|  |  |
| --- | --- |
| Học trực tuyến CNTT, học lập trình từ cơ bản đến nâng cao |  |

**IOT304 – MẠNG TRUYỀN THÔNG KHÔNG DÂY**

**THUYẾT MINH ASSIGNMENT 3**

**XÂY DỰNG THIẾT BỊ HOẠT ĐỘNG VỚI TÍNH NĂNG HOÀN CHỈNH, TÍCH HỢP VÀO HỆ THỐNG LUMI LIFE**

|  |  |
| --- | --- |
| Tên học viên:  Mã học viên:  Ngày báo cáo: | Dương Nhật Huy  FX29455  13/4/2024 |

HÀ NỘI, 2/2024

MỤC LỤC

[**1. Danh sách thư viện** 1](#_Toc164824186)

[**2. Các sự kiện được khai báo** 2](#_Toc164824187)

[**3. Danh sách các hàm** 2](#_Toc164824188)

[**4 . Hàm emberAfMainInitCallback(void)** 3](#_Toc164824189)

[**5. Hàm mainStateEventHandler**() 3](#_Toc164824190)

[**6. Hàm** **buttonCallback**() 3](#_Toc164824191)

[**7. Hàm emberAfStackStatusCallback**(EmberStatus status) 4](#_Toc164824192)

[**8. Hàm emberAfPreCommandReceivedCallback**(EmberAfClusterCommand\* cmd) 4](#_Toc164824193)

[**9. Hàm lightSensorEventHandler()** 4](#_Toc164824194)

[**10. Lưu đồ thuật toán** 6](#_Toc164824195)

NỘI DUNG THUYẾT MINH

# **1. Danh sách thư viện**

|  |  |  |
| --- | --- | --- |
| STT | Thư viện | Mô tả |
| 1 | af.h | - Cung cấp struct EmberEventControl để khai báo các hàm sự kiện.  - Cung cấp các hàm xử lý sự kiện:  + emberEventControlSetActive : Hàm khởi động sự kiện.  + emberEventControlSetInActive : Hàm vô hiệu hóa sự kiện  + emberEventControlSetDelayMS : Hàm khởi động sự kiện sau một thời gian nhất định. |
| 2 | led.h | Thư viện cung cấp các hàm xử lý LED |
|  | send.h | Thư viện cung cấp các hàm gửi dữ liệu đến thiết bị khác  + SEND\_onoffAttributeResponse() Gửi trạng thái bật tắt của LED  + SEND\_modelIdAttributeResponse() Gửi Model ID |
|  | zigbee-device-common.h | Thư viện cung cấp hàm emberSendZigDevRequest() |
|  | Light\_sensor.h | Thư viện cung các hàm xử lý cảm biến ánh sáng |
|  | find-and-bind-target/find-and-bind-target.h | Thư viện cung cấp hàm emberAfPluginFindAndBindTargetStart(1) |
|  | find-and-bind-target/find-and-bind-Initiator.h | Thư viện cung cấp hàm emberAfPluginFindAndBindInitiatorStart() |
|  | SW\_ZB\_V1\_0\_0.h | Thư viện cung cấp Marco EMBER\_BINDING\_TABLE\_SIZE để xác định kích thước của Blind Table |

# **2. Các sự kiện được khai báo**

|  |  |  |
| --- | --- | --- |
| STT | Sự kiện | Mô tả |
| 1 | joinNetworkEventControl | Sự kiện tìm và gia nhập mạng |
| 2 | mainStateEventControl | Sự kiện xử lý các trạng thái của chương trình chính |
| 3 | lightSensorEventControl | Sự kiện đọc dữ liệu từ cảm biến ánh sáng và gửi cho HC cứ sau mỗi 1 phút |

# **3. Danh sách các hàm**

|  |  |  |
| --- | --- | --- |
| STT | Danh sách hàm | Mô tả |
| 1 | **void emberAfMainInitCallback(void)** | Hàm được thực hiện mỗi khi khởi động chương trình chính |
| 2 | **void mainStateEventHandler**() | Hàm xử lý các trạng thái của chương trình chính |
| 3 | **void buttonCallback**(  uint8\_t button,  uint8\_t state  ) | Hàm ngắt mỗi khi trạng thái nút bấm thay đổi |
| 4 | **void joinNetworkEventHandler**() | Hàm sự kiện tìm và gia nhập mạng |
| 5 | **void emberAfStackStatusCallback**(  EmberStatus status) | Hàm ngắt mỗi khi trạng thái chương trình thay đổi. Ở đây ta chỉ sử dụng để kiểm tra trạng thái mạng |
| 6 | **void emberAfPreCommandReceivedCallback**(EmberAfClusterCommand\* cmd) | Hàm ngắt mỗi khi nhận được mệnh lệnh. |
| 7 | **void**  **lightSensorEventHandler**() | Hàm sự kiện đọc dữ liệu từ cảm biến và gửi đến HC |

# **4 . Hàm emberAfMainInitCallback(void)**

*a. Mô tả*

Hàm này sẽ được gọi ra khi chương trình khởi chạy. Gọi các hàm sau đây:

- ledInit() : Cấu hình cho LED RGB

- initIADC() : Cấu hình IADC để giao tiếp với cảm biến ánh sáng

- buttonInit() : Hàm cấu hình nút nhấn

Sau khi gọi các hàm trên, khởi động sự kiện mainStateEventControl;

# **5. Hàm mainStateEventHandler**()

*a. Mô tả*

Hàm sự kiện xử lý các trạng thái của chương trình.

*-* POWER\_ON\_STATE*:* Chương trình vừa mới được khởi chạy. Nếu thiết bị chưa tồn tại mạng thì nháy LED 1 màu đỏ 3 lần và khởi động sự kiện joinNetworkEventControl để tìm mạng.

- NETWORK\_ON\_STATE *:* Thiết bị đã có mạng. Nháy LED 1 màu hồng 3 lần và gửi Model\_ID đến HC, rồi khởi động sự kiện lightSensorEventControl để tiến hành gửi giá trị cường độ ánh sáng theo chu kỳ cho HC.

- NETWORK\_LEAVE\_STATE : Trạng thái thiết bị rời mạng. Tiến hành rời mạng, xóa bảng binding table rồi chuyển sang trạng thái REBOOT\_STATE.

- REBOOT\_STATE: tiến hành reboot thiết bị.

# **6. Hàm** **buttonCallback**()

*a. Mô tả*

Hàm ngắt mỗi khi nhấn nút, có nhiệm vụ đếm số lần nhấn nút SW1 và SW2, rồi xử lý theo yêu cầu bài toán.

*b. Giá trị truyền vào*

uint8\_t button : PIN của nút bấm

uint8\_t state: Trạng thái nút bấm (Nhấn hoặc thả)

# **7. Hàm emberAfStackStatusCallback**(EmberStatus status)

*a. Mô tả*

Hàm ngắt mỗi khi trạng thái mạng thay đổi.

*b. Giá trị truyền vào*

EmberStatus status là trạng thái mạng

# **8. Hàm emberAfPreCommandReceivedCallback**(EmberAfClusterCommand\* cmd)

*a. Mô tả*

Hàm ngắt được gọi ra mỗi khi nhận lệnh. Có vai trò xử lý các lệnh nhận được

*b.Giá trị truyền vào*

EmberAfClusterCommand\* cmd con trỏ chỉ đến cấu trúc chứa các dữ liệu liên quan tới mệnh lệnh được gửi.

*c. Giá trị trả về*

Có kiểu dữ liệu boolean. Nếu giá trị trả về là False thì mệnh lệnh này sẽ tiếp tục được đưa xuống các tầng dưới để xử lý. Nếu giá trị là True thì kết thúc tại đây.

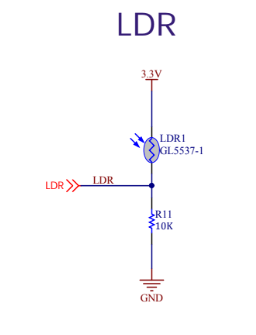
# **9. Hàm lightSensorEventHandler()**

*a. Mô tả*

- Sau mỗi một phút, hàm sự kiện này sẽ giao tiếp với cảm biến để đo cường độ ánh sáng.

Các bước đo cường độ ánh sáng:

+ Bước 1: Tính điện trở hiện tại của GL5537-1



RLDR = 10 x (4096 - VLDR) / VLDR

Trong đó VLDR là kết quả thu được từ IADC0 với độ phân giải 12bit.

RLDR là điện trở của GL5537-1(KΩ)

+ Bước 2:

Thiết lập công thức liên hệ giữa RLDR và Lux.

Tra datasheet GL5537-1, ta có:

Tại RLDR = 25 KΩ, Lux = 10

Lg(R10/R100) = 0.6 (Trong đó R10 và R100 là điện trở của cảm biến khi cường độ ánh sáng đạt 10 Lux và 100 Lux)

A graph of a line graph

Description automatically generated

Sau các bước tính toán, ta tìm ra công thức gần đúng:

Log(Lux) = - Log10(RLDR) \* 5 / 3 + 3.3299;

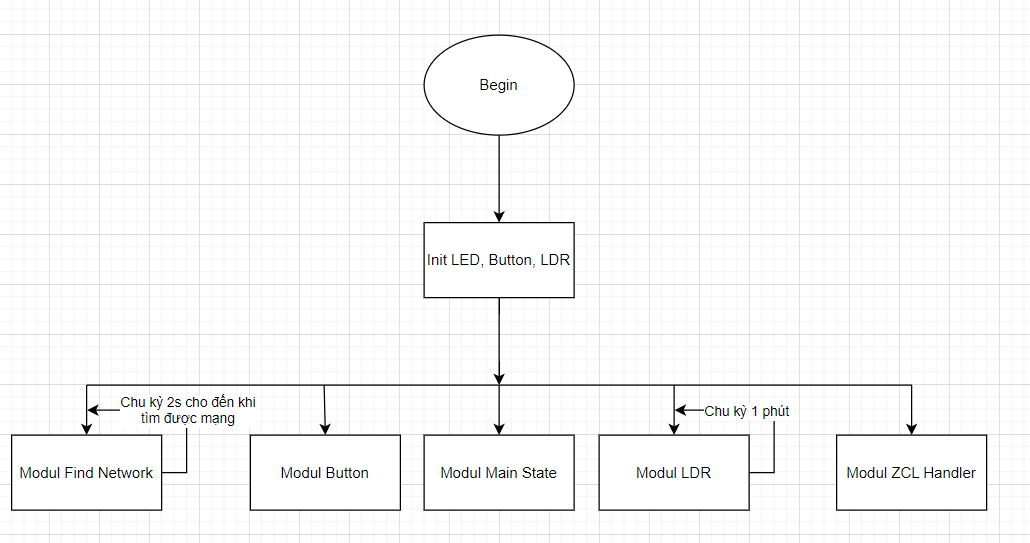
- Gửi giá trị cường độ ánh sáng lên HC nếu sự chênh lệch so với lần đo trước đó lớn hơn 30 Lux.

- Nếu cường độ ánh sáng ở thiết bị > 500 lux: LED 2 sáng xanh green (xử lý độc lập ở mỗi thiết bị).

- Nếu cường độ ánh sáng ở thiết bị < 500 lux: LED 2 tắt (xử lý độc lập ở mỗi thiết bị).

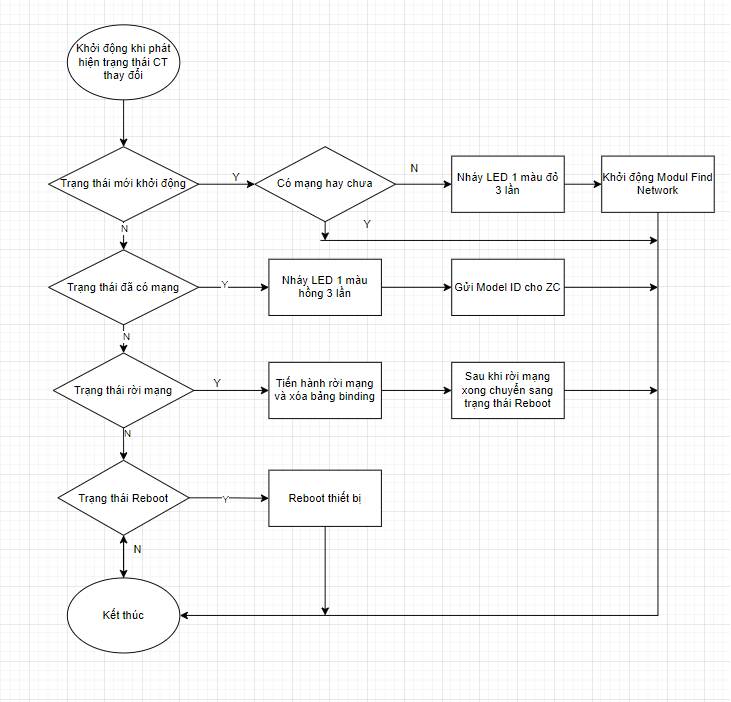
# **10. Lưu đồ thuật toán**

a) Sơ đồ tổng quan



b. Sơ đồ thuật toán cho từng modul

- Modul Main State

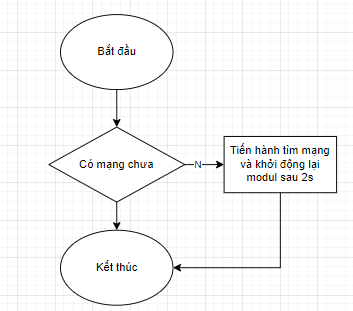


- Modul Button:

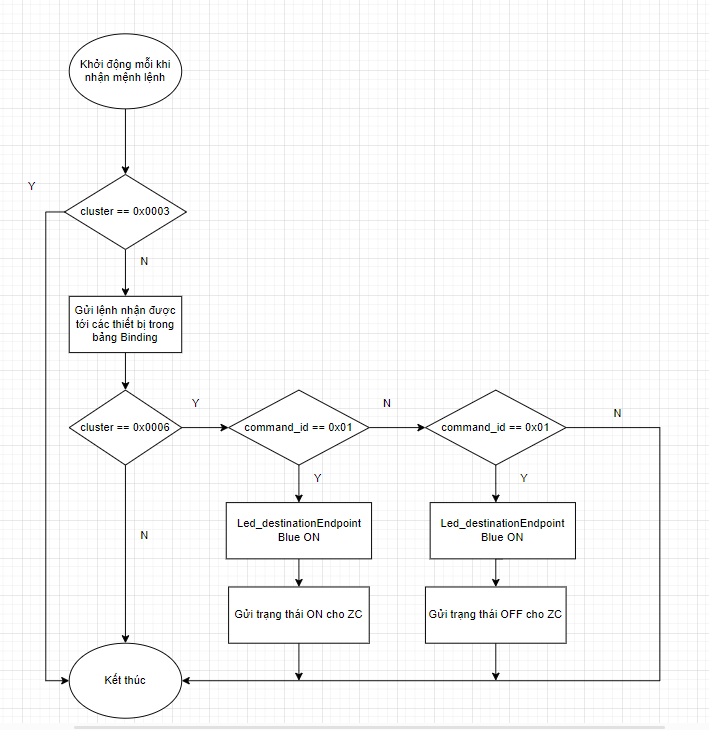
A diagram of a flowchart

Description automatically generated

- Modul Find Network



- Modul ZCL Handler



- Modul LDR

A diagram of a flowchart

Description automatically generated