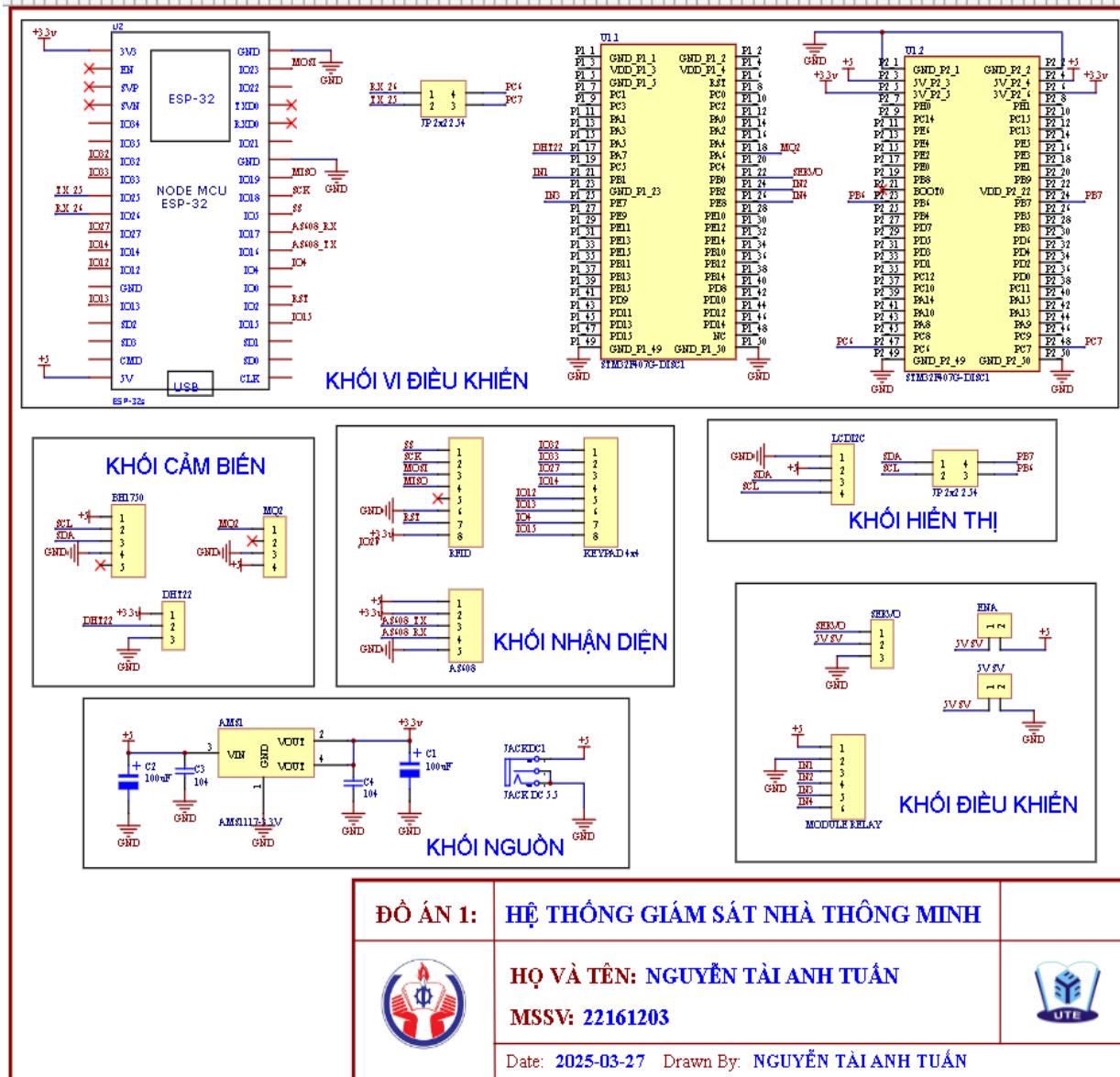
- Khối vi điều khiển: Gồm STM32F407 và ESP32, đóng vai trò trung tâm xử lý của hệ thống. STM32 kết nối với các cảm biến, khái niệm thị, và khái niệm điều khiển thiết bị. Trong khi ESP32 xử lý khái niệm nhận diện và giao tiếp với

cloud server. Hai vi điều khiển giao tiếp với nhau qua giao tiếp UART hai chiều, cho phép truyền dữ liệu môi trường và lệnh điều khiển giữa STM32 và ESP32.

- Khối cảm biến: Bao gồm cảm biến ánh sáng BH1750, cảm biến nhiệt độ và độ ẩm DHT22 và cảm biến khí gas MQ2. Các cảm biến này kết nối với STM32F407 qua giao tiếp I2C (BH1750), GPIO (DHT22) và ADC (MQ2), thu thập dữ liệu môi trường và gửi đến STM32 để xử lý trước khi truyền sang ESP32.
- Khối nhận diện: Gồm Keypad4x4 để nhập mật khẩu, cảm biến vân tay AS608 để quét vân tay và module RFID để đọc thẻ UID. Các thành phần này kết nối với ESP32 qua giao tiếp UART (AS608), SPI (RFID) và GPIO (Keypad), xử lý dữ liệu xác thực và so sánh với thông tin từ Firebase.
- Khối hiển thị: Sử dụng màn hình LCD20x4 I2C, kết nối với STM32 qua giao tiếp I2C, hiển thị dữ liệu cảm biến, trạng thái xác thực, thông báo hệ thống và menu cài đặt khi người dùng tương tác qua Keypad.
- Khối điều khiển thiết bị: Bao gồm servo để mở cửa và module relay 4 kênh để bật/tắt thiết bị như đèn. Các thành phần này kết nối với STM32 qua GPIO và PWM (servo), nhận lệnh từ STM32 để thực thi các tác vụ điều khiển dựa trên dữ liệu từ ESP32 hoặc Web Dashboard.
- Khối cloud server: Gồm Firebase Realtime Database và Web Dashboard. ESP32 kết nối với Firebase qua Wi-Fi, gửi dữ liệu cảm biến, thông tin xác thực và lịch sử điều khiển lên đám mây, đồng thời nhận lệnh từ Web Dashboard để truyền về STM32 qua UART.

3.3. THIẾT KẾ PHẦN CỨNG

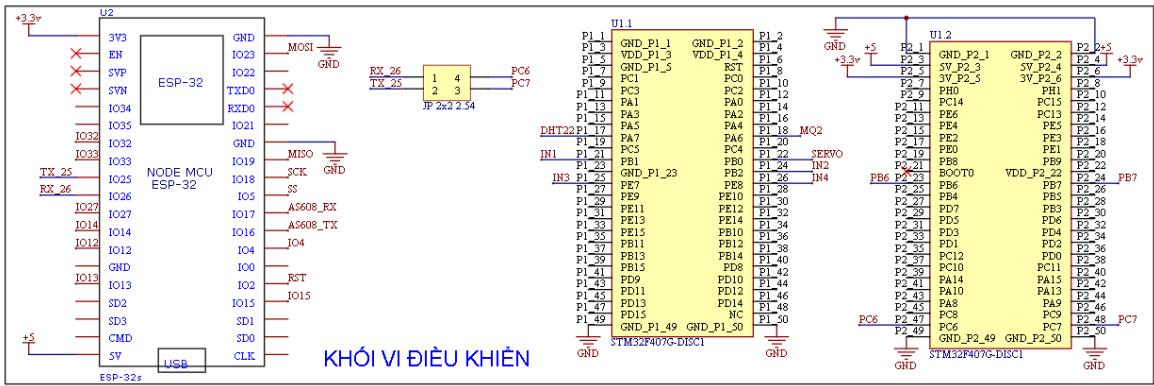
3.3.1. Sơ đồ nguyên lý toàn bộ hệ thống



Hình 3.2: Sơ đồ nguyên lý toàn bộ hệ thống

Hình 3.2 là sơ đồ nguyên lý toàn bộ hệ thống nhà thông minh, thể hiện mối quan hệ giữa các khối chức năng chính như vi điều khiển, cảm biến, nhận diện, hiển thị, điều khiển thiết bị và giao tiếp với cloud server.

3.3.2. Khối vi điều khiển



Hình 3.3: Sơ đồ nguyên lý khói vi điều khiển

Hình 3.3 là sơ đồ nguyên lý của khói vi điều khiển trong hệ thống nhà thông minh có hai vi điều khiển chính được sử dụng STM32 và ESP32, được kết nối và phân chia nhiệm vụ. Hình 3.3 cung cấp cái nhìn tổng quan về cấu trúc phần cứng của khói này, bao gồm giao tiếp UART hai chiều giữa hai vi điều khiển, cũng như các chân kết nối với các khói cảm biến, nhận diện, hiển thị và điều khiển thiết bị. Dựa trên sơ đồ này, về kết nối chân được trình bày trong Bảng 3.1 dưới đây:

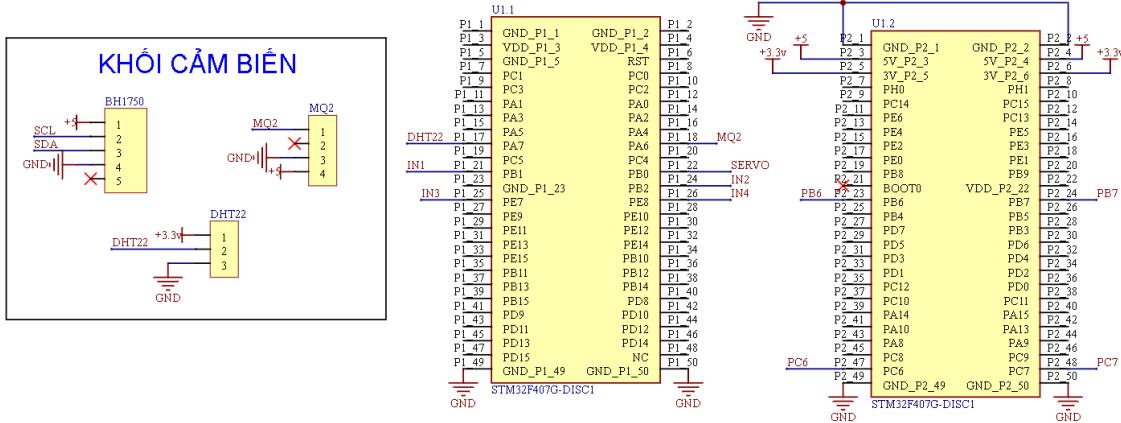
Bảng 3.1: Bảng kết nối chân giữa STM32 và ESP32

ESP32	STM32F407
3.3V	3.3V
GND	GND
TX	PC7
RX	PC6

Về tổng quan, khói vi điều khiển đóng vai trò trung tâm trong hệ thống nhà thông minh. STM32 được thiết kế để xử lý dữ liệu từ các cảm biến, điều khiển hiển thị trên LCD và thực thi các lệnh điều khiển cho servo và module relay 4 kênh, đảm bảo quản lý môi trường và thiết bị trong nhà. Trong khi đó, ESP32 chịu trách nhiệm xử lý nhận diện người dùng thông qua Keypad4x4, AS608 và RFID, đồng thời giao tiếp với Firebase Realtime Database và Web Dashboard qua Wi-Fi, hỗ trợ giám sát từ xa và cập nhật thông tin. Sự phối hợp giữa hai vi điều khiển qua

giao tiếp UART hai chiều đảm bảo hệ thống vận hành hiệu quả, hỗ trợ các chế độ giám sát (phím 'C'), xác thực (phím 'A') và cài đặt (phím 'D').

3.3.3. Khối cảm biến



Hình 3.4: Sơ đồ kết nối khói cảm biến với STM32

Hình 3.4 là sơ đồ nguyên lý của khói cảm biến trong hệ thống nhà thông minh, bao gồm các cảm biến BH1750, DHT22 và MQ2 được kết nối với vi điều khiển STM32 để thu thập dữ liệu môi trường. Hình 3.4 cung cấp cái nhìn tổng quan về cấu trúc phần cứng của khói này, bao gồm giao tiếp GPIO, I2C và ADC, cũng như cách các cảm biến được cấp nguồn. Dựa trên sơ đồ này, về kết nối chân được trình bày trong Bảng 3.2 dưới đây, làm rõ cách mỗi cảm biến được kết nối với STM32.

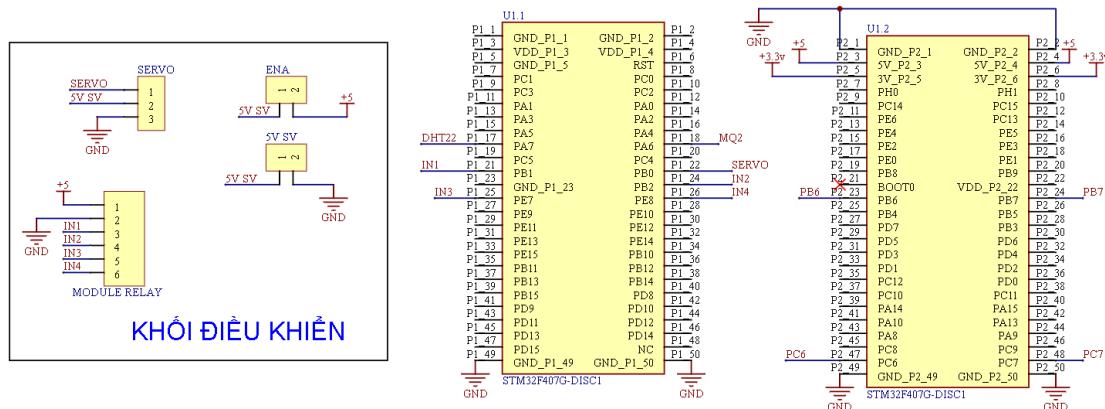
Bảng 3.2: Bảng kết nối chân khói cảm biến với STM32

Cảm biến	Chân	STM32F407
DHT22	3.3V	3.3V
	GND	GND
	DATA	PA7
MQ-2	5V	5V
	GND	GND
	AO	PA6

BH1750	5V	5V
	GND	GND
	SCL	PB6
	SDA	PB7

Về tổng quan, khói cảm biến đóng vai trò quan trọng trong việc giám sát môi trường của hệ thống nhà thông minh. Cảm biến BH1750 đo cường độ ánh sáng trong dải 0-65535 lux, DHT22 đo nhiệt độ (-40°C đến 80°C) và độ ẩm (0-100% RH), trong khi MQ2 phát hiện nồng độ khí gas (200-10000 ppm). Các cảm biến này kết nối với STM32 qua giao tiếp GPIO (DHT22), I2C (BH1750) và ADC (MQ2), thu thập dữ liệu thời gian thực và truyền qua UART đến ESP32 để gửi lên Firebase Realtime Database. Khối này hỗ trợ chức năng giám sát môi trường, phát hiện bất thường như rò rỉ khí gas và cung cấp thông tin quan trọng cho các chế độ giám sát từ xa.

3.3.4. Khối điều khiển thiết bị



Hình 3.5: Sơ đồ kết nối khối điều khiển với STM32

Hình 3.5 là sơ đồ nguyên lý của khói điều khiển thiết bị trong hệ thống nhà thông minh, bao gồm các servo và module relay 4 kênh được kết nối với vi điều khiển STM32 để thực thi các lệnh điều khiển. Hình 3.5, cung cấp cái nhìn tổng quan về cấu trúc phần cứng của khói này, bao gồm giao tiếp PWM và GPIO, cũng như cách các thiết bị được cấp nguồn. Dựa trên sơ đồ này, về kết nối chân được

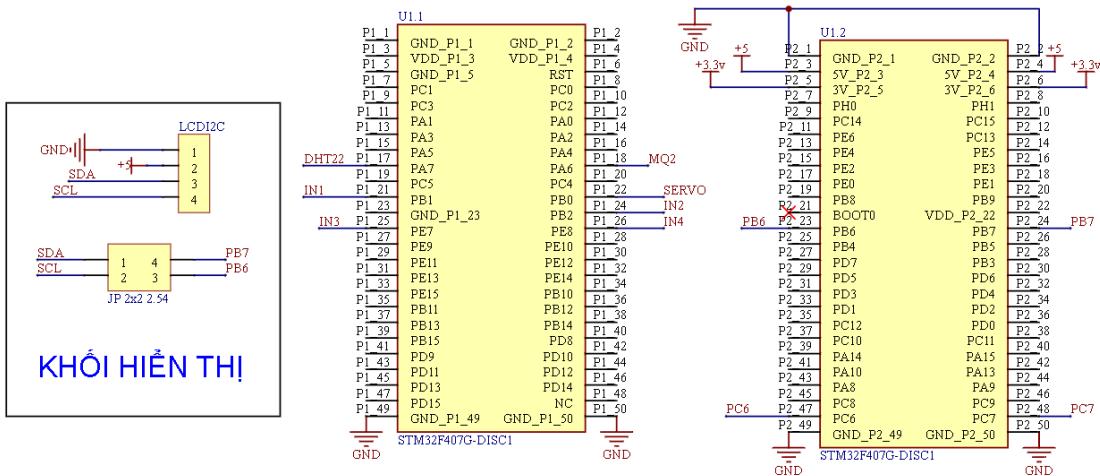
trình bày trong Bảng 3.3 dưới đây, làm rõ cách mỗi thiết bị điều khiển kết nối với STM32.

Bảng 3.3: Bảng kết nối chân khói điều khiển thiết bị với STM32

Cảm biến	Chân	STM32F407
Servo	5V	5V
	GND	GND
	PWM	PB0
Module Relay 4 kênh	5V	5V
	GND	GND
	IN1	PB1
	IN2	PB2
	IN3	PE7
	IN4	PE8

Về tổng quan, khói điều khiển thiết bị đóng vai trò quan trọng trong việc thực thi các lệnh điều khiển của hệ thống nhà thông minh. Servo được sử dụng để mở cửa tự động khi xác thực thành công, với góc quay từ 0° đến 180° , hỗ trợ kiểm soát truy cập an toàn. Module relay 4 kênh điều khiển bật/tắt các thiết bị gia dụng như đèn, với điện áp tải tối đa 250V/10A, mang lại sự tiện lợi trong quản lý ngôi nhà. Các thiết bị này được STM32 điều khiển thông qua tín hiệu PWM (servo) và GPIO (relay), nhận lệnh từ ESP32 qua UART hoặc từ Web Dashboard qua Firebase, hỗ trợ chức năng điều khiển từ xa và đảm bảo phản hồi nhanh chóng trong các chế độ vận hành.

3.3.5. Khối hiển thị



Hình 3.6: Sơ đồ kết nối khói hiển thị với STM32

Hình 3.6 là sơ đồ nguyên lý của khói hiển thị trong hệ thống nhà thông minh, có màn hình LCD20x4 I2C được kết nối với vi điều khiển STM32 để hiển thị thông tin. Hình 3.6, cung cấp cái nhìn tổng quan về cấu trúc phần cứng của khói này, bao gồm giao tiếp I2C và cách màn hình được cấp nguồn. Dựa trên sơ đồ này, về kết nối chân được trình bày trong Bảng 3.4 dưới đây, làm rõ cách LCD kết nối với STM32.

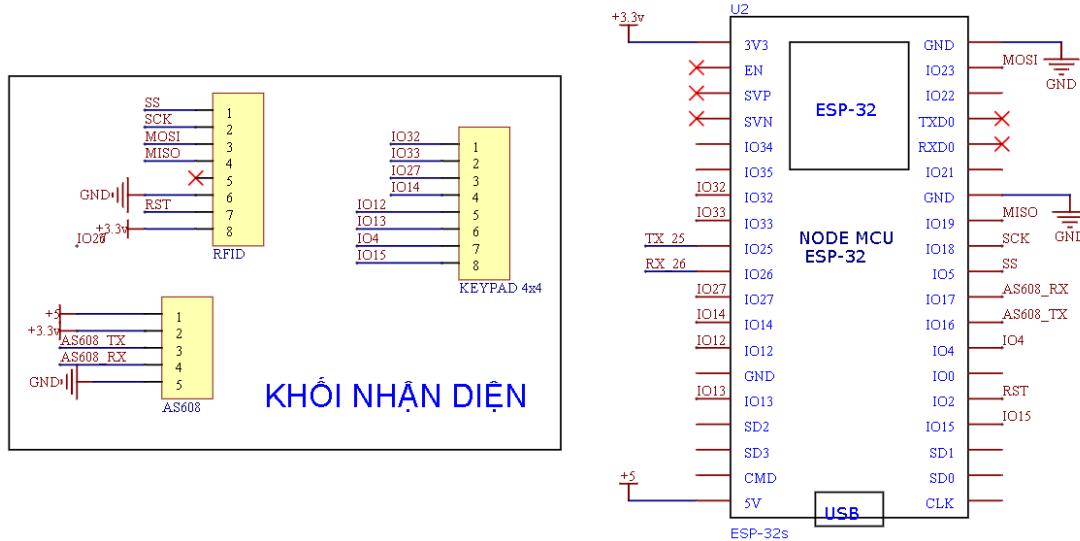
Bảng 3.4: Bảng kết nối chân khói hiển thị với STM32

LCD 20x4 I2C	STM32F407
5V	5V
GND	GND
SCL	PB6
SDA	PB7

Về tổng quan, khói hiển thị đóng vai trò quan trọng trong việc cung cấp giao diện trực quan tại chỗ của hệ thống nhà thông minh. Màn hình LCD, với khả năng hiển thị 20 ký tự trên 4 dòng, được sử dụng để hiển thị dữ liệu cảm biến như ánh sáng, nhiệt độ, độ ẩm và khí gas; trạng thái xác thực (thành công hay thất bại), thông báo hệ thống và menu cài đặt. LCD kết nối với STM32 qua giao tiếp I2C, sử dụng địa chỉ 0x27, cho phép STM32 cập nhật thông tin thời gian thực với độ

trễ dưới 100ms. Khối này hỗ trợ người dùng theo dõi tình trạng hệ thống mà không cần truy cập Web Dashboard, đảm bảo tính tiện lợi trong các chế độ vận hành như giám sát, xác thực, và cài đặt.

3.3.6. Khối nhận diện



Hình 3.7: Sơ đồ kết nối khói nhận diện với ESP32

Hình 3.7 là sơ đồ nguyên lý của khói nhận diện trong hệ thống nhà thông minh, bao gồm các thiết bị Keypad4x4, cảm biến vân tay AS608 và module RFID được kết nối với vi điều khiển ESP32 để thực hiện chức năng xác thực người dùng. Hình 3.7, cung cấp cái nhìn tổng quan về cấu trúc phần cứng của khói này, bao gồm giao tiếp GPIO, UART, và SPI, cũng như cách các thiết bị được cấp nguồn. Dựa trên sơ đồ này, về kết nối chân được trình bày trong Bảng 3.5 dưới đây, làm rõ cách mỗi thiết bị nhận diện kết nối với ESP32.

Bảng 3.5: Bảng kết nối chân khói nhận diện với ESP32

Cảm biến	Chân	ESP32
AS608	3.3V	3.3V
	GND	GND
	TX	IO16
	RX	IO17

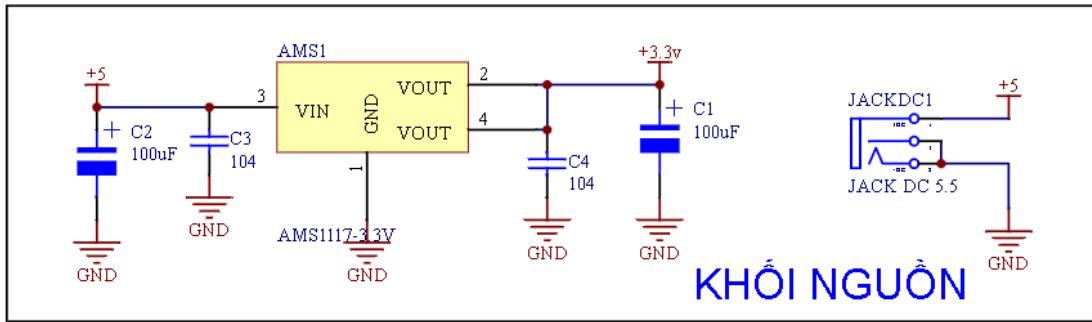
Keypad 4x4	R1	IO32
	R2	IO33
	R3	IO27
	R4	IO14
	C1	IO12
	C2	IO13
	C3	IO4
	C4	IO15
RFID	3.3V	3.3V
	GND	GND
	SS	IO5
	SCK	IO18
	MOSI	IO23
	MISO	IO19
	RST	IO2

Về tổng quan, khôi nhận diện đóng vai trò quan trọng trong việc kiểm soát truy cập của hệ thống nhà thông minh. Keypad4x4 cho phép người dùng nhập mật khẩu (4-6 ký tự) hoặc chọn chế độ (phím 'A', 'C', 'D'), cảm biến vân tay AS608 hỗ trợ xác thực sinh trắc học với khả năng lưu trữ đến 300 dấu vân tay và module RFID đọc thẻ UID với tần số 13.56 MHz. Các thiết bị này kết nối với ESP32 qua giao tiếp GPIO (Keypad), UART (AS608), và SPI (RFID), cho phép ESP32 xử lý dữ liệu xác thực và so sánh với thông tin trên Firebase Realtime Database. Khi xác thực thành công, ESP32 gửi lệnh qua UART đến STM32 để điều khiển servo mở cửa, hỗ trợ chức năng xác thực (phím 'A') và chế độ cài đặt (phím 'D'), đảm bảo an ninh và tiện lợi cho người dùng.

3.3.7. Khối nguồn

Khối nguồn điện đóng vai trò rất quan trọng trong việc cung cấp điện áp ổn định cho toàn bộ hệ thống, bao gồm các vi điều khiển STM32, ESP32 và các

module ngoại vi khác như: DHT22, MQ-2, BH1750, relay, servo, LCD, RFID, keypad 4x4 và AS608. Một module điều chỉnh điện áp AMS1117 được tích hợp để giảm điện áp 5V xuống 3.3V, với các tụ lọc 100 μ F tại đầu vào và đầu ra để giảm nhiễu và đảm bảo hoạt động liên tục. Và cũng cần phải tính toán đến đèn dòng tiêu thụ và công suất để tránh sụt áp hoặc quá tải. Dưới đây là (Hình 3.8) của khối nguồn cho toàn bộ hệ thống:



Hình 3.8: Sơ đồ nguyên lý khói nguồn

Bảng 3.6: Bảng thông số điện áp, dòng và công suất tiêu thụ của hệ thống

Linh kiện	Điện áp hoạt động (V)	Dòng tiêu thụ (mA)	Công suất tiêu thụ (mW)
STM32F407	5V	50mA	250mW
ESP32	3.3V	80mA	264mW
DHT22	3.3V	1.5mA	4.95mW
MQ-2	5V	150mA	750mW
BH1750	5V	0.5mA	2.5mW
Relay 4 kênh	5V	20mA	100mW
Servo	5V	100mA	500mW
LCD 20x4	5V	10mA	50mW
RFID	3.3V	13mA	42.9mW
Keypad	3.3V	1mA	3.3mW

AS608	3.3V	50mA	165mW
-------	------	------	-------

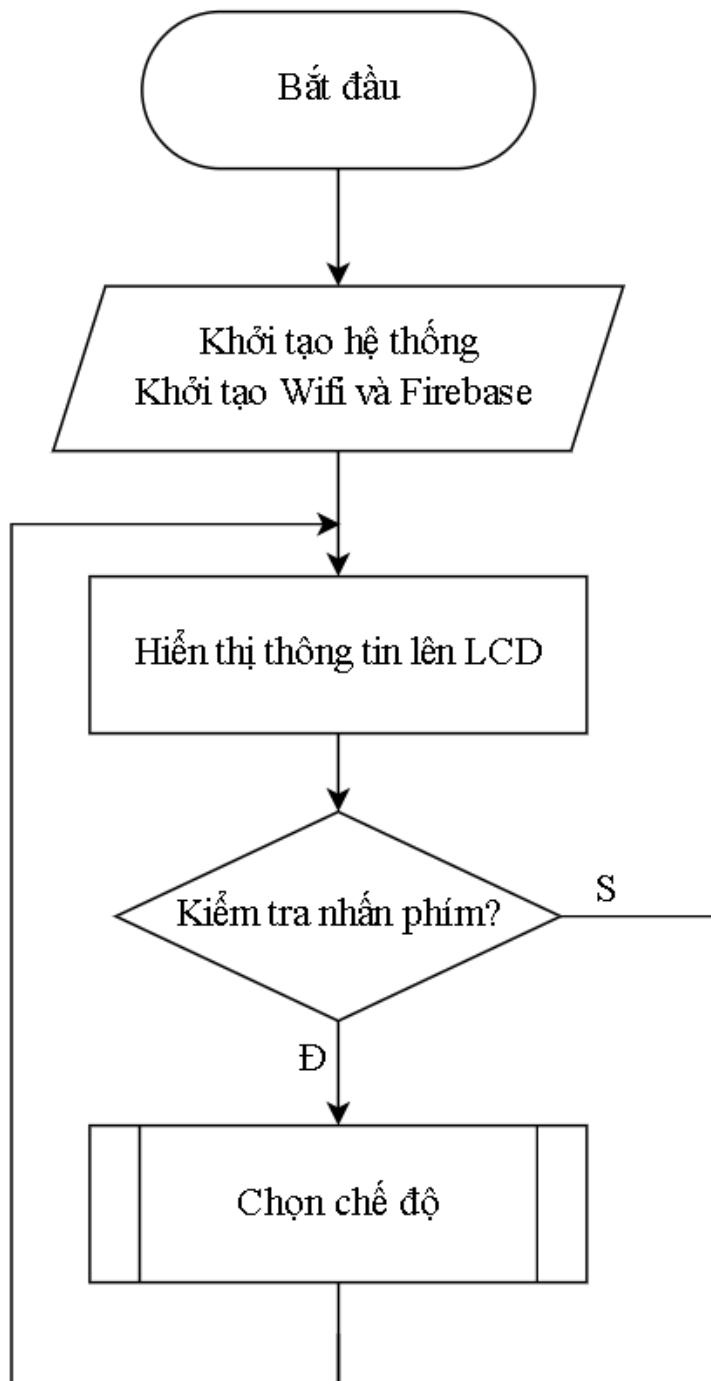
Bảng 3.6 trình bày các thông số về dòng điện, điện áp và công suất tiêu thụ của từng linh kiện trong hệ thống nhà thông minh, thể hiện nhu cầu năng lượng của từng khối chức năng. Bảng 3.6 cung cấp cái nhìn tổng quan về mức tiêu thụ điện của từng linh kiện, giúp xác định nguồn cấp phù hợp để đảm bảo hệ thống hoạt động ổn định.

Nguồn DC 5V – 3A được chọn để cung cấp nguồn cho toàn hệ thống nhằm đảm bảo dòng điện tối đa 3000mA, đáp ứng nhu cầu tiêu thụ của tất cả các khối chức năng. Tổng dòng tiêu thụ của hệ thống, bao gồm STM32 (50mA), ESP32 (80mA), cảm biến (152mA), nhận diện (64mA), relay (80mA cho 4 kênh), servo (100mA) và LCD (10mA), là khoảng 536mA, tương ứng với công suất 2677.5mW (2.68W), thấp hơn nhiều so với khả năng cung cấp của nguồn. Nguồn 5V – 3A không chỉ đáp ứng được yêu cầu về dòng điện, công suất mà còn có chi phí thấp, dễ thay thế và tương thích với các linh kiện phổ biến, giúp hệ thống hoạt động ổn định và dễ bảo trì.

Về tổng quan, khối nguồn đóng vai trò quan trọng trong việc đảm bảo cung cấp năng lượng ổn định cho toàn bộ hệ thống nhà thông minh. Nguồn DC 5V – 3A cung cấp trực tiếp cho STM32, module relay, servo, và LCD, trong khi IC AMS1117-3.3 hạ áp xuống 3.3V để cấp cho ESP32, các cảm biến như BH1750, DHT22 và các thiết bị nhận diện như Keypad4x4, AS608 và RFID, tránh hỏng hóc do chênh lệch điện áp.

3.4. THIẾT KẾ PHẦN MỀM

3.4.1. Lưu đồ giải thuật cho toàn hệ thống



Hình 3.9: Lưu đồ giải thuật toàn hệ thống

Hình 3.9 lưu đồ giải thuật tổng quát cho toàn hệ thống nhà thông minh, thể hiện các bước khởi tạo ban đầu và cách hệ thống xử lý lệnh từ người dùng thông qua Keypad. Lưu đồ này mô tả quy trình khởi động ban đầu và xử lý lệnh người

dùng của hệ thống nhà thông minh, đảm bảo hệ thống sẵn sàng vận hành và phản hồi các yêu cầu từ người dùng một cách hiệu quả.

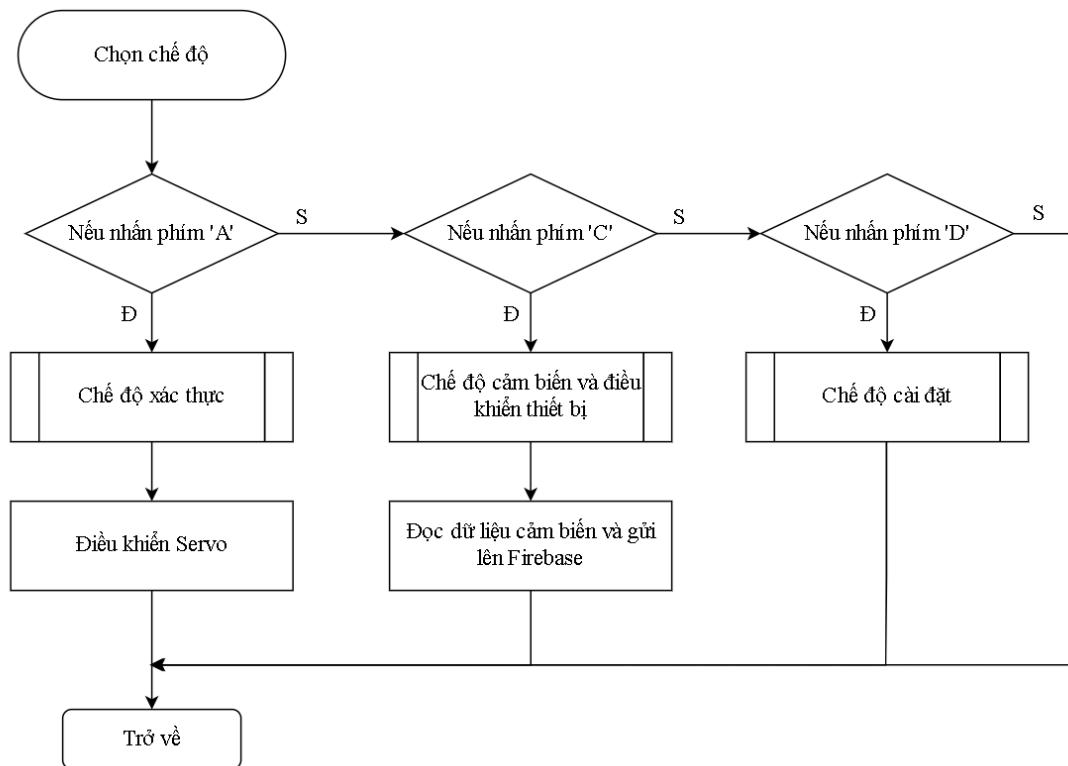
Quá trình bắt đầu với Bắt đầu, đánh dấu điểm khởi động của hệ thống. Tiếp theo, bước Khởi tạo hệ thống / Khởi tạo Wi-Fi và Firebase được thực hiện để khởi động các vi điều khiển STM32 và ESP32, cấu hình các giao tiếp cần thiết (GPIO, I2C, UART, SPI) và thiết lập kết nối Wi-Fi giữa ESP32 và Firebase Realtime Database. Bước này đảm bảo hệ thống có thể giao tiếp với cloud server và sẵn sàng cho các chức năng như giám sát từ xa, xác thực người dùng và điều khiển thiết bị.

Sau đó, Hiển thị thông báo trên LCD được thực hiện bởi STM32, hiển thị trên màn hình LCD, nhằm thông báo cho người dùng biết rằng hệ thống đã khởi động thành công và sẵn sàng nhận lệnh. Bước này giúp người dùng tại chỗ nắm được trạng thái ban đầu của hệ thống mà không cần truy cập Web Dashboard.

Tiếp theo, hệ thống thực hiện kiểm tra nhấn phím, trong đó ESP32 đọc tín hiệu từ Keypad4x4 để xác định xem có phím nào được nhấn không. Bước này cho phép hệ thống phản hồi các lệnh từ người dùng, như chọn chế độ vận hành. Nếu có thì hệ thống chuyển sang Chọn chế độ, nơi người dùng có thể chọn chế độ vận hành dựa trên phím được nhấn (phím 'C' cho giám sát, 'A' cho xác thực, hoặc 'D' cho cài đặt). Nếu không thì lưu đồ không chỉ định bước tiếp theo, nhưng hệ thống sẽ tiếp tục kiểm tra phím, duy trì trạng thái chờ lệnh từ người dùng.

3.4.2. Lưu đồ giải thuật cho các chế độ

- **Lưu đồ giải thuật chọn chế độ:**



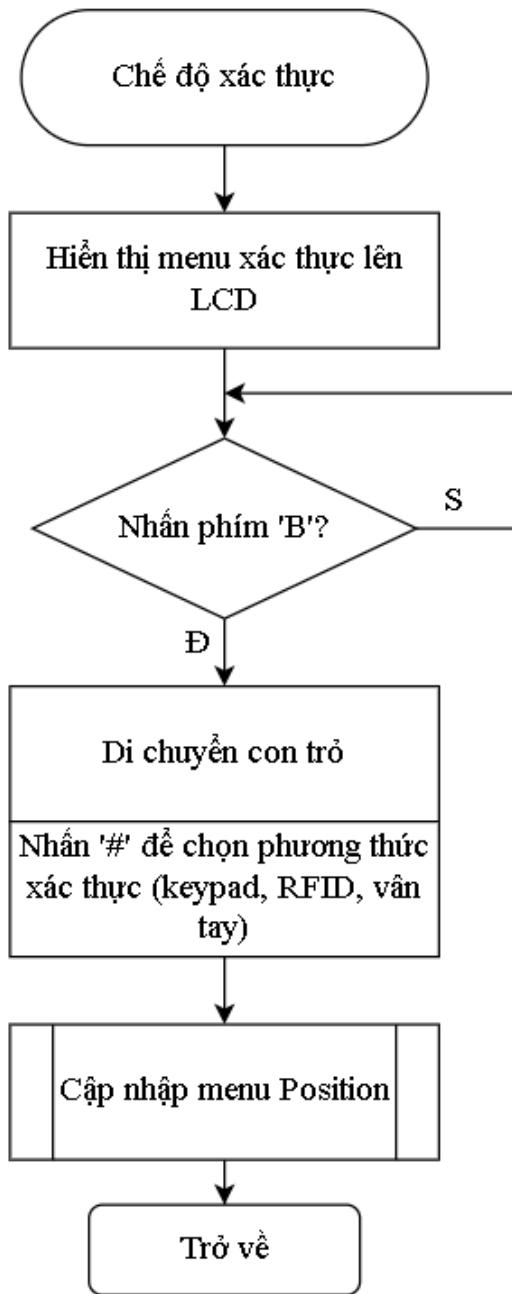
Hình 3.10: Lưu đồ giải thuật cho chọn chế độ

Hình 3.10 lưu đồ giải thuật cho hệ thống bắt đầu tại bước “Chọn chế độ” sau khi nhận được tín hiệu từ Keypad, nơi người dùng nhấn phím để chọn các chế độ hoạt động. Đầu tiên, hệ thống sẽ kiểm tra xem phím nào được nhấn:

- Nếu là phím ‘A’ thì nó sẽ chuyển qua chế độ xác thực, nơi cửa được mở bằng cách điều khiển servo.
- Nếu là phím ‘C’ thì nó sẽ chuyển qua chế độ cảm biến và điều khiển. Sau đó sẽ đọc dữ liệu từ các cảm biến và gửi thông tin này lên Firebase để lưu trữ hoặc hiển thị.
- Nếu là phím ‘D’ thì nó sẽ chuyển qua chế độ cài đặt (setup), cho phép người dùng cài đặt lại các thông tin như là mật khẩu, vân tay hoặc thẻ UID.

Sau khi hoàn thành mỗi chế độ trên thì hệ thống sẽ quay trở về trạng thái ban đầu và sẵn sàng nhận thêm lệnh mới.

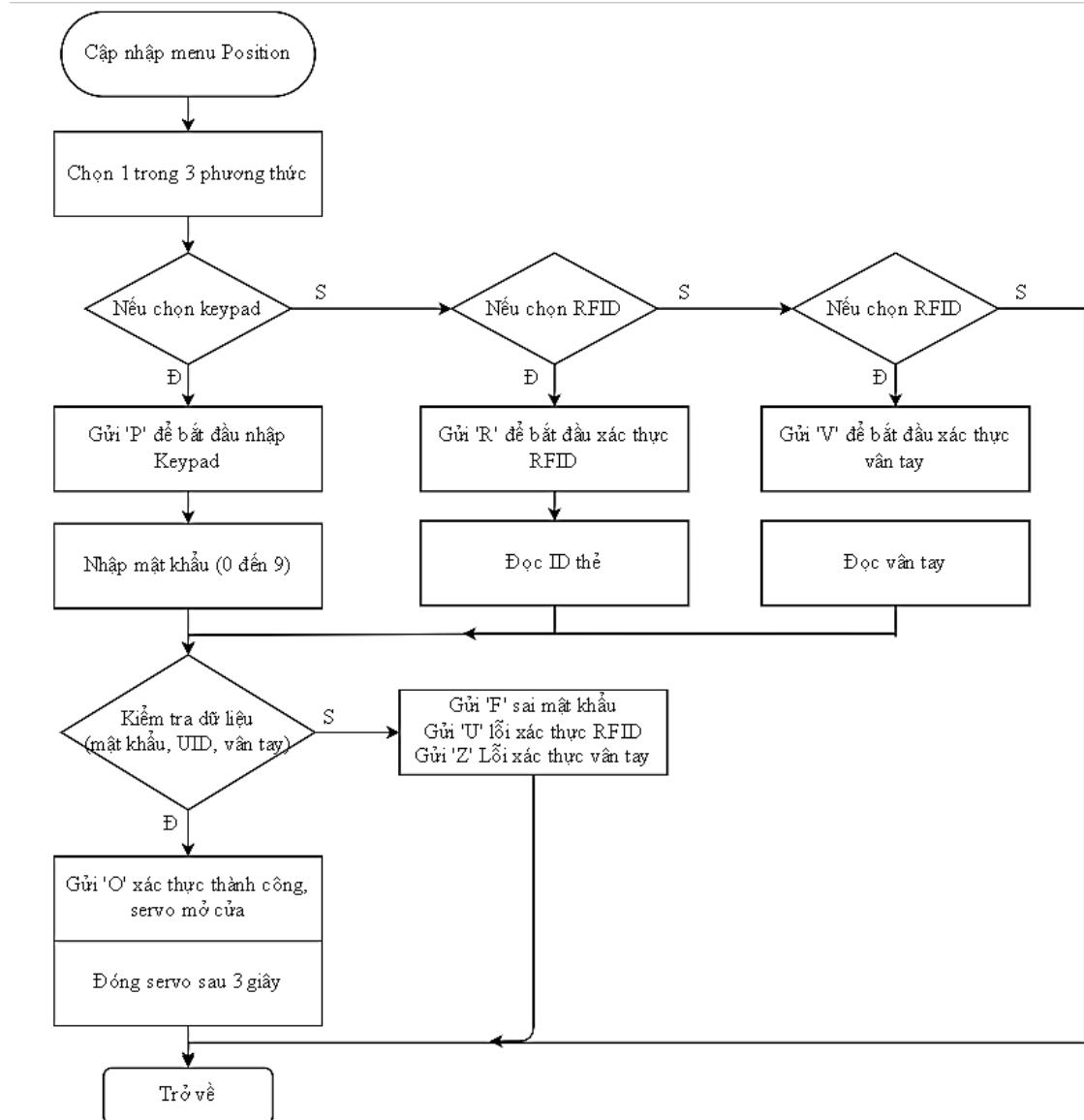
- **Lưu đồ giải thuật chế độ xác thực**



Hình 3.11: Lưu đồ giải thuật chế độ xác thực

Hình 3.11 lưu đồ giải thuật khi chế độ xác thực được dùng hệ thống bắt đầu bằng việc hiển thị menu trên LCD, liệt kê các phương thức xác thực như nhập mật khẩu, quét thẻ RFID, hoặc quét vân tay. Tiếp theo hệ thống sẽ chờ cho người dùng nhấn phím chọn phương thức. Vị trí con trỏ được cập nhật để xác định phương thức được chọn. Sau khi người dùng xác nhận lực chọn thì hệ thống sẽ thực hiện xác thực tương ứng. Cuối cùng sau khi hoàn tất thì hệ thống sẽ trở về trạng thái ban đầu.

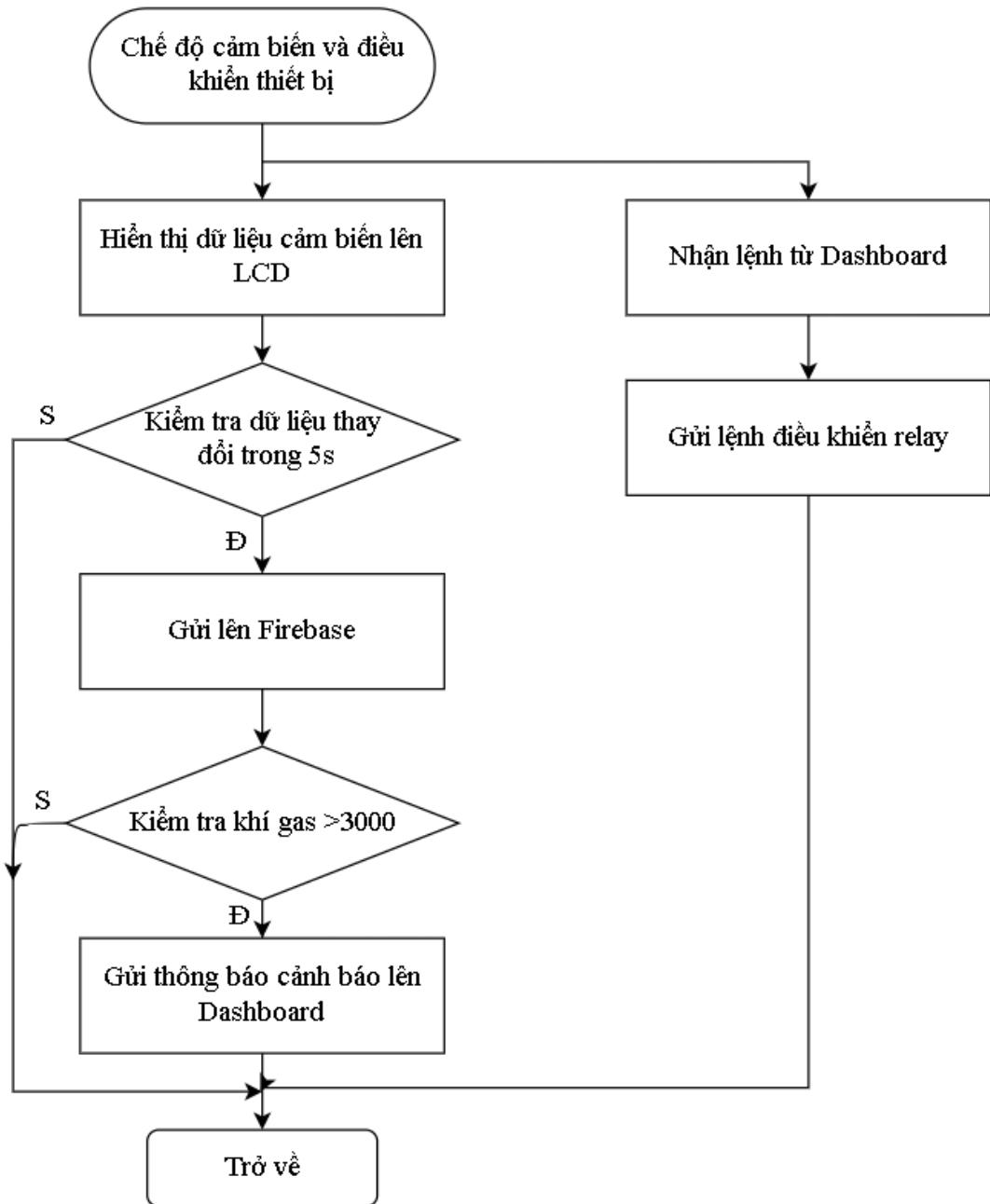
- Lưu đồ giải thuật cập nhật menuPosition



Hình 3.12: Lưu đồ giải thuật cập nhập menuPosition

Hình 3.12 lưu đồ giải thuật khi người dùng vào chế độ xác thực và bắt đầu cập nhập menuPosition, hệ thống kiểm tra phím nào được nhấn để chọn phương thức: Nếu là keypad thì hệ thống yêu cầu nhập mật khẩu; Nếu là RFID thì hệ thống yêu cầu quét thẻ; Hoặc nếu là vân tay thì hệ thống yêu cầu quét vân tay. Sau khi nhập dữ liệu thì hệ thống sẽ so sánh với thông tin đã lưu trước đó để kiểm tra tính hợp lệ. Nếu là đúng thì servo sẽ mở cửa và nếu là sai thì sẽ báo lỗi. Cuối cùng sau khi hoàn tất thì hệ thống quay về trạng thái ban đầu.

- Lưu đồ giải thuật chế độ cảm biến và điều khiển thiết bị**

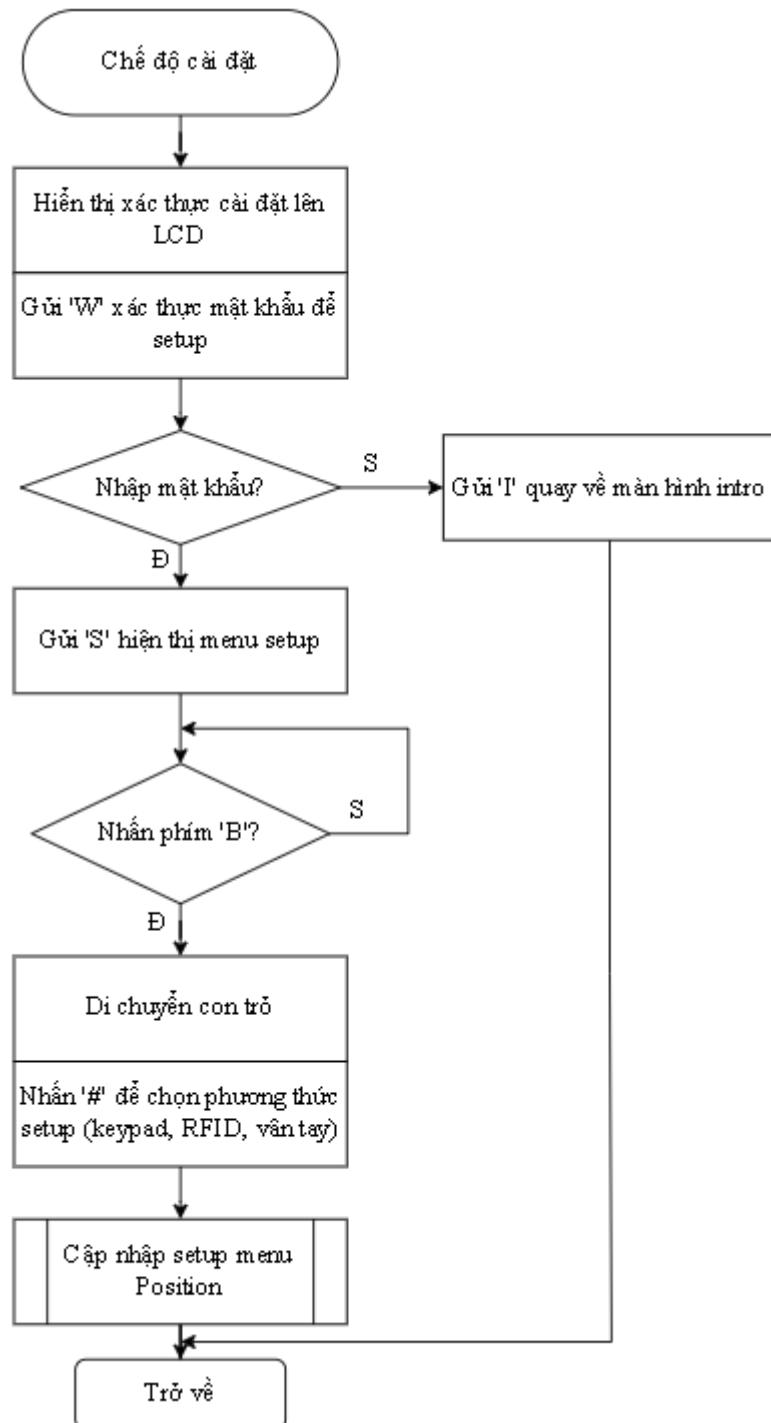


Hình 3.13: Lưu đồ giải thuật chế độ cảm biến và điều khiển thiết bị

Hình 3.13 lưu đồ giải thuật khi chế độ cảm biến và điều khiển thiết bị được dùng thì hệ thống bắt đầu bằng việc đọc dữ liệu từ các cảm biến như nhiệt độ, độ ẩm, khí gas và ánh sáng. Dữ liệu này được hiển thị trên LCD, sau đó sẽ kiểm tra nếu có sự thay đổi đáng kể và sau 5 giây sẽ được gửi lên Firebase để lưu trữ và cập nhập trên dashboard cho phép người dùng theo dõi từ xa. Đồng thời dashboard

cũng sẽ gửi lệnh xuống để điều khiển các thiết bị như relay. Sau đó, hệ thống lặp lại chu kỳ này liên tục để giám sát và điều khiển.

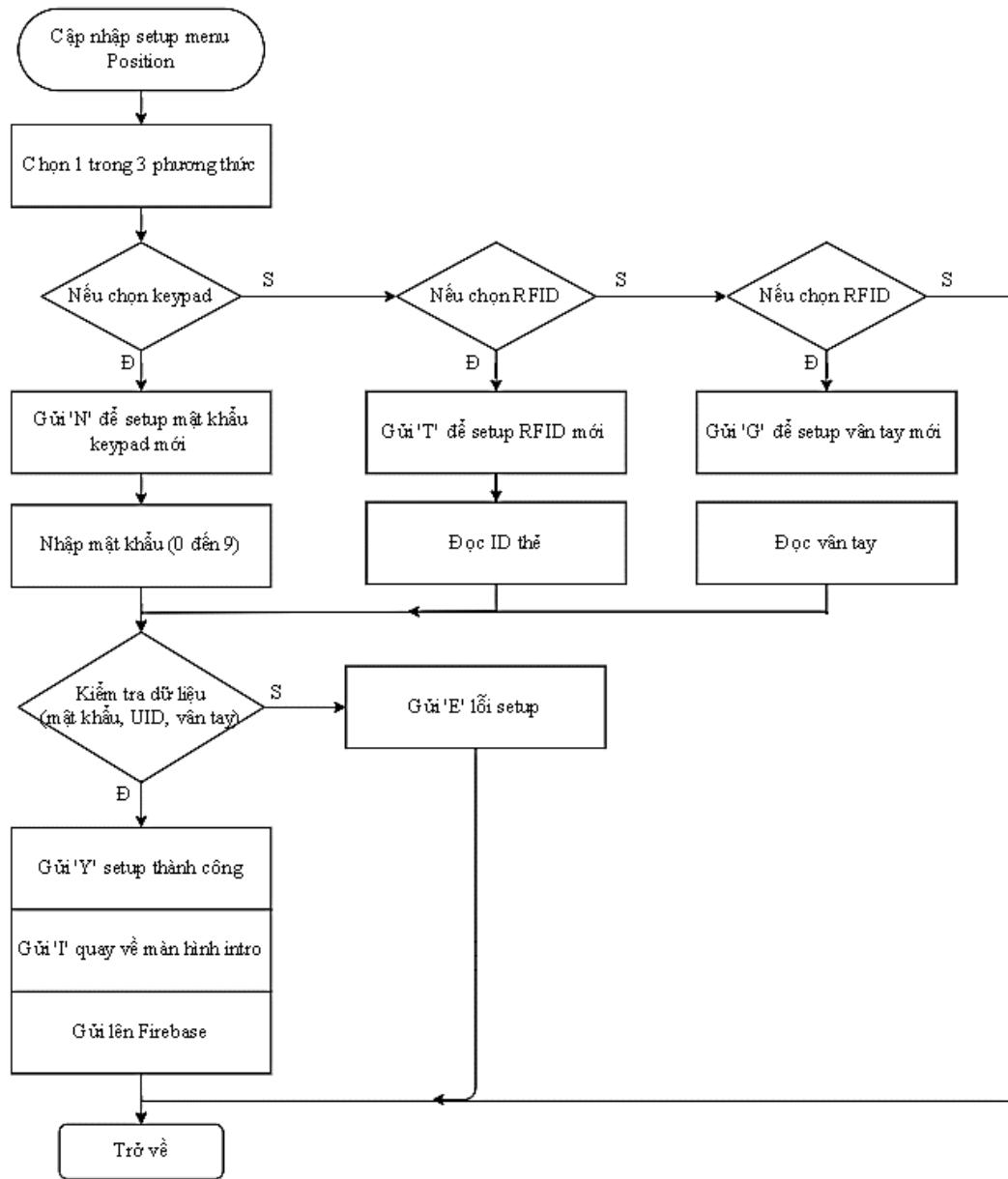
- **Lưu đồ giải thuật chế độ cài đặt (setup)**



Hình 3.14: Lưu đồ giải thuật chế độ cài đặt (setup)

Hình 3.14 lưu đồ giải thuật khi chế độ cài đặt được dùng thì hệ thống yêu cầu người dùng xác thực quyền truy cập bằng cách nhập mật khẩu, đảm bảo chỉ có người có quyền mới có thể thay đổi cài đặt. Nếu mật khẩu đúng, hệ thống hiển thị menu cài đặt trên LCD, liệt kê các tùy chọn như thay đổi mật khẩu, đăng ký vân tay mới hoặc thêm thẻ RFID. Người dùng sử dụng phím để tùy chọn chọn mong muốn. Sau khi hoàn tất thì hệ thống sẽ hiển thị kết quả lên LCD và quay về trạng thái ban đầu.

- Lưu đồ giải thuật cập nhập setup menuPosition



Hình 3.15: Lưu đồ giải thuật cập nhập setup menuPosition

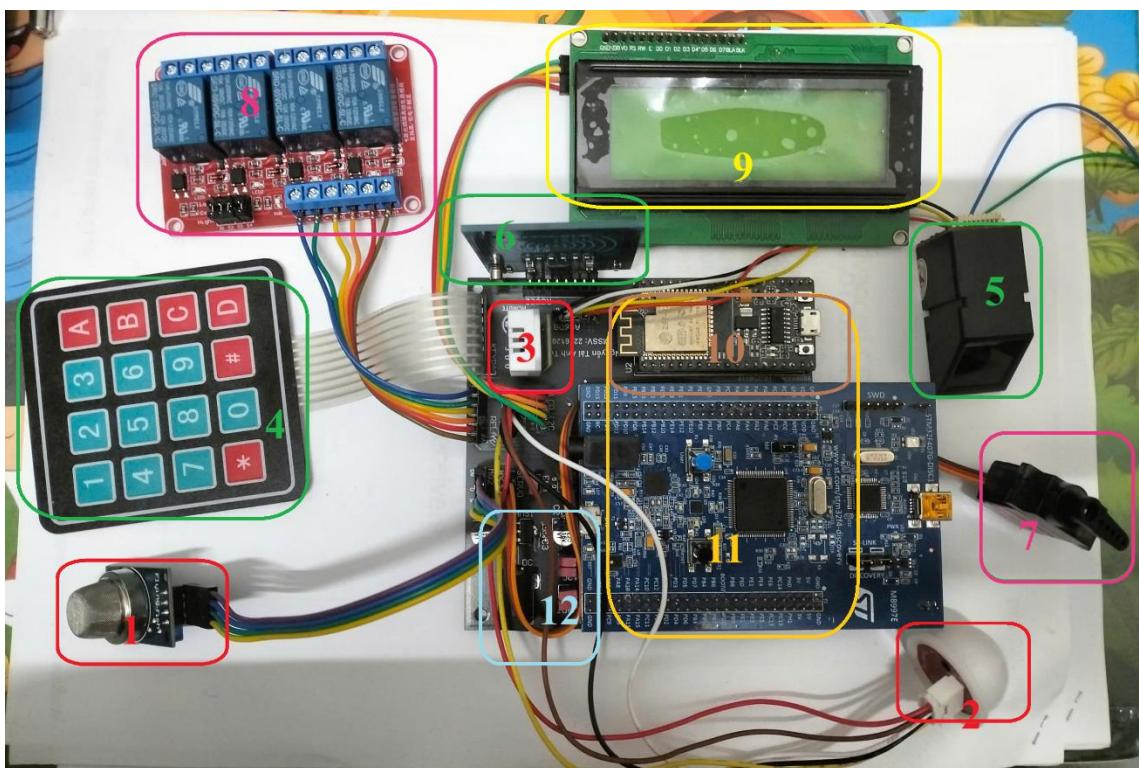
Hình 3.15 lưu đồ giải thuật khi người dùng đã vào chế độ cài đặt và bắt đầu cập nhập setup menuPosition thì hệ thống kiểm tra phím nhấn để xác định tùy chọn được chọn: Nếu chọn thay đổi mật khẩu, người dùng nhập dãy số mới; Nếu chọn đăng ký vân tay, hệ thống yêu cầu quét vân tay mới; Nếu chọn thêm thẻ RFID, hệ thống sẽ đọc mã thẻ. Sau khi thu thập dữ liệu đó xong thì sẽ gửi lên Firebase để lưu trữ. Hệ thống sau đó hiển thị kết quả trên LCD và quay trở lại trạng thái ban đầu.

CHƯƠNG 4: KẾT QUẢ VÀ ĐÁNH GIÁ

4.1. KẾT QUẢ PHẦN CỨNG

4.1.1. Mô hình phần cứng thực tế

Mô hình phần cứng thực tế của hệ thống giám sát nhà thông minh được xây dựng dựa trên bo mạch chính là STM32 và ESP32, kết hợp với các cảm biến và thiết bị ngoại vi. Cụ thể, STM32 được dùng để điều khiển các thiết bị tại chỗ bao gồm cảm biến nhiệt độ và độ ẩm (DHT22); Cảm biến khí gas (MQ-2); Cảm biến ánh sáng (BH1750); Servo (đóng/mở cửa) và LCD 20x4 (hiển thị thông tin). Còn ESP32 đảm nhiệm vai trò kết nối mạng, xử lý xác thực và giao tiếp với Firebase, tích hợp các module như keypad 4x4, cảm biến vân tay AS608 và module RFID. Các thành phần linh kiện được lắp ráp trên một bo mạch PCB nhỏ gọn với các chân cắm module để thuận tiện cho việc thay đổi linh kiện bị hư hỏng.



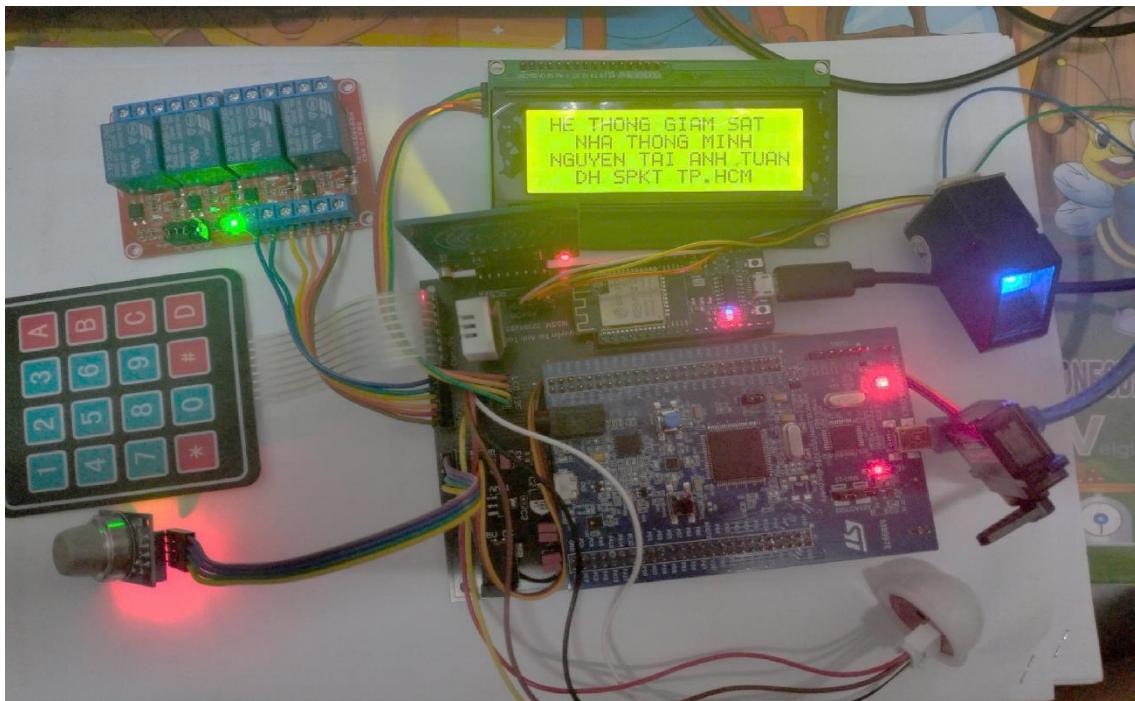
Hình 4.1: Mạch PCB thực tế của hệ thống

Trong hình ảnh bô trí về phần cứng (Hình 4.1), có thể thấy rõ các thành phần chính có trong hệ thống bao gồm:

1. Cảm biến khí gas MQ-2.

2. Cảm biến cường độ ánh sáng BH1750.
3. Cảm biến nhiệt độ, độ ẩm DHT22.
4. Bàn phím ma trận 4x4.
5. Cảm biến vân tay AS608.
6. Đầu đọc RFID RC522.
7. Servo cơ học MG90S.
8. Module relay 4 kênh.
9. Màn hình LCD20x4 I2C.
10. Mạch ESP32-S.
11. Mạch STM32F407.
12. Khối nguồn.

Khi cấp nguồn cho hệ thống thì LCD sẽ hiển thị thông tin như sau:



Hình 4.2: Khi cấp nguồn LCD sẽ hiển thị thông tin

Hình 4.2, hiển thị thông tin khi hệ thống được khởi động thành công với các thông tin cơ bản trên màn hình LCD như:

- Hàng 0: “HE THONG GIAM SAT”
- Hàng 1: “NHA THONG MINH”

- Hàng 2: “NGUYEN TAI ANH TUAN”
- Hàng 3: “DH SPKT TP.HCM”

4.1.2. Kiểm tra chế độ xác thực

Chế độ xác thực được kiểm tra để đánh giá khả năng mở cửa bằng ba phương thức: Keypad, RFID và vân tay.

Đầu tiên, khi nhấn phím ‘A’ trên keypad để vào chế độ xác thực, LCD sẽ hiển thị menu như ảnh sau:

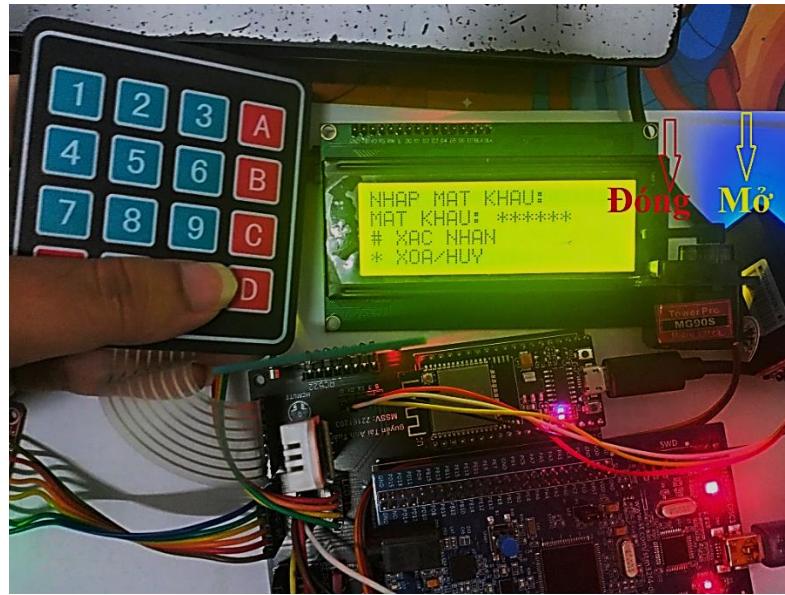


Hình 4.3: Nhấn ‘A’ vào chế độ xác thực

Hình 4.3 khi vào chế độ xác thực, sẽ có ba lựa chọn phương thức xác nhận đó là keypad, thẻ RFID và vân tay. Để chọn được các phương thức trên thì chỉ cần nhấn phím ‘B’ để di chuyển con trỏ tới vị trí tương ứng. Sau đó nhấn '#' để xác nhận chọn phương thức đó.

Phương thức mật khẩu

Di chuyển con trỏ tới vị trí keypad và nhấn '#' để chọn keypad là phương thức xác thực. Ban đầu trạng thái cửa sẽ là đóng và phải nhập mật khẩu đúng để mở cửa như Hình 4.4.



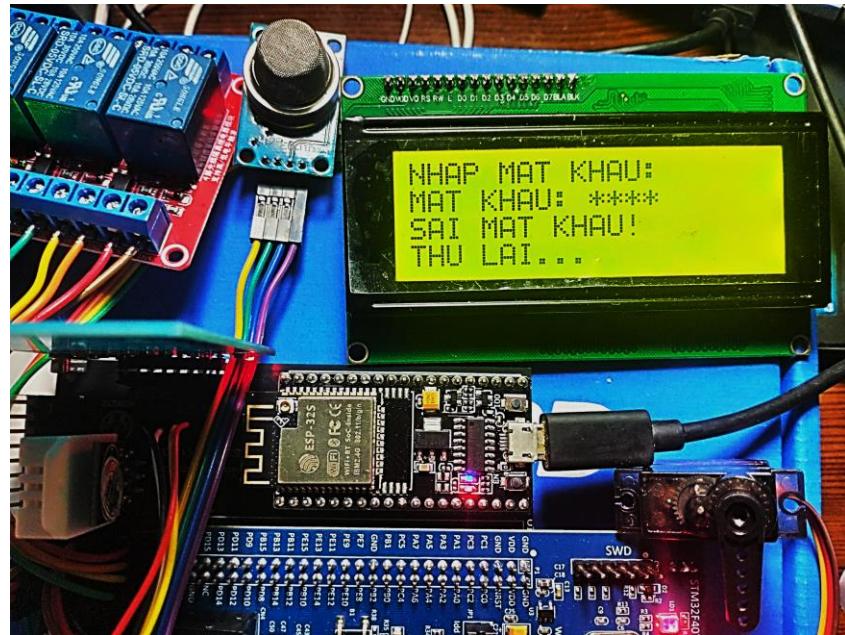
Hình 4.4: Trạng thái ban đầu là đóng cửa

Sau khi đã nhập mật khẩu đúng và xác nhận thì LCD sẽ hiển thị “XAC THUC THANH CONG”, trạng thái “DANG MO CUA” servo sẽ được kích hoạt để mở cửa như Hình 4.5.



Hình 4.5: Khi mật khẩu đúng thì cửa sẽ mở

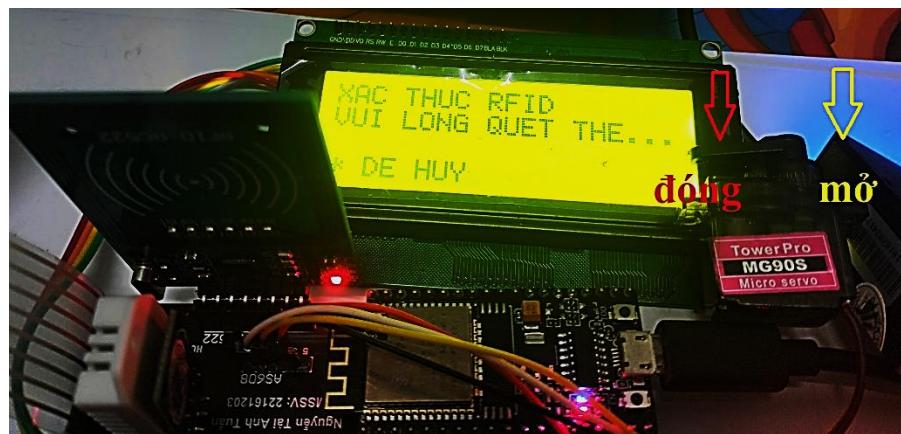
Còn với trường hợp nhập mật khẩu sai thì LCD sẽ hiển thị “SAI MAT KHAU” “THU LAI...” như Hình 4.6.



Hình 4.6: Khi nhập mật khẩu sai

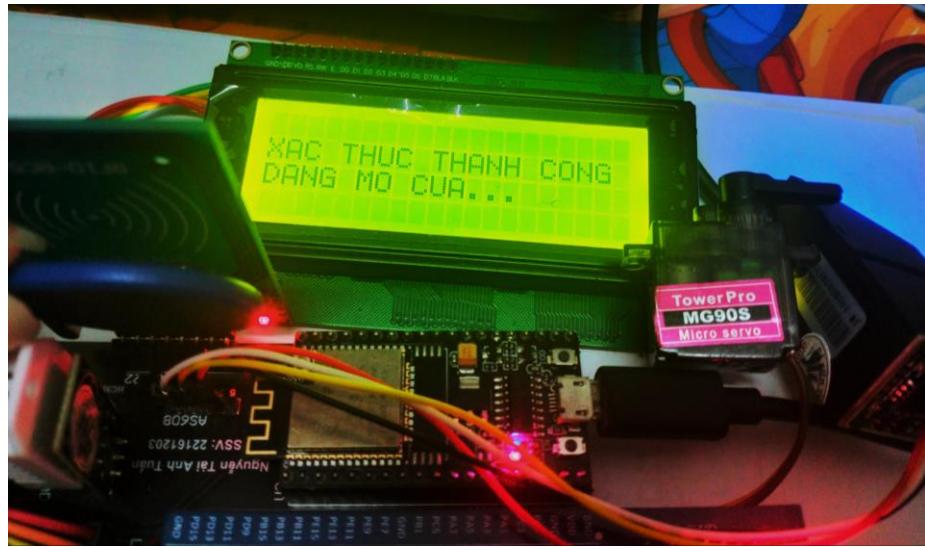
Phương thức thẻ RFID

Di chuyển con trỏ tới vị trí thẻ RFID và nhấn ‘#’ để chọn RFID là phương thức xác thực. Ban đầu trạng thái cửa sẽ là đóng và phải quét thẻ đúng để mở cửa như Hình 4.7.



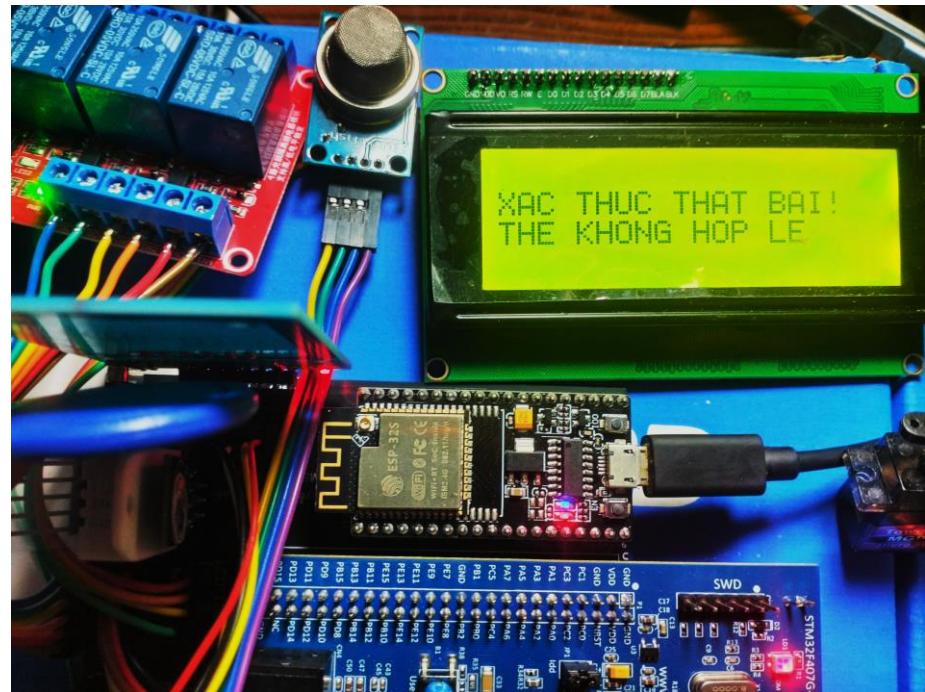
Hình 4.7: Trạng thái ban đầu là cửa đóng và yêu cầu quét thẻ

Sau khi đã quét thẻ ID đúng và xác nhận thì LCD sẽ hiển thị “XAC THUC THANH CONG”, trạng thái “DANG MO CUA” servo sẽ được kích hoạt để mở cửa như Hình 4.8.



Hình 4.8: Khi quét đúng thẻ đã lưu cửa sẽ mở

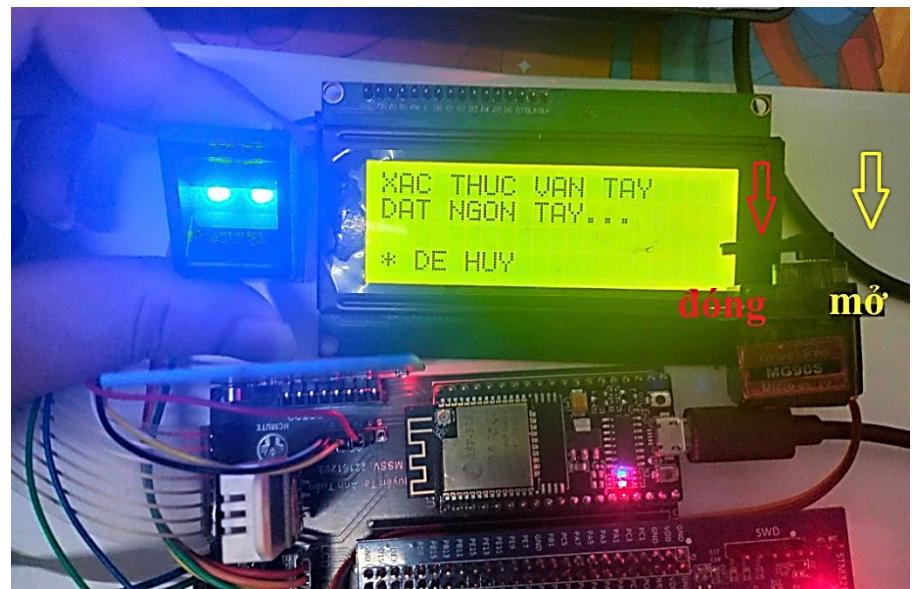
Còn với trường hợp quét thẻ ID sai thì LCD sẽ hiển thị “XAC THUC THAT BAI” “THE KHONG HOP LE” như Hình 4.9.



Hình 4.9: Khi quét thẻ sai

Phương thức vân tay

Di chuyển con trỏ tới vị trí vân tay và nhấn '#' để chọn vân tay là phương thức xác thực. Ban đầu trạng thái cửa sẽ là đóng và phải quét vân tay đúng để mở cửa như Hình 4.10.



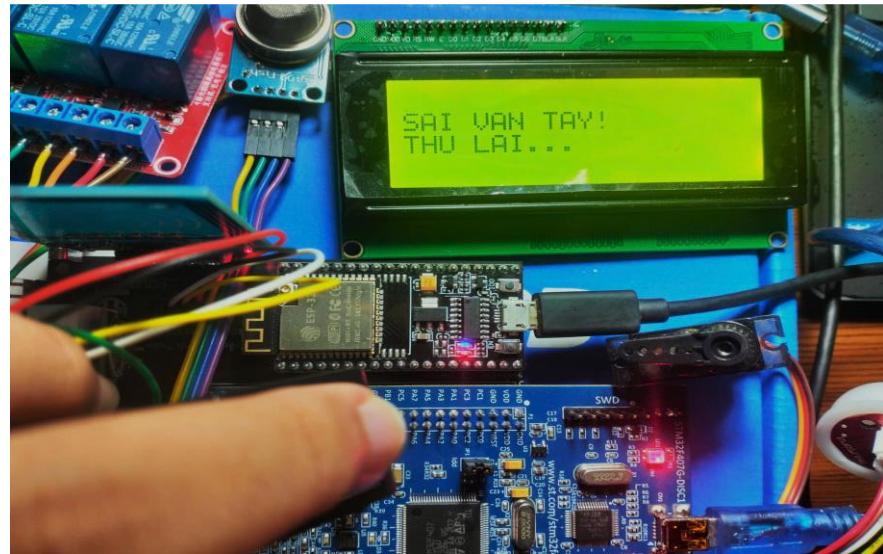
Hình 4.10: Ban đầu đóng và yêu cầu đặt vân tay

Sau khi đã quét vân tay đúng và xác nhận thì LCD sẽ hiển thị “XAC THUC THANH CONG”, trạng thái “DANG MO CUA...” servo sẽ được kích hoạt để mở cửa như Hình 4.11.



Hình 4.11: Vân tay đúng cửa sẽ mở

Còn với trường hợp quét vân tay sai thì LCD sẽ hiển thị “SAI MAT KHAU” “THU LAI...” như Hình 4.12.



Hình 4.12: Khi vân tay sai

4.1.3. Kiểm tra chế độ cảm biến và điều khiển thiết bị

Chế độ cảm biến và điều khiển thiết bị được kiểm tra và đánh giá khả năng giám sát môi trường và điều khiển từ xa.

Để vào chế độ giám sát và điều khiển thì nhấn phím ‘C’ trên bàn phím thì LCD sẽ chuyển sang hiển thị thông tin của các cảm biến.



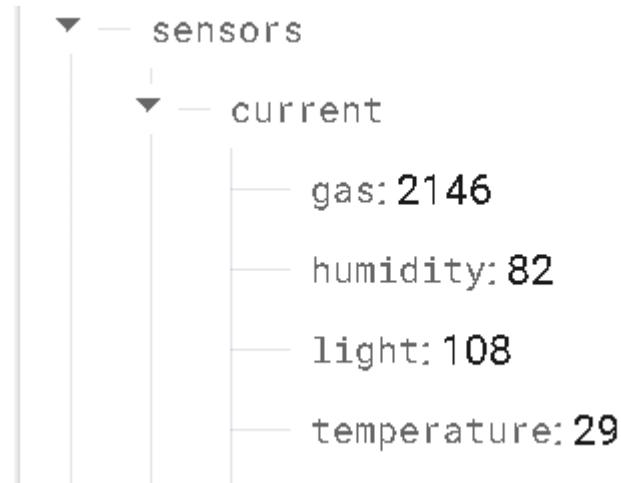
Hình 4.13: LCD hiển thị dữ liệu cảm biến

Hình 4.13, hiển thị thông tin cảm biến trên LCD với các giá trị sau:

- Hàng 0: “THONG TIN CAM BIEN”
- Hàng 1: “NHIET DO: 30.2°C
- Hàng 2: “DO AM: 81.9%

- Hàng 3: “GAS: 2368 LUX: 14”

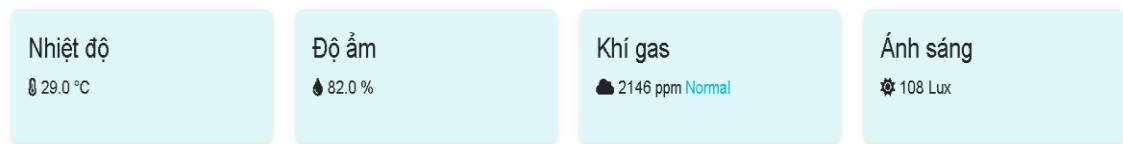
Trên Firebase cũng hiển thị dữ liệu từ các cảm biến như Hình 4.14:



Hình 4.14: Hiển thị dữ liệu cảm biến lên Firebase

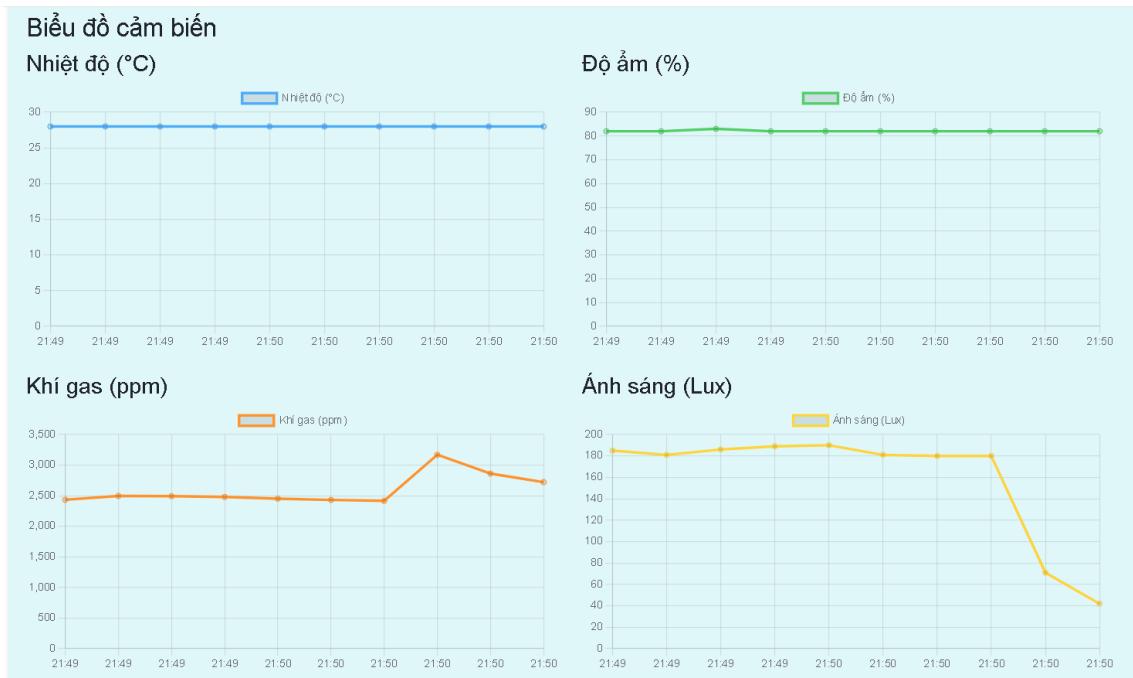
Trên giao diện dashboard sẽ nhận dữ liệu từ Firebase và hiển thị dữ liệu cảm biến như Hình 4.15:

Dashboard



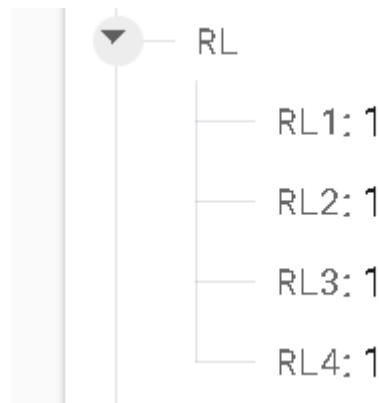
Hình 4.15: dashboard hiển thị dữ liệu cảm biến

Trên giao diện dashboard sẽ có biểu đồ thay đổi trạng thái của bốn dữ liệu, các dữ liệu này sẽ thay đổi liên tục theo thời gian thực như Hình 4.16.



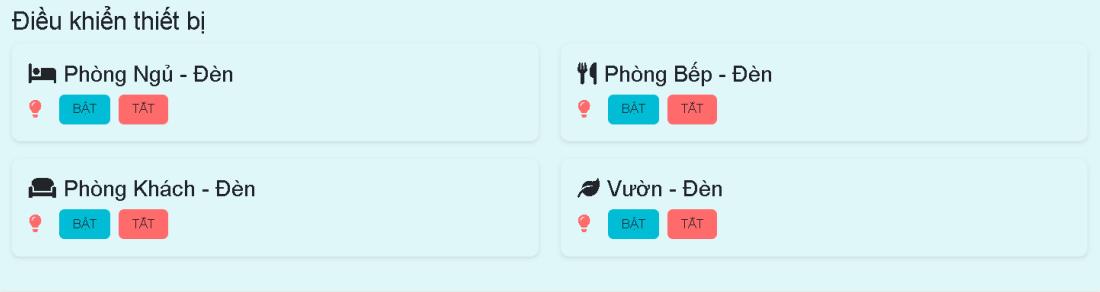
Hình 4.16: Biểu đồ cảm biến thay đổi theo dữ liệu

Trên Firebase sẽ có phần lưu trữ trạng thái điều khiển của bốn relay và tại đây cũng có thẻ điều khiển từ xa thông qua Firebase mà không cần tới giao diện web như Hình 4.17.



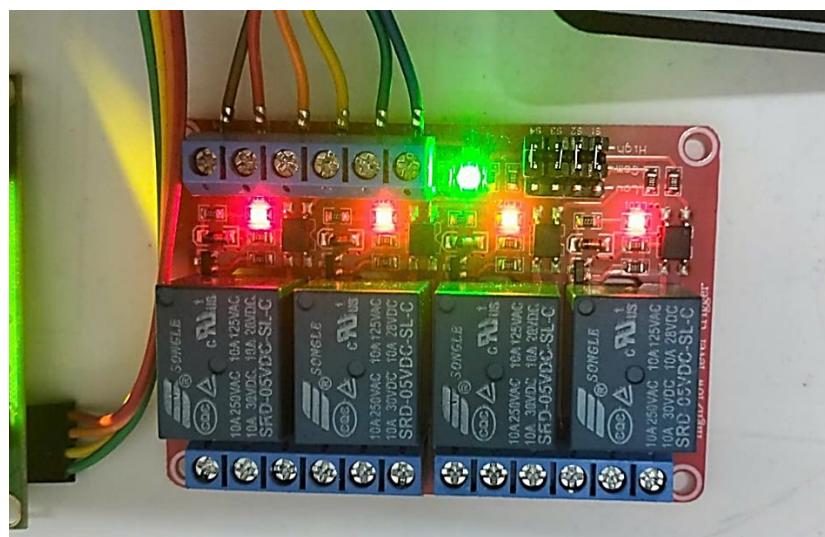
Hình 4.17: Hiện tại bốn thiết bị đang bật ‘1’.

Giao diện điều khiển thiết bị trên dashboard, gồm có bốn khu vực là phòng ngủ, phòng bếp, phòng khách và vườn như Hình 4.18.



Hình 4.18: Hiện tại bốn thiết bị đang bật.

Tương ứng với đó thì Hình 4.19 thể hiện hình ảnh của module relay có 4 relay đang ở mức HIGH tức là trạng thái của bốn relay là bật đèn.



Hình 4.19: Bốn LED trên Relay đang bật

4.1.4. Kiểm tra chế độ cài đặt

Chế độ cài đặt được kiểm tra và đánh giá khả năng thay đổi thông tin an ninh của hệ thống.

Để bắt đầu vào chế độ cài đặt thì nhấn phím 'D' trên bàn phím nhưng trước khi được cài đặt thì hệ thống sẽ bắt người dùng xác nhận danh tính bằng cách nhập mật khẩu như Hình 4.20.



Hình 4.20: Xác thực mật khẩu để vào chế độ cài đặt

Nếu đã xác minh danh tính bằng mật khẩu đúng thì sẽ vào được chế độ cài đặt Hình 4.21. Tại đây sẽ có ba phương thức cài đặt mới là: mật khẩu mới, vân tay mới và thẻ RFID mới. Chọn vào một trong ba phương thức mà muốn thay đổi hoặc thêm mới.



Hình 4.21: Chế độ cài đặt

Phương thức thêm mật khẩu mới

Hình 4.21, khi vào chế độ cài đặt và chọn thay đổi mật khẩu mới thì chọn bằng cách nhấn phím '#'. Sau đó hệ thống sẽ yêu cầu người dùng nhập mật khẩu mới như Hình 4.22.



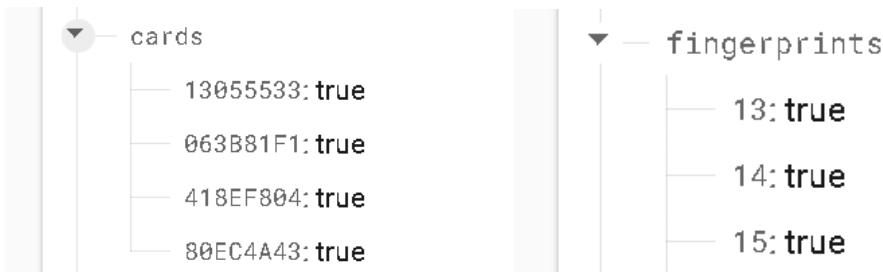
Hình 4.22: Nhập mật khẩu mới

Sau khi nhập mật khẩu mới thành công thì hệ thống sẽ ghi nhận lại và hiển thị lên LCD kết quả là “THANH CONG”, “CAI DAT THANH CONG”.



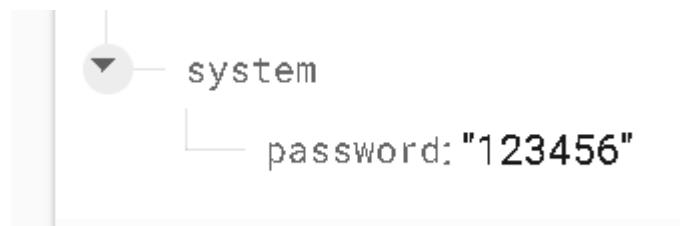
Hình 4.23: Khi thay đổi mật khẩu thành công

Và trên Firebase sẽ lưu trữ lại các dữ liệu cài đặt của thẻ RFID (Hình 4.24), vân tay (Hình 4.25) và mật khẩu (Hình 4.26), nếu người dùng muốn thay đổi thì có thể thay đổi trực tiếp trên Firebase:



Hình 4.24: Thông tin thẻ RFID

Hình 4.25: Thông tin vân tay



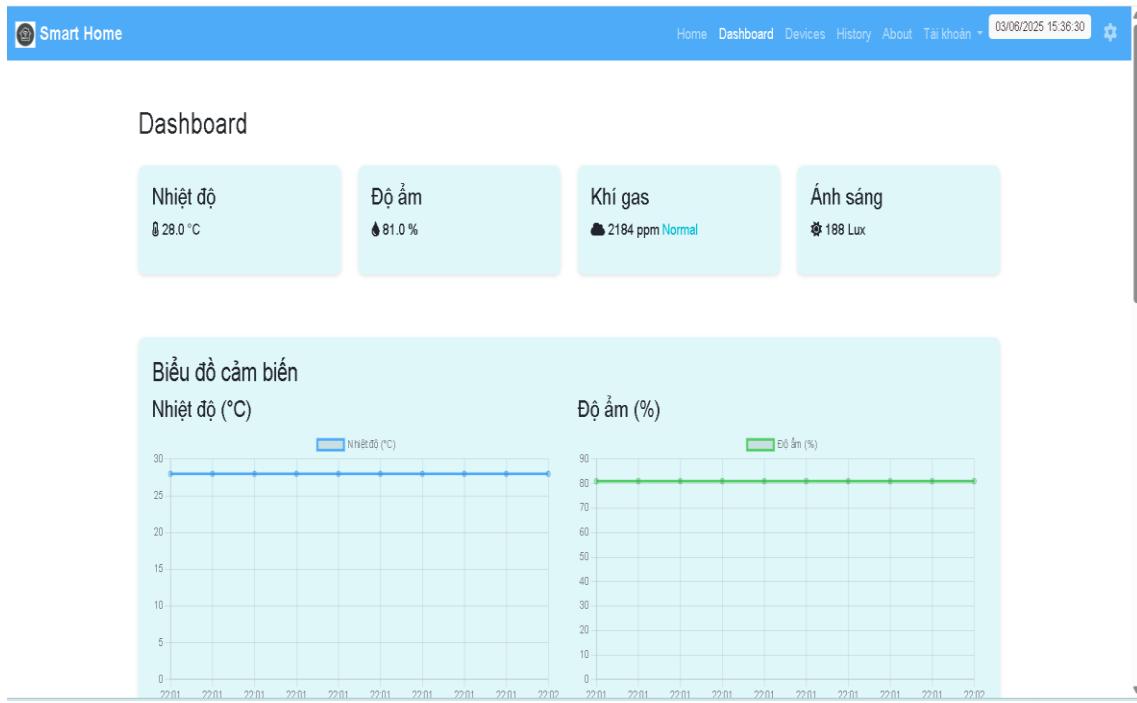
Hình 4.26: Thông tin mật khẩu

4.2. GIAO DIỆN HIỂN THỊ WEB

4.2.1. Giao diện Dashboard

Dưới đây là hình ảnh về giao diện dashboard lưu trữ lịch sử dữ liệu:

Hình 4.27 là giao diện giám sát và điều khiển thiết bị, thể hiện được tổng quan của trang chủ chính của hệ thống khi hoạt động. Giao diện được thiết kế gồm các thành phần chính sau đây:

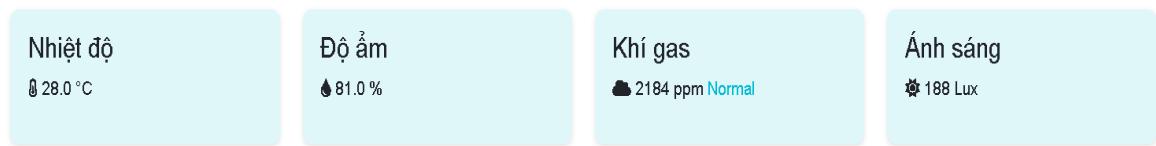


Hình 4.27: Giao diện chính

Ở phần Dashboard là nơi chứa các thông tin của dữ liệu cảm biến, biểu đồ và điều khiển thiết bị.

- Khu vực cảm biến

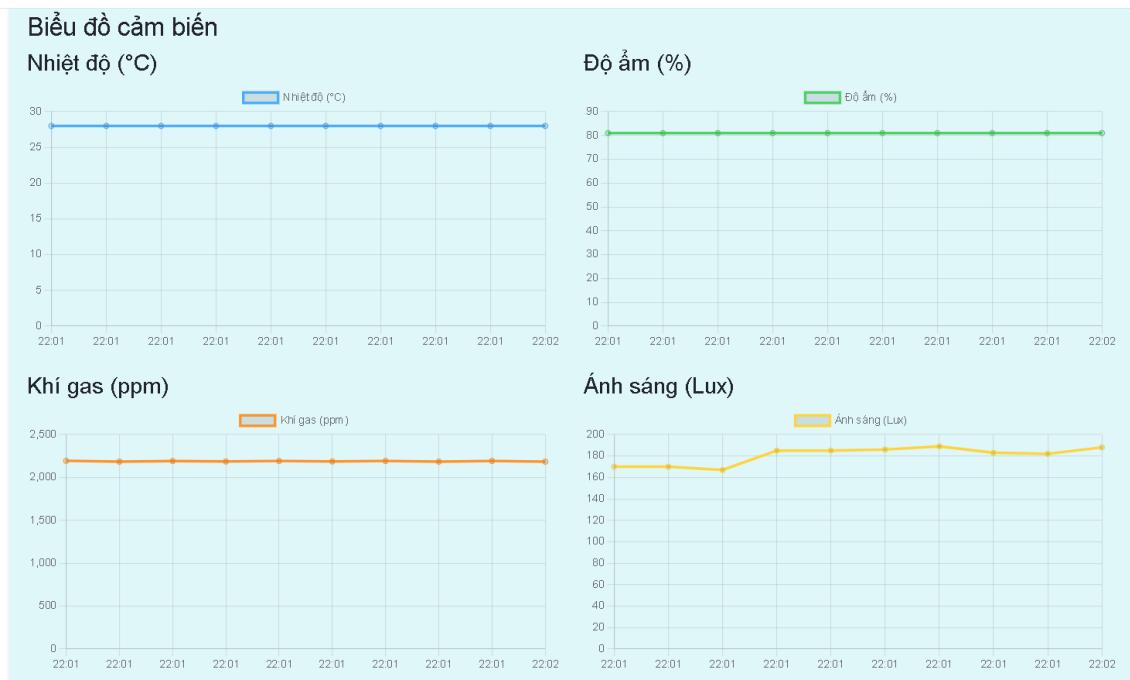
Dashboard



Hình 4.28: Khu vực hiển thị dữ liệu cảm biến

Hình 4.28, là nơi mà dữ liệu của các cảm biến hiển thị và thay đổi liên tục theo thời gian thực, dữ liệu này được cập nhập từ Firebase.

- Khu vực biểu đồ



Hình 4.29: Khu vực hiển thị biểu đồ

Hình 4.29, là nơi mà biểu đồ của các thiết bị sẽ được cập nhập liên tục và thay đổi theo thời gian thực. Để người dùng có thể theo dõi một cách chân thực nhất về sự thay đổi của dữ liệu.

- Khu vực điều khiển thiết bị

Điều khiển thiết bị

Phòng Ngủ - Đèn
? BẬT TẮT

Phòng Bếp - Đèn
? BẬT TẮT

Phòng Khách - Đèn
? BẬT TẮT

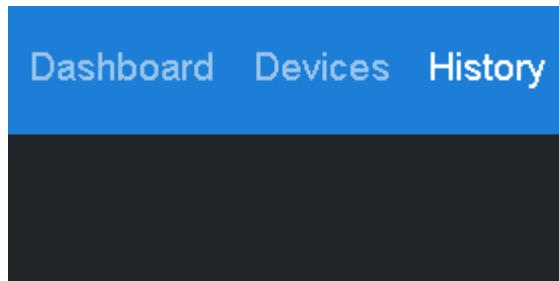
Vườn - Đèn
? BẬT TẮT

Hình 4.30: Khu vực điều khiển thiết bị

Hình 4.30, tại đây sẽ thực hiện việc điều khiển thiết bị từ xa thông qua Dashboard. Người dùng có thể truy cập vào web và bật/tắt các thiết bị trong nhà từ xa.

4.2.2. Giao diện lịch sử lưu trữ

Ngay tại giao diện chính ở Hình 4.27, nếu chọn vào phần History như Hình 4.31 thì sẽ vào phần lưu trữ lịch sử, tại đây thì sẽ là nơi mà các dữ liệu cảm biến mô tả ở Hình 4.32, điều khiển thiết bị mô tả ở Hình 4.33 và lịch sử cảnh báo khí gas được mô tả ở Hình 4.34 được lưu trữ lại giúp cho người dùng có thể kiểm tra lại bất cứ khi nào.



Hình 4.31: Giao diện History

- Khu vực lịch sử cảm biến

Bảng dữ liệu cảm biến

Thời gian	Nhiệt độ (°C)	Độ ẩm (%)	Khí gas	Cường độ ánh sáng (Lux)
15:02 03/06/2025	28.0	81.0	Normal	188
15:01 03/06/2025	28.0	81.0	Normal	182
15:01 03/06/2025	28.0	81.0	Normal	183
15:01 03/06/2025	28.0	81.0	Normal	189
15:01 03/06/2025	28.0	81.0	Normal	186

Hình 4.32: Bảng dữ liệu cảm biến

- Khu vực lịch sử điều khiển thiết bị

Lịch sử điều khiển thiết bị

Thời gian	Thiết bị	Hành động	Người thực hiện
10:12 02/06/2025	Garden	Bật	Nguyen T.A.Tuan
10:12 02/06/2025	Kitchen	Bật	Nguyen T.A.Tuan
10:12 02/06/2025	Livingroom	Bật	Nguyen T.A.Tuan
10:12 02/06/2025	Bedroom	Bật	Nguyen T.A.Tuan
10:12 02/06/2025	Garden	Tắt	Nguyen T.A.Tuan

Lịch sử cảnh báo

Hình 4.33: Lịch sử điều khiển thiết bị

- Khu vực lịch sử cảnh báo

Lịch sử cảnh báo

Thời gian	Loại cảnh báo	Mức độ	Hành động
14:50 03/06/2025	Khí gas vượt ngưỡng	Cao	Gửi thông báo
10:15 02/06/2025	Khí gas vượt ngưỡng	Cao	Gửi thông báo
10:15 02/06/2025	Khí gas vượt ngưỡng	Cao	Gửi thông báo
10:15 02/06/2025	Khí gas vượt ngưỡng	Cao	Gửi thông báo
10:14 02/06/2025	Khí gas vượt ngưỡng	Cao	Gửi thông báo

Hình 4.34: Lịch sử cảnh báo

- Tính năng xuất file Excel



Hình 4.35: Các tính năng khác

Hình 4.35, tại đây sẽ có các tính năng nổi bật đó là làm mới lại dữ liệu, xóa dữ liệu cũ và xuất file Excel. Nếu tích vào ba ô xanh như (Hình 4.35) thì khi xuất Excel thì sẽ xuất toàn bộ các dữ liệu đó (Hình 4.36), còn nếu chọn một trong ba thì chỉ xuất được dữ liệu mà đã chọn.

A	B	C	D	E
Thời gian	Nhiệt độ (°C)	Độ ẩm (%)	Khí gas	Cường độ ánh sáng (Lux)
3/6/2025 15:02	28	81	Normal	188
3/6/2025 15:01	28	81	Normal	182
3/6/2025 15:01	28	81	Normal	183
3/6/2025 15:01	28	81	Normal	189
3/6/2025 15:01	28	81	Normal	186
Thời gian	Thiết bị	Hành động	Người thực hiện	
2/6/2025 10:12	Garden	Bật	Nguyen T.A.Tuan	
2/6/2025 10:12	Kitchen	Bật	Nguyen T.A.Tuan	
2/6/2025 10:12	Livingroom	Bật	Nguyen T.A.Tuan	
2/6/2025 10:12	Bedroom	Bật	Nguyen T.A.Tuan	
2/6/2025 10:12	Garden	Tắt	Nguyen T.A.Tuan	
Thời gian	Loại cảnh báo	Mức độ	Hành động	
3/6/2025 14:50	Khí gas vượt ngưỡng	Cao	Gửi thông báo	
2/6/2025 10:15	Khí gas vượt ngưỡng	Cao	Gửi thông báo	
2/6/2025 10:15	Khí gas vượt ngưỡng	Cao	Gửi thông báo	
2/6/2025 10:15	Khí gas vượt ngưỡng	Cao	Gửi thông báo	
2/6/2025 10:14	Khí gas vượt ngưỡng	Cao	Gửi thông báo	

Hình 4.36: Xuất Excel toàn bộ dữ liệu

4.3. ĐÁNH GIÁ VÀ NHẬN XÉT HỆ THỐNG

Hệ thống giám sát nhà thông minh được phát triển dựa trên sự kết hợp STM32 và ESP32 đã được thử nghiệm và đánh giá qua các chức năng chính: giám sát môi trường, xác thực người dùng, điều khiển thiết bị và cài đặt an ninh. Dựa trên kết quả thực nghiệm, hệ thống hoạt động tốt, đáp ứng được các yêu cầu cơ bản về giám sát và quản lý ngôi nhà thông minh. Tuy nhiên, vẫn còn tồn tại một số hạn chế cần được cải thiện để tăng tính ổn định và hiệu quả trong môi trường thực tế.

4.3.1. Đánh giá hiệu suất

- Hiệu suất phần cứng: Mô hình phần cứng thực tế hoạt động ổn định, với các thành phần như cảm biến (DHT22, MQ-2, BH1750), servo và LCD được tích hợp tốt với STM32. Giao tiếp UART giữa STM32 và ESP32 đạt độ chính xác cao với tỷ lệ truyền chính xác liên tục trong quá trình hệ thống hoạt động. Các chế độ xác thực (keypad, RFID, vân tay) và chế độ cài đặt cũng đạt độ chính xác tầm khoảng ~90%, đảm bảo khả năng vận hành ổn định trong điều kiện thử nghiệm.
- Hiệu suất phần mềm: Giao diện dashboard trên Firebase hiển thị dữ liệu cảm biến và lịch sử điều khiển chính xác trong khoảng ~97%, với độ trễ

trung bình là 2 giây. Tính năng điều khiển từ xa qua dashboard có thời gian phản hồi trung bình khoảng 3 giây, cho phép người dùng bật/tắt thiết bị hoặc mở cửa từ xa một cách cơ bản.

4.3.2. Ưu điểm và những hạn chế của hệ thống

Ưu điểm:

- Tích hợp đa năng: Hệ thống kết hợp hiệu quả giữa phần cứng và phần mềm, cho phép giám sát, xác thực, điều khiển và cài đặt trong một nền tảng duy nhất, tạo ra một giải pháp toàn diện cho ngôi nhà thông minh.
- Bảo mật cao: Các phương thức xác thực mật khẩu, quét vân tay, quét thẻ và khả năng cài đặt linh hoạt đảm bảo bảo mật cho ngôi nhà, chỉ cho phép những người được ủy quyền truy cập vào hệ thống.
- Tích hợp từ xa: Giao diện dashboard cho phép người dùng theo dõi và điều khiển ngôi nhà mọi lúc, mọi nơi thông qua kết nối Internet, tăng sự tiện nghi và an toàn cho người dùng.
- Chi phí hợp lý: Sử dụng các linh kiện phổ biến và nền tảng Firebase miễn phí giúp giảm chi phí triển khai, phù hợp với các dự án cá nhân hoặc quy mô nhỏ.

Hạn chế:

- Độ nhạy của cảm biến vân tay: Khi đặt ngón tay không đúng vị trí hoặc trên bề mặt cảm biến có bụi thì sẽ dẫn đến xác thực thất bại, làm giảm độ tin cậy trong một số trường hợp.
- Độ trễ kết nối: Với độ trễ trung bình khoảng 3 giây trong việc điều khiển thiết bị từ xa qua dashboard có thể trải nghiệm của người dùng bị ảnh hưởng, đặc biệt là trong các tính hướng khẩn cấp.
- Khả năng mở rộng hạn chế: Hiện tại hệ thống chỉ sử dụng relay để điều khiển đèn cơ bản và servo để mở cửa, cần bổ sung thêm các thiết bị khác như quạt, điều hòa,...

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1. KẾT LUẬN

Hệ thống nhà thông minh đã được thiết kế và triển khai thành công, tích hợp hiệu quả giữa phần cứng và phần mềm để đáp ứng các yêu cầu cơ bản về giám sát môi trường, xác thực người dùng, điều khiển thiết bị, và cài đặt an ninh. Hệ thống sử dụng STM32 để xử lý dữ liệu từ các cảm biến, điều khiển thiết bị và hiển thị thông tin trên LCD, trong khi ESP32 đảm nhiệm nhận diện người dùng và kết nối Wi-Fi và giao tiếp với Firebase Realtime Database.

Kết quả thực nghiệm cho thấy hệ thống hoạt động ổn định, với giao tiếp UART giữa STM32 và ESP32 đạt tỷ lệ truyền dữ liệu chính xác gần 100%. Các chức năng giám sát môi trường và điều khiển thiết bị hoạt động hiệu quả, với độ trễ trung bình 2 giây khi cập nhật dữ liệu trên dashboard và 3 giây khi điều khiển từ xa. Chế độ xác thực đạt độ chính xác khoảng 90%, và giao diện dashboard trên Firebase hiển thị dữ liệu với độ chính xác 97%, mang lại trải nghiệm tiện lợi cho người dùng cả tại chỗ và từ xa. Hệ thống cũng đảm bảo bảo mật cao với các phương thức xác thực đa dạng (mật khẩu, vân tay, thẻ RFID) và khả năng cài đặt linh hoạt, cùng với chi phí triển khai hợp lý nhờ sử dụng linh kiện phổ biến và nền tảng Firebase miễn phí.

Tuy nhiên, hệ thống vẫn còn một số hạn chế như độ nhạy của cảm biến vân tay, độ trễ kết nối khi điều khiển từ xa và khả năng mở rộng hạn chế. Dù vậy, hệ thống đã đạt được mục tiêu đề ra, cung cấp một giải pháp toàn diện và thực tế cho ngôi nhà thông minh, phù hợp với nhu cầu giám sát và quản lý của người dùng hiện đại.

5.2. HƯỚNG PHÁT TRIỂN

Dựa trên kết quả thực nghiệm và các hạn chế đã được xác định, hệ thống nhà thông minh có thể được cải thiện và mở rộng trong tương lai để nâng cao hiệu quả và đáp ứng tốt hơn nhu cầu thực tế. Một số hướng phát triển cụ thể bao gồm:

- **Cải thiện độ nhạy của cảm biến vân tay:** Tích hợp thuật toán xử lý tín hiệu nâng cao hoặc sử dụng cảm biến vân tay thế hệ mới để giảm tỷ lệ xác thực thất bại khi ngón tay đặt sai vị trí hoặc bề mặt cảm biến có bụi.
- **Giảm độ trễ kết nối:** Tối ưu hóa giao thức truyền dữ liệu giữa ESP32 và Firebase hoặc bổ sung bộ nhớ đệm cục bộ để giảm độ trễ khi điều khiển từ xa, cải thiện trải nghiệm người dùng xuống dưới 1 giây.
- **Mở rộng khả năng điều khiển thiết bị:** Tích hợp thêm các thiết bị gia dụng như quạt, điều hòa, cửa tự động bằng cách nâng cấp module relay hoặc sử dụng các giao thức điều khiển không dây như Zigbee, tăng tính ứng dụng của hệ thống.
- **Phát triển tính năng thông báo thời gian thực:** Nâng cấp giao diện dashboard trên Firebase để hỗ trợ gửi thông báo qua email hoặc ứng dụng di động khi phát hiện sự cố (như rò rỉ khí gas hoặc xác thực thất bại), tăng tính an toàn.
- **Tích hợp ứng dụng di động:** Xây dựng ứng dụng trên Android/iOS để thay thế hoặc bổ sung cho Web Dashboard, giúp người dùng quản lý hệ thống tiện lợi hơn trên thiết bị di động.
- **Nâng cấp bảo mật và lưu trữ:** Tăng dung lượng lưu trữ vân tay trên AS608 hoặc sử dụng cơ chế mã hóa dữ liệu trên Firebase để tăng cường bảo mật thông tin người dùng.

PHỤ LỤC

Các chương trình điều khiển, chương trình thiết kế web của hệ thống được đặt trong đường dẫn sau:

<https://github.com/nguyen-tai-anh-tuan/DO-AN-1>

Link truy cập vào giao diện web dashboard:

<https://nguyen-tai-anh-tuan.github.io/web-dashboard-DA1/>

Link video demo phần cứng:

<https://youtu.be/7rocfGUtAUA?si=6O8CC3n-9ZJzJ-tH>

Link Github cá nhân:

<https://github.com/nguyen-tai-anh-tuan>

TÀI LIỆU THAM KHẢO

- [1] Simar Singh, Sourabh Anand, and M. K. Satyarthi, "A Comprehensive Review of Smart Home Automation Systems," *Advances in Computer Science and Information Technology (ACSIT)*, vol. 10, pp. 61-66, 2023.
- [2] A. S. al-Sumaiti, M. H. Ahmed, and M. M. A. Salama, "Smart Home Activities: A Literature Review," *Electric Power Components and Systems*, tập 42, pp. 294-305, 2014.
- [3] Nguyễn Hùng Long và Trịnh Xuân Hùng, "Thiết kế hệ thống nhà thông minh dùng chuẩn không dây Zigbee," Đò án tốt nghiệp, Khoa Kỹ thuật - Công nghệ, Trường ĐH Bà Rịa - Vũng Tàu, Vũng Tàu, 2022.
- [4] Nguyễn Văn Thắng, Nguyễn Trọng Đức, "Ứng dụng Internet of Things xây dựng ngôi nhà thông," Đề tài NCKH cấp Trường, Trường Đại học Hàng hải, 2016.
- [5] Robles, R. J., & Kim, T. H., "Applications, systems and methods in smart home technology: a review," *International Journal of Advanced Science and Technology*, p. 15, 2010.
- [6] Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., & Ayyash, "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, pp. 2347-2376, 2015.
- [7] STMicroelectronics, "STM32F407/417 Technical Reference Manual," UM1472, STMicroelectronics, 2023.
- [8] E. Systems, "ESP32 Technical Reference Manual," Espressif System, 2023.
- [9] T. Liu, "Digital-output relative humidity & temperature sensor DHT22," Aosong , China, 2020.
- [10] R. Semiconductor, "BH1750 Ambient Light Sensor IC Datasheet," Rohm Semiconductor, 2022.
- [11] H. Electronics, "MQ-2 Semiconductor Sensor for Combustible Gas," Hanwei Electronics, 2021.

- [12] N. Semiconductors, “MFRC522,” NXP Semiconductors, 2010.
- [13] Alientek, “ATK-AS608 指纹模块用户手册,” ALIENTEK, China, 2016.
- [14] H. Teachnology, “4-Channel Relay Module Datasheet,” Hongfa Teachnology, 2023.
- [15] T. Pro, “MG90S Micro Sensor Motor Datasheet,” Tower Pro, 2022.
- [16] N. Semiconductors, “PCF8574/PCF8574A Remote 8-bit I/O expander for I2C-bus,” NXP Semiconductors, 2011.
- [17] Hitachi, “HD44780U (LCD-II),” Hitachi, Ltd, Japan.
- [18] Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, “Internet of Things (IoT): A vision, architectural elements, and future directions,” *Future Generation Computer Systems*, pp. 1645-1660, 2013.