

ĐẠI HỌC BÁCH KHOA HÀ NỘI
TRƯỜNG ĐIỆN – ĐIỆN TỬ



BÁO CÁO ĐỒ ÁN MÔN HỌC
MẠNG CẢM BIẾN KHÔNG DÂY

ĐỀ TÀI: MẠNG CẢM BIẾN ĐO NHIỆT ĐỘ MÔI TRƯỜNG

Sinh viên thực hiện:	NGUYỄN VĂN KHÔI	20202647
	BÙI VŨ DUY TRƯỜNG	20202543
	VŨ MẠNH HÙNG	20200265
Lớp học:	145444	

Giảng viên hướng dẫn: PGS. TS LÊ MINH THÙY

Hà Nội, 10-2023

MỞ ĐẦU

Sản phẩm này có mục đích phục vụ cho môn học Mạng cảm biến không dây, kỳ 20231. Mục tiêu của sản phẩm là có thể thu thập được thông số môi trường như là nhiệt độ, ở nhiều vị trí ở nhiều khoảng cách xa khác nhau. Trong báo cáo này sẽ trình bày chi tiết quá trình phát triển ý tưởng đến sản phẩm hoàn thiện cuối cùng. Các bước có thể kể tới như tìm hiểu các giao thức truyền tin, thiết kế mạch nguyên lý, PCB, lập trình cho thiết bị, kiểm tra và sửa lỗi.

Bản báo cáo này gồm 7 chương:

- **CHƯƠNG 1: Phân tích yêu cầu dự án**
- **CHƯƠNG 2: Kế hoạch thực hiện chi tiết**
- **CHƯƠNG 3: Phân chia công việc cụ thể cho từng thành viên**
- **CHƯƠNG 4: Tìm hiểu các công nghệ không dây**
- **CHƯƠNG 5: Mô tả chi tiết**
- **CHƯƠNG 6: Kết quả và đánh giá**
- **CHƯƠNG 7: Tài liệu tham khảo**

Trước hết chúng em xin gửi lời cảm ơn tới cô Lê Minh Thùy đã hướng dẫn về mặt kiến thức đã giúp chúng em hoàn thành báo cáo môn học này. Vì hiểu biết còn hạn hẹp và thời gian còn hạn chế nhóm em còn nhiều thiếu sót, nhóm mong nhận được sự góp ý, sửa đổi từ phía cô để báo cáo nhóm em hoàn thiện hơn.

Chúng em xin chân thành cảm ơn cô!

MỤC LỤC

MỞ ĐẦU	2
MỤC LỤC	3
DANH MỤC HÌNH ẢNH.....	5
DANH MỤC BẢNG.....	6
CHƯƠNG 1. PHÂN TÍCH YÊU CẦU DỰ ÁN	7
1.1. Mức độ đáp ứng yêu cầu.	8
1.1.1. Dải đo nhiệt độ.....	8
1.1.2. Thời lượng hoạt động	8
1.1.3. Kích thước và trọng lượng.....	9
1.1.4. Thời gian đo một mẫu	9
1.1.5. Kết nối máy tính.....	10
1.1.6. Quản lý thiết bị.....	10
1.1.7. Phần mềm máy tính.....	10
1.1.8. Nút bấm nhiệt độ bắt đầu đo.....	11
CHƯƠNG 2. KẾ HOẠCH THỰC HIỆN CHI TIẾT	12
CHƯƠNG 3. PHÂN CHIA CÔNG VIỆC CỤ THỂ CHO TỪNG THÀNH VIÊN...	13
3.1. Nguyễn Văn Khởi.....	13
3.2. Bùi Vũ Duy Trường.....	13
3.3. Vũ Mạnh Hùng.....	14
CHƯƠNG 4. TÌM HIỂU CÁC CÔNG NGHỆ KHÔNG DÂY	16
4.1. Tổng quan về công nghệ truyền thông không dây.....	16
4.1.1. Mô hình truyền thông không dây.....	16
4.1.2. Phân loại mạng không dây.....	16
4.2. WiFi	18
4.3. Zigbee	19
4.4. Bluetooth.....	19

4.4.1 Bluetooth Low Energy (BLE).....	20
4.4.2 Bluetooth Mesh	27
4.5 Lora	32
CHƯƠNG 5: MÔ TẢ CHI TIẾT.....	33
5.1. Kiến trúc chung của nút cảm biến.....	33
5.2. Năng lượng tiêu thụ của nút cảm biến	33
5.2.1 Các chế độ hoạt động của nút cảm biến.....	34
5.2.2 Công suất tiêu thụ của nút cảm biến không dây	35
5.3 Sơ đồ kiến trúc hệ thống và chức năng	37
5.4 Sơ đồ triển khai chi tiết từng khối.....	38
5.4.1 Sơ đồ triển khai chi tiết Sensor Node.....	38
5.4.2 Sơ đồ triển khai chi tiết Gateway.....	44
5.5 Lưu đồ hoạt động của chương trình	47
CHƯƠNG 6: KẾT QUẢ VÀ ĐÁNH GIÁ.....	48
6.1 Mạch trong thực tế	48
6.2 Phân tích và đánh giá.....	50
CHƯƠNG 7: TÀI LIỆU THAM KHẢO	52

DANH MỤC HÌNH ẢNH

Hình 1.1: Thời gian đổi ứng với độ phân giải của cảm biến LM35	9
Hình 1.2: Chi tiết thời gian lấy mẫu ở độ phân giải 12 bit	9
Hình 4.1: Mô hình truyền thông không dây	16
Hình 4.2: Các chuẩn tương ứng với các mạng truyền thông tương ứng.....	17
Hình 4.3: So sánh tốc độ truyền và khoảng cách truyền của Bluetooth, Zigbee, và Wifi	19
Hình 4.4: Mô hình Broadcasting và Observing trong BLE	21
Hình 4.5: Mô hình connection	22
Hình 4.6: BLE Protocols Stack.....	24
Hình 4.7: Tổ chức phân cấp trong cấu trúc dữ liệu BLE	27
Hình 4.8: Node trong Bluetooth Mesh.....	29
Hình 4.9: Element trong Bluetooth Mesh	29
Hình 4.10: Cơ chế Publish và Subscribe	31
Hình 5.1: Kiến trúc chung của nút cảm biến không dây.....	33
Hình 5.2: Công suất tiêu thụ trong các chế độ hoạt động	35
Hình 5.3: Sơ đồ khối mô tả hệ thống	37
Hình 5.4: Sơ đồ khối sạc pin	38
Hình 5.5: Sơ đồ nguyên lý khối nguồn LDO RT9013	39
Hình 6.6: Sơ đồ nguyên lý vi điều khiển NRF52832	39
Hình 5.7: Module NRF52832.....	41
Hình 5.8: Khối giám sát điện áp pin Lipo.....	42
Hình 5.9: Sơ đồ nguyên lý khối debug	42
Hình 5.10: Sơ đồ nguyên lý các nút nhấn và LED	43
Hình 5.11: Sơ đồ nguyên lý khối hiển thị LED 7 thanh	43
Hình 5.12: Sơ đồ nguyên lý khối nguồn LDO	44
Hình 5.13: Sơ đồ nguyên lý vi điều khiển ESP32	45
Hình 5.14: Sơ đồ nút nguyên lý các nút nhấn và LED	45
Hình 5.15: Sơ đồ nguyên lý khối debug và nạp code	46
Hình 5.16: Sơ đồ nguyên lý khối hỗ trợ nạp	46
Hình 5.17: Lưu đồ hoạt động của Gateway và Node.....	47
Hình 6.1: Mạch Node sau khi hoàn thiện.....	48
Hình 6.2: Mạch Node ở trạng thái pin được sạc đầy(Led trạng thái chuyển xanh)	48
Hình 6.3: Mạch Node treenn thực tế đang hiển thị nhiệt độ	49

Hình 6.4: Dữ liệu log ra từ GATEWAY	50
Hình 6.5: Màn hình hiển thị dữ liệu từ các node trên web ThingsBoard	50

DANH MỤC BẢNG

Bảng 1.1 Yêu cầu bài toán.....	8
Bảng 2.1 Phân chia công việc.....	12
Bảng 3.1: Phân chia công việc Nguyễn Văn Khởi	13
Bảng 3.2: Phân chia công việc Bùi Vũ Duy Trường	14
Bảng 3.3: Phân chia công việc Vũ Mạnh Hùng	15
Bảng 5.1: các chế độ của thiết bị	34
Bảng 5.2: Công suất tiêu thụ của từng phần tử	35

CHƯƠNG 1. PHÂN TÍCH YÊU CẦU DỰ ÁN

TT	Yêu cầu	Nội dung thực hiện	Mức độ ưu tiên	Kịch bản đánh giá
1	- Dải đo: $25^{\circ}\text{C} \div 150^{\circ}\text{C}$ - Độ chính xác: 1°C - Độ phân giải hiển thị: 0.1°C	Đo nhiệt độ môi trường Hiển thị nhiệt độ bằng LED 7 thanh	Đáp ứng được yêu cầu của dự án Mức 1	-Dải đo: nhiệt độ cần phải đo là được đo ở nhiều nơi khác nhau: trong nhà, ngoài trời. -Độ chính xác: So sánh nhiệt độ của 2 hay nhiều Node khi đo ở cùng 1 địa điểm -Độ phân giải hiển thị: kiểm tra hiển thị độ phân giải trên các Node
2	Nguồn pin, thời gian hoạt động của thiết bị là 4h (nâng cao: 8h). Pin có thể sạc trực tiếp trên máy hoặc tháo ra ngoài	Sử dụng pin Li-Po công suất khoảng 2100mAh cùng với mạch bảo vệ đi kèm	Đáp ứng được yêu cầu của dự án Mức 1	-Thử nghiệm thời gian sử dụng thiết bị trong 4 tiếng, trong 8 tiếng, thời gian chạy tối đa của thiết bị. - Thử nghiệm việc sạc pin trực tiếp khi thiết bị đang hoạt động, sạc đầy khi đạt mức pin tối thiểu trong bao nhiêu giờ.
3	- Kích thước (dự kiến): - Trọng lượng (dự kiến):	Kích thước(dự kiến): $70 \times 50 \times 100$ mm (kiểu trụ để dễ thao tác) Trọng lượng < (150g)	Đáp ứng được yêu cầu của dự án Mức 2	
4	- Thời gian đo một mẫu : <20s. (nâng cao < 5s)		Đáp ứng được yêu cầu của dự án Mức 1	-Kiểm tra thời gian đo một mẫu là bao nhiêu -Gửi về Gateway bằng Bluetooth Mesh với thời gian có đúng theo mức dự định là khoảng 5s cho từng Node hay không, bản tin nhận được có lỗi hay không.
5	- Kết nối máy tính: RF Khoảng cách truyền trong phạm vi 20m từ hệ thống đo đến trạm thu RF có nối nguồn và mạng	Thiết bị có thể giao tiếp với máy tính thông qua cổng USB bằng module chuyển đổi COM-TTL	Đáp ứng được yêu cầu của dự án Mức 2	-Kiểm tra xem dữ liệu từ Gateway có thể truyền ra được laptop hay không. -Thử nghiệm truyền thông: +Thử nghiệm với khoảng cách 20m +Thử nghiệm với vật cản thì khoảng cách tối đa là bao nhiêu +Tăng khoảng cách để đo tối đa
6	Quản lý tối thiểu cho 11 thiết bị đo. (nâng cao: phương án mở rộng số thiết bị với khoảng cách từ thiết bị đến trạm tiếp	Kiến trúc mạng dự kiến là hình lưới Tối thiểu 10 thiết bị	Đáp ứng được yêu cầu của dự án Mức 1	-Tăng thêm 2 Node vào mạng và xem hoạt động trong Mesh hay không. -Đo khoảng cách tối đa giữa các Node và giữa Node và Gateway sau khi thêm các Node mới.

	nhận có thể lên tới 100m)	Thiết bị gỡ ra hoặc thêm vào có thể được phát hiện		
7	- Phần mềm máy tính: thu thập giá trị đo từ thiết bị đo, quản lý dữ liệu, xuất báo cáo dạng excel, giao diện theo mẫu thống nhất.	Webserver riêng	Đáp ứng được yêu cầu của dự án Mức 2	-Với website Thingboard: thu thập giá trị từ toàn bộ Node gửi lên website. -Xuất file excel báo cáo dữ liệu. -Hiển thị dữ liệu thời gian thực, có kèm đồ thị.
8	Có nút bấm bắt đầu đo; Đèn LED báo ngưỡng nhiệt độ (3 LED); Các ngưỡng nhiệt độ có thể cập nhật từ máy tính	Có đèn LED báo ngưỡng nhiệt độ Tích hợp các nút bấm	Đáp ứng được yêu cầu của dự án Mức 1	-Thử nghiệm nút bấm Reset của Node. -Thử nghiệm 3 nút bấm cảnh báo và hiển thị lên Led 7 thanh có hoạt động đúng hay không. -Ngưỡng đo nhiệt độ có thể thay đổi từ laptop qua cổng UART to TTL được kết nối với Gateway và truyền xuống từng Node
9	OTA (nâng cao)	Tích hợp chức năng OTA	Đáp ứng được yêu cầu của dự án Mức 3	-Kiểm tra tính năng OTA có trên Gateway đã hoạt động được chưa.

Bảng 1.1 Yêu cầu bài toán

1.1. Mức độ đáp ứng yêu cầu.

1.1.1. Dải đo nhiệt độ

Sau khi tìm kiếm và tổng hợp các loại cảm biến, nhóm đã tiến hành lựa chọn DS18B20 với các thông số như sau:

- Dải đo: -55C tới 130C
- Độ chính xác: +-0.5C

Như vậy, nhóm đã đáp ứng được yêu cầu 1.

1.1.2. Thời lượng hoạt động

Sau khi tính toán ở phần 6.2.2 nhóm đã tính toán được thời gian hoạt động trên lý thuyết tối đa của thiết bị là 16 ngày.

Và sau qua trình đo thực nghiệm để cho Node hoạt động liên tục, thì sau khoảng 12 ngày Node vẫn hoạt động ổn định với mức điện áp sau 12 ngày là 3.92V từ điện áp cao nhất là 4.2V.

Do đó nhóm đã đáp ứng yêu cầu đề bài đưa ra.

1.1.3. Kích thước và trọng lượng

Kích thước của gateway:

Trọng lượng của gateway:

Kích thước của node: 72x56x30(mm)

Trọng lượng của node: 150g

Như vậy, nhóm đã đáp ứng được yêu cầu 3.

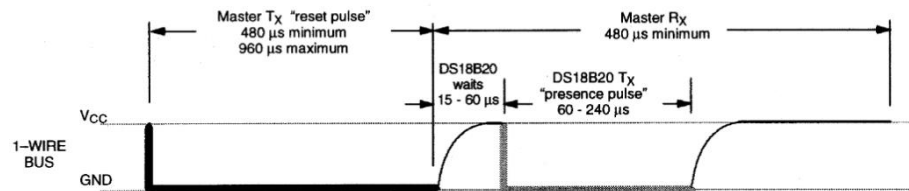
1.1.4. Thời gian đo một mẫu

Cảm biến DS18B20 được cấu hình ở độ phân giải 12bit.

R1	R0	Thermometer Resolution	Max Conversion Time
0	0	9 bit	93.75 ms ($t_{conv}/8$)
0	1	10 bit	187.5 ms ($t_{conv}/4$)
1	0	11 bit	375 ms ($t_{conv}/2$)
1	1	12 bit	750 ms (t_{conv})

Hình 1.1: Thời gian đổi ứng với độ phân giải của cảm biến LM35

Đầu tiên, host sẽ kéo đường Tx xuống trong khoảng thời gian 480us-960us, cảm biến LM35 sẽ đợi trong khoảng từ 15-60us rồi bắt đầu gửi lại 1 xung trong khoảng tối đa 240us. Trong khoảng thời gian đó, host sẽ chỉ đợi tối đa là 480us.



Hình 1.2: Chi tiết thời gian lấy mẫu ở độ phân giải 12 bit

Với cấu hình 12bit ở trên, một lần lấy mẫu cảm biến sẽ mất khoảng 750ms.

Tổng cộng thời gian để hoàn thành 1 lần đọc cảm biến sẽ là 750ms+960us+60us+480us và sẽ là 752ms.

Như vậy nhóm đã đáp ứng yêu cầu 4.

1.1.5. Kết nối máy tính

Thiết bị Gateway có khả năng kết nối máy tính thông qua chuẩn chuyển đổi COM-TTL để có thể tiện cho việc sao lưu lại dữ liệu trong trường hợp việc kết nối với server gặp trục trặc.

Nhóm đã thiết kế lưu lại dữ liệu vào một ổ đĩa riêng để có thể log ra mỗi khi cần.

Như vậy nhóm đã đáp ứng yêu cầu 5.

1.1.6. Quản lý thiết bị

Do sử dụng truyền thông không dây Bluetooth để truyền nhận dữ liệu giữa Gateway và Sensor Node, nhóm em sẽ sử dụng cấu trúc mạng hình lưới, số lượng các Sensor Node cần quản lý tối thiểu là 10. Các Sensor Node sẽ thực hiện giao tiếp trực tiếp với Gateway.

Các Sensor Node sẽ được cấp phát cho một địa chỉ nhờ vào quá trình provision trên mạng Bluetooth Mesh. Chúng em sử dụng cơ chế public và subscribe của mô hình Bluetooth Mesh để đẩy dữ liệu của các Sensor Node lên mạng. Ở đây các Sensor Node sẽ public properties của chính nó lên một địa chỉ chung hay còn gọi là Group Address, sau đó Gateway sẽ subscribe vào Group Address này và lấy tất cả các thông tin cần thiết của từng Sensor Node về.

Để có thể thêm một hay nhiều Sensor Node vào mạng thì hiện tại chúng em sẽ config cứng bằng phần mềm điện thoại do hãng cung cấp, và cần gỡ Sensor Node ra khỏi mạng cũng có thể thao tác trên này. Về việc có thể phát hiện Node còn hoạt động hay không sẽ nhờ vào phần RSSI mà Gateway nhận được từ Node, nếu nó ở khoảng -100 tức là Node này đã bị shutdown tức là không còn hoạt động, vì nếu nó còn hoạt động thì chắc chắn chỉ số RSSI này sẽ là tốt nhất là -30 -40. Và thời gian để xác định Node này bị shutdown hay không phụ thuộc vào thời gian mà một gói bản tin gửi đến là khoảng 5s. Hoặc với một bản tin Heartbeat truyền về Gateway cũng có thể xác định được Node đó có còn hoạt động trong Mesh hay không. Với thời gian để cho phép vào mạng là khoảng 1 phút bao gồm quá trình quét thiết bị và cho phép vào mạng.

Hiện tại đã phát triển Gateway làm 1 provisioner với tính năng tương tự như trên để có thể quản lý mạng thuận tiện hơn.

1.1.7. Phần mềm máy tính

Nhóm đã thiết kế một Webserver riêng và một nền tảng bên thứ 3 là Thingsboard để phục vụ cho việc hiển thị các thông số thông qua giao thức MQTTs.

Các dữ liệu sẽ được đẩy lên MQTTs bằng chuẩn cJSON, từ đó làm database cho webserver cũng như là Thingsboard.

Giao diện Thingsboard được thiết kế với các template mẫu, phù hợp cho việc truyền tải các dữ liệu về nhiệt độ.

Thiết bị Gateway sẽ được thiết kế cổng kết nối MicroUSB và có thể kết nối với cổng USB – COM của máy tính. Sau khi kết nối dây thành công, chọn cổng kết nối, người dùng có thể theo dõi thông tin thời gian thực được hiển thị trên phần mềm máy tính. Dữ liệu được hiển thị theo dạng đồ thị của từng Node có mặt và đang hoạt động trong mạng lưới. Đồ thị dữ liệu sẽ được cập nhật từng giây.

Với những mô tả trên, yêu cầu 7 đã được bọn em đáp ứng.

1.1.8. Nút bấm nhiệt độ bắt đầu đo

Trên Sensor Node có 3 nút nhấn để bắt đầu/dừng đo, hiển thị điện áp pin và 4 LED 7 thanh để hiển thị các thông số cần thiết.

Như vậy yêu cầu 8 đã được đáp ứng.

CHƯƠNG 2. KẾ HOẠCH THỰC HIỆN CHI TIẾT

Nội dung	Kết quả đạt được	Thời gian(tuần)	Ghi chú thêm
Tìm hiểu các bài toán, mô hình có liên quan	Tìm hiểu được các mô hình hiện có, tham khảo các dự án có chung mô hình.	T1 đến T4	
Tổng hợp thông tin, chốt phương án sơ bộ	Tổng hợp thông tin cả nhóm. Hoàn thiện sơ bộ các đầu mục nội dung	T4 tới tuần T5	
Thiết kế phần cứng	Sau khi tìm các linh kiện có sẵn trong nước, lên phương án thiết kế chi tiết. Thiết kế mạch nguyên lí và bo mạch in.	T5 tới T7	
Thiết kế phần mềm	Thử nghiệm lại kết nối của các thiết bị trong mạng, kiểm tra lại độ hiệu quả của mô hình mạng.	T7 tới T11	
Thử nghiệm sản phẩm	Đưa ra các kịch bản khác nhau, từ đó đánh giá được thông số cụ thể của mạng đối với từng kịch bản.	T12 tới T14	
Hoàn thiện hệ thống	Hoàn thiện vỏ hộp và đưa giá đánh giá tổng quan chung	T15 tới T16	

Bảng 2.1 Phân chia công việc

CHƯƠNG 3. PHÂN CHIA CÔNG VIỆC CỤ THỂ CHO TỪNG THÀNH VIÊN

3.1. Nguyễn Văn Khởi

Nội dung	Kết quả đạt được	Thời gian(tuần)	Đánh giá
Tìm hiểu các bài toán, mô hình có liên quan	Tham khảo các dự án hiện có. Phân tích ưu nhược điểm của từng dự án.	T1 tới T4	10/10
Tổng hợp thông tin, chốt phương án sơ bộ	Đề xuất được mô hình mạng cần triển khai là mô hình Bluetooth Mesh. Chốt được tổng quan các đầu mục cần thực hiện.	T4 tới T5	10/10
Thiết kế phần cứng	Thực hiện triển khai mạch nguyên lí và mạch in. Mua linh kiện và lắp ráp thành bo mạch hoàn chỉnh. Đã thiết kế lại phiên bản mới cho mạch nguyên lí.	T5 tới T7	10/10
Thiết kế phần mềm	Tìm hiểu cấu trúc mạng của giao thức, kết hợp với nhóm để đưa ra hướng giải quyết chung. Thiết kế phần mềm hiển thị trên máy tính và trên server.	T7 tới T11	10/10
Thử nghiệm sản phẩm và hoàn thiện hệ thống	Đi kiểm nghiệm thực tế tại các môi trường và địa điểm khác nhau để có được thông số cụ thể cho từng loại kịch bản đã đề ra.	T12 tới T16	10/10
Tổng	Hoàn thành nhiệm vụ đề ra		10/10 (đ)

Bảng 3.1: Phân chia công việc Nguyễn Văn Khởi

3.2. Bùi Vũ Duy Trường

Nội dung	Kết quả đạt được	Thời gian(tuần)	Đánh giá
Tìm hiểu các bài toán, mô hình có liên quan	Tham khảo các dự án hiện có. Phân tích ưu nhược điểm của từng dự án.	T1 tới T4	10/10
Tổng hợp thông tin, chốt phương án sơ bộ	Tìm hiểu về mô hình mạng đã được chốt	T4 tới T5	10/10
Thiết kế phần cứng	Đề xuất linh kiện, check lại mạch PCB	T5 tới T6	10/10
Thiết kế phần mềm	Tìm hiểu và sử dụng SDK cho vi điều khiển đã chốt, Lập trình cho Node Sensor và Gateway về phần Bluetooth Mesh	T7 tới T14	10/10
Thử nghiệm sản phẩm	Đưa ra các kịch bản khác nhau, từ đó đánh giá được thông số cụ thể của mạng đối với từng kịch bản.	T12 tới T14	10/10
Hoàn thiện hệ thống	Đi kiểm nghiệm thực tế tại các môi trường và địa điểm khác nhau để có được thông số cụ thể cho từng loại kịch bản đã đề ra.	T12 tới T16	10/10
Tổng	Hoàn thành nhiệm vụ đề ra		10/10 (đ)

Bảng 3.2: Phân chia công việc Bùi Vũ Duy Trường

3.3. Vũ Mạnh Hùng

Nội dung	Kết quả đạt được	Thời gian(tuần)	Ghi chú thêm
Tìm hiểu các bài toán, mô hình có liên quan	Tham khảo các dự án	T1-t4	9/10
Tổng hợp thông tin, chốt phương án sơ bộ	đề xuất và tham khảo các phương án mô hình mạng	T4-t5	9/10
Thiết kế phần cứng	Tìm hiểu phương án thiết kế mạch nguyên lý	T6	8/10
Thiết kế phần mềm			0/10

Thử nghiệm sản phẩm			0/10
Hoàn thiện hệ thống			0/10
Tổng	Chưa hoàn thành nhiệm vụ đề ra		8/10 (đ)

Bảng 3.3: Phân chia công việc Vũ Mạnh Hùng

CHƯƠNG 4. TÌM HIỂU CÁC CÔNG NGHỆ KHÔNG DÂY

Mạng truyền thông được phân loại như sau:

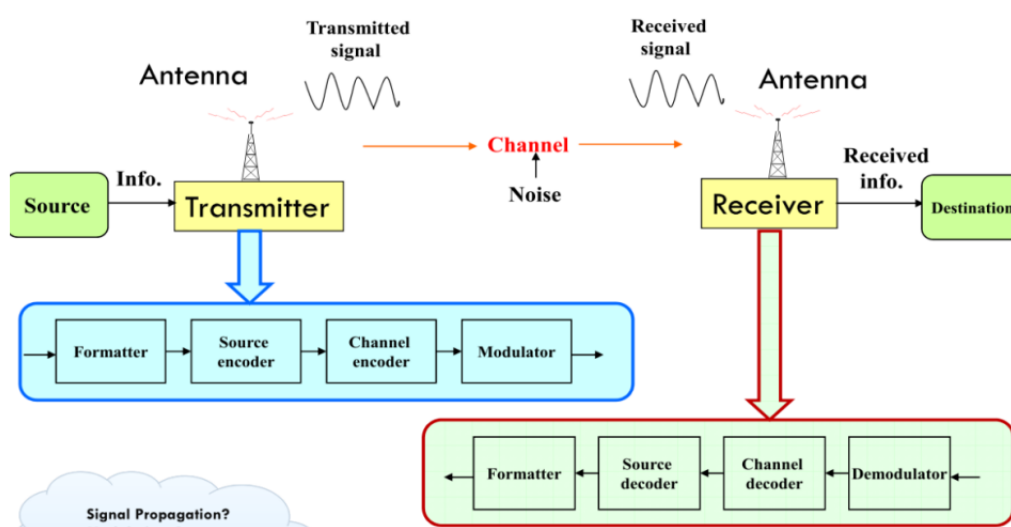
- Dựa vào dạng tín hiệu thông tin (Information Signals): Digital, Analog
 - Dựa vào phạm vi (Scale): LAN, WAN, MAN, Internet
- Dựa vào công nghệ truyền dẫn (Transmission Technology): Broadcast, Point –to–Point
- Dựa vào phương tiện truyền dẫn (Transmission Medium): Wired, Wireless

4.1. Tổng quan về công nghệ truyền thông không dây

Wireless: vô tuyến, không dùng dây dẫn

Truyền thông không dây là việc truyền tải thông tin qua một khoảng cách mà không cần dây dẫn làm môi trường truyền.

4.1.1. Mô hình truyền thông không dây



Hình 4.1: Mô hình truyền thông không dây

4.1.2 Phân loại mạng không dây

Mạng không dây được phân loại dựa theo:

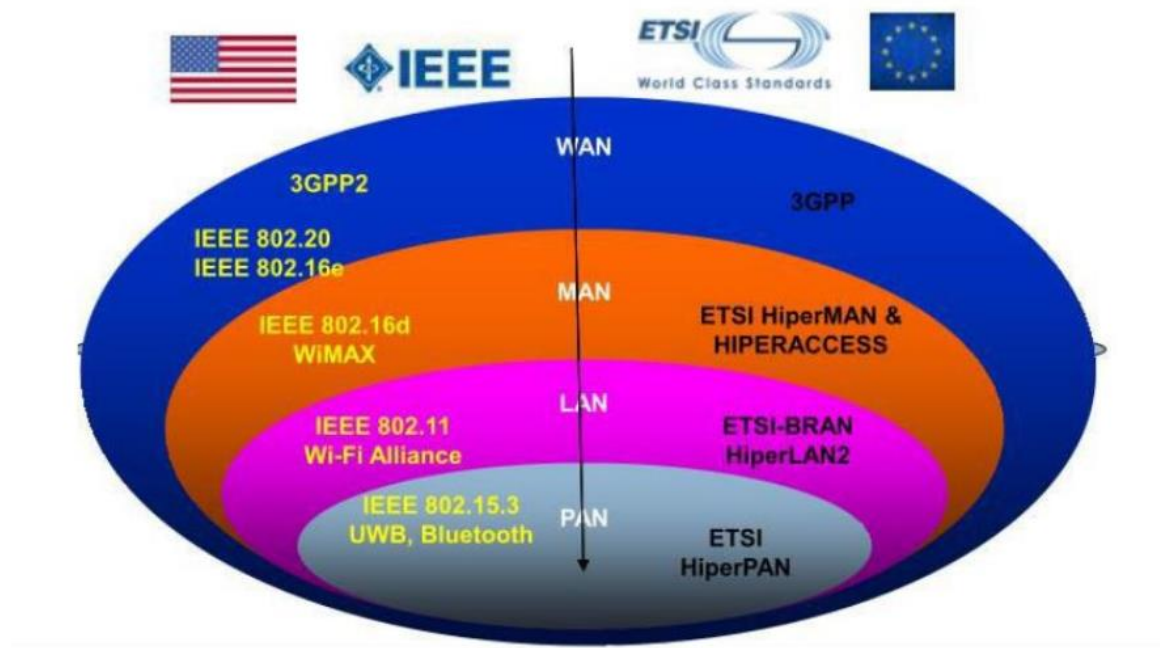
(1) Dựa vào kiến trúc (Infrastructure)

- Mạng có kiến trúc:

- + Sử dụng các node và gateway có dây và không dây
- + Mỗi node chính được định nghĩa trước vai trò trong mạng
- + Mạng có kiến trúc: Mạng tế bào, WLAN dùng access point, ...
- Mạng không có kiến trúc (Ad – hoc network)
 - + Không có sự sắp xếp trước
 - + Tập hợp ngẫu nhiên các node lại với nhau tạo thành một mạng

(2) Dựa vào chuẩn (Standardizations)

- 3GPP: WCDMA, HSDPA, 4G LTE
- 3GPP2: CDMA2000, EV – DO, UMB
- IEEE: IEEE802.11, IEEE802.16.



Hình 4.2: Các chuẩn tương ứng với các mạng truyền thông tương ứng

(3) Dựa vào phạm vi (Scale)

- Body Area Networks: như cảm biến được gắn vào người
- Personal Area Networks: Home networking, Bluetooth và ZigBee

- Local Area Networks: Wireless LAN, WiFi
- Metropolitan Area Networks (Mạng đô thị): WiMAX
- Wide Area Networks: Cellula và Satellite networks

(4) Dựa vào sự di động (Mobility)

- Mạng không dây cố định: người dùng đã kết nối được xem như ở yên một chỗ.
- Mạng không dây di động: một phần trong các thiết bị không dây là di động
 - + Việc thay đổi điểm kết nối vào mạng có thể được thực hiện khi thiết bị vẫn online.
 - + VD: Cellula network, mobile ad hoc network (MANET).

(5) Dựa vào dạng tín hiệu (Signal)

(6) Dựa vào băng thông (Bandwidth)

- Narrowband Wireless Network: Mạng băng hẹp.
- Broadband Wireless Network: Mạng băng rộng (UWB, WiMax, LTE,..)

Các tần số tiêu biểu: Bluetooth ~ 2.4 GHz hay WiFi ~ 2.4 GHz

4.2 WiFi

Wi-Fi là viết tắt của Wireless Fidelity. Đây là công nghệ kết nối mạng cục bộ không dây với các thiết bị dựa trên tiêu chuẩn IEEE 802.11. Các thiết bị tương thích với Wi-Fi có thể kết nối với Internet qua mạng WLAN và điểm truy cập không dây viết tắt là AP (Access Point). Mọi mạng WLAN đều có một điểm truy cập chịu trách nhiệm nhận và truyền dữ liệu từ / tới người dùng. IEEE đã xác định các thông số kỹ thuật nhất định cho mạng LAN không dây, được gọi là IEEE 802.11 bao gồm các lớp liên kết vật lý và dữ liệu.

Một số điểm quan trọng về Wi-Fi:

- (1) Nó yêu cầu bộ điều hợp không dây (Wireless Adapter) trên tất cả các thiết bị và bộ định tuyến không dây (Wireless Router) để kết nối với nó.
- (2) Tiêu thụ năng lượng lớn
- (3) Cung cấp bảo mật tốt
- (4) Hỗ trợ lượng lớn người dùng
- (5) Phạm vi tín hiệu ~ 100m
- (6) Yêu cầu băng thông cao

4.3 Zigbee

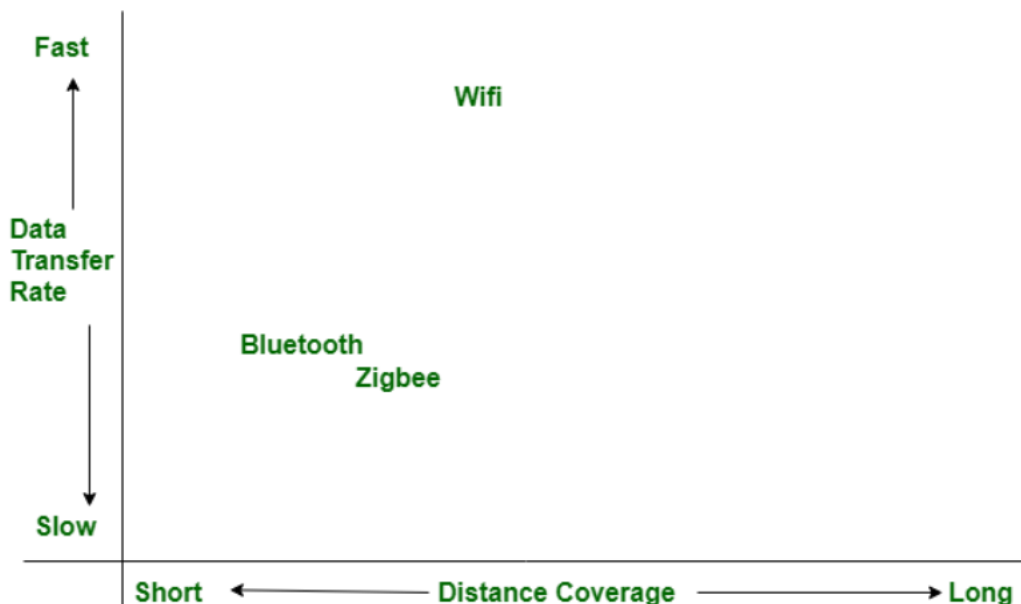
Zigbee giống như UWB (Băng tần siêu rộng). dải tần được hỗ trợ trong Zigbee hầu hết là 2,4 GHz trên toàn thế giới, có nghĩa là 2,4 GHz không được hỗ trợ mọi lúc. Nó bao gồm nhiều khoảng cách hơn so với Bluetooth. Có 16 kênh RF trong Zigbee.

Một số điểm quan trọng về Zigbee:

- (1) Phát triển theo chuẩn IEEE 802.15.4
- (2) Dải tần hỗ trợ ~ 2,4 GHz
- (3) Kỹ thuật điều chế BPSK và QPSK như UWB
- (4) Hơn 65.000 node trong Zigbee
- (5) 16 kênh RF
- (6) Zigbee cũng yêu cầu băng thông thấp nhưng cao hơn băng thông của Bluetooth.
- (7) Phạm vi tín hiệu ~ 10 – 100m.

4.4 Bluetooth

Bluetooth được phát triển theo chuẩn IEEE 802.15.1, được sử dụng để cung cấp giao tiếp không dây thông qua tín hiệu vô tuyến. Dải tần số được hỗ trợ trong Bluetooth thay đổi từ 2,4 GHz đến 2,483 GHz. Phạm vi tín hiệu của Bluetooth ngắn hơn Zigbee. Trong Bluetooth, kỹ thuật điều chế GFSK được sử dụng.



Hình 4.3: So sánh tốc độ truyền và khoảng cách truyền của Bluetooth, Zigbee, và Wifi

Cả Bluetooth và Zigbee đều có nhiều điểm chung, đó là mỗi kiểu đơn vị khu vực của IEEE 802.15 WPANs. mỗi loại đều chạy trong cặp băng tần không được cấp phép 4 GHz và mỗi loại đều tiêu thụ công suất thấp.

4.4.1 Bluetooth Low Energy (BLE)

Bluetooth Low Energy (BLE) là một chuẩn kết nối không dây tiết kiệm năng lượng, được phát triển bởi Bluetooth SIG (Special Interest Group), một tổ chức chuyên về việc phát triển các công nghệ không dây. BLE đã được giới thiệu lần đầu tiên vào năm 2010 và nhanh chóng trở thành một trong những chuẩn kết nối không dây phổ biến nhất trên thế giới.

BLE được thiết kế để hoạt động trong các ứng dụng IoT, như các thiết bị y tế thông minh, thiết bị đeo tay, thiết bị định vị GPS và các thiết bị gia đình thông minh. Với khả năng tiết kiệm năng lượng, BLE cho phép các thiết bị hoạt động trong thời gian dài hơn, giúp tiết kiệm pin và tăng tuổi thọ của pin.

NGUYÊN LÝ HOẠT ĐỘNG CỦA BLE:

- BLE hoạt động bằng cách sử dụng tần số 2,4 GHz để truyền thông tin giữa các thiết bị. Nó sử dụng một giao thức kết nối hai bước để thiết lập kết nối giữa hai thiết bị BLE: Giao thức kết nối và Giao thức giao tiếp.
- Giao thức kết nối là quá trình mà hai thiết bị BLE trao đổi các thông tin để thiết lập kết nối giữa chúng. Giao thức này bao gồm hai giai đoạn: quảng bá và quá trình kết nối.
- Trong giai đoạn quảng bá, thiết bị BLE sẽ gửi các thông điệp quảng bá đến các thiết bị khác trong vùng phủ sóng của nó. Các thông điệp này chứa các thông tin cơ bản về thiết bị như tên, địa chỉ MAC và loại thiết bị. Các thiết bị khác trong vùng phủ sóng có thể nhận thông điệp này và quyết định liệu chúng có muốn kết nối với thiết bị gửi thông điệp hay không.
- Trong giai đoạn kết nối, khi một thiết bị nhận được yêu cầu kết nối từ một thiết bị khác, nó sẽ phản hồi bằng một thông điệp kết nối và thiết lập kết nối giữa hai thiết bị. Sau khi kết nối được thiết lập, hai thiết bị sẽ chuyển sang giai đoạn giao tiếp.
- Giao thức giao tiếp là quá trình mà hai thiết bị BLE trao đổi các thông tin dữ liệu. Nó sử dụng một giao thức chuyển giao dữ liệu hai chiều, cho phép hai thiết bị gửi và nhận dữ liệu đồng thời. Các thông điệp dữ liệu này có thể chứa các thông tin về trạng thái của thiết bị, dữ liệu cảm biến, âm thanh hoặc hình ảnh.

Các giới hạn chính của BLE:

- Thông lượng dữ liệu nhỏ

Tần số điều chế của sóng BLE trong không gian là 1Mbps. Đây là giới hạn trên của thông lượng theo lý thuyết. Tuy nhiên trong thực tế tham số này nhỏ hơn do ảnh hưởng của nhiều yếu tố.

Để minh họa ta đặt giả thiết là một thiết bị trung tâm (Master) được khởi tạo và thiết lập kết nối đến một ngoại vi (Slave) qua giao diện BLE:

- Ta có khái niệm về chu kỳ kết nối (Conneciton interval), đây là khoảng thời gian giữa 2 sự kiện kết nối liên tiếp. Với BLE, khi một sự kiện kết nối diễn ra, các thiết bị trong kết nối sẽ trao đổi dữ liệu với nhau, sau đó trở về trạng thái IDLE để tiết kiệm năng lượng, và chờ đến thời điểm thì thực hiện sự kiện kết nối tiếp theo. Tham số này nằm trong khoảng 7.5ms đến 4s.
- nRF51822 có thể truyền đến 6 gói dữ liệu trong mỗi sự kiện kết nối. Mỗi gói có thể chứa 20 bytes dữ liệu của người dùng.
- Giả sử tần số sự kiện kết nối là lớn nhất (chu kỳ kết nối nhỏ nhất = 7.5ms). Khi đó mỗi giây có thể xảy ra tối đa 133 sự kiện kết nối

➔ Công thức tính thông lượng: $133 * 120 = 15960 \text{ bytes/s}$ (125Kbit/s)

Theo khuyến cáo thì ta nên lựa chọn thông lượng khoảng 80 Kbit/s. Điều này cũng nói lên vì sao Wifi, classic bluetooth vẫn có chỗ đứng.

- Khoảng cách gần

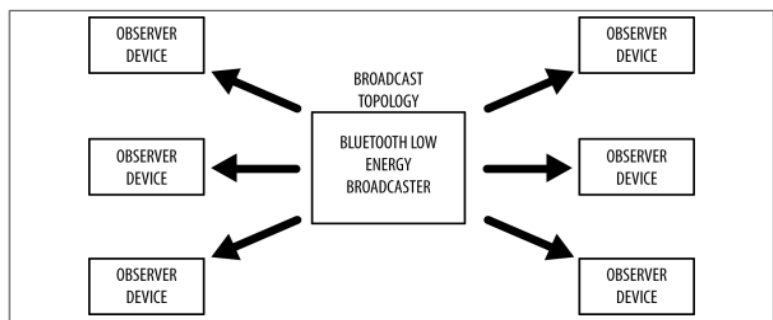
Các yếu tố ảnh hưởng đến khoảng cách truyền thông như môi trường hoạt động, thiết kế anten, vật cản, hướng thiết bị,... BLE tập trung vào các ứng dụng truyền thông trong phạm vi gần.

- Với BLE ta có:
 - Khoảng cách lý thuyết: 100m (điều kiện tốt).
 - Khoảng cách khả thi: 30m.
 - Khoảng cách thường được sử dụng: 2-5m.

MÔ HÌNH MẠNG TRUYỀN THÔNG CHO BLE:

Một thiết bị BLE có thể giao tiếp với bên ngoài thông qua 2 cơ chế: Broadcasting hoặc Connection. Mỗi cơ chế có thể mạnh và giới hạn riêng, cả hai được thiết lập bởi GAP (Generic Access Profile).

Broadcasting và Observing

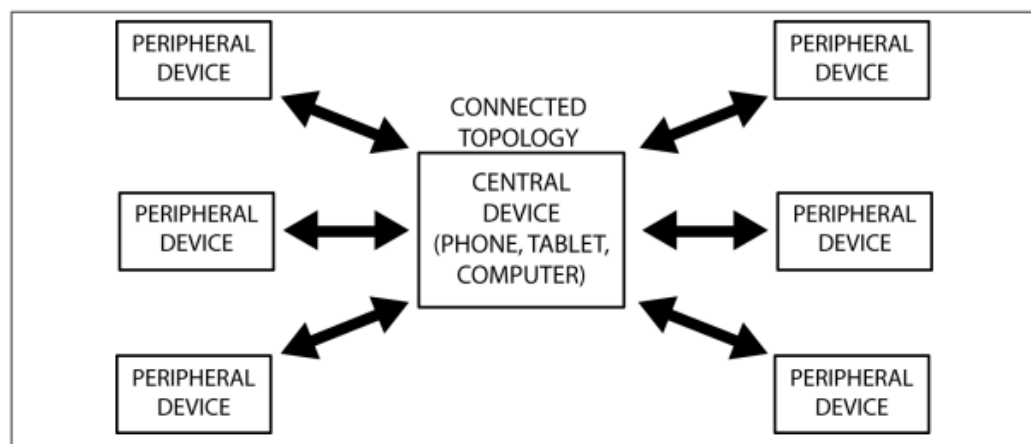


Hình 4.4: Mô hình Broadcasting và Observing trong BLE

- Thiết bị Broadcaster: Gửi các gói tin quảng bá phi kết nối đến bất kỳ thiết bị nào có thể nhận.
- Thiết bị Observer: Quét liên tục theo tần số đặt trước để nhận các gói tin quảng bá phi kết nối

Đây là kiểu truyền thông cho phép một thiết bị có thể truyền dữ liệu đến nhiều thiết bị khác nhau cùng lúc (một chiều). Đây là cơ chế nhanh chóng và dễ sử dụng, là lựa chọn tốt nếu muốn truyền lượng nhỏ dữ liệu đến nhiều thiết bị cùng lúc. Hạn chế là dữ liệu không được bảo đảm an ninh, vì thế không phù hợp để truyền các dữ liệu nhạy cảm.

Connection



Hình 4.5: Mô hình connection

- Thiết bị Central (Master): Quét các gói tin quảng bá hướng kết nối theo tần số đặt trước, khi phù hợp thì khởi tạo một kết nối với một peripheral. Central quản lý timing và bắt đầu những sự trao đổi dữ liệu theo chu kỳ.
- Thiết bị Peripheral (Slave): Phát các gói tin quảng bá hướng kết nối theo chu kỳ và chấp nhận kết nối do central yêu cầu.
- Khởi tạo kết nối:
 - Khi muốn kết nối, slave phát các gói tin quảng bá ra không gian.

- Central nhận được các gói tin quảng bá của slave, trong đó chứa các thông tin cần thiết cho phép kết nối với slave đó.
- Dựa trên đó, central gửi yêu cầu kết nối đến slave để thiết lập một kết nối riêng giữa hai thiết bị.
- Khi kết nối được thiết lập, slave dừng quảng bá và hai thiết bị có thể bắt đầu trao đổi dữ liệu hai chiều. Vai trò Master và Slave không ảnh hưởng đến việc truyền dữ liệu, mặc dù Master là bên quản lý thiết lập kết nối.
- Lợi thế của Connection so với Broadcasting là:
 - Thông lượng cao.
 - Khả năng thiết lập kết nối liên kết mã hóa an ninh.
 - Khả năng quản lý năng lượng tối ưu hơn.
 - Khả năng tổ chức dữ liệu với nhiều sự thay đổi phù hợp cho các thuộc tính thông qua việc sử dụng các lớp giao thức bổ sung thêm, đặc biệt là Generic Attribute Profile (GATT). Dữ liệu được tổ chức theo cấu trúc có ý nghĩa xoay quanh cái gọi là services và characteristics.

Protocols và Profiles

Để hai thiết bị có thể giao tiếp với nhau thông qua chuẩn BLE, các thiết bị BLE cần tuân thủ một số quy định. Các quy định này được khái quát hóa thành các giao thức và cấu hình.

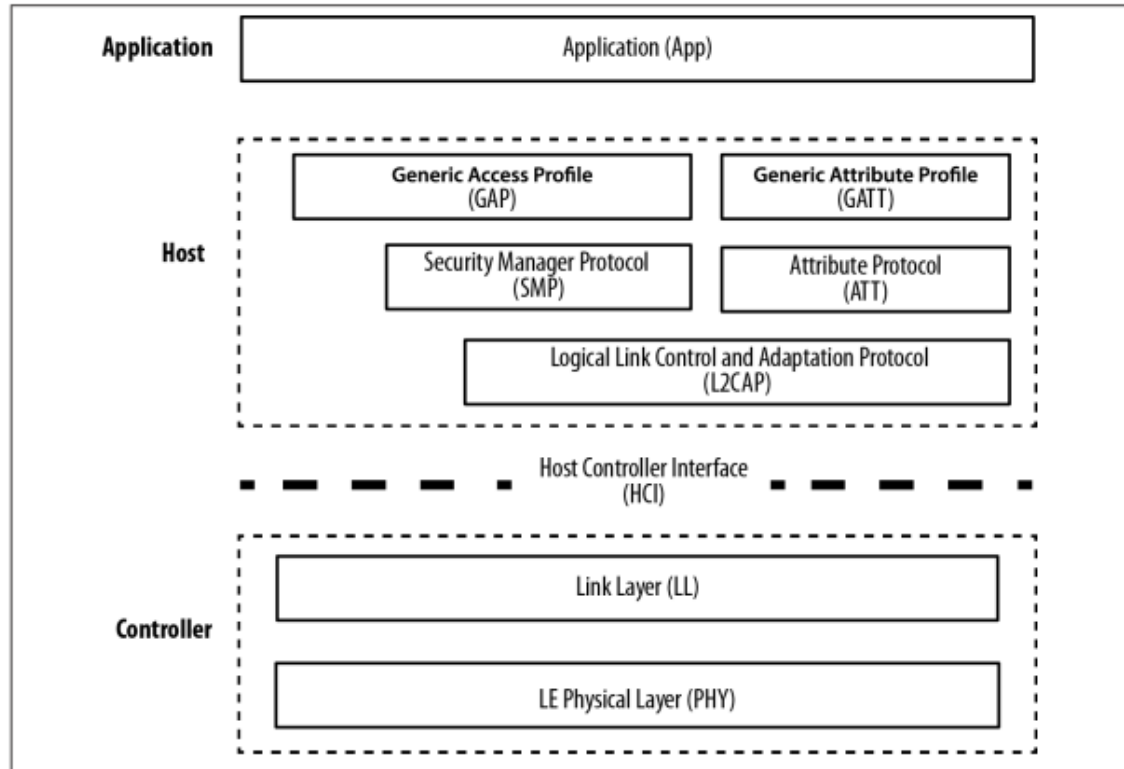
- **Protocol** (Giao thức): Tập các luật quy định việc định dạng gói tin, định tuyến, dồn kênh, mã hóa,... để trao đổi dữ liệu giữa các bên.
- **Profile** (Cấu hình): Định nghĩa cách mà giao thức được dùng để đạt các mục tiêu cụ thể. Có hai loại cấu hình là cấu hình chung (generic profiles) và cấu hình cụ thể theo trường hợp sử dụng (use-case profiles)
 - *Generic profiles*: các profile cơ sở được định nghĩa trong tài liệu Bluetooth Specifications, đặc biệt là hai profiles không thể thiếu giúp các thiết bị BLE kết nối và trao đổi dữ liệu với nhau, GAP và GATT.
 - *Use-case profile*: Các profile cho các trường hợp sử dụng cụ thể
 - Các profile do Bluetooth Special Interest Group (SIG) định nghĩa
 - Các profile do vendor tự định nghĩa

The BLE Protocols Stack

Để lập trình cho thiết bị BLE, có thể chỉ cần quan tâm đến các hàm API ở lớp trên của bộ giao thức BLE (BLE protocol stack), nhưng tốt hơn hết chúng ta nên bắt đầu với một cái

nhìn cơ bản về bộ giao thức cho BLE, giúp cung cấp kiến thức nền tảng để có thể nghiên cứu sâu hơn về BLE.

Hình dưới thể hiện đầy đủ về các thành phần bên trong bộ giao thức BLE cho thiết bị Bluetooth Smart.



Hình 4.6: BLE Protocols Stack

- Bộ giao thức cho thiết bị BLE được chia thành 3 phần chính: controller, host và application. Mỗi phần bao gồm một hoặc nhiều lớp (layer) theo chức năng:
 - Application: Là lớp cao nhất của bộ giao thức, cung cấp giao diện người dùng, xử lý logic, và điều khiển dữ liệu của mọi thứ liên quan đến các trường hợp hoạt động của ứng dụng. Kiến trúc của application phụ thuộc nhiều vào từng bài toán cụ thể.
 - Host: bao gồm các lớp sau
 - Generic Access Profile (GAP)
 - Generic Attribute Profile (GATT)
 - Attribute Protocol (ATT)
 - Security Manager (SM)

- Logical Link Control and Adaptation Protocol (L2CAP)
- Host Controller Interface (HCI), Host side
- Controller: bao gồm các lớp sau
 - Host Controller Interface (HCI), Controller side
 - Link Layer (LL)
 - Physical Layer (PHY)

Tóm lại, BLE protocol stack bao gồm nhiều lớp, mỗi lớp đảm nhiệm một vài chức năng nhất định giúp thực hiện quá trình giao tiếp giữa các thiết bị BLE với nhau.

Các profiles cơ sở: GAP và GATT

GAP (Advertising and Connections)

- GAP (Generic Access Profile) là nền tảng cho phép các thiết bị BLE giao tiếp với nhau. Nó cung cấp một framework mà bất cứ thiết bị BLE nào cũng phải tuân theo để có thể tìm kiếm các thiết bị BLE (Bluetooth) khác, quảng bá dữ liệu, thiết lập kết nối an ninh, thực hiện nhiều hoạt động nền tảng theo một chuẩn.
- Tài liệu BLE Specifications định nghĩa các khái niệm sau khi xét đến sự tương tác giữa các thiết bị:
 - Roles: Mỗi thiết bị có thể hoạt động theo một hoặc nhiều vai trò khác nhau tại cùng một thời điểm: broadcaster, observer, central, peripheral.
 - Modes: Một mode là một trạng thái mà thiết bị có thể chuyển đến trong một khoảng thời gian để đạt được một mục đích cụ thể hoặc nhiều điều đặc biệt, để cho phép một peer thực hiện một thủ tục cụ thể.
 - Procedures: Là các thủ tục (thường thì Link Layer điều khiển sự trao đổi gói tin) để cho phép một thiết bị đạt được một mục đích chắc chắn. Một thủ tục thường được liên kết với một mode, nên mode và procedure thường được xem xét cùng nhau.
 - Security: GAP xây dựng dựa trên Security Manager và Security Manager Protocol (định nghĩa các modes và procedures an ninh để xác định cách mà các thiết bị đặt mức an ninh khi trao đổi dữ liệu). Ngoài ra GAP định nghĩa thêm các tính năng an ninh cao hơn mà không gắn với modes và procedures cụ thể nào, tăng cường mức bảo vệ dữ liệu được yêu cầu bởi mỗi ứng dụng.

Xét một cách bản chất thì GAP là lớp điều khiển cao nhất của BLE (Topmost control layer) và là cấu hình bắt buộc cho tất cả thiết bị BLE.

GATT (Services and Characteristics)

- GATT thiết lập chi tiết cách trao đổi tất cả profile và dữ liệu người dùng qua kết nối BLE. Ngược lại với GAP (định nghĩa sự tương tác mức thấp với các thiết bị), GATT chỉ trình bày các thủ tục truyền và định dạng dữ liệu thực tế
- GATT sử dụng ATT và giao thức truyền của nó để trao đổi dữ liệu giữa các thiết bị. Dữ liệu này được tổ chức phân cấp thành các phần gọi là services, nó nhóm các phần khái niệm liên quan của dữ liệu người dùng gọi là characteristic. Nói một cách ngắn gọn thì dữ liệu truyền qua BLE là dữ liệu có cấu trúc, mà cụ thể là được tổ chức phân cấp thành services và characteristics.

Roles

- GATT Client: tương ứng với ATT client, gửi yêu cầu đến server và nhận kết quả phản hồi. Ban đầu, GATT Client không biết server hỗ trợ những thuộc tính nào vì thế nó cần phải thực hiện service discovery.
- GATT Server: tương ứng ATT server, nhận yêu cầu từ client và gửi những nội dung tương ứng.
- Chú ý rằng các vai trò của GATT không phụ thuộc vào vai trò của GAP. Có nghĩa là cả GAP Central và GAP Peripheral có thể hoạt động như GATT Client hoặc GATT Server hoặc thậm chí là cả hai tại cùng một thời điểm.

UUIDs

- Là một số định danh thiết bị, dài 128 bit (16 byte) duy nhất trên thế giới. Vì độ dài quá lớn, chiếm phần lớn trong gói dữ liệu, BLE Specification định nghĩa thêm 2 định dạng UUID: 16bit và 32 bit. Các định dạng ngắn này có thể chỉ được sử dụng với UUID được định nghĩa trong BT Specification.

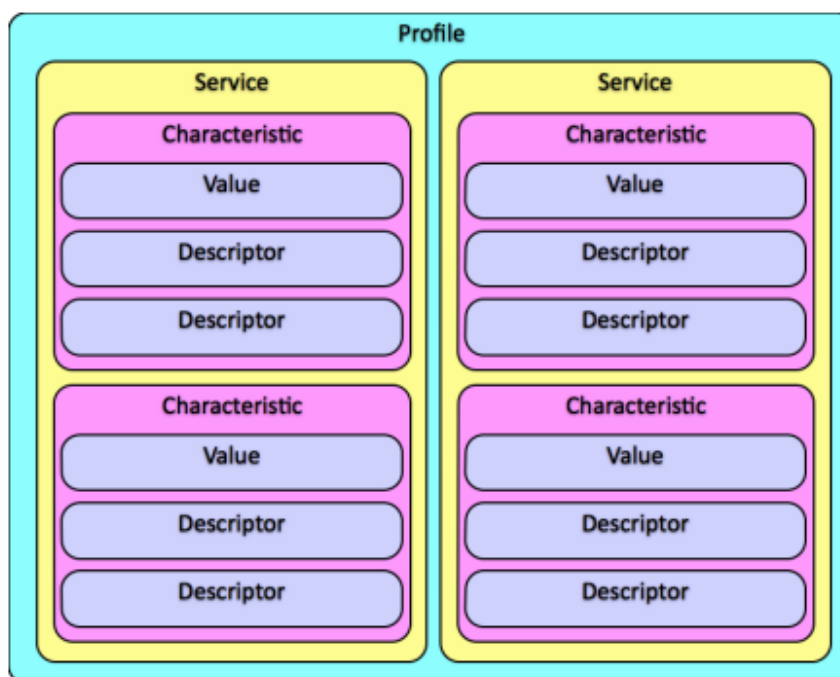
Attributes

- Là thực thể dữ liệu nhỏ nhất được định nghĩa bởi GATT (và ATT).
- Cả GATT và ATT chỉ làm việc với attributes nên để tương tác giữa client và server tất cả dữ liệu phải được tổ chức theo định dạng này.
- Mỗi attribute chứa thông tin về chính nó là dữ liệu người dùng và được mô tả như sau:
 - Handle: số 16 bit duy nhất trên mỗi server để địa chỉ hóa attribute
 - Type: là kiểu UUID, 16bit – 32bit – 128 bit
 - Permission: xác định các ATT operation có thể thực thi trên attribute cụ thể

- Value: chứa phần dữ liệu thực tế trong attribute, giới hạn 512 byte

Services và Characteristics

Dữ liệu trao đổi thông qua kết nối BLE là dữ liệu có cấu trúc, được tổ chức phân cấp thành các services, bản thân services lại bao gồm các characteristics.



Hình 4.7: Tổ chức phân cấp trong cấu trúc dữ liệu BLE

GAP và GATT là hai profiles nền tảng cho mọi ứng dụng BLE. Ngoài ra, tùy ứng dụng mà các thiết bị sẽ cung cấp các profile khác (dựa trên GAP và GATT). Các services chính là các dịch vụ mà thiết bị cung cấp như: Heart Rate Monitor, Battery, Health Thermometer, HID,...

4.4.2 Bluetooth Mesh

- Bluetooth mesh do chuẩn Bluetooth SIG quy định và được phát hành vào tháng 7 năm 2017 với mục tiêu mở rộng kết nối của các thiết bị Bluetooth.
- Bluetooth mesh xây dựng dựa trên BLE và sử dụng nhiều khái niệm của BLE
- Có 2 trạng thái hoạt động chính của các thiết bị BLE là: Advertising (Scanning), hoặc Connection. Còn Bluetooth mesh chỉ sử dụng trạng thái Advertising/Scanning của thiết bị BLE. Điều này có nghĩa là các thiết bị nằm trong mạng Bluetooth mesh sẽ không kết nối lẫn nhau như cách truyền thống

mà BLE đã làm. Thay vào đó, các thiết bị sẽ chuyển tiếp bản tin cho nhau thông qua các gói tin Advertising và sẽ được nhận bởi các thiết bị khác trong mạng thông qua Scanning.

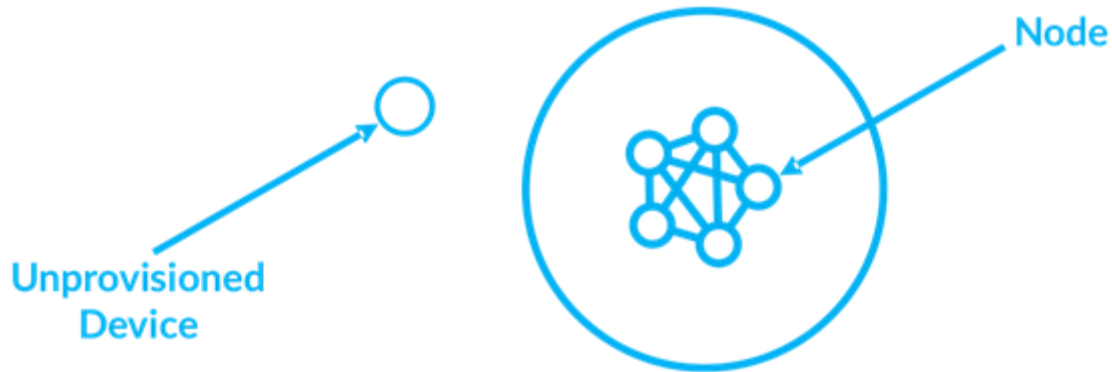
- Bluetooth mesh hỗ trợ từ version BLE 4.0 trở đi, với BLE 5.0 thì khoảng cách truyền xa hơn, tốc độ truyền dữ liệu cao hơn.
- Các phiên bản trước của Bluetooth hỗ trợ hai cấu trúc liên kết khác nhau: One-to-one (Khi hai thiết bị BLE được kết nối). One-to-many: Khi các thiết bị BLE hoạt động độc lập và đang ở trạng thái Broadcast.
- Với Bluetooth mesh, BLE sẽ sử dụng một cấu trúc liên kết mới. Các thiết bị hoạt động như mạng many-to-many, các thiết bị có thể thiết lập kết nối với nhiều thiết bị khác ở trong mạng.
- Số lượng node tối đa trong mạng Bluetooth mesh là 32.767 node.
- Bước nhảy tối đa mà bản tin có thể truyền trong mạng Bluetooth mesh là 127 bước.

Lợi ích của mesh network:

- **Extended range:** Các node có thể chuyển tiếp bản tin đến các node ở xa hơn thông qua node ở giữa chúng. Điều này cho phép mở rộng phạm vi hoạt động của mạng và mở rộng phạm vi hoạt động của các thiết bị.
- **Self-healing capabilities:** Nếu có một node ở trong mạng bị mất kết nối, thì các node còn lại vẫn có thể tham gia vào quá trình gửi bản tin, chuyển tiếp bản tin đến node khác. Tuy nhiên, điều này chỉ đúng một phần trong BLE mesh vì có rất nhiều loại node có chức năng khác nhau và một số loại node lại phụ thuộc lẫn nhau, mình sẽ đề cập những loại này sau trong bài viết sau.
-

Các khái niệm quan trọng trong Bluetooth Mesh

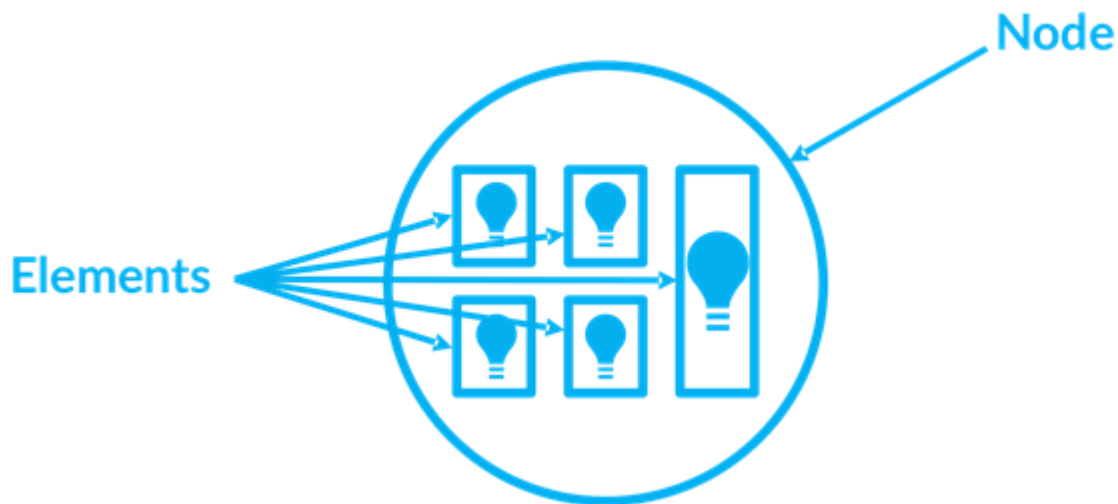
- **Node:** Node là một thiết bị đã tham gia mạng Bluetooth mesh. Các thiết bị không phải là một phần của mạng được gọi là thiết bị chưa được cấp phát (unprovisioned). Khi một thiết bị unprovisioned được cấp phát (provisioned) nó sẽ tham gia vào mạng Bluetooth mesh và trở thành một node.



Bluetooth mesh Network

Hình 4.8: Node trong Bluetooth Mesh

- **Element:** Một node có thể chứa nhiều phần có thể được kiểm soát độc lập. Ví dụ, một cụm bóng đèn có thể chứa nhiều bóng đèn có thể bật/tắt độc lập. Những phần khác nhau của một node được gọi là các phần tử.



Hình 4.9: Element trong Bluetooth Mesh

- **State:** Trong node có nhiều phần tử khác nhau và mỗi phần tử mang một trạng thái. Chẳng hạn bật và tắt là trạng thái của bóng đèn trong cụm bóng đèn. Một sự thay đổi từ trạng thái này sang trạng thái khác được gọi là sự chuyển đổi trạng thái. Điều này có thể là ngay lập tức hoặc có thể xảy ra theo thời gian. Một số trạng thái có thể bị ràng buộc với nhau, có nghĩa là một thay đổi trong một trạng thái kích hoạt một thay đổi ở trạng thái khác. Có thể có hai

hoặc nhiều trạng thái ràng buộc với nhau. Lấy ví dụ như một bóng đèn dimmer: Điều chỉnh độ sáng của đèn cần những mức độ khác nhau cũng như trạng thái bật/tắt. Nếu giá trị trạng thái hiện tại thay đổi thành 0, nó sẽ kích hoạt trạng thái bật/tắt để chuyển sang tắt. Nếu mức thay đổi từ 0 thành giá trị khác không, thì điều đó sẽ kích hoạt trạng thái bật/tắt để chuyển sang bật và sáng theo từng mức độ khác nhau.

• **Property:** Thuộc tính được thêm vào trong một số tình huống cho giá trị trạng thái. Chẳng hạn, định nghĩa giá trị nhiệt độ ở trong nhà hoặc ngoài trời có hai kiểu thuộc tính:

- Manufacturer property: Chỉ cung cấp quyền truy cập cho đọc
- Admin property: Chỉ cung cấp quyền truy cập cho ghi

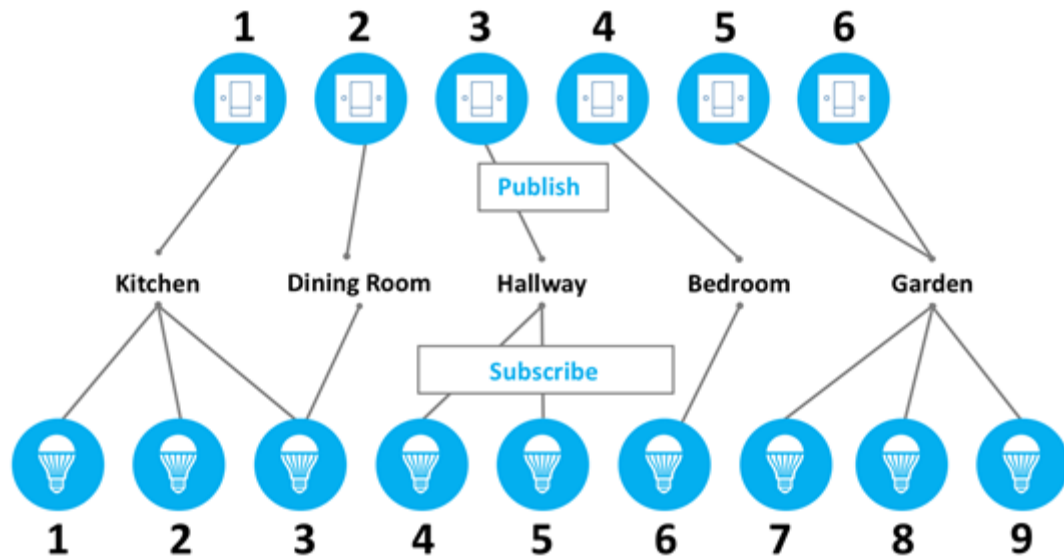
• **Message:** Trong Bluetooth mesh, tất cả các giao tiếp trong mạng đều được thực hiện theo bản tin. Các node gửi bản tin để điều khiển hoặc chuyển tiếp thông tin cho nhau. Có ba loại bản tin trong Bluetooth mesh, mỗi loại được xác định bởi một mã opcode:

- Một bản tin GET: Một thông báo để yêu cầu status từ một hoặc nhiều node.
- Một bản tin SET: Một thông báo để thay đổi value của một status nhất định.
 - Phản hồi bản tin GET: Chứa giá trị trạng thái.
 - Phản hồi bản tin SET: Xác nhận
- Một bản tin STATUS: Bản tin về status được sử dụng trong các tình huống khác nhau
- Một số bản tin yêu cầu thiết bị nhận xác nhận bản tin được gửi. Việc thông báo xác nhận này có hai mục đích như sau:
 - Xác nhận đã nhận được bản tin
 - Trả lại dữ liệu liên quan đến bản tin nhận được.
- Trong trường hợp thiết bị gửi không nhận được phản hồi cho bản tin hoặc nhận được phản hồi không mong muốn, nó có thể gửi lại bản tin. Nhiều bản tin được xác nhận nhận được bởi một node không ảnh hưởng đến quá trình hoạt động của node đó.

• **Address:** Bản tin trong mạng Bluetooth mesh phải được gửi đến hoặc gửi từ một địa chỉ. Có ba loại địa chỉ:

- Unicast Address: Một địa chỉ định danh duy nhất của 1 node được cung cấp trong quá trình provisioning
- Group Address: Một địa chỉ được sử dụng để xác định một nhóm các node. Một group address có thể hiểu như một căn phòng, các bóng đèn ở trong một căn phòng nhất định và mỗi bóng đèn là một node. Theo định nghĩa của BLE SIG, có 4 Group address cố định bao gồm All-proxies, All-friends, All-relays, and All-nodes

- Virtual Address: Địa chỉ mà có thể gán cho một hoặc nhiều phần tử, trên một hoặc nhiều nodes. Hoạt động này giống như là một nhãn và có định dạng 128-bit UUID có thể lấy bởi bất kì phần tử nào liên kết. Địa chỉ ảo này có khả năng được cấu hình sẵn tại điểm sản xuất.
- **Publish/Subscribe:** Cách trao đổi bản tin trong mạng Bluetooth mesh thông qua cơ chế Publish/Subscribe. Publish là hoạt động gửi bản tin đi. Subscribe là cấu hình đăng kí một địa chỉ nhất định để nhận các bản tin Pushlish sau đó xử lý. Thông thường, tin nhắn được gửi đến Group address hoặc Virtual address.



Hình 4.10: Cơ chế Publish và Subscribe

Ví dụ về mạng lưới trong một ngôi nhà mà gồm có 6 công tắc đèn và 9 bóng đèn. Mạng lưới sử dụng phương thức Publish/Subscribe để cho phép các nút gửi bản tin cho nhau. Các Node có thể Subscribe nhiều địa chỉ như bóng đèn 3 trong hình trên, được Subscribe vào cả địa chỉ nhóm Kitchen và Dining Room. Ngoài ra, nhiều nút có thể Publish đến cùng một địa chỉ, chẳng hạn như các công tắc 5 và 6 trong ví dụ này hai công tắc này điều khiển cùng một nhóm đèn Garden.

- **Managed flooding:** Bluetooth mesh chuyển tiếp bản tin thông từ node này đến node kia trong mạng mà không xem xét về đường truyền để đến được địa chỉ đích. Kỹ thuật này được gọi là Managed flooding, bản tin được phát tới tất cả các node trong phạm vi của node gửi với một vài kĩ thuật tối ưu hóa được thêm vào:
 - **Messages have a TTL assigned:** TTL là viết tắt của time-to-live, giới hạn số bước nhảy mà bản tin có thể truyền qua các nodes trong mạng. Nếu TTL = 0 có nghĩa là một node có thể gửi tin nhắn đến các nodes khác trong phạm vi phủ sóng của nó và chỉ ra rằng các nodes

không có khả năng chuyển tiếp bản tin khi $TTL = 0$. Nếu một tin nhắn được gửi với $TTL = 2$, thì mỗi lần nó được chuyển tiếp, giá trị TTL sẽ bị giảm. Giá trị TTL là 1 có nghĩa là tin nhắn có thể đã được chuyển tiếp ít nhất một lần nữa.

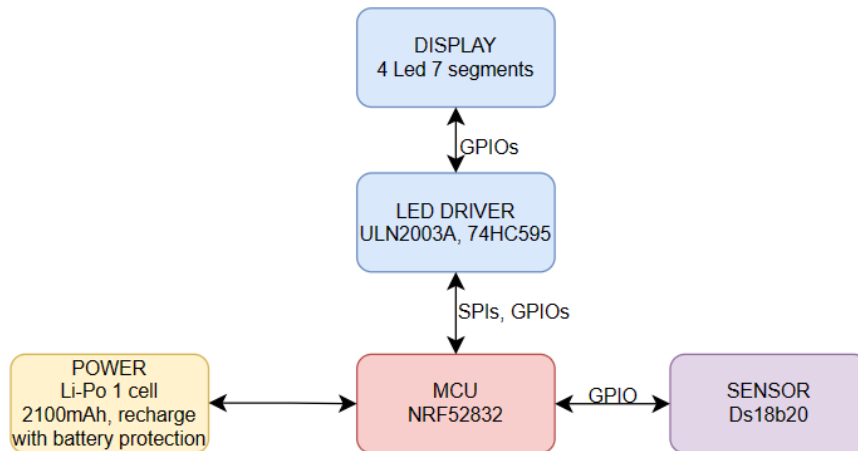
- **Messages are cached:** Bản tin được yêu cầu lưu trữ bởi tất cả các nodes và yêu cầu tất cả các bản tin tồn tại trong bộ nhớ đệm phải bị xóa ngay lập tức
- **Friendship:** là mối quan hệ của 2 node trong mạng, hai loại này là:
 - Low-power node (LPN) là node bị giới hạn nguồn và bảo tồn nguồn năng lượng và hầu hết không thể nhận được bản tin của mạng mesh trong hầu hết thời gian. Node này dành hầu hết thời gian để tắt bộ phát sóng.
 - Friend node luôn luôn được cấp năng lượng để hoạt động, nó có thể dùng để cấp phát dữ liệu cho LPN, Friend node sẽ lưu trữ bản tin cho LPN để tiết kiệm năng lượng, vì vậy LPN có dành hầu hết thời gian ở trạng thái ngủ và thỉnh thoảng thức dậy. Khi LPN thức dậy, nó sẽ đọc hết tất cả dữ liệu ở trong bộ nhớ của Friend node và gửi bất kỳ bản tin nào mà nó cần phải gửi đến mạng lưới.

4.5 Lora

LoRa (Long Range Radio) là một nền tảng công nghệ không dây có công suất thấp và phạm vi xa, sử dụng dải tần miễn phí. Mục đích của tạo ra công nghệ LoRa là nhằm loại bỏ repeater, giảm giá thành thiết bị. Nó là một lớp vật lý, được sử dụng cho giao tiếp với khoảng cách lớn. LoRa là một kỹ thuật điều chế dựa trên kỹ thuật trải phổ Spread-Spectrum và biến thể của Chirp Spread Spectrum (CSS), được nghiên cứu và phát triển bởi Cycleo và được mua lại bởi công ty Semtech vào năm 2012. Thay vì sử dụng công suất lớn và băng thông hẹp thì tín hiệu được trải ra băng thông rộng hơn với mức công suất nhỏ hơn (tổng mức công suất không đổi) mà bên thu có bộ giải mã thích hợp vẫn có thể nhận được chính xác dữ liệu. Nhờ áp dụng kỹ thuật trải phổ mà tín hiệu truyền đi ít bị suy hao và chống nhiễu tốt hơn, do đó có thể truyền tín hiệu đi xa trong khi truyền với mức công suất thấp. Dữ liệu truyền đi được mã hóa thành các “Chirp” là tín hiệu hình sin có tần số thay đổi tuyến tính theo thời gian, có hai loại tín hiệu Chirp: Up – Chirp (Tần số tín hiệu tăng dần theo thời gian) và Down – Chirp (Tần số tín hiệu giảm dần theo thời gian).

CHƯƠNG 5: MÔ TẢ CHI TIẾT

5.1. Kiến trúc chung của nút cảm biến



Hình 5.1: Kiến trúc chung của nút cảm biến không dây

Một nút cảm biến không dây: bản chất là một thiết bị đo mang tính nhỏ gọn, tiêu thụ năng lượng thấp, dễ di chuyển và lắp đặt, sử dụng công nghệ truyền tin không dây.

Một nút cảm biến không dây bao gồm các khối:

1. Khối nguồn cung cấp
2. Khối đầu đo cảm biến/ cơ cấu chấp hành
3. Khối điều khiển có thể tích hợp cả bộ nhớ và truyền thông không dây, antenna

5.2. Năng lượng tiêu thụ của nút cảm biến

Để đảm bảo công suất tiêu thụ thấp khi thiết kế, nhóm sẽ phân tích về các vấn đề tiêu thụ năng lượng của từng phần tử trong cảm biến và mạng.

Năng lượng tiêu thụ của mạng được tính bằng thời gian cảm biến truyền hay nhận hay ngủ.

Năng lượng tiêu thụ của cảm biến gồm hai phần:

1. Năng lượng của mỗi linh kiện cấu tạo nên nút cảm biến

2. Năng lượng truyền / nhận sóng khi các nút cảm biến trong mạng trao đổi thông tin với nhau

5.2.1 Các chế độ hoạt động của nút cảm biến

Thông thường, một nút cảm biến sẽ có 3 chế độ hoạt động bao gồm:

- Chế độ hoạt động tích cực (Active mode)
- Chế độ nghỉ (Idle mode/ System ON)
- Chế độ ngủ (Sleep mode/ System OFF)

Chế độ hoạt động tích cực (Active mode)	Nút sẽ thực hiện việc đo nhiệt độ, truyền nhận các gói tin trong mạng cũng như chuyển nó tới gateway
	Ta gọi công suất tiêu thụ ở chế độ này là P_{active}
Chế độ nghỉ (Idle mode)	Nút cảm biến sẽ không hoạt động trong lưới, không thực hiện bất kì nhiệm vụ nào
	Ta gọi công suất tiêu thụ ở chế độ này là P_{idle}
Chế độ ngủ (Sleep mode)	Nút cảm biến vẫn hoạt động trong lưới, nhưng không thực hiện nhiệm vụ nào
	Ta gọi công suất tiêu thụ ở chế độ này là P_{sleep}

Bảng 5.1: các chế độ của thiết bị

Sau đây là phần tính toán chi tiết của nhóm về công suất tiêu thụ ở từng kịch bản:

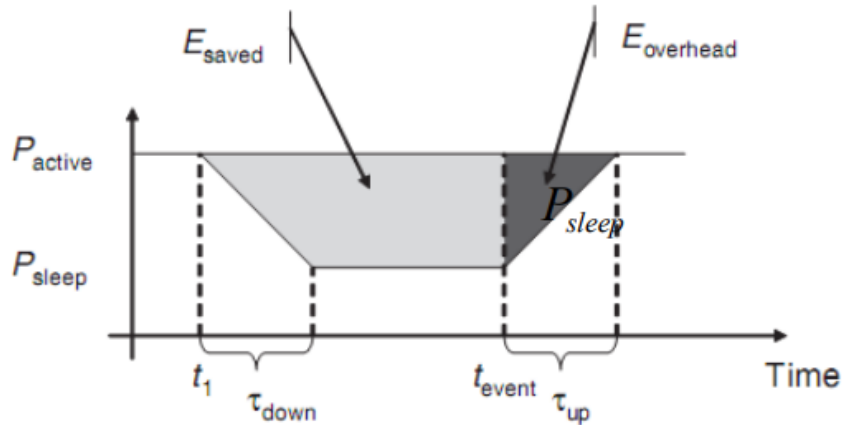
Khi nút cảm biến đang ở chế độ hoạt động tích cực, tại thời điểm t_1 là thời điểm đưa ra quyết định chuyển đổi (switch) sang trạng thái mới hoặc vẫn giữ lại ở chế độ cũ.

- Nếu nút vẫn duy trì trạng thái tích cực thì ta có năng lượng tiêu thụ được tính bằng công thức dưới đây:

$$E_{active} = P_{active} * (t_{event} - t_1)$$

- Nếu nút chuyển sang trạng thái ngủ, thì ta có năng lượng tiêu thụ được tính bằng công thức dưới đây:

$$E = P_{sleep} * (t_{event} - t_{down}) + \int_{t_1}^{t_{down}} P_{active} - P_{sleep}$$



Hình 5.2: Công suất tiêu thụ trong các chế độ hoạt động

5.2.2 Công suất tiêu thụ của nút cảm biến không dây

Nút cảm biến không dây bao gồm:

MCU: NRF52832

VCC: 3.3V

Tần số hoạt động:

Cảm biến: DS18B20

Khối hiển thị: 4 LED 7 SEG K chung

Sau đây sẽ là bảng tính toán công suất tiêu thụ của nút:

Linh kiện	Chế độ	Điện áp hoạt động(V)	Dòng điện tiêu thụ(mA)	Công suất tiêu thụ(mW)
MCU (NRF52832)	Tx(+4dBm)	3.3	7.5	24.75
	Rx(1Mbps)	3.3	5.4	17.82
	Idle	3.3	0.0016	0.00528
	Sleep	3.3	0.0019	0.00627
LM35		5	0.0056	0.028
LED 7 SEG		3.3	30	99

Bảng 5.2: Công suất tiêu thụ của từng phần tử

Như trên, công suất tiêu thụ của nút cảm biến ở chế độ sleep là 0.03427mW, ở chế độ idle là 0.00528, ở chế độ Tx là 1080.778mW, ở chế độ Rx là 1073.848mW.

Nhóm chọn thời gian lấy mẫu là 5s. Trong khoảng 5s đó thì thiết bị sẽ ở chế độ Tx trong khoảng 200ms. Sau đó sẽ chuyển sang mode Rx với transition time là rất ngắn

khoảng 140 μ s. Thiết bị sẽ ở mode Rx để đợi thông tin từ các node khác. Vậy trong khoảng 1 phút sẽ có 2.4s thiết bị ở chế độ Tx, 57s ở chế độ Rx. Về phía LED thì mỗi khi người dùng thực hiện việc bấm nút sẽ hiển thị các thông số lên màn trong khoảng 2s.

Do đó ta tính được công suất tiêu thụ của thiết bị tối đa là (với trường hợp LED on liên tục):

$$P_{avg} = \frac{24.75 * 2.4 + 17.82 * 57}{60} + 0.028 + 99 * 4 = 414 \text{ mW}$$

Suy ra năng lượng tiêu thụ trong 1 giờ là 414mWh. Với điện áp định mức của thiết bị là 3.3V thì tính ra được dòng tiêu thụ là $414/3.3 = 125\text{mAh}$. Thiết bị sẽ hoạt động liên tục trong khoảng $2100/125\text{mAh} = 16.8\text{h}$

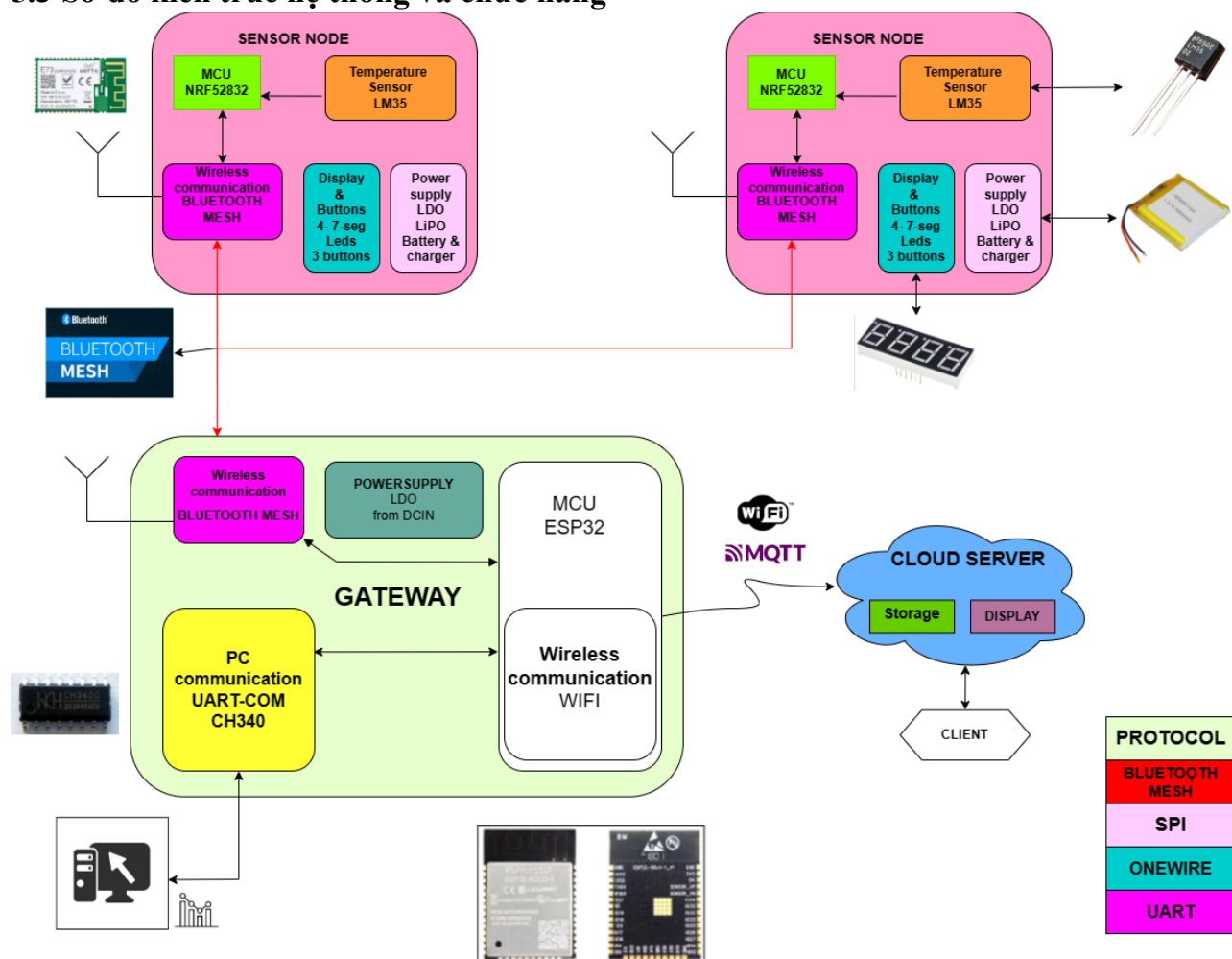
Với công suất tiêu thụ tối thiểu (khi LED được tắt và hoạt động chế độ bình thường chỉ khi người dùng thực hiện bấm mới hiển thị) thì ta có:

$$P_{avg} = \frac{24.75 * 2.4 + 17.82 * 57}{60} + 0.028 = 18 \text{ mW}$$

Suy ra năng lượng tiêu thụ trong 1 giờ là 18mWh. Với điện áp định mức của thiết bị là 3.3V thì dòng tiêu thụ là $18/3.3\text{V} = 5.5\text{mAh}$. Thiết bị hoạt động liên tục trong khoảng thời gian là $2100/5.5 = 381\text{h} = 16 \text{ ngày}$

Từ đó tính được thời gian hoạt động tối thiểu là 17h và tối đa là 16 ngày.

5.3 Sơ đồ kiến trúc hệ thống và chức năng



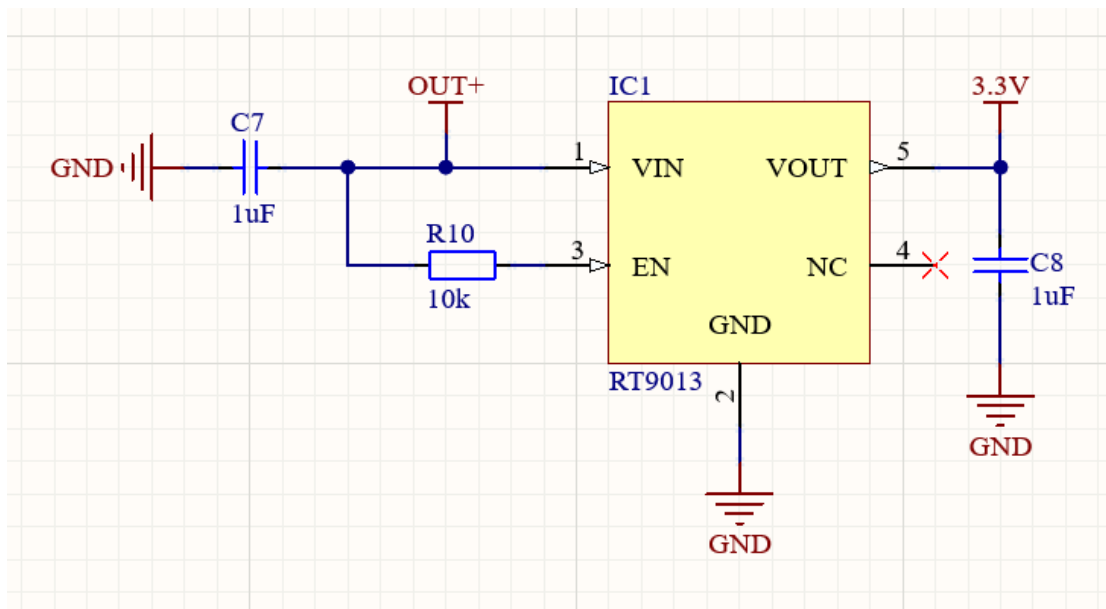
Hình 5.3: Sơ đồ khối mô tả hệ thống

Sơ đồ khối mô tả một cách khái quát về mối liên kết giữa các thành phần của hệ thống, chúng tách ra làm 3 phần riêng biệt:

- Khối Gateway: nhận dữ liệu nhiệt độ từ các Sensor Node và đẩy dữ liệu đó lên Cloud Server thông qua giao thức MQTT
- Khối Sensor Node: giám sát và thu thập dữ liệu nhiệt độ và gửi dữ liệu đó về Gateway để đánh giá thông qua công nghệ truyền thông không dây BLE, cụ thể là truyền tải dữ liệu giữa các Node bằng mạng BLE Mesh
- Khối cloud Server: quản lý và hiển thị dữ liệu theo giao diện thống nhất.

Nhóm chúng em sẽ đi vào demo sơ đồ triển khai chi tiết từng khối phía dưới và phân tích chức năng từng khối trong hệ thống này.

Với điện áp Pin từ 3.7 – 4.2V, nhóm em sử dụng IC LDO RT9013 để hạ áp xuống 3.3V để cấp nguồn cho các khối khác. IC RT9013 cho dòng cung cấp tối đa là 0.5A khi VDO tối thiểu 250mV. Tụ C7 và C8 dùng để lọc nhiễu, chân EN của IC RT9013 sẽ được kéo trở cao để luôn cho phép hoạt động.

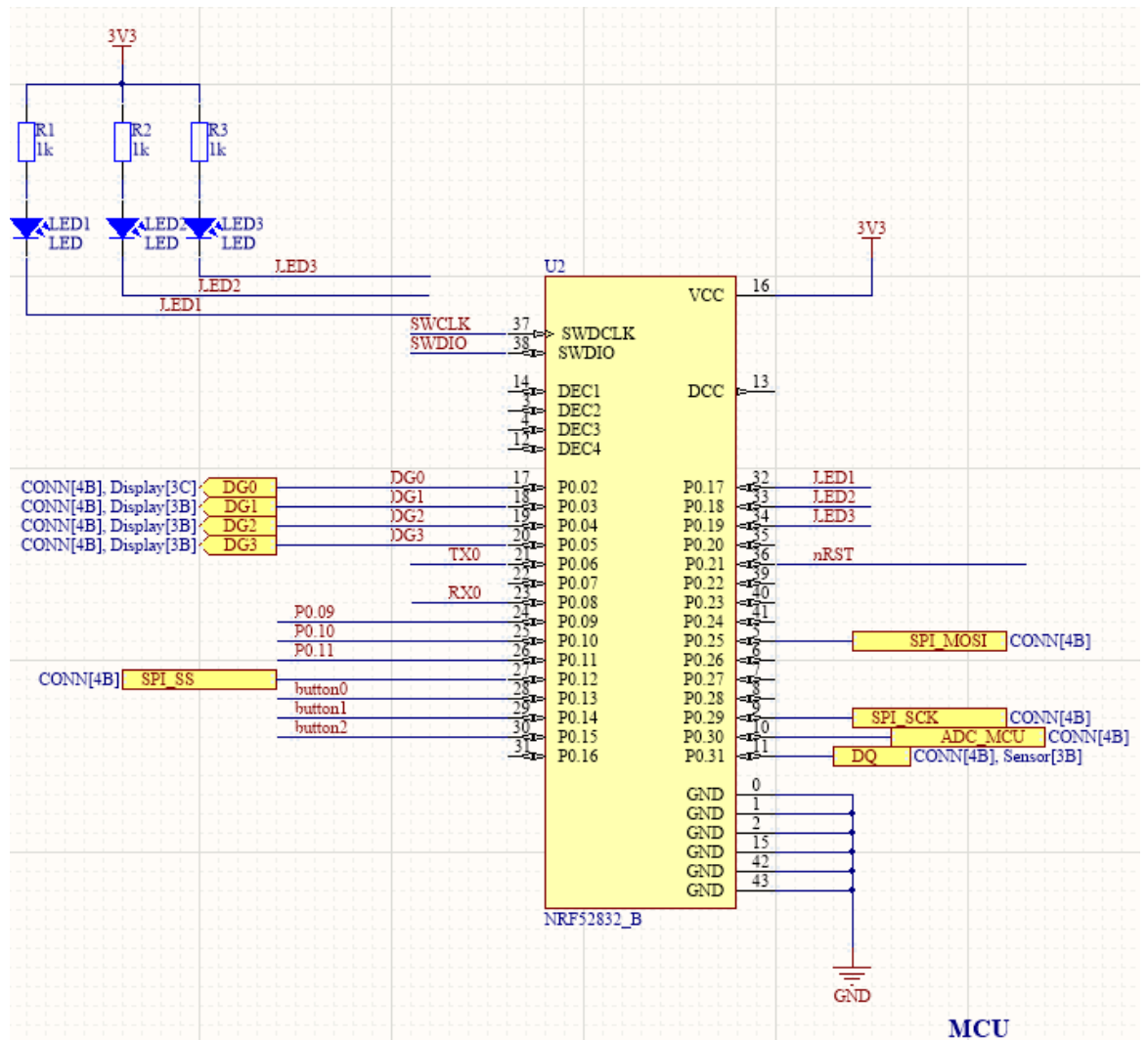


Hình 5.5: Sơ đồ nguyên lý khối nguồn LDO RT9013

5.4.1.2. Khối xử lý trung tâm

Điện áp out của IC RT9013 sẽ được cấp trực tiếp cho NRF52832.

Như đã nói trong phần VI.3, ở Sensor Node nhóm em sử dụng vi điều khiển là NRF52832 có lõi là một vi xử lý ARM Cortex-M4 32 bit, có tần số thạch anh 64MHz, bộ nhớ flash 512kB, 64 kB RAM. Vi điều khiển NRF52832 có thể nạp bằng mạch nạp J-Link thông qua chuẩn Serial wire debug (SWD). Chân DQ là chân để giao tiếp với cảm biến LM35. Các chân SPI_MOSI, SPI_SCK, SPI_SS để giao tiếp với IC 74HC595 để điều khiển 4 LED 7 thanh



Hình 5.6: Sơ đồ nguyên lý vi điều khiển NRF52832

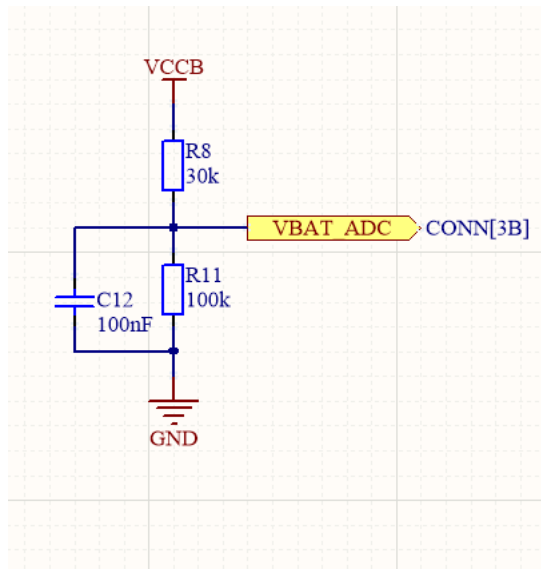
Do kiến thức của bọn em còn hạn chế, nên chúng em đã sử dụng một module NRF52832 bên trong đó đã được thiết kế đầy đủ và có thể sử dụng để thiết kế PCB layout ví dụ như bộ dao động tần số thạch anh, phần antenna phục vụ cho việc hoạt động cho mạng Bluetooth,... và nó sẽ được bao phủ bởi 1 lồng Faraday để bảo quản và chống nhiễu EMI cho chip. Hình ảnh bên dưới sẽ minh họa cho module chúng em sử dụng.



nRF52832/E73-2G4M04S1B

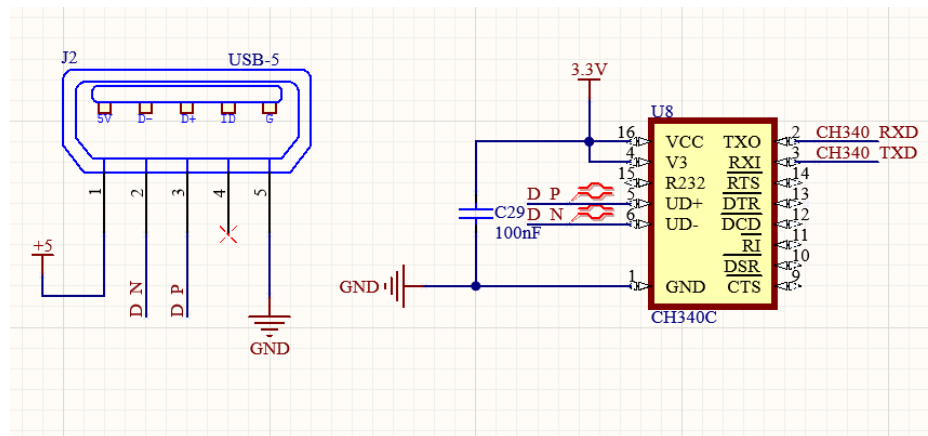
Hình 5.7: Module NRF52832

Do điện áp Pin có thể lên đến 3.8V vượt quá dải điện áp cho phép từ 0 – 3.3V của NRF52832 nên cần được phân áp trước khi đến chân ADC của NRF52832. Ở đây em chọn 2 điện trở là $R_8 = 30k$ và $R_{11} = 100k$, khi $V_{BAT} = 3.8V$ thì $V_{BAT_ADC} = \frac{100}{30+100} * 3.8 = 2.92$ nằm trong dải điện áp cho phép.



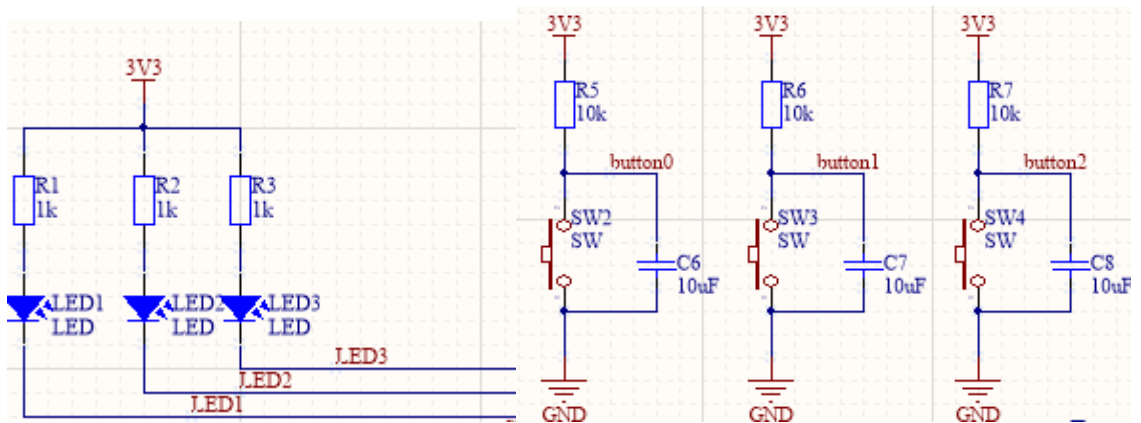
Hình 5.8: Khối giám sát điện áp pin Lipo

Ngoài ra nhóm em thiết kế thêm khối Debug để xuất dữ liệu cần thiết, thuận tiện khi lập trình và kiểm tra lỗi. USB Micro dùng để chuyển tiếp dữ liệu tới máy tính đồng thời cấp nguồn sạc cho Pin. Hai chân D_N và D_P sẽ được nối với chân UD+ và UD- ở IC CH340, IC này sẽ chuyển đổi dữ liệu sang UART để giao tiếp với vi điều khiển.



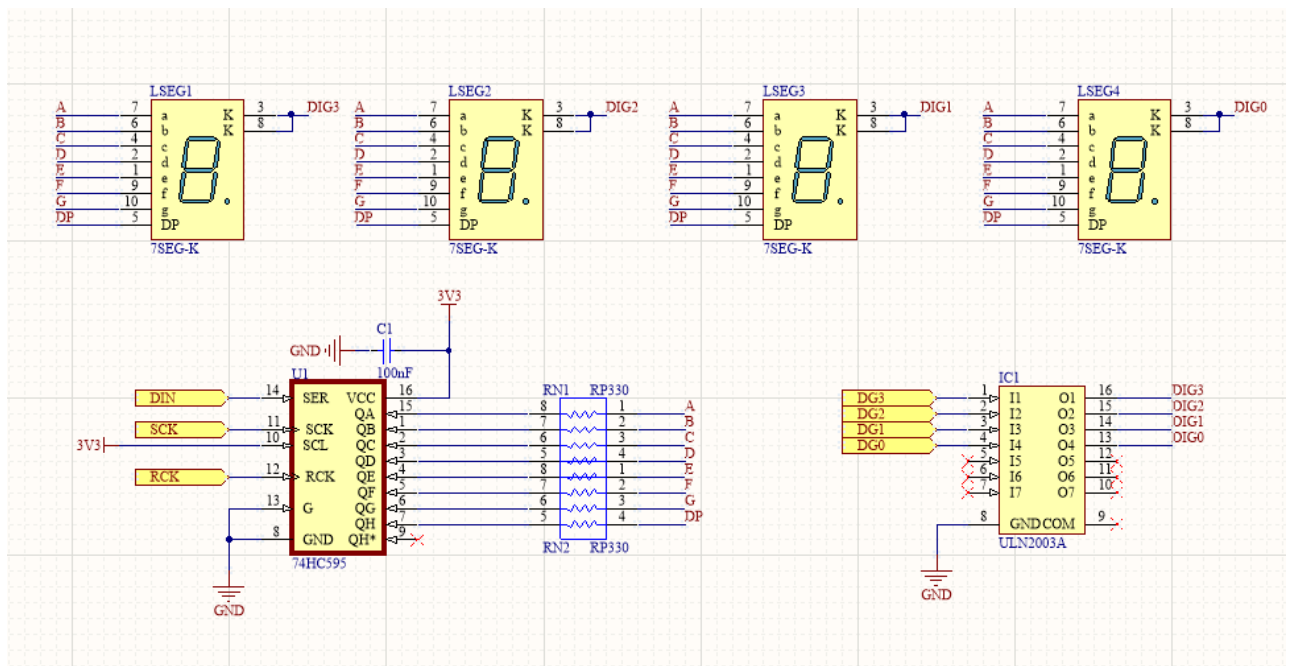
Hình 5.9: Sơ đồ nguyên lý khối debug

Theo như yêu cầu của dự án, ở Sensor Node sẽ có 3 LED cảnh báo ngưỡng và một nút nhấn chức năng. Vi điều khiển sẽ dùng ngắt ngoài để bắt được xung cạnh xuống khi nhấn nút nhấn để tiến hành hoạt động theo từng chức năng của nút nhấn.



Hình 5.10: Sơ đồ nguyên lý các nút nhấn và LED

5.4.1.3. Khối hiển thị



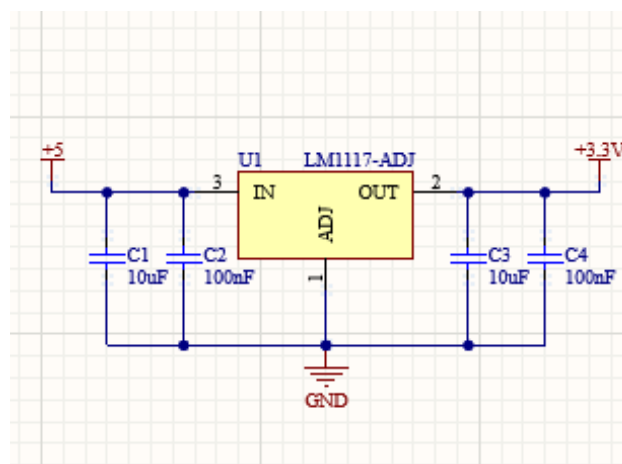
Hình 5.11: Sơ đồ nguyên lý khối hiển thị LED 7 thanh

Các LED 7 thanh được sử dụng là loại Katot chung, được điều khiển qua LED Driver 74HC595. IC 74HC595 giao tiếp với VĐK qua giao thức SPI. 74HC595 là Driver LED 7 thanh được thiết kế cho tối đa 8 số mỗi IC và có khả năng mở rộng bằng cách nối tiếp nhiều IC qua nhau. Trong ứng dụng này chúng em chỉ sử dụng 4 LED 7 thanh nên các cực Katot chung của các LED được nối vào các chân DIG0-3. Các tín hiệu của từng thanh trong các LED được nối song song với nhau và nối vào các chân SEGA-G và SEFDP. Nguồn điện áp pin được cấp trực tiếp cho các LED.

5.4.2 Sơ đồ triển khai chi tiết Gateway

5.4.2.1. Khối nguồn

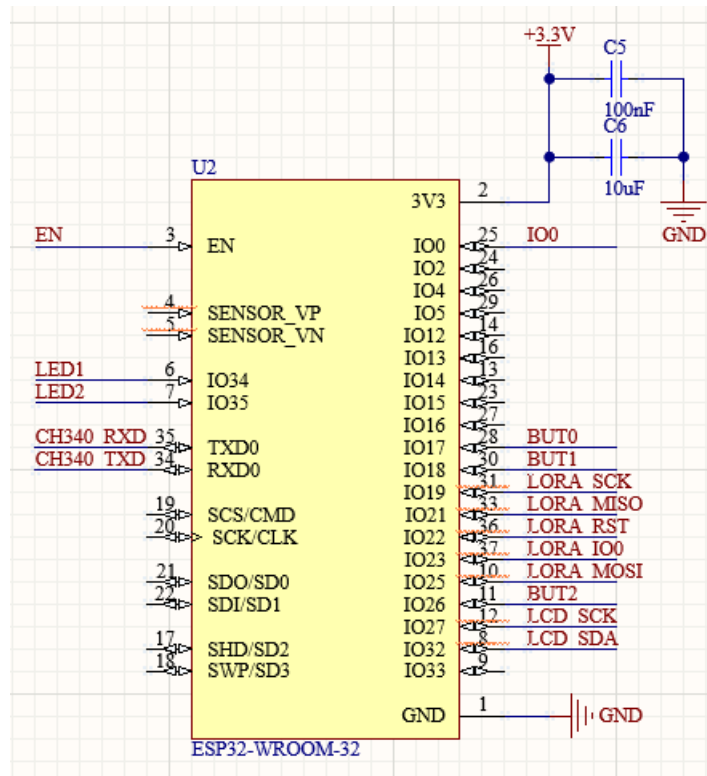
Gateway được cung cấp nguồn đầu vào 5V được lọc nhiễu qua tụ C_1 và C_2 , để hạ áp xuống 3.3V cấp nguồn cho khối xử lý trung tâm thì nhóm em sẽ sử dụng IC LDO LM1117 có khả năng cung cấp dòng điện lên đến 1A đủ đáp ứng cho các khối còn lại



Hình 5.12: Sơ đồ nguyên lý khối nguồn LDO

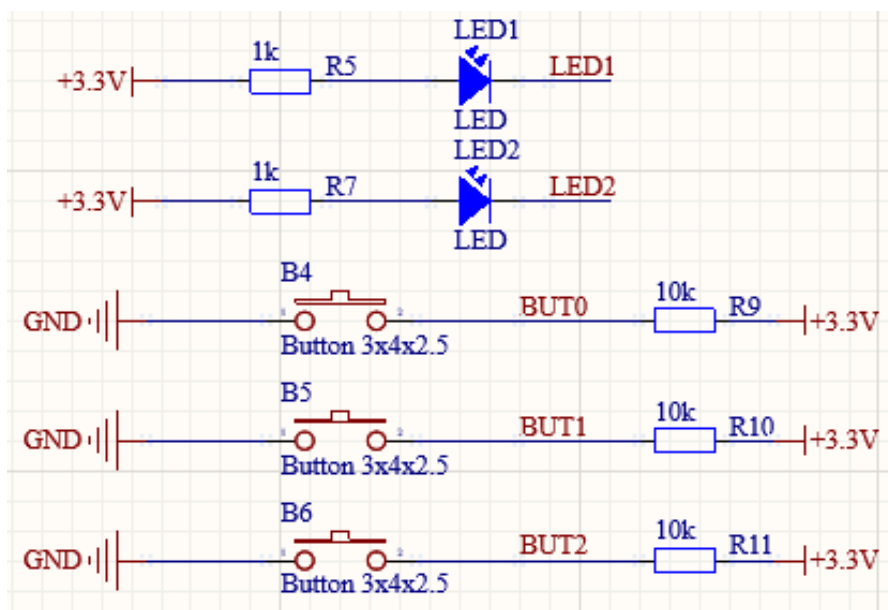
5.4.2.2. Khối xử lý trung tâm

Ở Gateway, nhóm em sử dụng vi điều khiển ESP32. Đây là dòng vi điều khiển hiệu năng rất tốt, tần số hoạt động lên tới 240MHz. Đồng thời hãng đã trang bị cho nó bộ nhớ Flash 4M, SRAM 520KB thích hợp để sử dụng hệ điều hành FreeRTOS. Vi điều khiển được cấp nguồn 3.3V được lọc nhiễu qua 2 tụ C_5 và C_6 theo yêu cầu trong datasheet. Hai chân LCD_SDA và LCD_SCK được dùng để giao tiếp với màn hình LCD 1602. Để Gateway đảm bảo kết nối với WiFi nhất thì có thể sử dụng antenna rời, không sử dụng antenna on – chip.



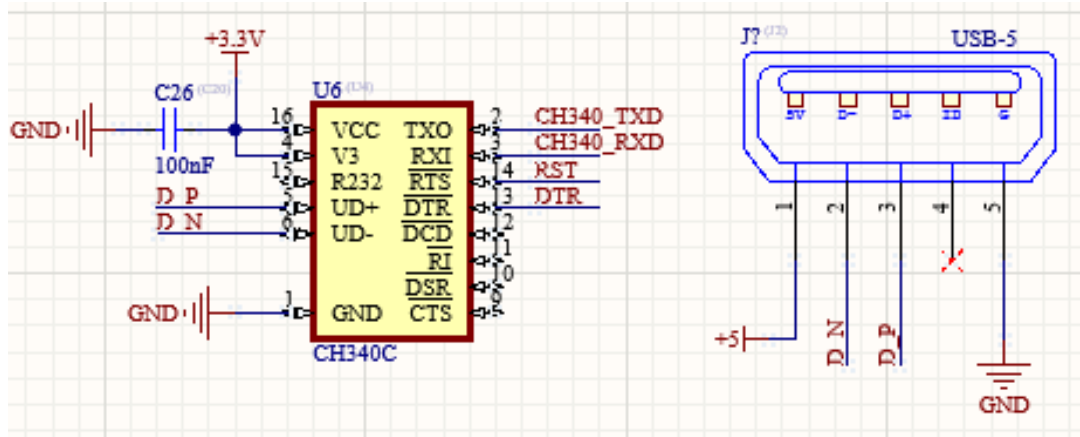
Hình 5.13: Sơ đồ nguyên lý vi điều khiển ESP32

Như đã cập ở yêu cầu 8, Gateway sẽ có 3 nút nhấn chức năng để thuận tiện thao tác vật lý, đồng thời có 2 LED để chỉ thị trạng thái hoạt động.



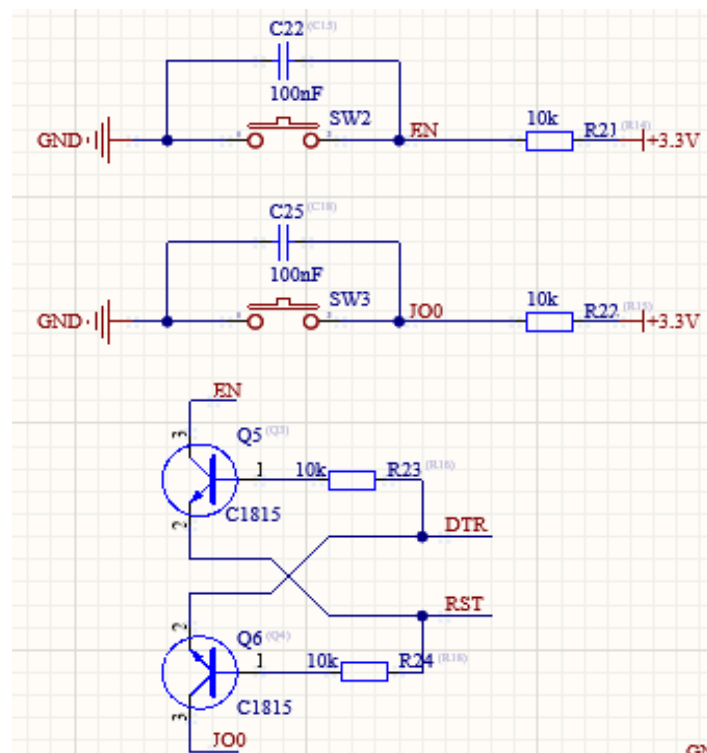
Hình 5.14: Sơ đồ nút nguyên lý các nút nhấn và LED

Để phục vụ quá trình nạp firmware và debug, một khối chuyển đổi UART sang USB và một cổng MicroUSB được sử dụng giống như ở Sensor Node.



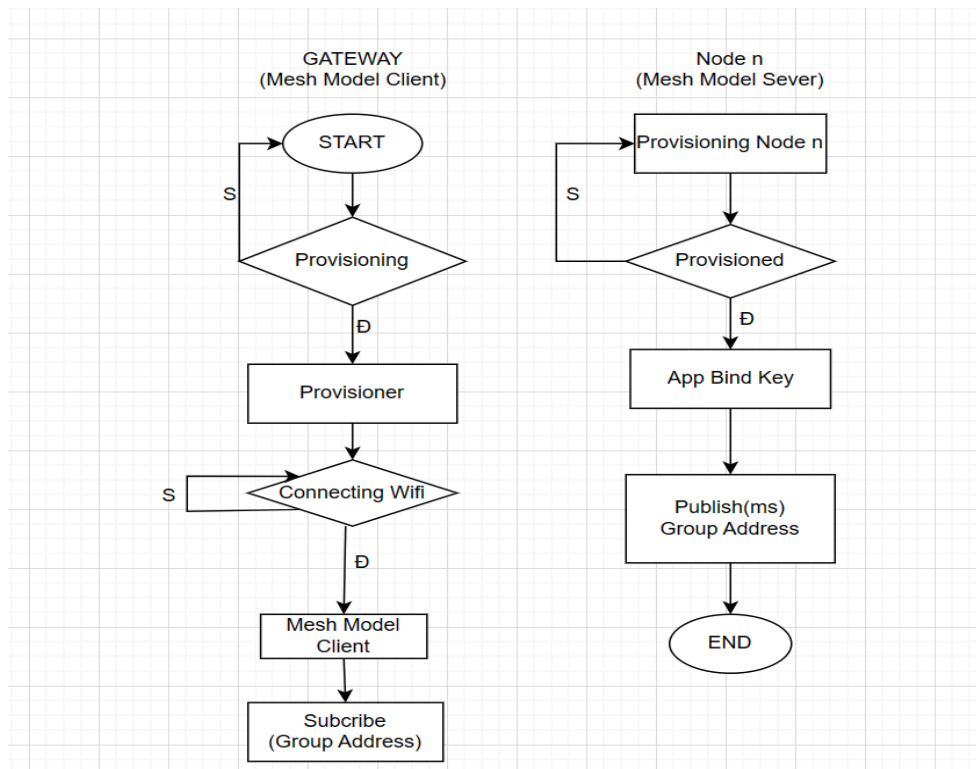
Hình 5.15: Sơ đồ nguyên lý khối debug và nạp code

Khối phụ trợ cho khối nạp firmware, có nhiệm vụ chính là đưa module ESP32 vào chế độ boot bằng cách lần lượt kéo chân EN và GPIO0 xuống GND. Hai nút bấm để reset module và để kéo GPIO0 xuống GND trong trường hợp khối nạp firmware hoạt động không đúng mong đợi.



Hình 5.16: Sơ đồ nguyên lý khối hỗ trợ nạp

5.5 Lưu đồ hoạt động của chương trình



Hình 5.17: Lưu đồ hoạt động của Gateway và Node

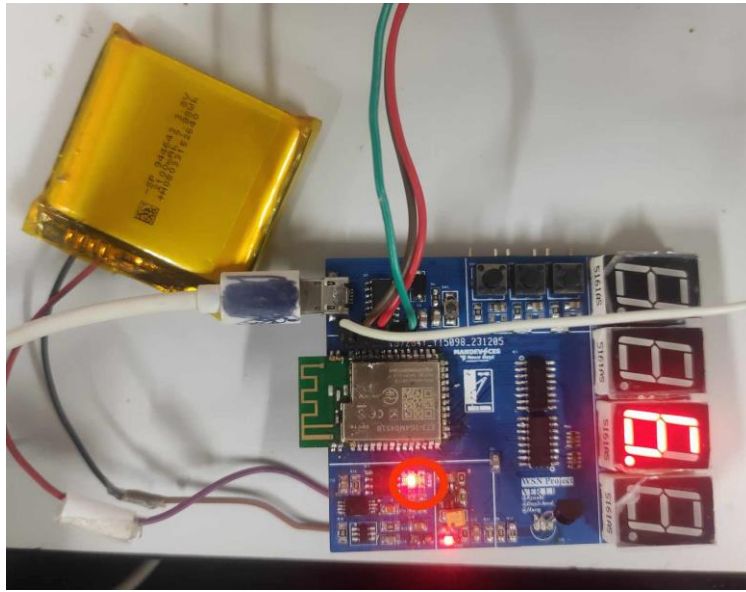
Có thể nói ngắn gọn qua về cách thức hoạt động của 2 thiết bị Gateway và Node như sau:

- Trước tiên là Gateway, đầu tiên chúng ta sẽ cần cho nó vào trong mạng mesh tức là provision nó bằng một app điện thoại tên là NRF Mesh. Sau khi đã vào mesh, nó sẽ trở thành luôn 1 provisioner để có thể provision cho những con Node unprovision. Khi đã provision hết toàn bộ Node cần thiết. Nó sẽ chuyển sang mode wifi và connect với wifi thông qua smartconfig. Khi đã kết nối xong với wifi khi đó nó sẽ chuyển sang mode Bluetooth để có thể truy cập vào mạng Bluetooth Mesh. Lúc này nó sẽ subscribe vào một Group Address để có thể lấy được dữ liệu của những con Node. Sự luân chuyển giữa Wifi và Bluetooth là cần thiết và không thể hoạt động đồng thời do đều sử dụng tần số radio 2.4Ghz. Nhưng việc đó không làm gián đoạn đến việc bị mất mát dữ liệu lấy từ Bluetooth Mesh.
- Tiếp theo là các Node. Sau khi Gateway provision toàn bộ Node, tức là cho phép Node vào mạng. Các Node này sẽ cần phải được cung cấp một app bind key, có thể tạm hiểu là một khóa chung giúp các Node có thể Publish bản tin lên cùng một mesh. Có được key này xong, toàn bộ Node sẽ được set publish bản tin mong muốn vào một Group Address chung để đẩy toàn bộ dữ liệu vào đó, để cho Gateway có thể Subscribe vào và lấy toàn bộ dữ liệu về.

CHƯƠNG 6: KẾT QUẢ VÀ ĐÁNH GIÁ

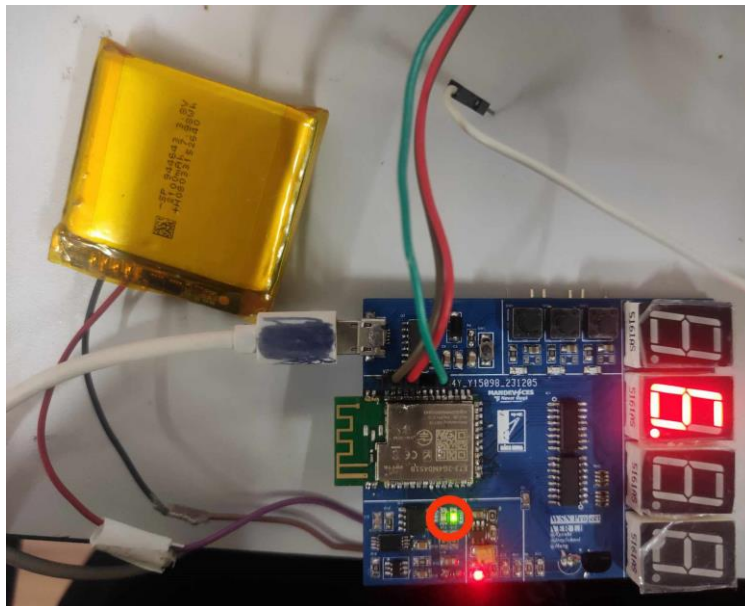
6.1 Mạch trong thực tế

Sau khi triển khai vẽ mạch nguyên lý và PCB trên phần mềm Altium chúng em đã đặt mạch in về và đặt những linh kiện cần thiết để hàn mạch, hoàn thiện sản phẩm. Dưới đây là một số hình ảnh sản phẩm trên thực tế sau khi đã hàn tất cả các khối và nạp Firmware cho nó.

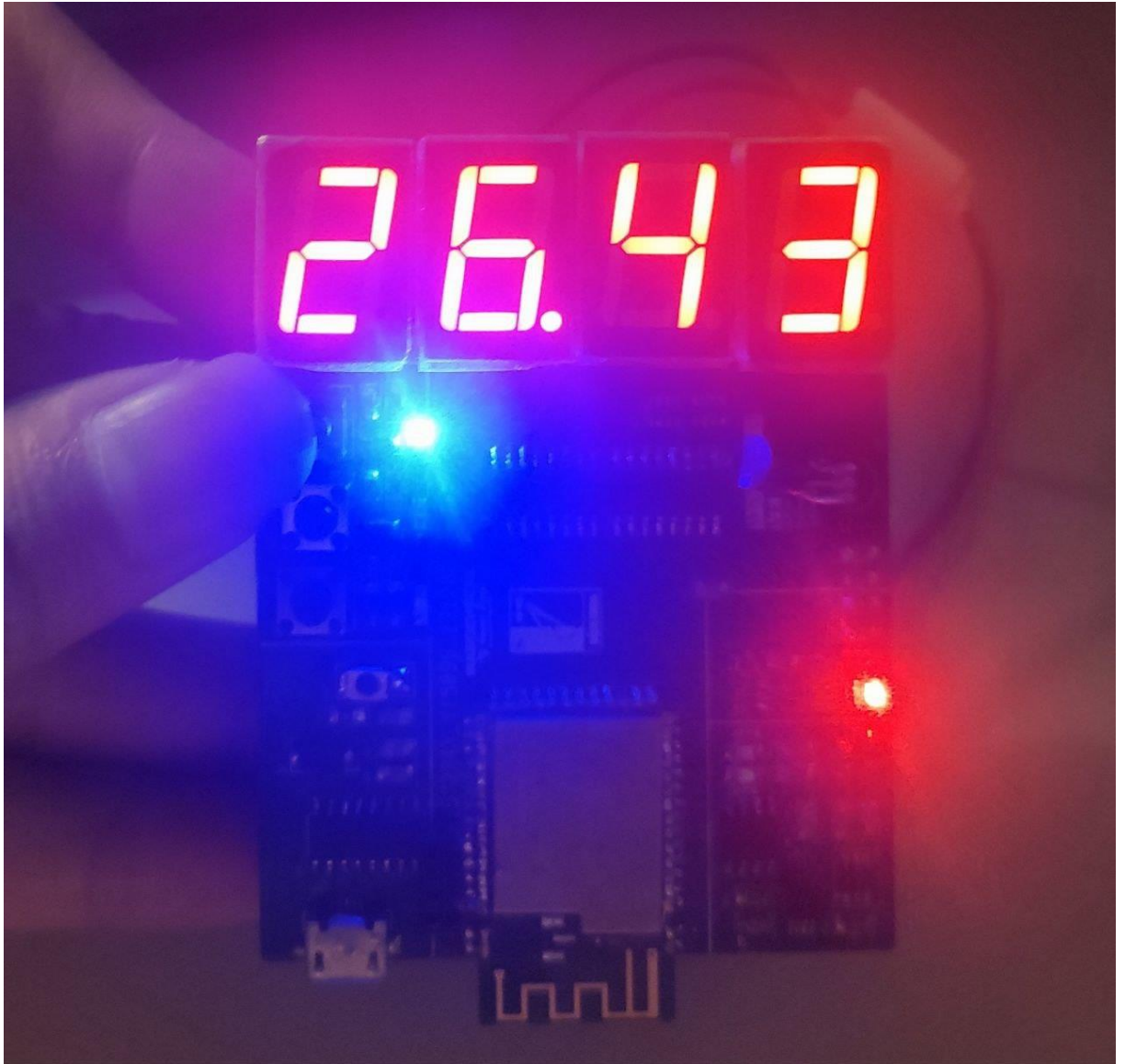


Hình 6.1: Mạch Node sau khi hoàn thiện

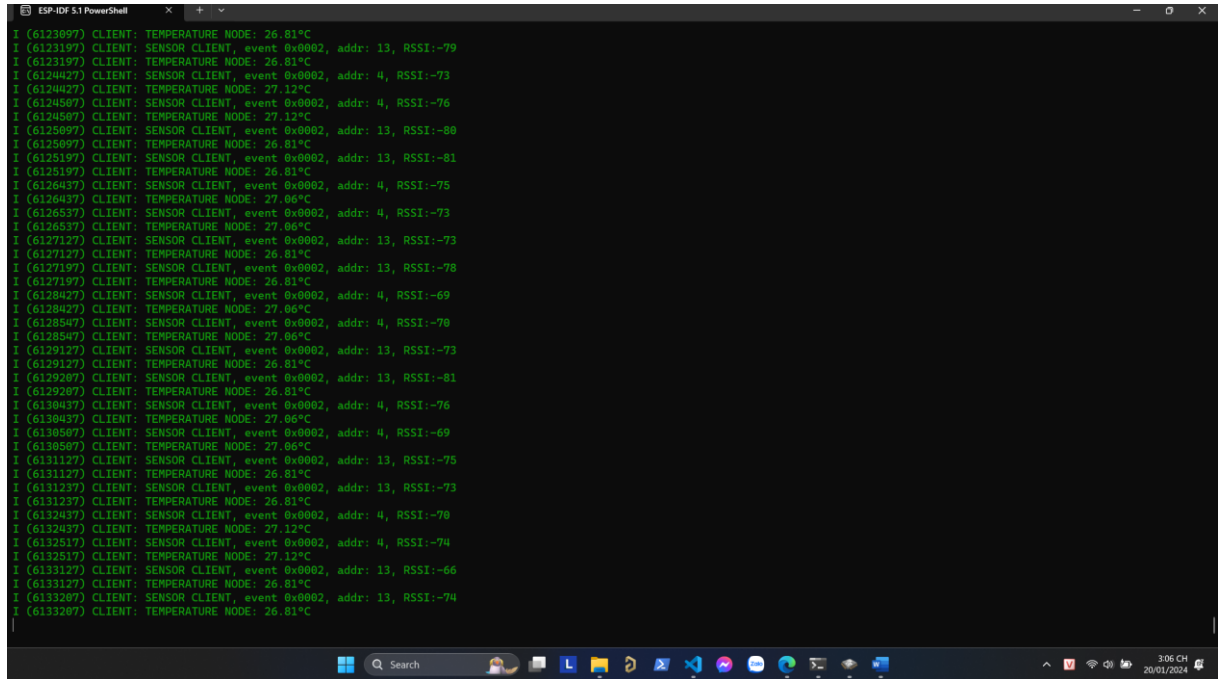
Hình 7.1 là hình ảnh của mạch Node sau khi đã hàn toàn bộ linh kiện và đang được chạy thử sử dụng nguồn là pin, phần khoanh tròn màu đỏ có 1 led đỏ đang sáng thể hiện mạch đang sạc cho pin.



Hình 6.2: Mạch Node ở trạng thái pin được sạc đầy (Led trạng thái chuyển xanh)

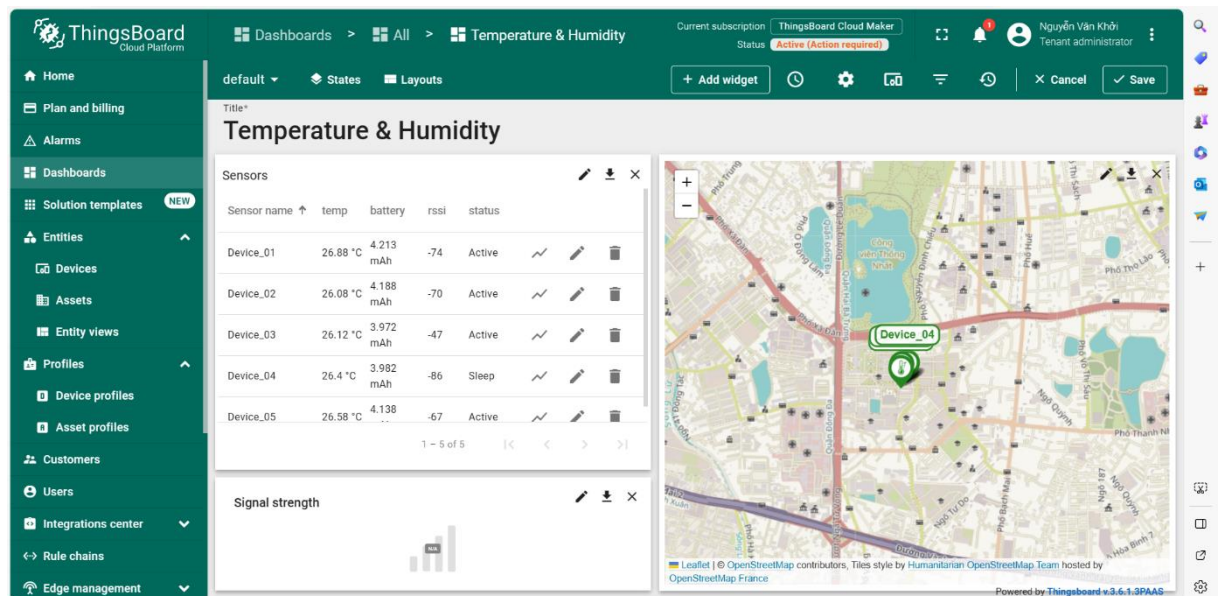


Hình 6.3: Mạch Node treenn thực tế đang hiển thị nhiệt độ



Hình 6.4: Dữ liệu log ra từ GATEWAY

Hình 7.3 là phần dữ liệu được log ra từ GATEWAY, dữ liệu được hiển thị ra là dữ liệu của 2 node có địa chỉ là 4 và 13.



Hình 6.5: Màn hình hiển thị dữ liệu từ các node trên web ThingsBoard

6.2 Phân tích và đánh giá

Tất cả các mạch Node, ở đây chúng em đang thử nghiệm 5 Node đã chạy hoàn toàn tốt, cảm biến đã đọc hoàn toàn chính xác theo thông số thiết lập, phần nút bấm và

LED 7 thanh đã có thể hiển thị và hoạt động theo yêu cầu đề bài. Các Node đều có thể vào mạng Bluetooth Mesh theo như lý thuyết và Public bản tin như mong muốn.

Khi tất cả các Node đã được provision tức là ở trong Bluetooth Mesh Network thì theo thực tế chúng em thử nghiệm, với nguồn pin thì Node sẽ dùng được trong khoảng 12 ngày đến lúc hết pin. Từ đó cho thấy toàn bộ thiết kế của bọn em đều đã đúng với dự tính lý thuyết.

Sau khi phân tích, nhóm đã chia ra các kịch bản thử nghiệm như sau:

- Ở điều kiện môi trường lý tưởng (không vật cản), các thiết bị đều nhìn thấy nhau, khoảng cách truyền tin tối thiểu giữa node và gateway là 70m, giữa các node với nhau là khoảng 60m, tỉ lệ lỗi bản tin là 0%. Khi đưa thiết bị Node càng ra xa ngoài với khoảng cách tối đa so với Gateway thì tỉ lệ lỗi bản tin nhận ở Gateway càng tăng. Ở điều kiện thử nghiệm này, những đánh giá ở mức ưu tiên 1 đều đã được đáp ứng một cách hoàn toàn và chính xác.
- Ở điều kiện môi trường có vật cản, đối với vật cản là cửa gỗ thì tối đa là 18m, đối với tường bê tông thì tối đa là 10.5m. Với điều kiện này thì mục khoảng cách sẽ không đáp ứng được mức ưu tiên như mong muốn.
- Thiết kế 2 thiết bị ở trong vùng khuất tầm nhìn sao cho không thể nhận dữ liệu với nhau. Sau đó đặt thêm 1 thiết bị thứ 3 ở giữa làm relay trung chuyển data (sao cho thiết bị đó đều tương tác được với thiết bị 1 và thiết bị 2). Nhóm thấy rằng thiết bị thứ 2 đã nhận được data từ thiết bị thứ 1, do đó đáp ứng được mô hình mạng

CHƯƠNG 7: TÀI LIỆU THAM KHẢO

- [1] D.A.Godse A.P.Godse, Microprocessors & Microcontroller Systems Technical Publications, 2008.
- [2] Lê Minh Thùy (2023), Bài giảng Mạng cảm biến không dây, Đại học Bách Khoa Hà Nội, trường Điện – Điện tử
- [3] Espressif (2022), “ESP32 Technical Reference Manual” Version 4.7.
- [4] SparkFun Electronics, “USB to Serial chip CH340,” 2022.
- [5] ElectronicWings (2022), “AVR ATmega Controllers”, từ <https://www.electronicwings.com/avr-atmega>
- [6] <https://www.geeksforgeeks.org/difference-between-bluetooth-and-zigbee/?ref=lb>
- [7] <https://rangdong.com.vn/faq/so-sanh-dac-diem-cua-3-cong-nghe-khong-day-wifizigbee-bluetooth.html>
- [8] <https://www.daihockhonggiay.com/blogs/post/cong-nghe-mang-khongday#:~:text=M%E1%BA%A1ng%20kh%C3%B4ng%20d%C3%A2y%20l%C3%A0%20m%E1%BA%A1ng,b%E1%BB%8B%20truy%E1%BB%81n%20th%C3%B4ng%20v%E1%BB%9B%20nhau.>
- [9] Datasheet LM35
- [10] Datasheet NRF52832
- [11] [1903_Mesh-Models-Overview_FINAL.pdf \(bluetooth.com\)](#)
- [12] [Mesh-Technology-Overview.pdf \(bluetooth.com\)](#)
- [13] [Tổng quan và các khái niệm quan trọng trong Bluetooth Mesh - TAPIT](#)
- [14] [Giới thiệu công nghệ BLE | VBLUno51 board \(vngiotlab.github.io\)](#)