

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA KỸ THUẬT ĐIỆN TỬ 2



BÁO CÁO TỔNG KẾT

Môn: Mạng cảm biến

CHỦ ĐỀ: HOME ASSISTANT
SỬ DỤNG RASPBERRY PI VÀ ZIGBEE

Giảng viên hướng dẫn: Thầy Hồ Nhật Minh

NHÓM 5

Tên thành viên	MSSV	Lớp
Nguyễn Đức Hùng	N21DCDT033	D21CQDT01-N
Phùng Đình Khôi	N21DCDT048	D21CQDT01-N

Ngày 26 tháng 10 năm 2024

MỤC LỤC

MỤC LỤC	1
DANH MỤC HÌNH ẢNH	2
1. GIỚI THIỆU.....	3
1.1. Khái quát đề tài	3
1.2. Mục tiêu.....	3
1.3. Phạm vi nghiên cứu.....	4
1.4. Phương pháp thực hiện.....	4
2. CÁC THIẾT BỊ SỬ DỤNG	5
2.1. Raspberry Pi 4 model B	5
2.2. Usb zigbee cc2531	6
2.3. Cảm biến nhiệt độ aqara t1 và công tắc tuya (zigbee)	7
3. CÁC PHẦN MỀM TRONG RASPBERRY PI	8
3.1. Home assistant.....	8
3.2. Zigbee2MQTT	8
3.3. MQTT Broker	9
3.4. Web bằng flask.....	10
3.4.1. <i>Thiết kế giao diện người dùng (UI)</i>	10
3.4.2. <i>Kết nối với Home Assistant API</i>	10
3.4.3. <i>Triển khai trang web trên Raspberry Pi</i>	11
4. SƠ ĐỒ KHỐI VÀ CÁCH THỨC HOẠT ĐỘNG	11
4.1. Sơ đồ khối	11
4.2. Cách thức hoạt động.....	11
4.3. Kết quả thực hiện	13
4.4. Đánh giá	13
KẾT LUẬN	15
TÀI LIỆU THAM KHẢO.....	15

DANH MỤC HÌNH ẢNH

Hình 1.1. Ngôi nhà thông minh	3
Hình 1.2. Mô hình nhà thông minh cơ bản.....	4
Hình 2.1. Bo Mạch Đơn Máy Tính Nhúng Raspberry Pi 4.....	5
Hình 2.2. Mạch Thu Phát ZigBee CC2531 USB 2.4Ghz.....	7
Hình 2.3. Cảm biến nhiệt độ, độ ẩm Aqara T1	7
Hình 2.4. Công tắc Tuya 4 nút	8
Hình 3.1. Giao diện web hoàn chỉnh	10
Hình 3.2. Code dùng API của home assistant để lấy data.....	10
Hình 3.3. Thư viện flask để code backend	11
Hình 4.1. Sơ đồ khối.....	11
Hình 4.2. Data lưu vào file .txt mỗi 30 phút.....	12

1. GIỚI THIỆU

1.1. Khái quát đề tài

Tự động hóa nhà thông minh đang trở thành một xu hướng không thể thiếu trong cuộc sống hiện đại, đang thay đổi cách chúng ta tương tác với không gian sống của mình. Với sự phát triển của công nghệ, việc kiểm soát và quản lý các thiết bị trong ngôi nhà trở nên dễ dàng hơn bao giờ hết. Nhờ vào một trang web điều khiển các thiết bị thông minh, bạn có thể

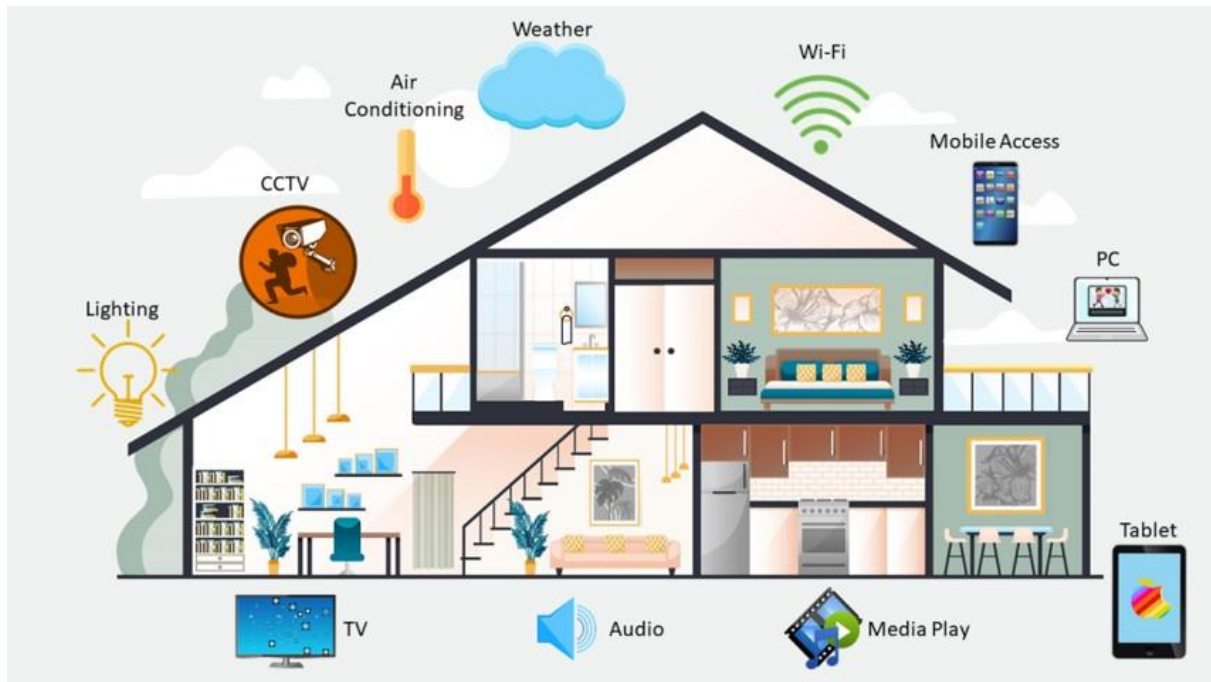


Hình 1.1. Ngôi nhà thông minh

theo dõi tình trạng và điều khiển từng thiết bị chỉ bằng một cú nhấp chuột ngay trên trình duyệt của mình, bất kể bạn đang ở đâu. Trong bài viết này, chúng ta sẽ cùng nhau khám phá những bước cần thiết để sử dụng Zigbee kết hợp với Raspberry Pi, tạo ra một trang web quản lý nhà thông minh hiệu quả. Chúng ta sẽ đi sâu vào các thành phần cần thiết, cách thiết lập hệ thống, cũng như những tính năng thú vị mà hệ thống này có thể mang lại cho cuộc sống hàng ngày của bạn. Hãy cùng nhau biến ngôi nhà của bạn trở thành một không gian sống thông minh và tiện nghi hơn.

1.2. Mục tiêu

Là sinh viên ngành Điện - Điện tử, nhóm tôi luôn tìm tòi và sáng tạo trong việc áp dụng công nghệ vào cuộc sống. Dự án của nhóm tập trung vào việc phát triển hệ thống nhà thông minh dựa trên sóng Zigbee và Raspberry Pi, nhằm biến những ngôi nhà truyền thống trở nên thông minh và tiện ích hơn. Khi không bận rộn với bài tập hoặc thực hiện các dự án cá nhân, tôi thường nghiên cứu các công nghệ tự động hóa, đặc biệt là cách các giao thức không dây có thể tối ưu hóa việc điều khiển thiết bị trong nhà.



Hình 1.2. Mô hình nhà thông minh cơ bản

1.3. Phạm vi nghiên cứu

Bài nghiên cứu tập trung vào việc thiết kế và triển khai một hệ thống nhà thông minh sử dụng Raspberry Pi và giao thức Zigbee để giám sát và điều khiển các thiết bị thông minh. Phạm vi nghiên cứu bao gồm:

- Xây dựng các thành phần phần cứng như Raspberry Pi, mô-đun Zigbee, các cảm biến nhiệt độ, độ ẩm và công tắc thông minh. Song song đó là phát triển giao diện web để điều khiển và hiển thị thông tin từ các cảm biến.
- Nghiên cứu và áp dụng Zigbee cho các thiết bị nhằm tạo mạng lưới truyền tải dữ liệu ổn định, tiết kiệm năng lượng và dễ dàng mở rộng.
- Xác định tính ổn định, độ tin cậy và hiệu suất của hệ thống khi vận hành, đặc biệt trong các tình huống giám sát và điều khiển từ xa.

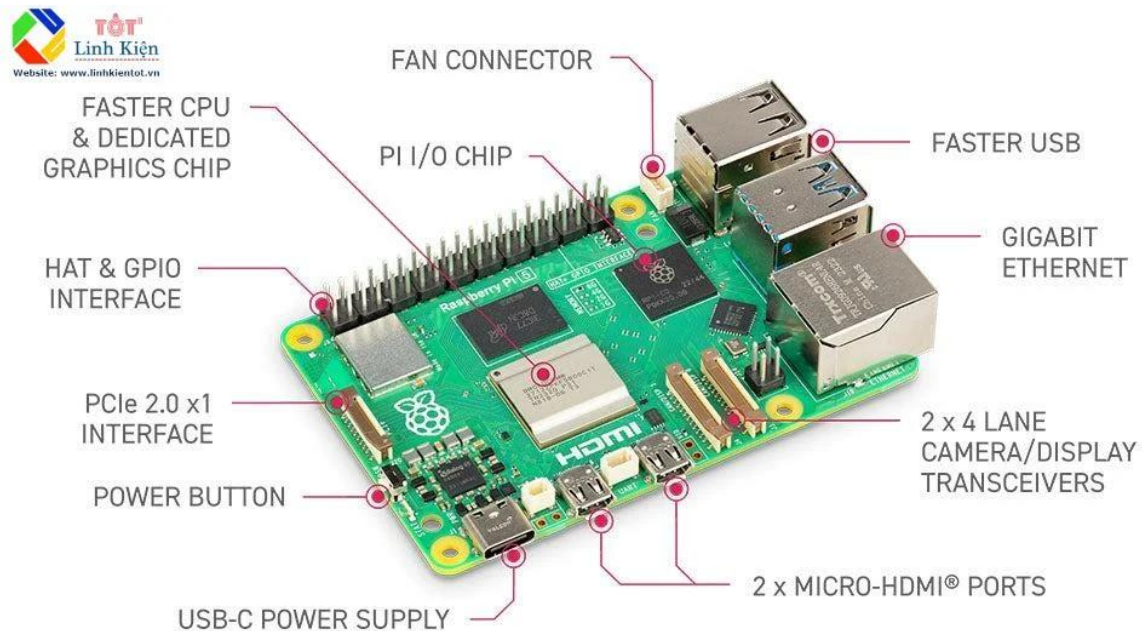
1.4. Phương pháp thực hiện

- Tìm hiểu về công nghệ nhà thông minh, các giao thức truyền thông như Zigbee, và các thiết bị như Raspberry Pi, cảm biến nhiệt độ, độ ẩm. Nghiên cứu cũng bao gồm các phương pháp bảo mật cho giao diện web và giao tiếp dữ liệu.

- Lắp đặt phần cứng gồm Raspberry Pi và các thiết bị Zigbee, cấu hình mạng Zigbee để các thiết bị có thể giao tiếp với Raspberry Pi. Phát triển giao diện web cho phép điều khiển công tắc, hiển thị dữ liệu từ cảm biến và lưu trữ lịch sử dữ liệu.
- Thực hiện các bài kiểm thử chức năng để đảm bảo các thiết bị hoạt động đúng như mong muốn. Kiểm tra độ trễ, tính ổn định, và khả năng đáp ứng của hệ thống khi số lượng thiết bị gia tăng hoặc khi khoảng cách giữa các thiết bị và trung tâm điều khiển thay đổi.
- Đánh giá các kết quả đạt được so với mục tiêu ban đầu. Dựa trên các chỉ số về hiệu suất, độ tin cậy, và khả năng mở rộng, đưa ra các đánh giá về ưu và nhược điểm của hệ thống.
- Dựa vào kết quả phân tích, đề xuất các cải tiến để nâng cao hiệu suất và khả năng ứng dụng của hệ thống.

2. CÁC THIẾT BỊ SỬ DỤNG

2.1. Raspberry Pi 4 model B



Hình 2.1. Bo Mạch Đơn Máy Tính Nhúng Raspberry Pi 4

- **Raspberry Pi 4 Model B** là phiên bản cải tiến mạnh mẽ của dòng máy tính mini Raspberry Pi, với hiệu suất và khả năng kết nối được cải thiện đáng kể so với các

phiên bản trước. Đây là một máy tính đơn bo mạch nhỏ gọn, giá thành thấp, nhưng có đủ sức mạnh để thực hiện nhiều tác vụ, từ việc lập trình, học tập, đến các ứng dụng IoT (Internet of Things) và nhà thông minh.

- **Bộ vi xử lý:** Broadcom BCM2711, vi xử lý ARM Cortex-A72 (ARM v8) 64-bit, 4 nhân, tốc độ 1.5GHz.
- **Bộ nhớ RAM:** Có các tùy chọn 2GB, 4GB hoặc 8GB LPDDR4-3200 SDRAM.
- **Đồ họa:** GPU VideoCore VI hỗ trợ OpenGL ES 3.0, giải mã video 4K H.265 ở tốc độ 60fps.
- **Cổng USB:** 2 cổng USB 3.0 và 2 cổng USB 2.0, tăng khả năng kết nối với các thiết bị ngoại vi tốc độ cao.
- **Cổng mạng:** Cổng Gigabit Ethernet, giúp tăng tốc độ truyền dữ liệu qua mạng lên đến 1 Gbps.
- **Wi-Fi và Bluetooth:** Tích hợp Wi-Fi chuẩn 802.11ac (băng tần kép 2.4 GHz và 5.0 GHz) và Bluetooth 5.0.
- **Cổng HDMI:** 2 cổng micro-HDMI hỗ trợ độ phân giải 4K kép, có thể xuất hình ảnh ra 2 màn hình cùng lúc.
- **Cổng GPIO:** 40 chân GPIO để kết nối với các thiết bị ngoại vi và cảm biến.
- **Bộ nhớ lưu trữ:** Không có bộ nhớ lưu trữ cố định; sử dụng thẻ microSD để chạy hệ điều hành và lưu trữ dữ liệu.

2.2. Usb zigbee cc2531

- **USB Zigbee dongle CC2531** là một thiết bị nhỏ cắm vào cổng USB, hoạt động như một bộ điều khiển Zigbee (coordinator), cho phép giao tiếp giữa hệ thống máy tính (như Raspberry Pi) và các thiết bị Zigbee (cảm biến, bóng đèn thông minh, khóa cửa, v.v.).
- **Mục đích:** Kết nối các thiết bị Zigbee với hệ thống nhà thông minh của bạn mà không cần hub riêng biệt từ các nhà sản xuất như Philips Hue, Xiaomi hay IKEA, Phối hợp với các phần mềm như Zigbee2MQTT, Home Assistant, OpenHAB, hoặc Node-RED để điều khiển thiết bị Zigbee.
- **Lợi ích** khi sử dụng USB Zigbee:

- Đa dạng thiết bị: Kết nối với các thiết bị Zigbee từ nhiều hãng khác nhau mà không cần mua nhiều hub riêng.
- Chi phí thấp: Giảm chi phí vì không cần các trung tâm điều khiển riêng biệt.
- Tính tùy biến: Có thể cấu hình và điều chỉnh theo nhu cầu riêng thông qua Zigbee2MQTT và MQTT broker.

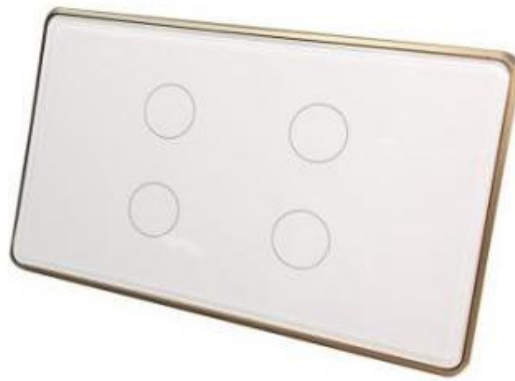


Hình 2.2. Mạch Thu Phát ZigBee CC2531 USB 2.4Ghz

2.3. Cảm biến nhiệt độ aqara t1 và công tắc tuya (zigbee)



Hình 2.3. Cảm biến nhiệt độ, độ ẩm Aqara T1



Hình 2.4. Công tắc Tuya 4 nút

3. CÁC PHẦN MỀM TRONG RASPBERRY PI

3.1. Home assistant

Home Assistant là một nền tảng mã nguồn mở dành cho hệ thống nhà thông minh, cho phép giám sát, điều khiển và tự động hóa các thiết bị trong nhà. Nó hỗ trợ rất nhiều thiết bị từ các nhà sản xuất khác nhau và có thể chạy trên nhiều nền tảng phần cứng, bao gồm Raspberry Pi.

– Các tính năng chính của Home Assistant:

- **Điều khiển thiết bị:** Quản lý và giám sát các thiết bị thông minh từ nhiều hãng.
- **Tự động hóa:** Tạo kịch bản tự động cho các thiết bị (ví dụ: bật đèn khi phát hiện chuyển động).
- **Tích hợp:** Hỗ trợ hàng ngàn thiết bị và dịch vụ như Zigbee, Z-Wave, Google Assistant, Alexa.
- **Giao diện trực quan:** Điều khiển qua ứng dụng di động hoặc giao diện web.
- **Bảo mật:** Lưu trữ dữ liệu cục bộ, đảm bảo quyền riêng tư.

3.2. Zigbee2MQTT

Zigbee2MQTT là phần mềm mã nguồn mở cho phép kết nối và điều khiển các thiết bị Zigbee thông qua giao thức MQTT.

- Chức năng của Zigbee2MQTT:
 - **Cầu nối:** Chuyển đổi tín hiệu từ thiết bị Zigbee sang giao thức MQTT, giúp tích hợp các thiết bị Zigbee vào hệ thống nhà thông minh mà không cần hub riêng biệt.
 - **Tương thích:** Hỗ trợ hàng ngàn thiết bị từ nhiều nhà sản xuất khác nhau như Philips Hue, Xiaomi, Sonoff, và IKEA.
 - **Tùy biến cao:** Có thể cấu hình và mở rộng các chức năng để phù hợp với nhu cầu của mình.

3.3. MQTT Broker

MQTT Broker là một thành phần chính trong giao thức MQTT (Message Queuing Telemetry Transport). Nó đóng vai trò như một "trung tâm điều phối" trung gian giữa các thiết bị hoặc ứng dụng trong hệ thống IoT, như Home Assistant và Zigbee2MQTT.

- Publisher (Người phát): Các thiết bị hoặc dịch vụ (như Zigbee2MQTT) gửi thông điệp đến MQTT broker với một chủ đề cụ thể.
- Subscriber (Người nhận): Các thiết bị hoặc dịch vụ khác (như Home Assistant) đăng ký lắng nghe (subscribe) các chủ đề mà chúng quan tâm.
- MQTT broker sẽ nhận thông điệp từ publisher và phân phối thông điệp đó cho các subscriber đang đăng ký lắng nghe cùng chủ đề.
- Giao tiếp dựa trên chủ đề (topic):
 - Zigbee2MQTT sẽ gửi thông tin từ cảm biến đến broker qua 1 chủ đề.
 - Home Assistant lắng nghe chủ đề đó để nhận dữ liệu và hiển thị hoặc xử lý thông tin cảm biến.
- Hỗ trợ kết nối bất đồng bộ: MQTT broker cho phép thiết bị và dịch vụ giao tiếp mà không cần kết nối liên tục. Nếu một thiết bị bị ngắt kết nối tạm thời, broker có thể lưu trữ thông điệp và gửi lại khi thiết bị quay trở lại.
- Quản lý kết nối giữa các thiết bị: MQTT broker cũng quản lý việc duy trì kết nối với các thiết bị IoT trong hệ thống, theo dõi ai đang online hoặc offline.

3.4. Web bằng flask

3.4.1. Thiết kế giao diện người dùng (UI)

Sử dụng các công cụ HTML, để tạo một giao diện dễ dùng, trực quan. Có thể tạo các nút điều khiển để bật/tắt thiết bị, điều chỉnh nhiệt độ, ánh sáng, hoặc hiển thị các thông tin từ cảm biến như nhiệt độ và độ ẩm.



Hình 3.1. Giao diện web hoàn chỉnh

3.4.2. Kết nối với Home Assistant API

Sử dụng các API của Home Assistant để lấy dữ liệu từ các thiết bị Zigbee và điều khiển chúng. Ví dụ, bạn có thể gửi yêu cầu từ trang web để bật/tắt bóng đèn thông minh hoặc theo dõi trạng thái của cảm biến cửa.

```

/
8 # Thông tin kết nối đến Home Assistant
9 HA_URL1 = "http://192.168.2.138:8123/api/states/switch.0xa4c138d7168fb636_l1"
10 HA_URL2 = "http://192.168.2.138:8123/api/states/switch.0xa4c138d7168fb636_l2"
11 HA_URL3 = "http://192.168.2.138:8123/api/states/switch.0xa4c138d7168fb636_l3"
12 HA_URL4 = "http://192.168.2.138:8123/api/states/switch.0xa4c138d7168fb636_l4"
13
14 # Thông tin cảm biến nhiệt độ, độ ẩm và áp suất
15 HA_URL_PRESSURE = "http://192.168.2.138:8123/api/states/sensor.0x54ef441000d1f33a_pressure"
16 HA_URL_HUMIDITY = "http://192.168.2.138:8123/api/states/sensor.0x54ef441000d1f33a_humidity"
17 HA_URL_TEMPERATURE = "http://192.168.2.138:8123/api/states/sensor.0x54ef441000d1f33a_temperature"
18

```

Hình 3.2. Code dùng API của home assistant để lấy data

3.4.3. Triển khai trang web trên Raspberry Pi

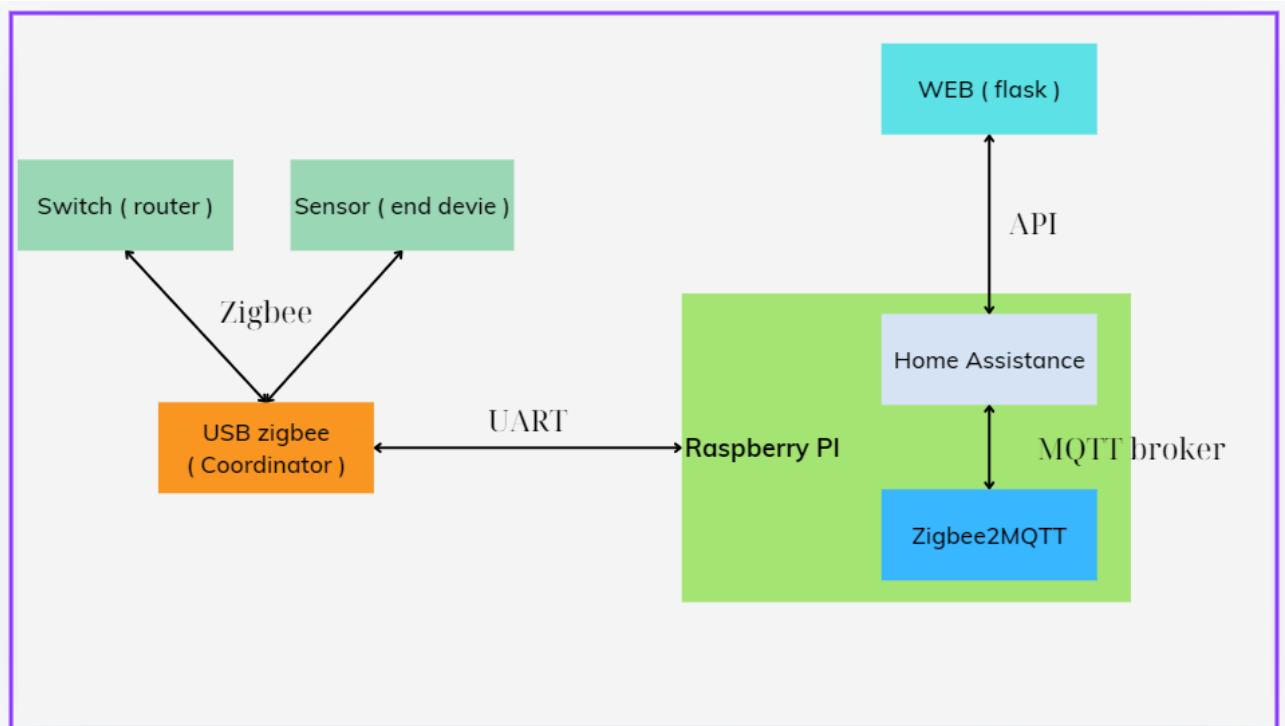
Sau khi phát triển trang web, bạn có thể triển khai nó trực tiếp trên Raspberry Pi. Dùng Flask để xây dựng backend, xử lý các yêu cầu từ người dùng và tương tác với hệ thống nhà thông minh.

```
1 from flask import Flask, render_template, jsonify
2 import requests
3 import time
4 import threading
5
6 app = Flask(__name__)
7
```

Hình 3.3. Thư viện flask để code backend

4. SƠ ĐỒ KHỐI VÀ CÁCH THỨC HOẠT ĐỘNG

4.1. Sơ đồ khối



Hình 4.1. Sơ đồ khối

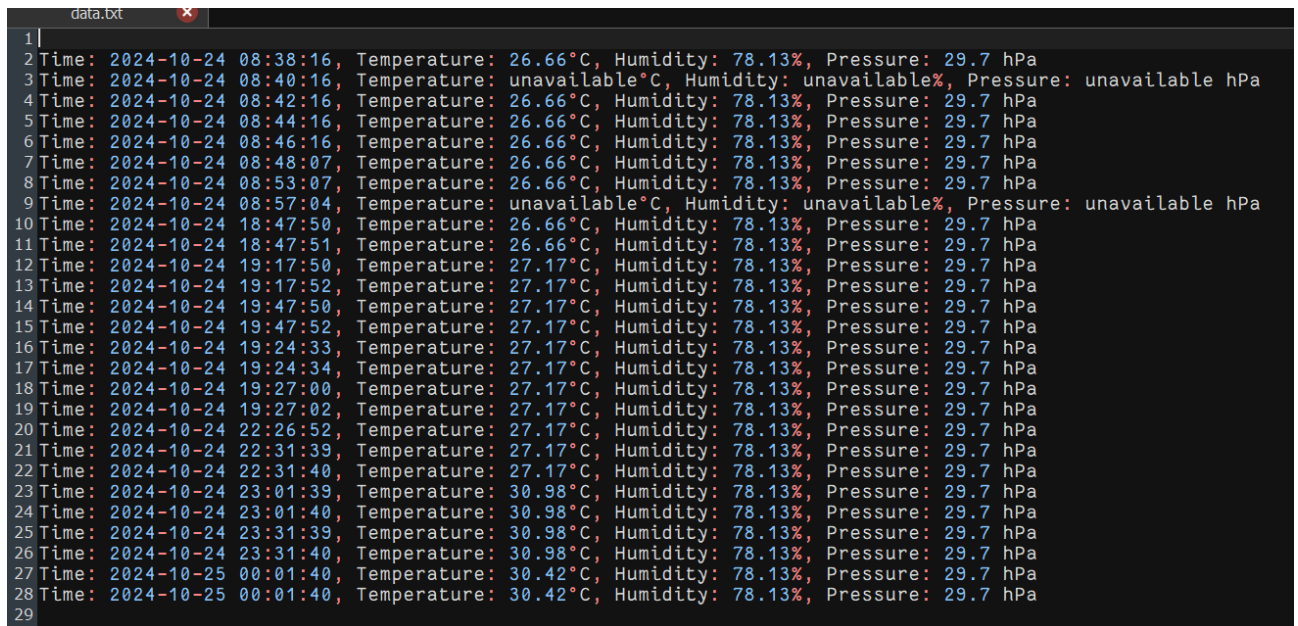
4.2. Cách thức hoạt động

Trong hệ thống này, các thiết bị như Router và các thiết bị đầu cuối (end devices) sẽ truyền dữ liệu đến Coordinator thông qua sóng Zigbee. Coordinator là trung tâm điều khiển trong mạng Zigbee, có nhiệm vụ thu thập và xử lý dữ liệu từ các thiết bị đầu cuối. Sau khi xử lý xong, Coordinator sẽ truyền dữ liệu này về Raspberry Pi thông qua cổng UART (Universal Asynchronous Receiver-Transmitter).

Khi dữ liệu đến Raspberry Pi, nó sẽ được chuyển tiếp đến Zigbee2MQTT. Zigbee2MQTT sẽ tiếp nhận và đẩy dữ liệu đã được xử lý lên Home Assistant qua MQTT broker. Home Assistant hoạt động như một nền tảng quản lý và tự động hóa thiết bị, cho phép người dùng theo dõi và điều khiển các thiết bị trong mạng.

Home Assistant cung cấp một API (Application Programming Interface) cho phép ứng dụng web mà bạn tạo ra có thể gửi và nhận các hoạt động điều khiển từ người dùng. Điều này có nghĩa là người dùng có thể tương tác với hệ thống thông qua giao diện web, và tất cả các đường truyền dữ liệu trong hệ thống đều có khả năng truyền tải hai chiều.

Ngoài ra, để tăng cường khả năng theo dõi, hệ thống sẽ tích hợp tính năng lưu trữ dữ liệu từ cảm biến. Cụ thể, sau mỗi 30 phút, dữ liệu nhiệt độ từ các cảm biến sẽ được lưu lại vào một file. File này sẽ cho phép người dùng tra cứu lịch sử nhiệt độ một cách dễ dàng, giúp họ có cái nhìn tổng quan về biến động nhiệt độ theo thời gian.



```

1 |
2 Time: 2024-10-24 08:38:16, Temperature: 26.66°C, Humidity: 78.13%, Pressure: 29.7 hPa
3 Time: 2024-10-24 08:40:16, Temperature: unavailable°C, Humidity: unavailable%, Pressure: unavailable hPa
4 Time: 2024-10-24 08:42:16, Temperature: 26.66°C, Humidity: 78.13%, Pressure: 29.7 hPa
5 Time: 2024-10-24 08:44:16, Temperature: 26.66°C, Humidity: 78.13%, Pressure: 29.7 hPa
6 Time: 2024-10-24 08:46:16, Temperature: 26.66°C, Humidity: 78.13%, Pressure: 29.7 hPa
7 Time: 2024-10-24 08:48:07, Temperature: 26.66°C, Humidity: 78.13%, Pressure: 29.7 hPa
8 Time: 2024-10-24 08:53:07, Temperature: 26.66°C, Humidity: 78.13%, Pressure: 29.7 hPa
9 Time: 2024-10-24 08:57:04, Temperature: unavailable°C, Humidity: unavailable%, Pressure: unavailable hPa
10 Time: 2024-10-24 18:47:50, Temperature: 26.66°C, Humidity: 78.13%, Pressure: 29.7 hPa
11 Time: 2024-10-24 18:47:51, Temperature: 26.66°C, Humidity: 78.13%, Pressure: 29.7 hPa
12 Time: 2024-10-24 19:17:50, Temperature: 27.17°C, Humidity: 78.13%, Pressure: 29.7 hPa
13 Time: 2024-10-24 19:17:52, Temperature: 27.17°C, Humidity: 78.13%, Pressure: 29.7 hPa
14 Time: 2024-10-24 19:47:50, Temperature: 27.17°C, Humidity: 78.13%, Pressure: 29.7 hPa
15 Time: 2024-10-24 19:47:52, Temperature: 27.17°C, Humidity: 78.13%, Pressure: 29.7 hPa
16 Time: 2024-10-24 19:24:33, Temperature: 27.17°C, Humidity: 78.13%, Pressure: 29.7 hPa
17 Time: 2024-10-24 19:24:34, Temperature: 27.17°C, Humidity: 78.13%, Pressure: 29.7 hPa
18 Time: 2024-10-24 19:27:00, Temperature: 27.17°C, Humidity: 78.13%, Pressure: 29.7 hPa
19 Time: 2024-10-24 19:27:02, Temperature: 27.17°C, Humidity: 78.13%, Pressure: 29.7 hPa
20 Time: 2024-10-24 22:26:52, Temperature: 27.17°C, Humidity: 78.13%, Pressure: 29.7 hPa
21 Time: 2024-10-24 22:31:39, Temperature: 27.17°C, Humidity: 78.13%, Pressure: 29.7 hPa
22 Time: 2024-10-24 22:31:40, Temperature: 27.17°C, Humidity: 78.13%, Pressure: 29.7 hPa
23 Time: 2024-10-24 23:01:39, Temperature: 30.98°C, Humidity: 78.13%, Pressure: 29.7 hPa
24 Time: 2024-10-24 23:01:40, Temperature: 30.98°C, Humidity: 78.13%, Pressure: 29.7 hPa
25 Time: 2024-10-24 23:31:39, Temperature: 30.98°C, Humidity: 78.13%, Pressure: 29.7 hPa
26 Time: 2024-10-24 23:31:40, Temperature: 30.98°C, Humidity: 78.13%, Pressure: 29.7 hPa
27 Time: 2024-10-25 00:01:40, Temperature: 30.42°C, Humidity: 78.13%, Pressure: 29.7 hPa
28 Time: 2024-10-25 00:01:40, Temperature: 30.42°C, Humidity: 78.13%, Pressure: 29.7 hPa
29

```

Hình 4.2. Data lưu vào file .txt mỗi 30 phút

Nhờ vào cấu trúc này, hệ thống không chỉ có khả năng thu thập và xử lý dữ liệu mà còn cho phép người dùng điều khiển các thiết bị một cách linh hoạt và hiệu quả, đồng thời cung cấp thông tin lịch sử cần thiết để phân tích và đưa ra quyết định.

4.3. Kết quả thực hiện

- **Đáp ứng chức năng của từng khối như mục tiêu đề ra:** Hệ thống đã được triển khai và chạy đúng các chức năng của từng khối, đảm bảo đáp ứng đầy đủ các mục tiêu ban đầu. Các thành phần điều khiển và giám sát đều hoạt động ổn định, đáp ứng yêu cầu trong suốt quá trình kiểm thử.
- **Điều khiển công tắc thông minh:** Các công tắc thông minh có thể được điều khiển trực tiếp thông qua trang web, phản hồi nhanh và không có độ trễ đáng kể. Người dùng có thể dễ dàng bật/tắt các thiết bị từ xa chỉ với một thao tác đơn giản trên trình duyệt.
- **Hiển thị dữ liệu cảm biến lên giao diện web:** Dữ liệu nhiệt độ từ cảm biến được cập nhật liên tục và hiển thị trực quan trên giao diện web, giúp người dùng theo dõi được nhiệt độ môi trường một cách nhanh chóng và chính xác. Hệ thống cũng đảm bảo độ chính xác của dữ liệu, với các thông số được lưu trữ để người dùng có thể kiểm tra lại lịch sử.

4.4. Đánh giá

Hệ thống hoạt động ổn định, các chức năng điều khiển và hiển thị dữ liệu cảm biến phản hồi nhanh và không có độ trễ đáng kể. Tốc độ truyền tải và xử lý dữ liệu giữa các thiết bị và giao diện web đáp ứng tốt yêu cầu giám sát theo thời gian thực. Tuy nhiên, khi số lượng thiết bị kết nối tăng cao, có thể xuất hiện một số hạn chế về băng thông và tốc độ xử lý, đặc biệt khi sử dụng các bộ vi xử lý hạn chế như Raspberry Pi.

Hệ thống cho thấy độ tin cậy cao trong quá trình thử nghiệm với khả năng duy trì kết nối ổn định và hoạt động chính xác trong các chức năng chính. Tuy nhiên, tính tin cậy có thể bị ảnh hưởng bởi kết nối mạng không ổn định hoặc yếu tố về phạm vi phủ sóng của sóng Zigbee. Khi mạng Zigbee gặp sự cố, dữ liệu cảm biến có thể không được cập nhật đầy đủ.

- **Ưu điểm:**

- **Dễ sử dụng:** Người dùng có thể dễ dàng điều khiển các thiết bị và theo dõi dữ liệu từ cảm biến thông qua giao diện web, phù hợp với người dùng không chuyên về kỹ thuật.
- **Tiết kiệm chi phí:** Sử dụng Raspberry Pi và giao thức Zigbee giúp giảm chi phí xây dựng hệ thống nhà thông minh, dễ dàng triển khai trong môi trường gia đình.
- **Mở rộng linh hoạt:** Hệ thống có thể mở rộng thêm nhiều thiết bị mà không cần thay đổi cấu trúc ban đầu, đáp ứng yêu cầu tùy biến theo nhu cầu sử dụng.

– **Nhược điểm:**

- **Phụ thuộc vào mạng nội bộ:** Hệ thống có thể bị gián đoạn nếu mạng nội bộ gặp sự cố hoặc nếu các thiết bị Zigbee nằm ngoài phạm vi phủ sóng.
- **Giới hạn xử lý của Raspberry Pi:** Nếu hệ thống quản lý nhiều thiết bị hoặc cần xử lý lượng dữ liệu lớn, Raspberry Pi có thể không đáp ứng được hiệu suất tối đa, cần cân nhắc các giải pháp nâng cao phần cứng.
- **Khả năng lưu trữ lịch sử dữ liệu hạn chế:** Hiện tại, dữ liệu cảm biến chỉ được lưu trữ cục bộ và trong thời gian ngắn. Để theo dõi lịch sử dữ liệu lâu dài, cần bổ sung phương án lưu trữ trên đám mây.

– **Đề xuất cải tiến:**

- **Nâng cấp phần cứng:** Có thể sử dụng các phiên bản Raspberry Pi mạnh mẽ hơn hoặc tích hợp máy chủ mini để tăng hiệu suất và mở rộng khả năng xử lý khi kết nối với nhiều thiết bị.
- **Mở rộng phạm vi mạng Zigbee:** Triển khai thêm các thiết bị Zigbee repeater để tăng phạm vi phủ sóng, giúp các cảm biến xa cũng có thể kết nối ổn định với hệ thống.
- **Lưu trữ dữ liệu trên đám mây:** Tích hợp dịch vụ lưu trữ đám mây cho phép lưu trữ dữ liệu cảm biến lâu dài, giúp người dùng dễ dàng xem lại lịch sử dữ liệu qua thời gian và phân tích xu hướng.
- **Bổ sung tính năng thông báo:** Thiết lập thông báo qua email hoặc ứng dụng khi phát hiện dữ liệu cảm biến vượt ngưỡng (ví dụ: nhiệt độ cao bất thường), giúp người dùng nhận biết nhanh chóng và có phản ứng kịp thời.

- **Tăng cường bảo mật:** Để đảm bảo an toàn cho hệ thống, nên triển khai các biện pháp bảo mật cho giao diện web, như mã hóa dữ liệu truyền tải và quản lý quyền truy cập, tránh nguy cơ xâm nhập trái phép.

KẾT LUẬN

Hệ thống điều khiển và giám sát nhà thông minh đã hoàn thành đúng các chức năng đặt ra, từ điều khiển công tắc thiết bị cho đến hiển thị dữ liệu cảm biến trên giao diện web. Hiệu suất ổn định và khả năng đáp ứng nhu cầu điều khiển từ xa đã giúp hệ thống đạt được mục tiêu ban đầu. Với chi phí hợp lý, dễ sử dụng và khả năng mở rộng linh hoạt, hệ thống này có tiềm năng ứng dụng rộng rãi trong các hộ gia đình và doanh nghiệp nhỏ, mang lại tiện nghi và tăng cường khả năng quản lý năng lượng một cách hiệu quả. Thông qua việc kiểm soát và tự động hóa các thiết bị trong nhà, người dùng không chỉ có thể tiết kiệm thời gian mà còn nâng cao chất lượng cuộc sống, đồng thời góp phần vào xu hướng phát triển nhà thông minh và bảo vệ môi trường.

TÀI LIỆU THAM KHẢO

Koenkk. (2024). Create a CC2530 router. Retrieved from November 1st 2024: https://www.zigbee2mqtt.io/advanced/zigbee/05_create_a_cc2530_router.html#result