

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA ĐIỆN TỬ VIỄN THÔNG

ĐỒ ÁN TỐT NGHIỆP
KHOA: ĐIỆN TỬ - VIỄN THÔNG

ĐỀ TÀI:
ỨNG DỤNG ESP32 TRONG HỆ THỐNG ĐỊNH VỊ VẬT THỂ

Người hướng dẫn	: TS. TRẦN VĂN LÍC TS. HỒ PHƯỚC TIẾN
Sinh viên thực hiện	: BÙI CÔNG KIÊN
Số thẻ sinh viên	: 106200265
Lớp	: 20KTMT2

Đà Nẵng, 06 /2025

NHẬN XÉT ĐỒ ÁN TỐT NGHIỆP

I. Thông tin chung:

1. Họ và tên sinh viên: Bùi Công Kiên
2. Lớp: 20KTMT2 Số thẻ SV: 106200265
3. Tên đề tài: Ứng dụng ESP32 trong hệ thống định vị vật thể
4. Người hướng dẫn: Trần Văn Líc Học hàm/ học vị: Tiến sĩ
Hồ Phước Tiến Học hàm/ học vị: Tiến sĩ

II. Nhận xét, đánh giá đồ án tốt nghiệp:

1. Về tính cấp thiết, tính mới, khả năng ứng dụng của đề tài: (điểm tối đa là 2đ)
.....
.....
2. Về kết quả giải quyết các nội dung nhiệm vụ yêu cầu của đồ án: (điểm tối đa là 4đ)
.....
.....
3. Về hình thức, cấu trúc, bố cục của đồ án tốt nghiệp: (điểm tối đa là 2đ)
.....
.....
4. Đề tài có giá trị khoa học/ có bài báo/ giải quyết vấn đề đặt ra của doanh nghiệp hoặc nhà trường: (điểm tối đa là 1đ)
.....
.....
5. Các tồn tại, thiếu sót cần bổ sung, chỉnh sửa:
.....
.....

III. Tinh thần, thái độ làm việc của sinh viên: (điểm tối đa 1đ)

.....

IV. Đánh giá:

1. Điểm đánh giá:/10 (lấy đến 1 số lẻ thập phân)
2. Đề nghị: ☐ Được bảo vệ đồ án ☐ Bổ sung để bảo vệ ☐ Không được bảo vệ

Đà Nẵng, ngày tháng năm 2025

Người hướng dẫn

NHẬN XÉT PHẢN BIỆN ĐỒ ÁN TỐT NGHIỆP

I. Thông tin chung:

1. Họ và tên sinh viên: Bùi Công Kiên
2. Lớp: 20KTMT2 Số thẻ SV:106200265
3. Tên đề tài: Ứng dụng ESP32 trong hệ thống định vị vật thể
4. Người phản biện: Học hàm/ học vị:.....

II. Nhận xét, đánh giá đồ án tốt nghiệp:

TT	Các tiêu chí đánh giá	Điểm tối đa	Điểm đánh giá
1	Sinh viên có phương pháp nghiên cứu phù hợp, giải quyết đủ nhiệm vụ đồ án được giao	80	
1a	- Tính mới (nội dung chính của ĐATN có những phần mới so với các ĐATN trước đây). - Đề tài có giá trị khoa học, công nghệ; có thể ứng dụng thực tiễn.	15	
1b	- Kỹ năng giải quyết vấn đề; hiểu, vận dụng được kiến thức cơ bản, cơ sở, chuyên ngành trong vấn đề nghiên cứu. - Chất lượng nội dung ĐATN (thuyết minh, bản vẽ, chương trình, mô hình,...).	50	
1c	- Có kỹ năng vận dụng thành thạo các phần mềm ứng dụng trong vấn đề nghiên cứu; - Có kỹ năng đọc, hiểu tài liệu bằng tiếng nước ngoài ứng dụng trong vấn đề nghiên cứu; - Có kỹ năng làm việc nhóm;	15	
2	Kỹ năng viết:	20	
2a	- Bố cục hợp lý, lập luận rõ ràng, chặt chẽ, lời văn súc tích	15	
2b	- Thuyết minh đồ án không có lỗi chính tả, in ấn, định dạng	5	
3	Tổng điểm đánh giá theo thang 100:		
	Quy về thang 10 (lấy đến 1 số lẻ)		

- Các tồn tại, thiếu sót cần bổ sung, chỉnh sửa:
- Câu hỏi đề nghị sinh viên trả lời trong buổi bảo vệ:.....
- Đề nghị: ☐ Được bảo vệ đồ án ☐ Bổ sung để bảo vệ ☐ Không được bảo vệ

Đà Nẵng, ngày tháng năm 2025

Người phản biện

TÓM TẮT

Tên đề tài: “Ứng dụng ESP32 trong hệ thống định vị vật thể”

Sinh viên thực hiện: Bùi Công Kiên

Số thẻ SV: 106200265

Lớp: 20KTMT2

Với đề tài “Ứng dụng ESP32 trong hệ thống định vị vật thể”, để thực hiện được đề tài này đồ án sẽ bao gồm 4 chương như sau:

Chương 1: Tổng quan về đề tài

Chương 2: Cơ sở lý thuyết

Chương 3: Phương pháp thực hiện

Chương 4: Kết quả và đánh giá hệ thống

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Họ tên sinh viên: Bùi Công Kiên Số thẻ sinh viên: 106200265

Lớp: 20KTMT2 Khoa: Điện tử Viễn thông Ngành: Kỹ thuật máy tính

1. Tên đề tài đồ án:

Ứng dụng ESP32 trong hệ thống định vị vật thể

2. Đề tài thuộc diện: ☐ Có ký kết thỏa thuận sở hữu trí tuệ đối với kết quả thực hiện

3. Các số liệu và dữ liệu ban đầu:

Sai số chỉ số RSSI đo được $< 10\%$, khoảng cách truyền nhận tín hiệu tối đa 20m, khả năng nhận biết và đưa ra khoảng cách chính xác $> 90\%$.

4. Nội dung các phần thuyết minh và tính toán:

Chương 1: Tổng quan về đề tài

Chương 2: Cơ sở lý thuyết

Chương 3: Phương pháp thực hiện

Chương 4: Kết quả và đánh giá hệ thống

5. Các bản vẽ, đồ thị (ghi rõ các loại và kích thước bản vẽ):

- Bản vẽ sơ đồ khối tổng quát hệ thống.
- Bản vẽ sơ đồ khối hệ thống.
- Bản vẽ sơ đồ thuật toán hệ thống.

6. Họ tên người hướng dẫn:

TS. Trần Văn Líc

TS. Hồ Phước Tiến

7. Ngày giao nhiệm vụ đồ án: 02/04/2025

8. Ngày hoàn thành đồ án: 09/06/2025

Đà Nẵng, ngày tháng năm 2025

Trưởng Bộ môn.....

Người hướng dẫn

LỜI NÓI ĐẦU

Tôi xin bày tỏ lòng biết ơn sâu sắc đến Ban Giám Hiệu Trường Đại học Bách Khoa – Đại học Đà Nẵng vì đã cung cấp cho tôi một môi trường học tập với cơ sở vật chất khang trang.

Tiếp đó, tôi xin gửi lời cảm ơn chân thành đến các thầy cô giảng viên trong khoa Điện tử - Viễn thông, những người đã hướng dẫn và dạy dỗ tôi suốt bốn năm qua. Thầy cô không những là người truyền đạt kiến thức mà còn là nguồn động viên to lớn, luôn giúp đỡ tôi một cách nhiệt tình và tận tâm.

Đặc biệt, tôi xin chân thành cảm ơn giảng viên Trần Văn Líc đã tận tình hướng dẫn và hỗ trợ tôi hoàn thành đề tài này.

Tôi cũng xin gửi lời cảm ơn đến các bạn học đã đồng hành, chia sẻ và giúp đỡ tôi trong suốt thời gian học tập tại đây.

Trong suốt quá trình học tập và thực hiện đồ án tốt nghiệp, tôi không thể tránh khỏi những thiếu sót. Rất mong nhận được những góp ý quý báu từ các thầy cô và các bạn để đồ án của tôi được hoàn thiện hơn.

Cuối cùng, tôi xin chúc quý thầy cô và các bạn sức khỏe dồi dào và ngày càng thành công trong cuộc sống.

CAM ĐOAN

Tôi xin cam đoan nội dung trong đồ án này “Ứng dụng ESP32 trong hệ thống định vị vật thể” là do tôi thực hiện dưới sự hướng dẫn trực tiếp của thầy Trần Văn Líc. Các tham khảo trong đồ án này đều được trích dẫn rõ ràng tên tác giả, tên công trình, địa điểm công bố. Mọi kết quả trình bày trong báo cáo đều trung thực, không sao chép toàn bộ từ bất kì nguồn nào. Nếu có những sao chép không hợp lệ, vi phạm liên chính học thuật, tôi xin chịu hoàn toàn trách nhiệm.

Đà Nẵng, ngày tháng năm 2025

Sinh viên thực hiện



Bùi Công Kiên

MỤC LỤC

TÓM TẮT	I
NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP	I
LỜI NÓI ĐẦU.....	II
CAM ĐOAN	III
DANH SÁCH CÁC BẢNG, HÌNH VẼ.....	VI
DANH SÁCH CÁC KÍ HIỆU, CHỮ VIẾT TẮT	VII
MỞ ĐẦU	1
CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI.....	2
1.1 Giới thiệu chương.....	2
1.2 Bối cảnh và vấn đề đặt ra	2
1.3 Các giải pháp hiện có	3
1.4 Phương pháp đề xuất	5
1.5 Mục tiêu đề tài	5
1.6 Kết luận chương	6
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT.....	7
2.1 Giới thiệu chương.....	7
2.2 Tổng quan về các công cụ và phương pháp sử dụng	7
2.2.1 Chỉ số RSSI	7
2.2.2 Mô hình suy hao đường truyền Log-Distance Path Loss Model	9
2.2.3 Cập nhật dữ liệu RSSI khoảng cách đo được lên Google Trang tính	12
2.3 Tìm hiểu phần cứng và phần mềm của hệ thống.....	13
2.3.1 Vi điều khiển ESP32	13
2.3.2 Nền tảng phát triển Arduino IDE	14
2.3.3 ESP32 Google Sheets	15
2.3.4 Các thư viện sử dụng.....	16
2.4 Kết luận chương	17
CHƯƠNG 3: PHƯƠNG PHÁP THỰC HIỆN.....	18
3.1 Giới thiệu chương.....	18
3.2 Các yêu cầu về tính năng và kỹ thuật.....	19
3.3 Quy trình thao tác thực hiện	21
3.4 Kết luận chương	22
CHƯƠNG 4: KẾT QUẢ VÀ ĐÁNH GIÁ HỆ THỐNG	23
4.1 Giới thiệu chương.....	23
4.2 Kết quả thực hiện.....	23
4.3 Cơ sở tiêu chuẩn đánh giá	29
4.4 Thực hiện đánh giá	30
4.4.1 Thực hiện kiểm tra và đánh giá độ chính xác.....	30

4.4.2 Thực hiện kiểm tra và đánh giá thời gian phản ứng.....	47
4.4.3 Thực hiện kiểm tra và đánh giá độ bao phủ	47
4.5 Kết luận chương	48
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	49
TÀI LIỆU THAM KHẢO	53
PHỤ LỤC	55

DANH SÁCH CÁC BẢNG, HÌNH VẼ

Bảng 1.1 Các giải pháp hệ thống định vị.....	3
Bảng 2.1 Bảng phân loại và ý nghĩa của các giá trị RSSI phổ biến [1].....	8
Bảng 4.1 Bảng dữ liệu đánh giá độ chính xác của trường hợp khoảng cách thực tế 1m	30
Bảng 4.2 Bảng dữ liệu đánh giá độ chính xác của trường hợp khoảng cách thực tế 2m	33
Bảng 4.3 Bảng dữ liệu đánh giá độ chính xác của trường hợp khoảng cách thực tế 3m	37
Bảng 4.4 Bảng dữ liệu đánh giá độ chính xác của trường hợp khoảng cách thực tế 4m	40
Bảng 4.5 Bảng dữ liệu đánh giá độ chính xác của trường hợp khoảng cách thực tế 5m	44
Hình 2.1 Bảng mạng trực quan hóa vị trí của các mạng được chọn trên băng tần 2,4 hoặc 5 GHz so với các mạng khác và cường độ tín hiệu của từng mạng [1].....	8
Hình 2.2 Biểu đồ cường độ tín hiệu theo thời gian cho thấy cường độ tín hiệu mạng của bạn thay đổi như thế nào khi bạn di chuyển trong phòng hoặc văn phòng [1].....	9
Hình 2.3 Esp32-devkitc [8].....	13
Hình 2.4 Sơ đồ chân của esp32-wroom-32 [8].....	14
Hình 2.5 Phần mềm arduino ide.....	14
Hình 2.6 Esp32 google sheet.....	15
Hình 3.1 Sơ đồ khối tổng quát của hệ thống định vị vật thể ứng dụng esp32.....	18
Hình 4.1 Sơ đồ khối.....	23
Hình 4.2 Sơ đồ thuật toán.....	24
Hình 4.3 Esp32 cách router cố định 0,5m.....	25
Hình 4.4 Esp32 cách router cố định 1m.....	25
Hình 4.5 Esp32 cách router cố định 2m.....	26
Hình 4.6 Esp32 cách router cố định 3m.....	26
Hình 4.7 Esp32 cách router cố định 4m.....	27
Hình 4.8 Esp32 cách router cố định 5m.....	27
Hình 4.9 Biểu đồ dạng sóng thể hiện sai số qua dao động sóng của các trường hợp (1m, 2m, 3m, 4m, 5m).....	28
Hình 4.10 Biểu đồ CDF phân phối chuẩn dữ liệu sai số(%).....	29
Hình 4.11 Vị trí esp32 cách router cố định 20m.....	48

DANH SÁCH CÁC KÍ HIỆU, CHỮ VIẾT TẮT

Kí hiệu, chữ viết tắt	Giải thích
IoT	Internet of Things
ESP32	Electronic Semiconductor Platform 32-bit
IDE	Integrated Development Environment
API	Application Programming Interface
NTP	Network Time Protocol
IPS	Indoor Positioning System
RSSI	Received Signal Strength Indicator
AP	Access Point
RFID	Radio-Frequency Identification
MQTT	Message Queuing Telemetry Transport
EMA	Exponential Moving Average
SD	Secure Digital
BLE	Bluetooth Low Energy
IMU	Inertial Measurement Unit
URL	Uniform Resource Locator
CDF	Cumulative Distribution Function
HTTP	HyperText Transfer Protocol
JSON	JavaScript Object Notation
GPS	Global Positioning System
GDOP	Geometry Dilution of Precision

UWB	Ultra-Wideband
NVS	Non-Volatile Storage
ToF	Time of Flight
IR	Infrared Ray
ID	Identification

MỞ ĐẦU

Ứng dụng ESP32 trong hệ thống định vị vật thể đóng vai trò quan trọng trong việc điều hướng robot tự hành trong nhà kho thông minh, theo dõi tài sản giá trị, đến hỗ trợ người dùng tìm kiếm đồ vật thất lạc hay xây dựng các giải pháp đỗ xe thông minh. Trong kỷ nguyên IoT phát triển mạnh mẽ, việc xác định vị trí của vật thể trong không gian, đặc biệt là trong nhà, đã trở thành một thách thức lớn. Hệ thống định vị toàn cầu (GPS) hoạt động kém hiệu quả trong môi trường trong nhà do tín hiệu bị cản trở bởi tường và các vật cản khác. Điều này đã thúc đẩy sự ra đời của nhiều công nghệ định vị trong nhà (Indoor Positioning System - IPS), trong đó, việc ứng dụng chỉ số cường độ tín hiệu nhận được (Received Signal Strength Indicator - RSSI) từ các tín hiệu Wi-Fi hoặc Bluetooth là một trong những phương pháp phổ biến và tiết kiệm chi phí nhất.

Đề tài này tập trung vào việc nghiên cứu và triển khai ứng dụng ESP32 trong hệ thống định vị khoảng cách vật thể sử dụng phương pháp RSSI. Với khả năng tích hợp Wi-Fi cùng hiệu suất xử lý mạnh mẽ, ESP32 là một nền tảng lý tưởng để xây dựng các hệ thống định vị dựa trên RSSI. Mặc dù phương pháp RSSI đối mặt với những thách thức về độ chính xác do sự biến động của tín hiệu trong môi trường trong nhà, đề tài sẽ xem xét các kỹ thuật xử lý tín hiệu và các thuật toán nâng cao để cải thiện độ tin cậy của hệ thống định vị.

Mục đích của đồ án tốt nghiệp này là nghiên cứu “Ứng dụng ESP32 trong hệ thống định vị vật thể”. Nội dung đề tài được trình bày trong 4 chương sau:

Chương 1: Tổng quan về đề tài

Chương 2: Cơ sở lý thuyết

Chương 3: Phương pháp thực hiện

Chương 4: Kết quả và đánh giá hệ thống

Do thời gian và kiến thức còn hạn chế nên tôi không thể tránh khỏi những thiếu sót, rất mong nhận được những sự bổ sung, góp ý của quý thầy cô cũng như các bạn để đề tài này được hoàn thiện hơn.

Đà Nẵng, ngày tháng năm 2025

Sinh viên thực hiện



Bùi Công Kiên

CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

1.1 Giới thiệu chương

Vấn đề định vị vật thể (Object Localization) đang ngày càng trở nên cấp thiết và đóng vai trò không thể thiếu trong cuộc sống hiện đại, từ những tiện ích hàng ngày đến các ứng dụng công nghiệp và an ninh phức tạp. Định vị vật thể không chỉ là một công nghệ tiện ích mà đã trở thành một phần không thể thiếu, mang lại những lợi ích to lớn về sự tiện lợi, an toàn, hiệu quả kinh tế và khả năng phát triển trong mọi lĩnh vực của cuộc sống hiện đại. Nó giúp chúng ta quản lý thế giới xung quanh một cách thông minh và kết nối hơn.

Mục tiêu chính của chương là đề xuất ứng dụng của ESP32 để định khoảng cách của vật thể bằng một phương pháp đo cường độ sóng Wi-Fi để xác định khoảng cách vật thể.

1.2 Bối cảnh và vấn đề đặt ra

Trong kỷ nguyên của cuộc cách mạng công nghiệp 4.0 và sự bùng nổ của Internet of Things (IoT), nhu cầu về việc quản lý và theo dõi các vật thể, tài sản, hoặc thậm chí con người trong không gian thực ngày càng trở nên cấp thiết. Khả năng biết được vị trí chính xác của một đối tượng có thể mang lại lợi ích to lớn trong nhiều lĩnh vực:

- Trong môi trường công nghiệp và logistics: Các nhà máy, kho bãi rộng lớn chứa hàng ngàn sản phẩm, thiết bị, và phương tiện vận chuyển. Việc tìm kiếm một công cụ, một lô hàng cụ thể, hoặc theo dõi luồng nguyên vật liệu là một thách thức lớn, ảnh hưởng trực tiếp đến hiệu quả sản xuất và chuỗi cung ứng.
- Trong lĩnh vực y tế: Các bệnh viện lớn thường đối mặt với việc quản lý tài sản y tế (giường bệnh, xe đẩy, thiết bị chuyên dụng), cũng như theo dõi vị trí của bệnh nhân (đặc biệt là bệnh nhân sa sút trí tuệ) và nhân viên để tối ưu hóa quy trình chăm sóc.
- Trong môi trường bán lẻ và dịch vụ: Theo dõi hành vi khách hàng, quản lý tồn kho trong các cửa hàng lớn, hoặc định vị thiết bị phục vụ trong các khu nghỉ dưỡng.
- Trong các tòa nhà thông minh và văn phòng: Tối ưu hóa việc sử dụng không gian, quản lý tài sản văn phòng, hoặc thậm chí hỗ trợ tìm đường trong các khu phức

hợp lớn.

- Giám sát an toàn: Trong các môi trường nguy hiểm hoặc rộng lớn, việc biết vị trí của nhân viên có thể cải thiện đáng kể an toàn và khả năng ứng phó khẩn cấp.

Trong khi các hệ thống định vị ngoài trời (như GPS - Global Positioning System) đã rất phát triển và được sử dụng rộng rãi, chúng lại gặp phải những hạn chế nghiêm trọng khi hoạt động trong nhà hoặc các khu vực có nhiều vật cản (nhà cao tầng, tầng hầm, nhà máy kín). Tín hiệu GPS quá yếu để xuyên qua các vật cản này, hoặc bị phản xạ, gây ra sai số lớn hoặc mất tín hiệu hoàn toàn.

Vấn đề đặt ra: Mặc dù nhu cầu về IPS là rõ ràng, việc xây dựng một hệ thống hiệu quả lại đối mặt với nhiều thách thức, dẫn đến các vấn đề cụ thể mà đề tài này muốn giải quyết.

1.3 Các giải pháp hiện có

Bảng 1.1 Các giải pháp hệ thống định vị

Tên phương pháp	Nguyên lý	RSSI-based	Ưu điểm	Nhược điểm
Hệ thống định vị dựa trên Wi-Fi	Chủ yếu dựa vào việc sử dụng cường độ tín hiệu Wi-Fi (RSSI - Received Signal Strength Indicator) từ các điểm truy cập Wi-Fi (Access Points - APs) đã biết vị trí để ước tính vị trí của một thiết bị	Đo RSSI từ nhiều điểm truy cập Wi-Fi (Access Points - APs) có vị trí đã biết	Tận dụng hạ tầng sẵn có (giảm chi phí), phổ biến, không yêu cầu thiết bị chuyên dụng đắt tiền cho thiết bị nhận	Độ chính xác tương đối thấp (vài mét đến hàng chục mét), rất nhạy cảm với nhiễu, đa đường, và sự thay đổi môi trường (người di chuyển, đồ đạc)
Hệ thống định vị dựa trên Bluetooth	Sử dụng các thiết bị phát tín hiệu Bluetooth năng lượng thấp (BLE	Tương tự Wi-Fi RSSI, đo cường độ tín	Tiêu thụ năng lượng thấp (cho beacon và thiết bị	Giới hạn về độ chính xác tương tự Wi-Fi RSSI

	Beacons) được đặt cố định. Thiết bị nhận (điện thoại thông minh, ESP32) quét RSSI từ các beacon này	hiệu từ các beacon	nhận), beacon nhỏ gọn, chi phí tương đối thấp	
Hệ thống định vị dựa trên Ultra-Wideband	Sử dụng các xung vô tuyến băng thông rộng ngắn để đo thời gian truyền tín hiệu (Time of Flight - ToF) giữa các thiết bị phát (anchors) và thiết bị nhận (tags)	Quét và thu thập cường độ tín hiệu RSSI hiện tại từ các anchor UWB xung quanh	Độ chính xác rất cao (thường ở mức centimet), ít bị ảnh hưởng bởi đa đường và vật cản hơn các công nghệ khác	Chi phí phần cứng cao hơn đáng kể (cho cả anchors và tags), không có hạ tầng sẵn có, vùng phủ sóng nhỏ hơn yêu cầu nhiều thiết bị
Hệ thống định vị dựa trên Hồng ngoại (IR) / Sóng siêu âm	Sử dụng các thiết bị phát/thu hồng ngoại hoặc sóng siêu âm để đo khoảng cách hoặc xác định vị trí	Tương tự như Wi-Fi, một cơ sở dữ liệu sẽ được xây dựng trước bằng cách thu thập các giá trị IR RSSI tại các điểm mẫu đã biết tọa độ	Có thể đạt độ chính xác cao trong điều kiện lý tưởng (đường ngắm - Line of Sight)	Yêu cầu đường ngắm trực tiếp, dễ bị chặn, phạm vi hoạt động thường ngắn
Hệ thống định vị dựa trên RFID (Radio-Frequency Identification)	Sử dụng các đầu đọc (readers) để phát hiện và đọc thông tin từ các thẻ RFID gắn trên vật thể	Tại mỗi điểm mẫu, hệ thống sẽ đọc và ghi lại cường độ tín hiệu RSSI	Tốt cho việc nhận dạng và theo dõi tài sản ở cự ly gần (passive)	Phạm vi đọc giới hạn, không cung cấp vị trí liên tục mà chỉ xác định vật

		nhận được từ các đầu đọc RFID (hoặc từ các thẻ RFID chủ động được đặt ở các vị trí cố định) trong khu vực	hoặc tâm trung (active)	thẻ đang ở trong vùng phủ sóng của đầu đọc
--	--	-----------------------------------------------------------------------------------------------------------	-------------------------	--------------------------------------------

1.4 Phương pháp đề xuất

Từ những vấn đề nêu trên, đề tài này đặt ra phương pháp sau:

- Nghiên cứu và triển khai một hệ thống định vị vật thể chi phí thấp sử dụng ESP32 và Wi-Fi RSSI để tận dụng tối đa tài nguyên sẵn có.
- Tìm hiểu và áp dụng Mô hình suy hao đường truyền Log-Distance để ước tính khoảng cách, đồng thời đề xuất các phương pháp xử lý dữ liệu RSSI để giảm thiểu ảnh hưởng của nhiễu và biến động, nhằm cải thiện độ tin cậy của ước tính khoảng cách.
- Xây dựng một cơ chế cập nhật và lưu trữ dữ liệu RSSI và khoảng cách đo được một cách trực tiếp, đơn giản và hiệu quả lên một nền tảng dễ tiếp cận như Google Sheets, nhằm tạo điều kiện thuận lợi cho việc theo dõi, phân tích và gỡ lỗi mà không cần hạ tầng phức tạp.

Mục tiêu cuối cùng là cung cấp một giải pháp thực tiễn cho bài toán định vị trong nhà ở mức độ "đủ chính xác" cho các ứng dụng không yêu cầu độ chính xác cao tuyệt đối, với chi phí và độ phức tạp triển khai tối thiểu.

1.5 Mục tiêu đề tài

Mục tiêu của đề tài là nghiên cứu, thiết kế, phát triển khai báo và đánh giá một hệ thống định vị vật thể hiệu quả về chi phí và dễ dàng phát triển khai báo, sử dụng vi điều khiển ESP32 và kỹ thuật RSSI dựa trên mô hình suy hao đường truyền Log-Distance Path Loss Model, đồng thời cung cấp khả năng ghi nhật ký và trực tuyến hóa định vị dữ

liệu một cách tiện lợi thông qua Google Sheets. Đề tài được thực hiện với các mục tiêu chi tiết sau:

- Thu thập dữ liệu RSSI đáng tin cậy.
- Xử lý dữ liệu RSSI để tính toán khoảng cách từ giá trị RSSI.
- Quản lý mạng kết nối và đồng bộ hóa: duy trì kết nối Wi-Fi ổn định, đồng bộ hóa thời gian chính xác.
- Giao tiếp dữ liệu trực tiếp với Google Sheets: chuẩn hóa dữ liệu, gửi dữ liệu trực tiếp đến Google Sheets, quản lý xác thực (tài khoản dịch vụ).
- Hiệu suất hoạt động và độ tin cậy cao.

1.6 Kết luận chương

Chương này đã nêu rõ vấn đề trong kỷ nguyên của cuộc cách mạng công nghiệp 4.0 và sự bùng nổ của Internet of Things (IoT), nhu cầu về việc quản lý và theo dõi các vật thể, tài sản, hoặc thậm chí con người trong không gian thực ngày càng trở nên cấp thiết. Trong khi các hệ thống định vị ngoài trời (như GPS - Global Positioning System) đã rất phát triển và được sử dụng rộng rãi, chúng lại gặp phải những hạn chế nghiêm trọng khi hoạt động trong nhà hoặc các khu vực có nhiều vật cản (nhà cao tầng, tầng hầm, nhà máy kín). Tín hiệu GPS quá yếu để xuyên qua các vật cản này, hoặc bị phản xạ, gây ra sai số lớn hoặc mất tín hiệu hoàn toàn.

Từ đó đưa ra phương pháp Hệ thống định vị sử dụng ESP32 và Wi-Fi RSSI áp dụng mô hình suy hao đường truyền Log-Distance để ước tính khoảng cách chi phí thấp, với tài nguyên sẵn có và dễ dàng thực hiện, phù hợp mục đích nghiên cứu, ứng dụng trong phạm vi nhỏ.

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

2.1 Giới thiệu chương

Từ những yêu cầu đặt ra về một Hệ thống định vị vật thể ứng dụng ESP32 tôi sẽ vẽ và trình bày sơ đồ khối của hệ thống từ đó lấy căn cứ, tìm hiểu đề xuất và liệt kê các thiết bị, các nền tảng lập trình và lưu trữ dữ liệu, các phương pháp sử dụng.

2.2 Tổng quan về các công cụ và phương pháp sử dụng

2.2.1 Chỉ số RSSI

RSSI là một giá trị đại diện cho cường độ của tín hiệu Wi-Fi, Bluetooth, hoặc bất kỳ tín hiệu vô tuyến nào khác mà thiết bị nhận được. Nó không phải là một đơn vị chuẩn hóa theo watt hoặc decibel-milliwatt (dBm) một cách tuyệt đối trên tất cả các thiết bị, nhưng thường được biểu thị bằng dBm.

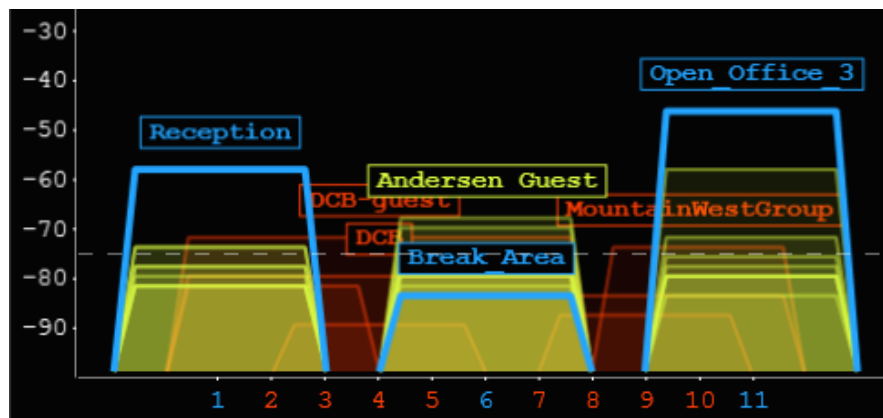
RSSI được đo bởi card mạng không dây hoặc module thu phát tín hiệu của thiết bị nhận. Nó phản ánh tổng công suất của tín hiệu sóng mang cộng với bất kỳ nhiễu nào khác có trong kênh. Trong hệ thống không dây, RSSI thường nằm trong khoảng từ -10 dBm (tín hiệu cực mạnh) đến -100 dBm hoặc thấp hơn (tín hiệu rất yếu hoặc không thể kết nối), giá trị RSSI càng gần 0 dBm thì tín hiệu càng tốt.

Giá trị RSSI (dBm)	Mô tả Tín hiệu	Chất lượng Kết nối	Ghi chú
-30 đến -40	Cực tốt	Xuất sắc	Gần như hoàn hảo, thường chỉ đạt được khi ở rất gần Access Point (AP) và không có vật cản.
-41 đến -50	Rất tốt	Rất ổn định	Tín hiệu tuyệt vời, cho phép truyền dữ liệu tốc độ cao nhất.
-51 đến -60	Tốt	Ổn định	Tín hiệu đáng tin cậy, phù hợp cho hầu hết các ứng dụng, bao gồm streaming video HD.

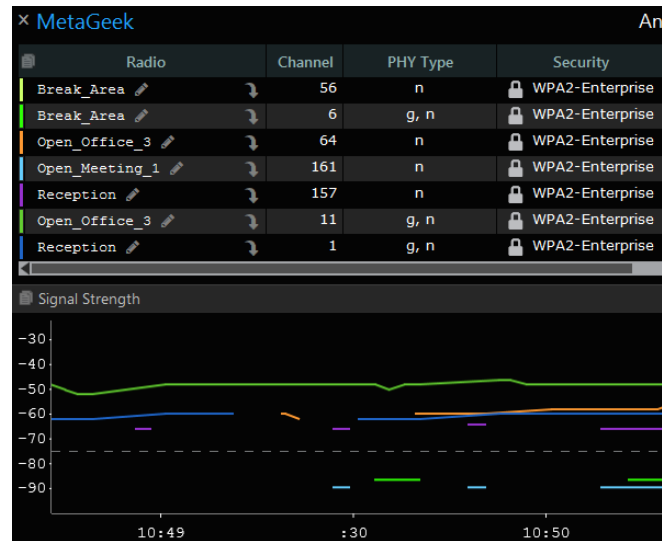
-61 đến -70	Trung bình	Chấp nhận được	Tín hiệu vẫn đủ mạnh cho các hoạt động duyệt web, email, một số streaming cơ bản. Có thể gặp hiện tượng giảm tốc độ ở các ứng dụng đòi hỏi băng thông cao.
-71 đến -80	Yếu	Kém/Gián đoạn	Kết nối có thể bị gián đoạn, tốc độ rất chậm. Không phù hợp cho các ứng dụng đòi hỏi độ tin cậy cao.
-81 đến -90	Rất yếu	Không ổn định	Kết nối gần như không thể thiết lập hoặc thường xuyên bị ngắt.
Dưới -90	Không đáng kể	Không thể kết nối	Tín hiệu quá yếu để thiết lập kết nối ổn định.

Bảng 2.1 Bảng phân loại và ý nghĩa của các giá trị RSSI phổ biến [1]

Để giúp tận dụng phép đo cường độ tín hiệu hiệu quả nhất để có thể đưa ra quyết định lập kế hoạch kênh, hiển thị cường độ tín hiệu theo hai cách.



Hình 2.1 Bảng Mạng trực quan hóa vị trí của các mạng được chọn trên băng tần 2,4 hoặc 5 GHz so với các mạng khác và cường độ tín hiệu của từng mạng [1]



Hình 2.2 Biểu đồ Cường độ tín hiệu theo thời gian cho thấy cường độ tín hiệu mạng của bạn thay đổi như thế nào khi bạn di chuyển trong phòng hoặc văn phòng [1]

2.2.2 Mô hình suy hao đường truyền Log-Distance Path Loss Model

“Mô hình suy hao đường truyền Log-Distance Path Loss Model” là một mô hình thực nghiệm (empirical model) hoặc bán thực nghiệm, dự đoán cường độ tín hiệu trung bình nhận được (hoặc suy hao tín hiệu) theo khoảng cách giữa thiết bị phát và thiết bị thu. Nó được gọi là "Log-Distance" vì sự suy hao tín hiệu được mô tả bằng một hàm logarit của khoảng cách.

Mục đích chính của mô hình này là:

- Ước tính cường độ tín hiệu: Dự đoán RSSI tại một khoảng cách cụ thể.
- Ước tính khoảng cách: Từ RSSI đo được, suy ra khoảng cách.
- Lập kế hoạch mạng: Giúp các kỹ sư dự đoán vùng phủ sóng và hiệu suất của mạng không dây.

Công thức của Mô hình suy hao đường truyền Log-Distance Path Loss Model
Công thức tổng quát thường được biểu diễn theo dB (decibel) hoặc dBm (decibel-milliwatts) cho cường độ tín hiệu nhận được:

Công thức cho cường độ tín hiệu nhận được (PR) (thường là RSSI):

$$PR_d(\text{dBm}) = PT(\text{dBm}) - PL_d(\text{dBm})$$

Trong đó PL_d là suy hao đường truyền tại khoảng cách d , được tính bằng công thức Log-Distance:

$$PL_d(dB) = PL_{d_0}(dB) + 10n \log_{10} \left(\frac{d}{d_0} \right)$$

Kết hợp hai công thức trên, ta có công thức thường thấy hơn khi sử dụng RSSI:

$$RSSI(-dBm) = RSSId_0 - 10n * \log_{10} \left(\frac{d}{d_0} \right)$$

Giải thích các thành phần:

- RSSI (hoặc PR_d): Cường độ tín hiệu nhận được tại khoảng cách d từ máy phát (đơn vị: dBm). Đây là giá trị mà thiết bị thu (ví dụ: ESP32) đo được.
- RSSId0 (hoặc $PT - PL_{d_0}$): Cường độ tín hiệu tham chiếu tại một khoảng cách chuẩn d_0 (đơn vị: dBm).
 - d_0 thường được chọn là 1 mét (m) trong môi trường trong nhà/thành phố hoặc 1 km trong môi trường nông thôn/xa.
 - RSSId0 là giá trị RSSI đo được tại khoảng cách d_0 . Giá trị này thường được xác định bằng cách đo thực nghiệm tại vị trí d_0 trong môi trường thực tế, vì nó đã bao gồm các yếu tố suy hao ban đầu (như tổn hao anten, tổn hao đường truyền gần).
- n : Hệ số suy hao đường truyền (Path Loss Exponent). Đây là một tham số quan trọng, đặc trưng cho môi trường truyền sóng và phản ánh mức độ suy giảm cường độ tín hiệu theo khoảng cách:
 - $n=2$: Trong môi trường không gian tự do (free space), tín hiệu chỉ suy hao do khoảng cách.
 - $n>2$: Trong các môi trường thực tế, tín hiệu gặp vật cản (tường, cây cối, con người) và bị hấp thụ, phản xạ, tán xạ, dẫn đến suy hao lớn hơn.
 - Trong nhà: n thường từ 2.5 đến 4 hoặc hơn.
 - Khu vực đô thị: n có thể từ 2.7 đến 5.
 - Nông thôn: n có thể gần với 2.
 - Giá trị của n cũng thường được xác định thông qua các phép đo thực nghiệm tại hiện trường.
- d : Khoảng cách từ máy phát đến máy thu (đơn vị: mét hoặc kilômét, tùy thuộc vào d_0).
- d_0 : Khoảng cách tham chiếu (đơn vị: mét hoặc kilômét).

Nguyên lý và Giả định:

- Nguyên lý Logarit: Mô hình này giả định rằng sự suy hao tín hiệu theo khoảng cách có mối quan hệ logarit, tức là sự suy hao tăng lên một lượng cố định (theo dB) khi khoảng cách tăng theo cấp số nhân.
- Giá trị trung bình: Mô hình này dự đoán giá trị RSSI trung bình. Trong thực tế, tín hiệu có thể dao động rất nhiều xung quanh giá trị trung bình này do các hiện tượng như fading (nhiều đa đường, bóng mờ).
- Xác định tham số: $RSSI_{d0}$ và n không phải là hằng số mà là các tham số phụ thuộc vào môi trường cụ thể và tần số hoạt động. Chúng cần được xác định thông qua các phép đo thực nghiệm tại hiện trường (quá trình hiệu chuẩn) để mô hình chính xác nhất.

Ưu điểm và Hạn chế:

Ưu điểm:

- Đơn giản và dễ hiểu: Công thức tương đối đơn giản, dễ áp dụng và tính toán.
- Thực tế: Mặc dù đơn giản, nó phản ánh khá tốt hành vi suy hao tín hiệu trung bình trong nhiều môi trường khác nhau.
- Chi phí thấp: Không yêu cầu thiết bị đo đạc quá phức tạp.
- Phù hợp với nhiều ứng dụng: Đặc biệt trong các ứng dụng định vị không yêu cầu độ chính xác cao (ví dụ: định vị trong nhà bằng RSSI) và lập kế hoạch mạng.

Hạn chế:

- Không chính xác hoàn toàn: Là một mô hình thực nghiệm, nó không thể mô tả chi tiết tất cả các hiện tượng vật lý phức tạp của sự truyền sóng (như khúc xạ, nhiễu xạ, phản xạ, tán xạ).
- Độ chính xác phụ thuộc vào hiệu chuẩn: Giá trị của n và $RSSI_{d0}$ phải được xác định chính xác cho từng môi trường. Nếu môi trường thay đổi (ví dụ: thêm tường, di chuyển đồ đạc), các tham số này cũng cần được điều chỉnh.
- Không tính đến dao động tín hiệu: Mô hình chỉ dự đoán giá trị trung bình, không tính đến các biến động ngắn hạn của tín hiệu (fading) có thể ảnh hưởng lớn đến hiệu suất thực tế.
- Độ chính xác định vị thấp: Do sự không ổn định của RSSI và các yếu tố môi trường, việc sử dụng mô hình này để ước tính khoảng cách cho mục đích định vị thường chỉ đạt độ chính xác ở mức mét chứ không phải centimet.

Ứng dụng trong Định vị RSSI

Trong hệ thống định vị vật thể bằng RSSI, Mô hình suy hao đường truyền Log-Distance Path Loss Model là công cụ chính để:

- Chuyển đổi RSSI thành khoảng cách: RSSI đo được từ mỗi điểm phát (AP/Beacon) được đưa vào công thức để ước tính khoảng cách đến điểm phát đó.
- Thực hiện thao tác: Sau khi có ước tính khoảng cách đến ít nhất ba điểm phát có vị trí đã biết, thuật toán sẽ được áp dụng để xác định tọa độ của thiết bị nhận.

2.2.3 Cập nhật dữ liệu RSSI khoảng cách đo được lên Google Trang tính

Để có khả năng cập nhật RSSI dữ liệu và cách đo khoảng cách lên Google Trang tính, thực hiện như sau:

Thiết lập Google Trang tính:

- Tạo một bảng tính mới của Google Trang tính.
- Tạo các cột tiêu đề phù hợp (ví dụ: "Thời gian", "ID điểm router", "RSSI", "Khoảng cách").

Viết Google Apps Script:

- Trong Google Sheets, vào "Công cụ" > "Lệnh chỉnh sửa tập lệnh".
- Viết một tập lệnh Apps Script để nhận dữ liệu qua HTTP POST và ghi vào bảng tính. Bạn có thể tìm kiếm các hướng dẫn trực tuyến về cách viết tập lệnh này (ví dụ: tìm kiếm "ESP32 Google Sheets Apps Script").
- Triển khai tập lệnh khai báo bên dưới dạng "Ứng dụng web" và lấy URL của trang web ứng dụng đó.

Sửa đổi firmware ESP32:

- Trong phần sụn của ESP32 (mở phần "Gửi dữ liệu lên máy chủ"), thay vì chỉ gửi dữ liệu đến vị trí xử lý máy chủ, bạn cũng gửi dữ liệu (RSSI và khoảng cách) đến URL của web ứng dụng Google Sheets bằng phương thức HTTP POST.
- Đảm bảo dữ liệu được gửi ở định dạng mà tập lệnh Apps Script có thể hiểu được (ví dụ: JSON).

Kiểm tra và điều chỉnh:

- Chạy hệ thống và kiểm tra xem dữ liệu có được ghi chính xác vào Google Trang tính hay không.
- Điều chỉnh tốc độ gửi dữ liệu từ ESP32 để tránh quá tải Google Sheets.

Công việc này cho phép bạn lưu trữ và trực tiếp hóa dữ liệu RSSI và cách thức theo thời gian, giúp phân tích và cải thiện độ chính xác của định vị hệ thống.

2.3 Tìm hiểu phần cứng và phần mềm của hệ thống

2.3.1 Vi điều khiển ESP32

ESP32 là một series các vi điều khiển trên một vi mạch giá rẻ, năng lượng thấp có hỗ trợ WiFi và dual-mode Bluetooth. Dòng ESP32 sử dụng bộ vi xử lý Tensilica Xtensa LX6 có hai biến thể lõi kép và lõi đơn, và bao gồm các công tắc antenna tích hợp, RF balun, bộ khuếch đại công suất, bộ khuếch đại thu nhiễu thấp, bộ lọc và module quản lý năng lượng.



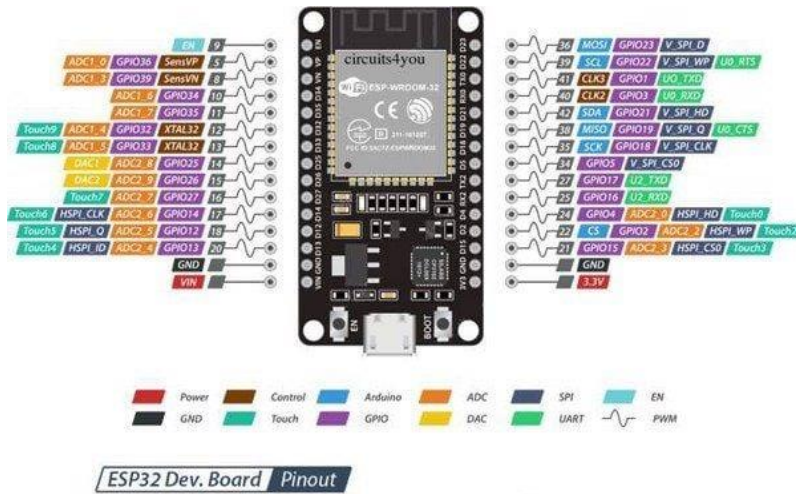
Hình 2.3 ESP32-DevKitC [8]

Thông số kỹ thuật:

- Vi xử lý: Từ 2 đến 240 MHz dual-core Tensilica LX6 microprocessor
- Bộ nhớ: 520 KB SRAM, 448 KB ROM, 4 MB flash memory
- Giao tiếp không dây: Wi-Fi 802.11 b/g/n, Bluetooth v4.2 BR/EDR và BLE (Bluetooth Low Energy)
- Giao tiếp dây: 2x SPI, 2x I2S, 2x I2C, 3x UART, 1x CAN 2.0, 1x Ethernet MAC interface with dedicated DMA and IEEE 1588 Precision Time Protocol support
- Giao tiếp ADC: 12-bit SAR ADC tới 18 kênh
- Giao tiếp PWM: 16 kênh PWM
- Giao tiếp đồng hồ thời gian thực: RTC built-in
- Tiêu thụ năng lượng: 5 μ A trong chế độ sleep, 20 mA - 240 mA trong chế độ hoạt động
- Điện áp hoạt động: 2.2V - 3.6V

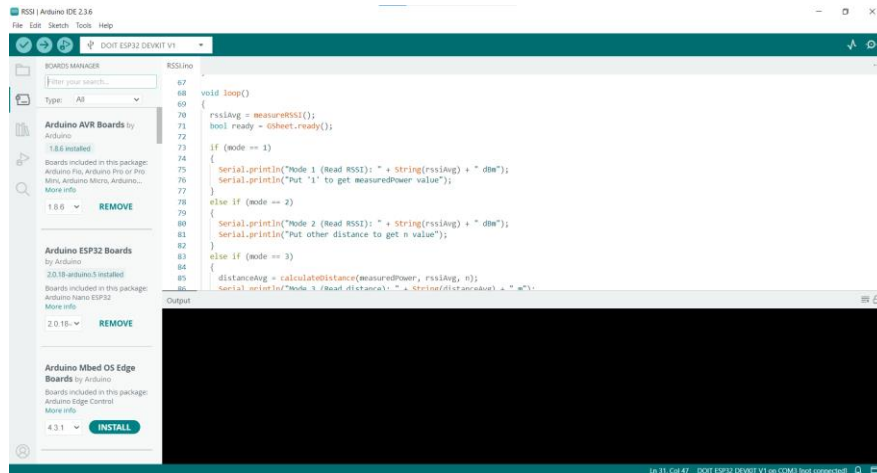
Ứng dụng ESP32 trong hệ thống định vị vật thể

- Kích thước: 18mm x 25.5mm x 3.1mm
- Trọng lượng: khoảng 20g



Hình 2.4 Sơ đồ chân của ESP32-WROOM-32 [8]

2.3.2 Nền tảng phát triển Arduino IDE



Hình 2.5 Phần mềm Arduino IDE

Arduino IDE là một phần mềm mã nguồn mở chủ yếu được sử dụng để viết và biên dịch mã vào module Arduino.

Đây là một phần mềm Arduino chính thức, giúp cho việc biên dịch mã trở nên dễ dàng mà ngay cả một người bình thường không có kiến thức kỹ thuật cũng có thể làm được.

Nó có các phiên bản cho các hệ điều hành như MAC, Windows, Linux và chạy trên nền tảng Java đi kèm với các chức năng và lệnh có sẵn đóng vai trò quan trọng để gỡ lỗi, chỉnh sửa và biên dịch mã trong môi trường.

Có rất nhiều các module Arduino như Arduino Uno, Arduino Mega, Arduino Leonardo, Arduino Micro và nhiều module khác.

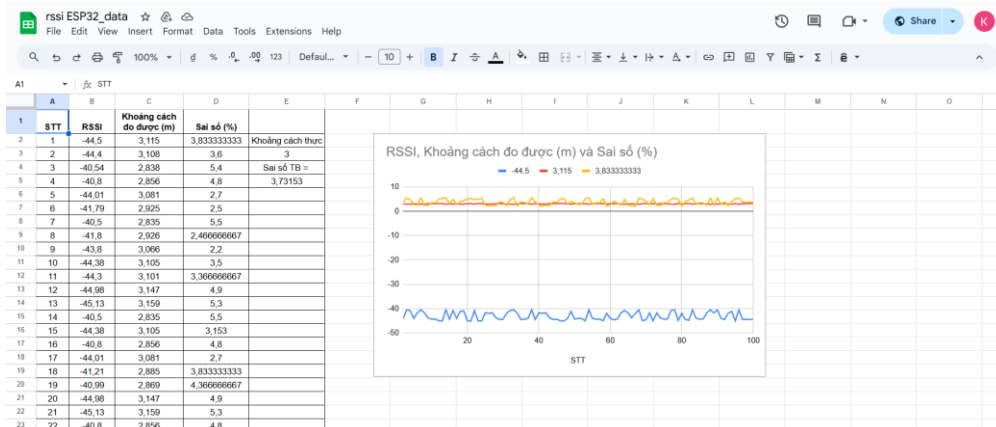
Mỗi module chứa một bộ vi điều khiển trên bo mạch được lập trình và chấp nhận thông tin dưới dạng mã.

Mã chính, còn được gọi là sketch, được tạo trên nền tảng IDE sẽ tạo ra một file Hex, sau đó được chuyển và tải lên trong bộ điều khiển trên bo.

Môi trường IDE chủ yếu chứa hai phần cơ bản: Trình chỉnh sửa và Trình biên dịch, phần đầu sử dụng để viết mã được yêu cầu và phần sau được sử dụng để biên dịch và tải mã lên module Arduino. Môi trường này hỗ trợ cả ngôn ngữ C và C++.

Khi người dùng viết mã và biên dịch, IDE sẽ tạo file Hex cho mã. File Hex là các file thập phân Hexa được Arduino hiểu và sau đó được gửi đến bo mạch bằng cáp USB. Mỗi bo Arduino đều được tích hợp một bộ vi điều khiển, bộ vi điều khiển sẽ nhận file hex và chạy theo mã được viết.

2.3.3 ESP32 Google Sheets



Hình 2.6 ESP32 Google Sheet

Google Sheets là một ứng dụng bảng tính trực tuyến, hoạt động trên nền tảng đám mây của Google (một phần của Google Drive). Nó có thể truy cập thông tin qua trình duyệt web và lưu trữ dữ liệu trên máy chủ của Google.

Để ESP32 có thể gửi dữ liệu lên Google Sheet, chúng tôi cần có "cầu kết nối" hoặc "trung gian" vì ESP32 không thể trực tiếp gọi các API của Google Sheet theo cách mà một trình duyệt web hay một ứng dụng máy tính làm.

ESP_Google_Sheet_Client.h: Đây là một thư viện chuyên biệt được thiết kế để đơn giản hóa quá trình xác thực và gửi dữ liệu từ ESP32 đến Google Sheets thông qua Google Apps Script. Thư viện này sẽ xử lý các chi tiết phức tạp của công việc tạo yêu cầu xác thực OAuth2 (sử dụng Email khách hàng và Khóa riêng của Tài khoản dịch vụ) và HTTP yêu cầu dạng.

=> "Google Sheet trong Arduino IDE" thực sự là công việc bạn cài đặt một thiết bị ESP32 bằng Arduino IDE để nó có thể gửi dữ liệu không dây qua Internet, sử dụng Google Apps Script làm trung gian để ghi dữ liệu trực tuyến vào bảng tính Google Sheets. Arduino IDE là công cụ tạo phần mềm cho ESP32, còn Google Sheets là đích đến của dữ liệu và Google Apps Script là "người vận hành" thông tin.

2.3.4 Các thư viện sử dụng

- WiFi.h: Thư viện này cung cấp các chức năng để điều khiển hợp nhất mô-đun Wi-Fi trên ESP32. Nó cho phép ESP32 kết nối với mạng Wi-Fi, hoạt động như Access Point (AP), quét mạng và thực thi cấp độ mạng tiếp theo cao hơn (như HTTP, TCP/UDP).
- Arduino.h: Đây là thư viện cốt lõi của nền tảng Arduino. Nó cung cấp các hàm và định nghĩa cơ bản nhất mà mọi chương trình Arduino đều cần có.
- Time.h: Đây là một thư viện chuẩn của C (C Standard Library) được sử dụng để làm việc theo thời gian và ngày. Nó cung cấp dữ liệu cấu trúc và chức năng để quản lý thời gian, chuyển đổi định dạng thời gian.
- ESP_Google_Sheet_Client.h: Dựa trên tên gọi, đây là một thư viện cụ thể được thiết kế để đơn giản hóa công việc gửi dữ liệu từ ESP32 trực tiếp đến Google Sheets thông qua Google Apps Script. Nó thường sẽ đóng gói các chi tiết phức tạp của công việc tạo HTTP POST yêu cầu, JSON định dạng và giao tiếp với URL của Apps Script.

- GS_SDHelper.h: Dựa trên tên gọi, "GS" có thể là Google Trang tính, và mọ "SDHelper" về thẻ nhớ SD được hỗ trợ. Đây có thể là một thư viện cụ thể được thiết kế để:
- Lưu trữ tạm thời dữ liệu trên thẻ SD: Nếu mạng kết nối không ổn định hoặc bạn muốn ghi lại dữ liệu ngay khi không có kết nối internet, thư viện này có thể giúp lưu RSSI/khoảng cách dữ liệu vào thẻ nhớ SD.
- Đồng bộ hóa dữ liệu với Google Trang tính từ thẻ SD: Khi mạng kết nối được khôi phục, nó có thể hỗ trợ đọc dữ liệu từ thẻ SD và gửi lên Google Trang tính.

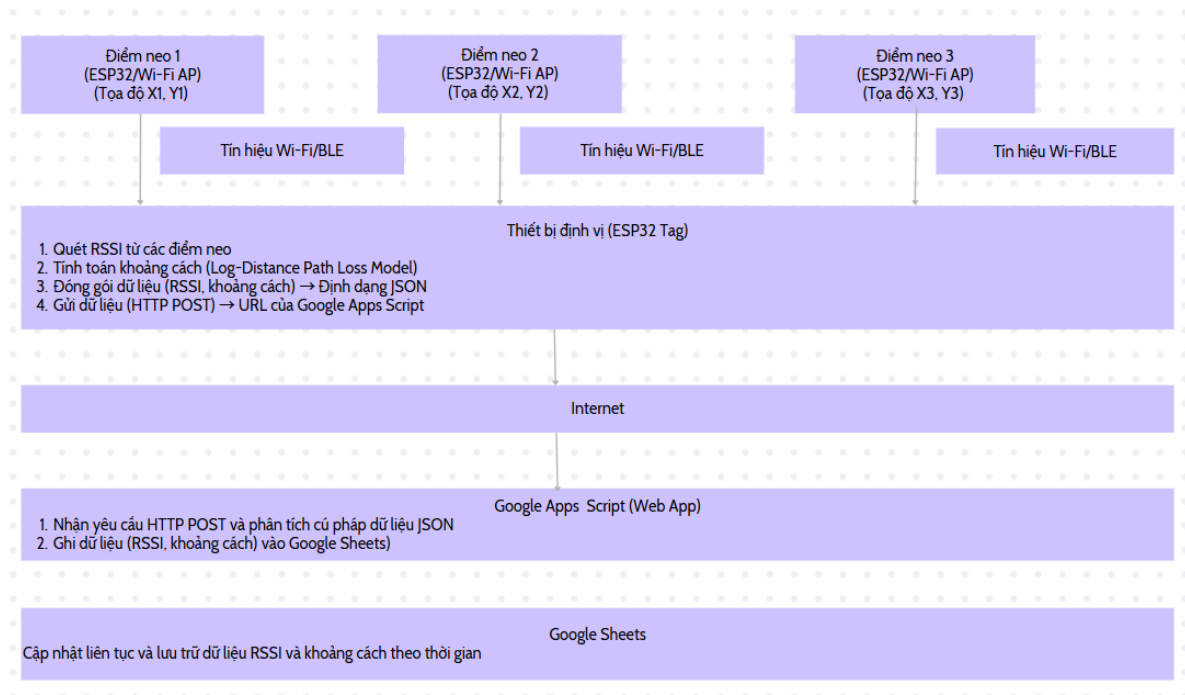
2.4 Kết luận chương

Qua chương này cung cấp cho chúng ta kiến thức cơ bản, quan trọng để hiểu và phát triển hệ thống định vị vật thể ứng dụng ESP32. Tôi đã tìm hiểu về mô hình tổng quát của hệ thống định vị vật thể, các giải pháp linh kiện phần cứng, chuẩn truyền dữ liệu và phần mềm lập trình có thể được sử dụng trong hệ thống, cũng như phần mềm lưu trữ dữ liệu. Các kiến thức này sẽ giúp tôi thiết kế và triển khai “ Ứng dụng ESP32 trong hệ thống định vị vật thể ” một cách hiệu quả và đáp ứng được các yêu cầu của một hệ thống định vị tiên tiến với độ chính xác cao.

CHƯƠNG 3: PHƯƠNG PHÁP THỰC HIỆN

3.1 Giới thiệu chương

Từ sơ đồ khối hệ thống, tôi sẽ phân tích và trình bày những vấn đề về tính năng và yêu cầu kỹ thuật của hệ thống từ đó sẽ đưa ra được những yêu cầu của hệ thống. Tiếp theo sẽ nghiên cứu mô hình suy hao đường truyền Log-Distance Path Loss Model RSSI và ESP32 (lập trình với Arduino IDE) giao tiếp với Google Sheets. Từ đó làm cơ sở để xây dựng nên một hệ thống hoàn chỉnh.



Hình 3.1 Sơ đồ khối tổng quát của hệ thống

Giải thích sơ đồ khối:

Các điểm router cố định:

- Là các thiết bị phát tín hiệu Wi-Fi hoặc Bluetooth (ví dụ: AP32 cấu hình AP, bộ định tuyến Wi-Fi hoặc BLE Beacon).
- Đã xác định vị trí và tốc độ cố định.

Định vị thiết bị (Thẻ ESP32):

- Quét RSSI từ các điểm mới.
- Tính toán khoảng cách đến từng điểm router bằng mô hình Log-Distance.

- Đóng gói dữ liệu và gửi lên Google Trang tính.

Internet: Mạng truyền dữ liệu từ ESP32 đến Google Sheets.

Tập lệnh Google Apps:

- Hoạt động như một webbook, nhận dữ liệu từ ESP32.
- Ghi dữ liệu vào Google Sheets.

Google Trang tính:

- Lưu trữ RSSI dữ liệu và khoảng cách theo thời gian.
- Có thể sử dụng để phân tích và trực quan hóa dữ liệu.

Sơ đồ này có thể tạo luồng dữ liệu chính của hệ thống, từ công việc thu thập tín hiệu đến nơi lưu trữ và xử lý dữ liệu trên đám mây.

3.2 Các yêu cầu về tính năng và kỹ thuật

Để thiết kế một ứng dụng ESP32 hiệu quả cho hệ thống định vị vật thể bằng phương pháp mô hình suy hao đường truyền RSSI Log-Distance Path Loss Model và cập nhật dữ liệu lên Google Sheets, chúng ta cần xác định rõ các yêu cầu về tính năng (Functional Requirements) của phần mềm ứng dụng ESP32. Đây là những gì mà ESP32 phải có khả năng thực hiện để hệ thống hoạt động như mong đợi.

Dưới đây là các yêu cầu tính năng chi tiết:

Yêu cầu về thu thập và xử lý dữ liệu RSSI

- Khả năng quét tín hiệu không dây:

Ứng dụng ESP32 phải có khả năng quét các mạng Wi-Fi (hoặc Bluetooth Low Energy - BLE nếu dùng BLE Beacon) trong môi trường xung quanh một cách định kỳ (ví dụ: mỗi X giây).

Nó phải có khả năng nhận diện và lọc ra các tín hiệu từ các điểm router (Anchor Nodes) đã được định nghĩa trước (ví dụ: dựa trên địa chỉ MAC - BSSID của các AP Wi-Fi hoặc UUID/Major/Minor ID của BLE Beacon).

- Thu thập dữ liệu RSSI:

Ứng dụng phải có khả năng trích xuất giá trị RSSI (Received Signal Strength Indicator) từ mỗi gói tin nhận được từ các điểm router đã nhận diện.

Nó phải thu thập một số lượng mẫu RSSI nhất định (ví dụ: 5 đến 10 mẫu) từ mỗi điểm router trong một chu kỳ quét để có dữ liệu đa dạng cho xử lý.

- Tiền xử lý dữ liệu RSSI:

Ứng dụng phải có khả năng áp dụng thuật toán làm mịn dữ liệu cho các giá trị RSSI đã thu thập (ví dụ: tính giá trị trung bình cộng hoặc trung vị của các mẫu RSSI) để giảm thiểu nhiễu và biến động tín hiệu.

- Tính toán khoảng cách theo mô hình Log-Distance:

Ứng dụng phải có khả năng sử dụng giá trị RSSI cùng với các tham số hiệu chuẩn (RSSI₀ và n) để tính toán ước tính khoảng cách từ ESP32 đến từng điểm router theo công thức Log-Distance Path Loss Model.

Yêu cầu về Giao tiếp và Truyền tải dữ liệu

- Đóng gói dữ liệu.
- Ứng dụng phải có khả năng tổng hợp tất cả các thông tin cần thiết vào một gói dữ liệu có cấu trúc (ví dụ: định dạng JSON), bao gồm.
- ID duy nhất của thiết bị ESP32.
- Timestamp (thời gian chính xác của lần đo).
- ID của từng điểm router được phát hiện.
- Giá trị RSSI từ mỗi điểm router.
- Khoảng cách đã được tính toán đến từng điểm router.
- Gửi dữ liệu đến Google Sheets.
- Ứng dụng phải có khả năng gửi gói dữ liệu đã đóng gói này trực tiếp đến một URL của Google Apps Script (Web App) thông qua yêu cầu HTTP POST.
- Ứng dụng phải sử dụng thông tin xác thực của Tài khoản dịch vụ (Service Account) (Client Email và Private Key) để ủy quyền cho các yêu cầu gửi dữ liệu đến Google Sheets.

Yêu cầu về Kết nối mạng và Đồng bộ hóa

- Kết nối Wi-Fi.
- Ứng dụng phải có khả năng kết nối và duy trì kết nối ổn định với một mạng Wi-Fi được cấu hình trước.
- Nó phải có cơ chế tự động kết nối lại khi kết nối Wi-Fi bị gián đoạn.
- Đồng bộ thời gian (NTP).
- Ứng dụng phải có khả năng đồng bộ hóa thời gian của mình với máy chủ NTP để đảm bảo timestamp của dữ liệu là chính xác và nhất quán.

Yêu cầu về Cấu hình và Quản lý lỗi

- Cấu hình tham số.
- Ứng dụng phải có khả năng lưu trữ và sử dụng các tham số hiệu chuẩn (RSSId0 cho từng điểm router, hệ số n cho môi trường) đã được xác định trước.
- (Tùy chọn) Có thể có khả năng cập nhật các tham số này từ xa hoặc thông qua Serial Monitor.
- Xử lý lỗi truyền dữ liệu.
- Ứng dụng phải có khả năng phát hiện và báo cáo (ví dụ: qua Serial Monitor) các lỗi xảy ra trong quá trình gửi dữ liệu lên Google Apps Script/Google Sheets (ví dụ: lỗi kết nối, lỗi xác thực, lỗi phản hồi từ server).
- Tình trạng hoạt động.
- Ứng dụng nên cung cấp một số hình thức chỉ thị trạng thái cơ bản (ví dụ: qua LED) để báo hiệu ESP32 đang hoạt động, đã kết nối Wi-Fi, hoặc đang gửi dữ liệu.

3.3 Quy trình thao tác thực hiện

Đây là bước cực kỳ quan trọng để xác định các tham số RSSId0 và hệ số môi trường n cho môi trường cụ thể, giúp ESP32 tính toán khoảng cách chính xác.

- Cài đặt môi trường: Đặt cố định vị trí router trong môi trường không có vật cản trực tiếp.
- Đặt ESP32 ở vị trí cách router 1m mà không có vật cản trực tiếp và nhập “1” vào Serial để tính RSSId0.
- Dịch ESP32 đến vị trí khác và nhập khoảng cách tại vị trí đó vào Serial để tính hệ số môi trường n.

- Từ giá trị RSSI và hệ số môi trường ta tính giá trị RSSI.
- Từ giá trị RSSI tính được ta tính giá trị khoảng cách tương ứng.
- Cập nhật dữ liệu RSSI và khoảng cách liên tục lên nền tảng lưu trữ Google Sheet.

3.4 Kết luận chương

Trong chương này, tôi đã tìm hiểu về các yêu cầu tính năng sử dụng và yêu cầu kỹ thuật của hệ thống. Đồ án đã đi vào chi tiết về các chức năng và đặc tính cần có của hệ thống để đạt được hiệu suất tối ưu. Sau đó, tôi đã thiết kế kiến trúc tổng thể của hệ thống, bao gồm mô hình tổng thể.

Tiếp theo, tôi sẽ thiết kế hệ thống định vị vật thể ứng dụng ESP32 bằng mô hình suy hao đường truyền Log-Distance Path Loss Model RSSI, sau đó cập nhật quản lý dữ liệu lưu trữ lên Google Sheet để theo dõi hoạt động và sự tối ưu của hệ thống.

Chương 3 rất cần thiết và quan trọng trong quá trình phát triển sản phẩm. Chương này đưa ra các yêu cầu về chức năng và thiết kế cho hệ thống, từ đó mà tôi xây dựng sơ đồ khối hoạt động của hệ thống và sơ đồ thuật toán toàn hệ thống để thiết kế một hệ thống định vị vật thể ứng dụng ESP32 bằng mô hình suy hao đường truyền Log-Distance Path Loss Model RSSI tối ưu với độ chính xác cao.

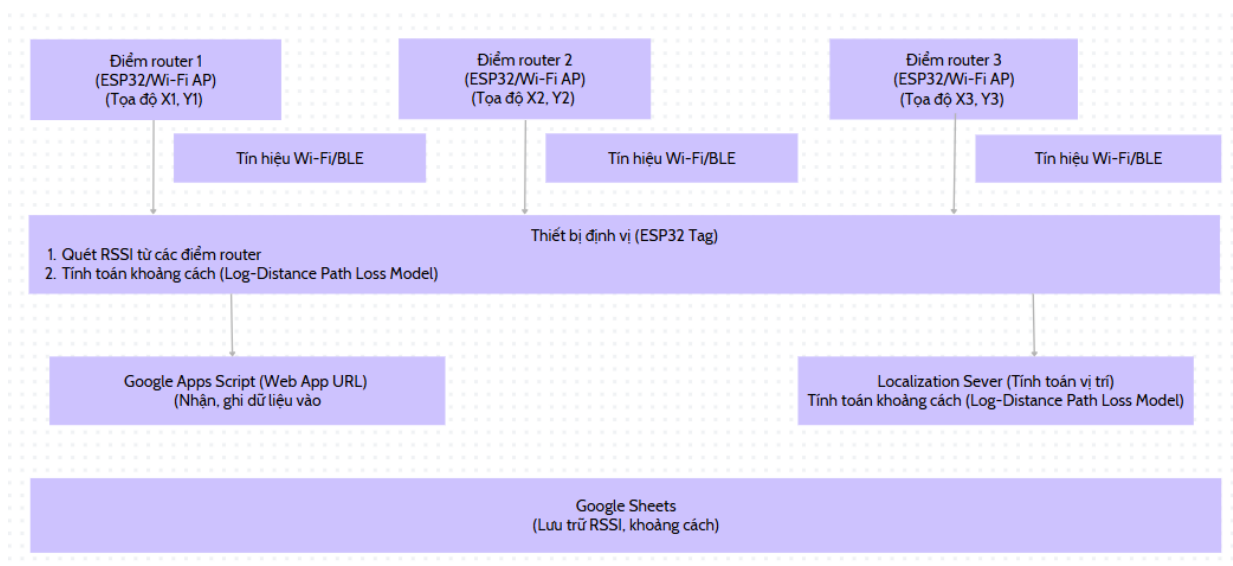
CHƯƠNG 4: KẾT QUẢ VÀ ĐÁNH GIÁ HỆ THỐNG

4.1 Giới thiệu chương

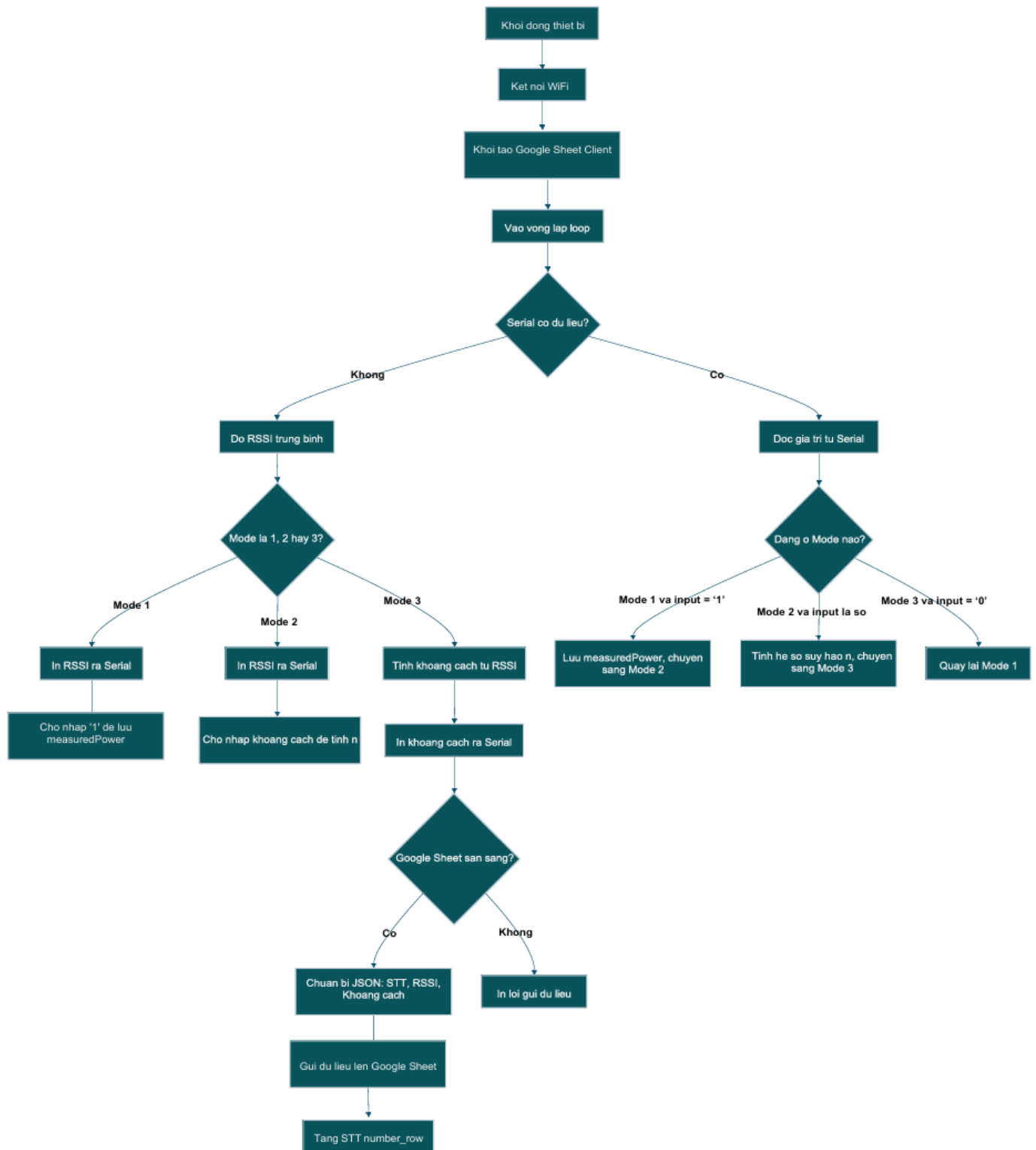
Trong chương này, tôi sẽ trình bày kết quả thực hiện hệ thống định vị khoảng cách vật thể ứng dụng ESP32 bằng phương pháp mô hình Log-Distance Path Loss Model RSSI. Bên cạnh đó, Ứng dụng ESP32 trong hệ thống định vị vật thể với phương pháp RSSI Log-Distance Path Loss Model là một giải pháp chi phí thấp, linh hoạt, và tương đối dễ triển khai, đặc biệt phù hợp cho các ứng dụng định vị trong nhà (Indoor Positioning Systems - IPS) hoặc các khu vực có giới hạn.

Quy trình đánh giá được chia thành 4 bước chính, bao gồm xác định mục tiêu đánh giá, xác định tiêu chuẩn đánh giá, thực hiện đánh giá và đưa ra kết luận, đề xuất cải tiến và hướng phát triển. Mỗi bước đều có vai trò quan trọng và sẽ giúp đảm bảo quá trình đánh giá được thực hiện chính xác và đầy đủ.

4.2 Kết quả thực hiện



Hình 4.1 Sơ đồ khối



Hình 4.2 Sơ đồ thuật toán

Thực nghiệm đo đạc:

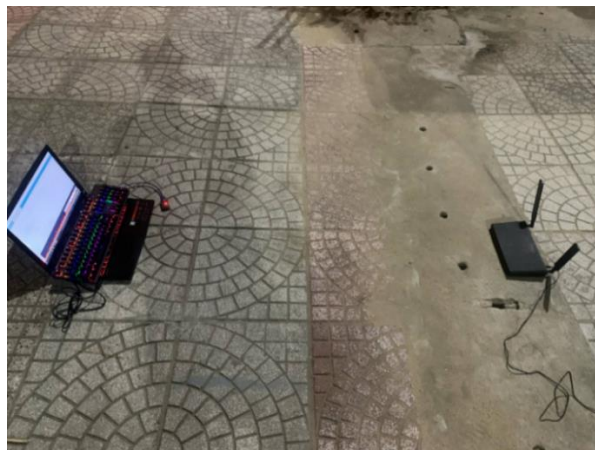
Xác định giá trị RSSI và hệ số môi trường để Hệ thống tính toán giá trị RSSI và giá trị khoảng cách đo được.



Hình 4.3 ESP32 cách router cố định 0,5m

Tại vị trí ESP32 cách router cố định 1m:

- Lấy 100 giá trị RSSI tương ứng với 100 giá trị khoảng cách tại vị trí 1m. Thời gian để lấy được 100 giá trị RSSI và khoảng cách tương ứng là 300 giây.
- Tính sai số % của từng giá trị khoảng cách so với giá trị 1m.
- Vẽ biểu đồ dạng sóng để thể hiện sự biến động sai số của 100 giá trị RSSI, khoảng cách và % sai số vừa tính.



Hình 4.4 ESP32 cách router cố định 1m

Tại vị trí ESP32 cách router cố định 2m:

- Lấy 100 giá trị RSSI tương ứng với 100 giá trị khoảng cách tại vị trí 2m. Thời gian để lấy được 100 giá trị RSSI và khoảng cách tương ứng là 300 giây.
- Tính sai số % của từng giá trị khoảng cách so với giá trị 2m.
- Vẽ biểu đồ dạng sóng để thể hiện sự biến động sai số của 100 giá trị RSSI, khoảng cách và % sai số vừa tính.



Hình 4.5 ESP32 cách router cố định 2m

Tại vị trí ESP32 cách router cố định 3m:

- Lấy 100 giá trị RSSI tương ứng với 100 giá trị khoảng cách tại vị trí 3m. Thời gian để lấy được 100 giá trị RSSI và khoảng cách tương ứng là 300 giây.
- Tính sai số % của từng giá trị khoảng cách so với giá trị 3m.
- Vẽ biểu đồ dạng sóng để thể hiện sự biến động sai số của 100 giá trị RSSI, khoảng cách và % sai số vừa tính.



Hình 4.6 ESP32 cách router cố định 3m

Tại vị trí ESP32 cách router cố định 4m:

- Lấy 100 giá trị RSSI tương ứng với 100 giá trị khoảng cách tại vị trí 4m. Thời gian để lấy được 100 giá trị RSSI và khoảng cách tương ứng là 300 giây.
- Tính sai số % của từng giá trị khoảng cách so với giá trị 4m.
- Vẽ biểu đồ dạng sóng để thể hiện sự biến động sai số của 100 giá trị RSSI, khoảng cách và % sai số vừa tính.



Hình 4.7 ESP32 cách router cố định 4m

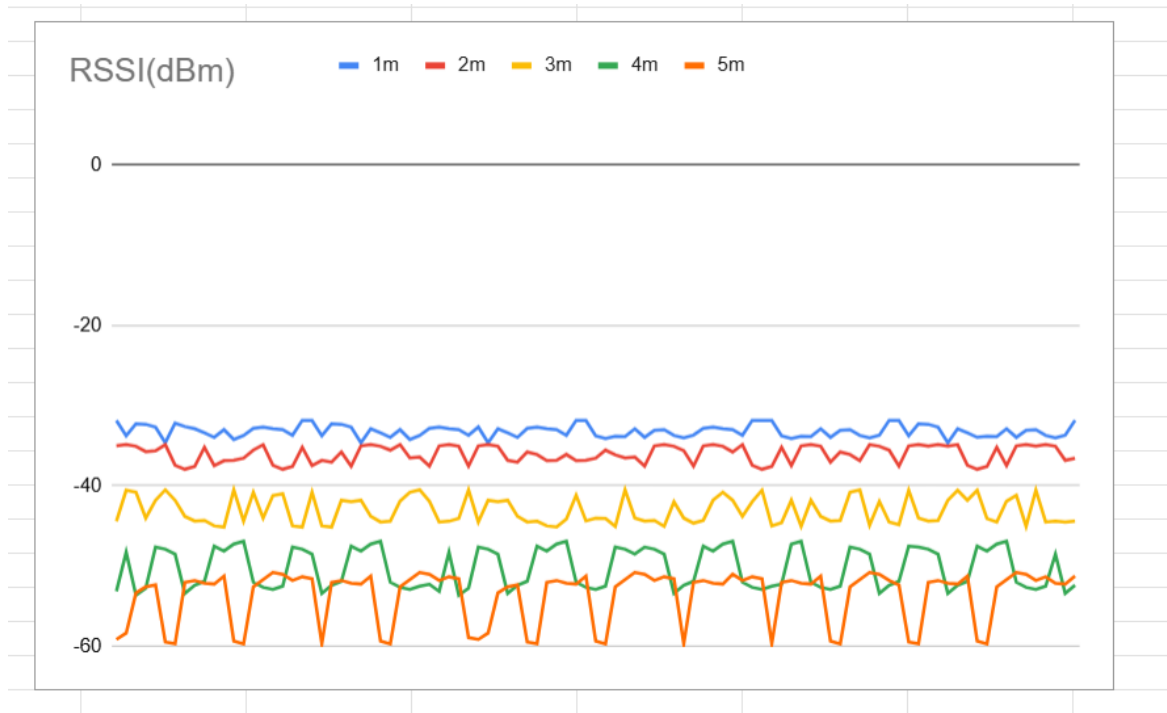
Tại vị trí ESP32 cách router cố định 5m:

- Lấy 100 giá trị RSSI tương ứng với 100 giá trị khoảng cách tại vị trí 5m. Thời gian để lấy được 100 giá trị RSSI và khoảng cách tương ứng là 300 giây.
- Tính sai số % của từng giá trị khoảng cách so với giá trị 5m.
- Vẽ biểu đồ dạng sóng để thể hiện sự biến động sai số của 100 giá trị RSSI, khoảng cách và % sai số vừa tính.



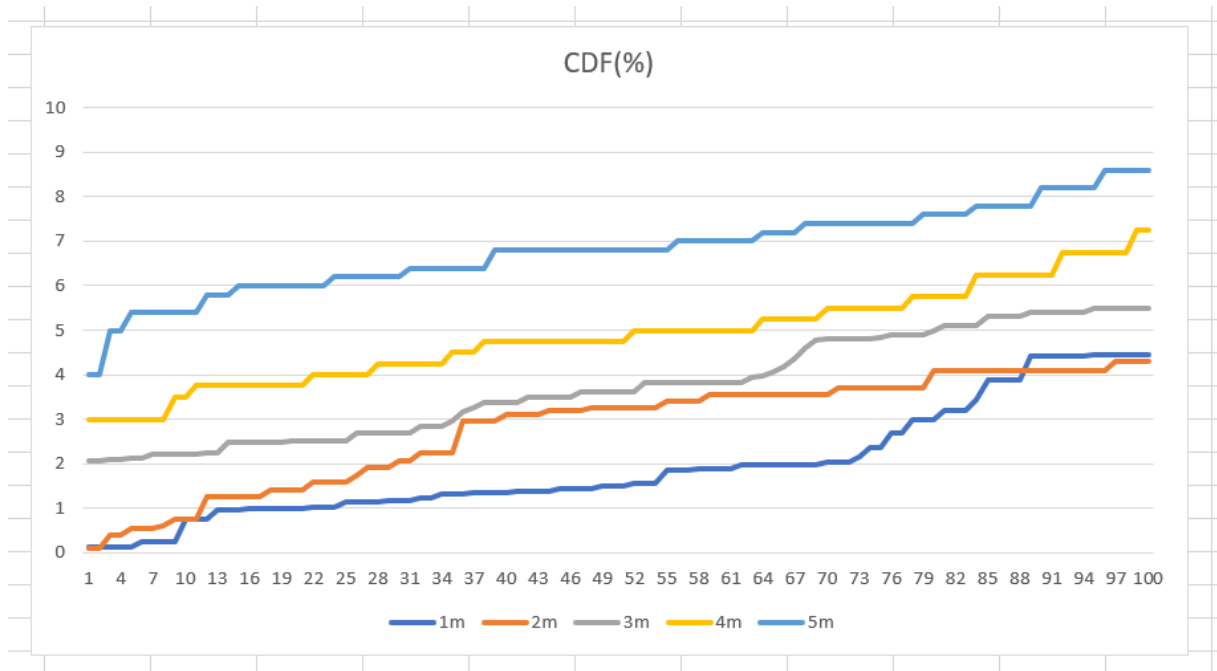
Hình 4.8 ESP32 cách router cố định 5m

Tổng hợp các giá trị RSSI của 5 trường hợp trong một đồ thị dạng sóng để biểu hiện rõ sai số thông qua sự dao động lên xuống của giá trị RSSI trong mỗi trường hợp so với các trường hợp còn lại. Đưa ra kết luận về sự tối ưu của hệ thống.



Hình 4.9 Biểu đồ dạng sóng thể hiện sai số qua dao động sóng của các trường hợp (1m, 2m, 3m, 4m, 5m)

Nhận xét biểu đồ: Ở khoảng cách nhỏ thì biên độ sóng dao động nhỏ, và khoảng cách càng tăng dần thì biên độ sóng dao động ngày càng lớn dần. Điều đó thể hiện khoảng cách càng xa thì giá trị RSSI thu được sai số càng lớn. Tại vị trí ESP32 càng gần router sóng wifi càng mạnh thì ta thu được giá trị RSSI càng lớn dẫn đến sai số đo khoảng cách càng nhỏ.



Hình 4.10 Biểu đồ CDF phân phối chuẩn dữ liệu sai số(%)

Nhận xét biểu đồ CDF:

- Trường hợp 1m: Có trên 50% giá trị có sai số dưới 2% và dưới 50% giá trị còn lại có sai số từ 2% đến dưới 4,5%.
- Trường hợp 2m: Có 40% giá trị có sai số dưới 30% và 60% giá trị còn lại có sai số từ trên 3% đến dưới 5%.
- Trường hợp 3m: Có trên 60% giá trị có sai số từ 2% đến 4% và dưới 40% giá trị còn lại có sai số từ 4% đến dưới 6%.
- Trường hợp 4m: Có trên 50% giá trị có sai số từ 3% đến 5% và dưới 50% giá trị còn lại có sai số từ trên 5% đến dưới 7,5%.
- Trường hợp 5m: Có trên 50% giá trị có sai số từ 4% đến 7% và dưới 50% giá trị còn lại có sai số từ trên 7% đến dưới 9%.

4.3 Cơ sở tiêu chuẩn đánh giá

- Độ chính xác: Hệ thống định vị vật thể với khả năng cập nhật dữ liệu giá trị RSSI, thông qua đó đưa ra khoảng cách vật thể đo được với độ chính xác tối ưu và lưu trữ tại nền tảng web Google Sheet tại thời gian thực tế hiển thị trong Serial của nền tảng phát triển Arduino IDE.

- Thời gian phản ứng: Hệ thống định vị vật thể cập nhật dữ liệu giá trị RSSI và giá trị khoảng cách đo được liên tục trong một khoảng thời gian để xác định.
- Độ nhạy: Hệ thống định vị vật thể cần có khả năng theo dõi khoảng cách vật thể trong phạm vi định vị liên tục cập nhật giá trị RSSI và khoảng cách vật thể để giám sát và đưa ra dữ liệu lưu trữ. Độ nhạy phát hiện chính xác dữ liệu khoảng cách vật thể trong phạm vi định vị liên tục.
- Độ bao phủ: Hệ thống định vị vật thể cần có phạm vi định vị khoảng cách phủ sóng rộng trong một phạm vi không gian phòng học, sân trường, công viên, nhà kho để ứng dụng vào các hệ thống theo dõi và định vị vật thể.
- Tin cậy và sẵn sàng: Hệ thống định vị vật thể cần được bảo trì và kiểm tra định kì để có được sự chính xác cao nhất.

4.4 Thực hiện đánh giá

4.4.1 Thực hiện kiểm tra và đánh giá độ chính xác

Tiến hành đánh giá theo phương pháp đo đặc kích thước trong thực tế và so sánh với khoảng cách mà hệ thống đo và lưu trữ được, độ chính xác cao sai số không quá 10%. Kết quả đánh giá được thể hiện trong các bảng dữ liệu từng trường hợp sau:

Bảng 4.1 Bảng dữ liệu đánh giá độ chính xác của trường hợp khoảng cách thực tế 1m

Lần kiểm tra	Khoảng cách thực tế(m)	Khoảng cách hệ thống đo được(m)	Sai số(%)
1	1	0,95557	4,443
2	1	0,95586	4,414
3	1	1,00128	0,128
4	1	0,96802	3,198
5	1	0,97023	2,977
6	1	0,98025	1,975
7	1	1,03883	3,883
8	1	0,96557	3,443
9	1	0,97857	2,143
10	1	0,98655	1,345
11	1	1,00236	0,236
12	1	1,01883	1,883

Ứng dụng ESP32 trong hệ thống định vị vật thể

13	1	0,99023	0,977
14	1	1,02683	2,683
15	1	1,01228	1,228
16	1	0,98559	1,441
17	1	0,98026	1,974
18	1	0,98608	1,392
19	1	0,99018	0,982
20	1	1,01129	1,129
21	1	0,95557	4,443
22	1	0,95586	4,414
23	1	1,00128	0,128
24	1	0,96802	3,198
25	1	0,97023	2,977
26	1	0,98025	1,975
27	1	1,03883	3,883
28	1	0,98655	1,345
29	1	1,00236	0,236
30	1	1,01883	1,883
31	1	0,99023	0,977
32	1	1,02683	2,683
33	1	1,01228	1,228
34	1	0,98559	1,441
35	1	0,98026	1,974
36	1	0,98608	1,392
37	1	0,99018	0,982
38	1	1,01129	1,129
39	1	0,98025	1,975
40	1	1,03883	3,883
41	1	0,98655	1,345
42	1	1,00236	0,236
43	1	1,01883	1,883
44	1	0,98559	1,441

Ứng dụng ESP32 trong hệ thống định vị vật thể

45	1	0,98026	1,974
46	1	0,98608	1,392
47	1	0,99018	0,982
48	1	1,01129	1,129
49	1	0,95557	4,443
50	1	0,95586	4,414
51	1	1,00128	0,128
52	1	1,02365	2,365
53	1	1,01509	1,509
54	1	1,01556	1,556
55	1	0,98683	1,317
56	1	1,01866	1,866
57	1	0,99254	0,746
58	1	0,99026	0,974
59	1	1,01165	1,165
60	1	1,02023	2,023
61	1	1,01025	1,025
62	1	0,98559	1,441
63	1	0,98026	1,974
64	1	0,98608	1,392
65	1	0,99018	0,982
66	1	1,01129	1,129
67	1	0,95557	4,443
68	1	0,95586	4,414
69	1	0,95586	4,414
70	1	1,00128	0,128
71	1	1,02365	2,365
72	1	1,01509	1,509
73	1	1,01556	1,556
74	1	0,98683	1,317
75	1	1,01866	1,866
76	1	0,99254	0,746

77	1	0,99026	0,974
78	1	1,01165	1,165
79	1	1,02023	2,023
80	1	1,01025	1,025
81	1	0,95557	4,443
82	1	0,95586	4,414
83	1	1,00128	0,128
84	1	0,96802	3,198
85	1	0,97023	2,977
86	1	0,98025	1,975
87	1	1,03883	3,883
88	1	0,98655	1,345
89	1	1,00236	0,236
90	1	1,01883	1,883
91	1	1,01509	1,509
92	1	1,01556	1,556
93	1	0,98683	1,317
94	1	1,01866	1,866
95	1	0,99254	0,746
96	1	0,99026	0,974
97	1	1,01165	1,165
98	1	1,02023	2,023
99	1	1,01025	1,025
100	1	0,95557	4,443

=>Trung bình sai số(%) của 100 lần kiểm tra là 1,93902%.

Bảng 4.2 Bảng dữ liệu đánh giá độ chính xác của trường hợp khoảng cách thực tế 2m

Lần kiểm tra	Khoảng cách thực tế(m)	Khoảng cách hệ thống đo được(m)	Sai số(%)
--------------	------------------------	---------------------------------	-----------

Ứng dụng ESP32 trong hệ thống định vị vật thể

1	2	2,065	3,25
2	2	1,926	3,7
3	2	1,918	4,1
4	2	1,929	3,55
5	2	1,965	1,75
6	2	1,959	2,05
7	2	1,918	4,1
8	2	2,059	2,95
9	2	2,086	4,3
10	2	2,068	3,4
11	2	1,936	3,2
12	2	2,062	3,1
13	2	2,028	1,4
14	2	2,025	1,25
15	2	2,011	0,55
16	2	1,955	2,25
17	2	1,918	4,1
18	2	2,059	2,95
19	2	2,086	4,3
20	2	2,068	3,4
21	2	1,936	3,2
22	2	2,062	3,1
23	2	2,025	1,25
24	2	2,038	1,9
25	2	1,968	1,6
26	2	2,065	3,25
27	2	1,926	3,7
28	2	1,918	4,1
29	2	1,929	3,55
30	2	1,955	2,25
31	2	1,918	4,1
32	2	2,008	0,4

Ứng dụng ESP32 trong hệ thống định vị vật thể

33	2	2,002	0,1
34	2	2,065	3,25
35	2	1,926	3,7
36	2	1,918	4,1
37	2	1,929	3,55
38	2	2,065	3,25
39	2	1,926	3,7
40	2	1,918	4,1
41	2	1,929	3,55
42	2	2,025	1,25
43	2	2,038	1,9
44	2	1,968	1,6
45	2	1,985	0,75
46	2	2,028	1,4
47	2	2,025	1,25
48	2	1,985	0,75
49	2	2,028	1,4
50	2	2,025	1,25
51	2	2,011	0,55
52	2	1,955	2,25
53	2	1,988	0,6
54	2	2,008	0,4
55	2	2,002	0,1
56	2	2,065	3,25
57	2	1,926	3,7
58	2	1,918	4,1
59	2	1,929	3,55
60	2	1,959	2,05
61	2	2,065	3,25
62	2	1,926	3,7
63	2	1,918	4,1
64	2	1,929	3,55

Ứng dụng ESP32 trong hệ thống định vị vật thể

65	2	1,968	1,6
66	2	1,918	4,1
67	2	2,059	2,95
68	2	2,086	4,3
69	2	2,068	3,4
70	2	1,936	3,2
71	2	2,062	3,1
72	2	1,926	3,7
73	2	1,918	4,1
74	2	1,929	3,55
75	2	2,038	1,9
76	2	1,968	1,6
77	2	1,985	0,75
78	2	2,028	1,4
79	2	1,918	4,1
80	2	1,929	3,55
81	2	1,955	2,25
82	2	2,065	3,25
83	2	1,926	3,7
84	2	1,918	4,1
85	2	1,929	3,55
86	2	1,918	4,1
87	2	1,929	3,55
88	2	1,918	4,1
89	2	2,059	2,95
90	2	2,086	4,3
91	2	2,068	3,4
92	2	1,936	3,2
93	2	2,062	3,1
94	2	1,926	3,7
95	2	1,918	4,1
96	2	1,929	3,55

97	2	1,918	4,1
98	2	1,929	3,55
99	2	2,025	1,25
100	2	2,011	0,55

=>Trung bình sai số(%) của 100 lần kiểm tra là 2,817%

Bảng 4.3 Bảng dữ liệu đánh giá độ chính xác của trường hợp khoảng cách thực tế 3m

Lần kiểm tra	Khoảng cách thực tế(m)	Khoảng cách hệ thống đo được(m)	Sai số(%)
1	3	3,115	3,833333333
2	3	3,108	3,6
3	3	2,838	5,4
4	3	2,856	4,8
5	3	3,081	2,7
6	3	2,925	2,5
7	3	2,835	5,5
8	3	2,926	2,466666667
9	3	3,066	2,2
10	3	3,105	3,5
11	3	3,101	3,366666667
12	3	3,147	4,9
13	3	3,159	5,3
14	3	2,835	5,5
15	3	3,105	3,153
16	3	2,856	4,8
17	3	3,081	2,7
18	3	2,885	3,833333333
19	3	2,869	4,366666667
20	3	3,147	4,9

Ứng dụng ESP32 trong hệ thống định vị vật thể

21	3	3,159	5,3
22	3	2,856	4,8
23	3	3,147	4,9
24	3	3,159	5,3
25	3	2,925	2,5
26	3	2,938	2,066666667
27	3	2,926	2,466666667
28	3	3,066	2,2
29	3	3,115	3,833333333
30	3	3,108	3,6
31	3	2,933	2,233333333
32	3	2,856	4,8
33	3	2,835	5,5
34	3	2,936	2,133333333
35	3	3,115	3,833333333
36	3	3,108	3,6
37	3	3,085	2,833333333
38	3	2,835	5,5
39	3	3,115	3,833333333
40	3	2,925	2,5
41	3	2,938	2,066666667
42	3	2,926	2,466666667
43	3	3,066	2,2
44	3	3,115	3,833333333
45	3	3,108	3,6
46	3	3,147	4,9
47	3	3,159	5,3
48	3	3,089	2,966666667
49	3	2,881	3,966666667
50	3	3,105	3,5
51	3	3,081	2,7
52	3	3,085	2,833333333

Ứng dụng ESP32 trong hệ thống định vị vật thể

53	3	3,153	5,1
54	3	2,835	5,5
55	3	3,081	2,7
56	3	3,105	3,5
57	3	3,101	3,366666667
58	3	3,153	5,1
59	3	2,937	2,1
60	3	3,081	2,7
61	3	3,125	4,166666667
62	3	3,098	3,266666667
63	3	2,925	2,5
64	3	2,856	4,8
65	3	2,926	2,466666667
66	3	3,066	2,2
67	3	2,937	2,1
68	3	2,838	5,4
69	3	3,15	5
70	3	3,122	4,066666667
71	3	2,925	2,5
72	3	3,153	5,1
73	3	2,926	2,466666667
74	3	3,066	2,2
75	3	3,105	3,5
76	3	3,101	3,366666667
77	3	2,855	4,833333333
78	3	2,835	5,5
79	3	3,143	4,766666667
80	3	2,936	2,133333333
81	3	3,115	3,833333333
82	3	3,138	4,6
83	3	2,838	5,4
84	3	3,081	2,7

85	3	3,105	3,5
86	3	3,101	3,366666667
87	3	2,925	2,5
88	3	2,838	5,4
89	3	2,926	2,466666667
90	3	2,838	5,4
91	3	3,085	2,833333333
92	3	3,115	3,833333333
93	3	2,933	2,233333333
94	3	2,882	3,933333333
95	3	3,153	5,1
96	3	2,838	5,4
97	3	3,115	3,833333333
98	3	3,108	3,6
99	3	3,115	3,833333333
100	3	3,108	3,6

=>Trung bình sai số(%) của 100 lần kiểm tra là 3,73153%.

Bảng 4.4 Bảng dữ liệu đánh giá độ chính xác của trường hợp khoảng cách thực tế 4m

Lần kiểm tra	Khoảng cách thực tế(m)	Khoảng cách hệ thống đo được(m)	Sai số(%)
1	4	4,18	4,5
2	4	4,25	6,25
3	4	3,86	3,5
4	4	4,29	7,25
5	4	4,22	5,5
6	4	3,81	4,75
7	4	3,83	4,25
8	4	3,88	3

Ứng dụng ESP32 trong hệ thống định vị vật thể

9	4	4,27	6,75
10	4	4,19	4,75
11	4	4,15	3,75
12	4	3,8	5
13	4	3,85	3,75
14	4	3,78	5,5
15	4	3,75	6,25
16	4	4,16	4
17	4	4,21	5,25
18	4	4,23	5,75
19	4	4,2	5
20	4	3,81	4,75
21	4	3,83	4,25
22	4	3,88	3
23	4	4,27	6,75
24	4	4,19	4,75
25	4	4,15	3,75
26	4	3,8	5
27	4	3,85	3,75
28	4	3,78	5,5
29	4	3,75	6,25
30	4	4,16	4
31	4	4,21	5,25
32	4	4,23	5,75
33	4	4,2	5
34	4	4,18	4,5
35	4	4,25	6,25
36	4	3,86	3,5
37	4	4,29	7,25
38	4	4,22	5,5
39	4	3,81	4,75
40	4	3,83	4,25

Ứng dụng ESP32 trong hệ thống định vị vật thể

41	4	3,88	3
42	4	4,27	6,75
43	4	4,19	4,75
44	4	4,15	3,75
45	4	3,8	5
46	4	3,85	3,75
47	4	3,78	5,5
48	4	3,75	6,25
49	4	4,16	4
50	4	4,21	5,25
51	4	4,23	5,75
52	4	4,2	5
53	4	3,81	4,75
54	4	3,83	4,25
55	4	3,88	3
56	4	3,81	4,75
57	4	3,83	4,25
58	4	3,88	3
59	4	4,27	6,75
60	4	4,19	4,75
61	4	4,15	3,75
62	4	3,8	5
63	4	3,85	3,75
64	4	3,78	5,5
65	4	3,75	6,25
66	4	4,16	4
67	4	4,21	5,25
68	4	4,23	5,75
69	4	4,2	5
70	4	4,18	4,5
71	4	3,78	5,5
72	4	3,75	6,25

Ứng dụng ESP32 trong hệ thống định vị vật thể

73	4	4,16	4
74	4	4,21	5,25
75	4	4,23	5,75
76	4	4,2	5
77	4	3,81	4,75
78	4	3,83	4,25
79	4	3,88	3
80	4	4,27	6,75
81	4	4,19	4,75
82	4	4,15	3,75
83	4	3,8	5
84	4	3,81	4,75
85	4	3,83	4,25
86	4	3,88	3
87	4	4,27	6,75
88	4	4,19	4,75
89	4	4,15	3,75
90	4	3,8	5
91	4	3,85	3,75
92	4	3,78	5,5
93	4	3,75	6,25
94	4	4,16	4
95	4	4,21	5,25
96	4	4,23	5,75
97	4	4,2	5
98	4	3,88	3
99	4	4,27	6,75
100	4	4,19	4,75

=>Trung bình sai số(%) của 100 lần kiểm tra là 4,8775%.

Bảng 4.5 Bảng dữ liệu đánh giá độ chính xác của trường hợp khoảng cách thực tế 5m

Lần kiểm tra	Khoảng cách thực tế(m)	Khoảng cách hệ thống đo được(m)	Sai số(%)
1	5	5,3	6
2	5	5,32	6,4
3	5	5,25	5
4	5	4,8	4
5	5	4,73	5,4
6	5	4,71	5,8
7	5	5,35	7
8	5	5,37	7,4
9	5	4,68	6,4
10	5	4,66	6,8
11	5	4,69	6,2
12	5	4,7	6
13	5	4,61	7,8
14	5	5,34	6,8
15	5	5,37	7,4
16	5	4,73	5,4
17	5	4,65	7
18	5	4,57	8,6
19	5	4,59	8,2
20	5	4,66	6,8
21	5	4,62	7,6
22	5	4,64	7,2
23	5	5,37	7,4
24	5	4,68	6,4
25	5	4,66	6,8
26	5	4,69	6,2
27	5	4,7	6

Ứng dụng ESP32 trong hệ thống định vị vật thể

28	5	4,61	7,8
29	5	5,34	6,8
30	5	5,37	7,4
31	5	4,73	5,4
32	5	4,65	7
33	5	4,57	8,6
34	5	4,59	8,2
35	5	4,66	6,8
36	5	4,62	7,6
37	5	4,64	7,2
38	5	5,3	6
39	5	5,32	6,4
40	5	5,25	5
41	5	4,8	4
42	5	4,73	5,4
43	5	4,71	5,8
44	5	5,35	7
45	5	5,37	7,4
46	5	4,68	6,4
47	5	4,66	6,8
48	5	4,69	6,2
49	5	4,7	6
50	5	4,61	7,8
51	5	5,34	6,8
52	5	5,37	7,4
53	5	4,73	5,4
54	5	4,65	7
55	5	4,57	8,6
56	5	4,59	8,2
57	5	4,66	6,8
58	5	4,62	7,6
59	5	4,64	7,2

Ứng dụng ESP32 trong hệ thống định vị vật thể

60	5	5,37	7,4
61	5	4,68	6,4
62	5	4,66	6,8
63	5	4,69	6,2
64	5	4,7	6
65	5	4,59	8,2
66	5	4,66	6,8
67	5	4,62	7,6
68	5	4,64	7,2
69	5	5,37	7,4
70	5	4,68	6,4
71	5	4,66	6,8
72	5	4,69	6,2
73	5	4,7	6
74	5	4,61	7,8
75	5	5,34	6,8
76	5	5,37	7,4
77	5	4,73	5,4
78	5	4,65	7
79	5	4,57	8,6
80	5	4,59	8,2
81	5	4,66	6,8
82	5	4,71	5,8
83	5	5,35	7
84	5	5,37	7,4
85	5	4,68	6,4
86	5	4,66	6,8
87	5	4,69	6,2
88	5	4,7	6
89	5	4,61	7,8
90	5	5,34	6,8
91	5	5,37	7,4

92	5	4,73	5,4
93	5	4,65	7
94	5	4,57	8,6
95	5	4,59	8,2
96	5	4,66	6,8
97	5	4,62	7,6
98	5	4,69	6,2
99	5	4,7	6
100	5	4,61	7,8

=>Trung bình sai số(%) của 100 lần kiểm tra là 6,806%.

Kết quả kiểm tra cho thấy được hệ thống định vị vật thể đo được khoảng cách trong phạm vi bán kính 5m với độ chính xác cao, sai số nhỏ hơn 10%. Kết luận, vì đo kiểm tra ở khoảng cách bán kính 5m nên độ chính xác đo được khá cao, khoảng cách càng xa thì giá trị RSSI sẽ càng lớn và độ chính xác khoảng cách đo được sẽ ngày càng giảm đi với biên độ sai số lớn.

4.4.2 Thực hiện kiểm tra và đánh giá thời gian phản ứng

Thực hiện thao tác nạp code vào ESP32 thời gian nạp thành công khoảng 10s.

Thực hiện thao tác trong thư viện giao tiếp Serial tính RSSI và hệ số môi trường n, mỗi giá trị tính trong thời gian 2 giây.

Cập nhật giá trị RSSI tính được sau khi tính RSSI và n, từ đó đưa ra khoảng cách đo được tương ứng cập nhật lên nền tảng web lưu trữ Google Sheet, giá trị cập nhật liên tục mỗi 3 giây.

4.4.3 Thực hiện kiểm tra và đánh giá độ bao phủ

Thực hiện di chuyển ESP32 xa dần so với vị trí Router cố định, di chuyển đến khi nào ESP32 không còn nhận được tín hiệu Wi-Fi từ router cố định và xác định vị trí xa nhất ESP32 nhận được tín hiệu Wifi từ router cố định.

=>Độ bao phủ của hệ thống định vị vật thể là 20m (khoảng cách tối đa để ESP32 thu được sóng Wi-Fi và giá trị RSSI).



Hình 4.11 Vị trí ESP32 cách router cố định 20m

4.5 Kết luận chương

Trong chương này đã trình bày về những kết quả đã đạt được trong phần thi công của hệ thống. Cùng với đó là trình bày về các cơ sở đánh giá hệ thống định vị vật thể. Qua đó cũng thể hiện rõ được những kết quả của mục tiêu ban đầu đề ra cho hệ thống, hoàn thành các nhiệm vụ cho 1 hệ thống hoàn chỉnh. Từ đó đưa ra các nhận xét, đánh giá và hướng phát triển cho việc phát triển và cải tiến hệ thống sau này.

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

A. KẾT LUẬN

Sau khoảng 10 tuần nghiên cứu và thực hiện đề tài tôi đã hoàn thành mô hình hệ thống theo yêu cầu và thu được những kết quả nhất định sau quá trình thực hiện:

- Hệ thống đạt yêu cầu về đo đạc và thu thập thông tin dữ liệu giá trị và khoảng cách chip ESP32 với router cố định.
- Hệ thống thực hiện được việc giao tiếp cập nhật dữ liệu lưu trữ từ Arduino IDE lên Google Sheet.
- Hệ thống có biện pháp bảo mật giúp dữ liệu an toàn trong quá trình truyền tải.
- Phạm vi hoạt động bán kính 20m (khoảng cách tối đa để ESP32 thu được sóng Wi-Fi và giá trị RSSI).
- Hệ thống có độ trễ thấp.
- Toàn bộ hệ thống chạy tương đối ổn định, đạt được kết quả độ chính xác cao đạt yêu cầu ban đầu đề ra. Tuy nhiên, do nhiễu bởi gió, vật cản,... trong môi trường truyền dẫn khiến hệ thống có lúc kết nối không ổn định.

Tổng hợp lại đề tài “Ứng dụng ESP32 trong hệ thống định vị vật thể” với chi phí thấp; dễ dàng triển khai; quản lý dữ liệu Google Sheet tiện lợi; tính khả thi cao cho cả mục đích học tập, nghiên cứu và các ứng dụng quy mô nhỏ.

Hệ thống cần được cải thiện nhằm nâng cao độ chính xác hạn chế - Đây là nhược điểm lớn nhất. Phương pháp RSSI vốn rất nhạy cảm với các yếu tố môi trường (nhiều, đa đường, vật cản, sự thay đổi của con người/đồ vật) làm cho cường độ tín hiệu không phản ánh chính xác khoảng cách. Do đó, độ chính xác định vị thường chỉ ở mức vài mét đến hàng chục mét, không phù hợp cho các ứng dụng đòi hỏi độ chính xác cao.

Việc xây dựng thành công hệ thống này là sự nỗ lực hết mình của tôi cùng sự dạy dỗ, chỉ bảo tận tình của các thầy cô. Xin chân thành cảm ơn giảng viên Trần Văn Líc đã rất tận tình hướng dẫn, chỉ bảo tôi trong suốt thời gian thực hiện đồ án vừa qua.

B. HƯỚNG PHÁT TRIỂN

Sau khi hoàn thành đề tài và đạt được những kết quả nhất định, tôi đặt ra các hướng phát triển cho hệ thống để có những hoàn thiện tốt hơn cho các ứng dụng vào thực tế. Nâng cao độ chính xác và độ bền của định vị:

- Bộ lọc tín hiệu nâng cao.

- Bộ lọc Kalman (Kalman Filter): Kết hợp các phép đo RSSI với mô hình động học của vật thể (ví dụ: thông tin từ cảm biến gia tốc/con quay hồi chuyển nếu có) để đưa ra ước tính vị trí chính xác và ổn định hơn theo thời gian.
- Bộ lọc Particle (Particle Filter): Phù hợp với các hệ thống phi tuyến tính, có thể xử lý tốt hơn các nhiễu trong môi trường thực.
- Kỹ thuật làm mịn nâng cao: Áp dụng các thuật toán như Exponential Moving Average (EMA) hoặc Robust Average để giảm ảnh hưởng của các giá trị ngoại lai.

Phương pháp định vị khác:

- Fingerprinting (Bản đồ cường độ tín hiệu): Thay vì sử dụng mô hình toán học, phương pháp này thu thập một "bản đồ" RSSI tại các điểm đã biết trong không gian. Khi định vị, hệ thống so sánh RSSI hiện tại với bản đồ để tìm vị trí gần đúng. Điều này thường mang lại độ chính xác cao hơn RSSI mô hình, nhưng tốn công sức thu thập dữ liệu ban đầu.
- Kết hợp nhiều công nghệ:

UWB (Ultra-Wideband): Tích hợp thêm module UWB vào ESP32 hoặc sử dụng song song để định vị chính xác hơn trong các khu vực nhỏ, nơi UWB có thể đạt độ chính xác đến centimet. RSSI có thể dùng để định vị thô, sau đó UWB tinh chỉnh.

BLE (Bluetooth Low Energy): Kết hợp với BLE Beacons cho định vị tầm gần hoặc trong các khu vực dày đặc.

IMU (Inertial Measurement Unit): Sử dụng dữ liệu từ cảm biến gia tốc và con quay hồi chuyển trên ESP32 để ước tính vị trí giữa các lần đo RSSI (dead reckoning), sau đó dùng RSSI để hiệu chỉnh tích lũy sai số.

- Định vị đa điểm neo: Mặc dù đã dùng nhiều điểm router, có thể tối ưu hóa vị trí đặt các điểm router để cải thiện "hình học" của hệ thống (Geometry Dilution of Precision - GDOP), làm cho ước tính vị trí ít bị ảnh hưởng bởi sai số đo.

Cải thiện quản lý dữ liệu và khả năng mở rộng:

- Chuyển đổi nền tảng dữ liệu.

- Cơ sở dữ liệu chuyên dụng: Đối với các ứng dụng lớn hơn, thay thế Google Sheets bằng các cơ sở dữ liệu mạnh mẽ hơn như Firebase Realtime Database/Firestore (cũng của Google, dễ tích hợp và có API thời gian thực), InfluxDB (cho dữ liệu chuỗi thời gian), hoặc PostgreSQL. Điều này cung cấp khả năng truy vấn, lưu trữ và xử lý dữ liệu tốt hơn.
- Nền tảng IoT chuyên nghiệp: Sử dụng các nền tảng IoT như AWS IoT Core, Google Cloud IoT Core, hoặc ThingsBoard để quản lý thiết bị, truyền dữ liệu và tích hợp với các dịch vụ phân tích dữ liệu khác một cách hiệu quả hơn.
- Xử lý dữ liệu ở biên (Edge Computing):

Thay vì gửi RSSI thô và khoảng cách lên Google Sheets để xem, có thể thực hiện thuật toán định vị (tam giác hóa, đa giác hóa) trên một máy chủ cục bộ (ví dụ: Raspberry Pi) hoặc thậm chí ngay trên ESP32 (nếu tài nguyên cho phép). Điều này giảm độ trễ và giảm tải cho đám mây.

Giao thức truyền thông hiệu quả hơn: Đối với lượng dữ liệu lớn hoặc cập nhật thường xuyên, xem xét sử dụng MQTT (Message Queuing Telemetry Transport) thay vì HTTP POST. MQTT nhẹ hơn, hiệu quả hơn và được thiết kế cho các ứng dụng IoT.

Phát triển giao diện người dùng và tính năng nâng cao:

- Giao diện trực quan thời gian thực: Xây dựng một ứng dụng web hoặc di động để hiển thị vị trí vật thể trên một bản đồ (floor plan) trong thời gian thực, thay vì chỉ là dữ liệu trong bảng tính.
- Hàng rào địa lý (Geofencing): Thiết lập các khu vực ảo và tạo cảnh báo khi vật thể ra/vào các khu vực đó.
- Lịch sử di chuyển và phân tích: Hiển thị lịch sử đường đi của vật thể, phân tích thời gian dừng/đỗ, và các mẫu di chuyển để tối ưu hóa quy trình.
- Quản lý năng lượng thông minh: Triển khai các thuật toán quản lý năng lượng động, cho phép ESP32 tự điều chỉnh tần suất đo/gửi dữ liệu dựa trên mức pin hoặc tình trạng hoạt động.

Cải thiện bảo mật:

- Lưu trữ khóa bảo mật: Khám phá các phương pháp bảo mật hơn để lưu trữ Private Key của Service Account trên ESP32 (ví dụ: sử dụng NVS - Non-Volatile Storage, hoặc phần cứng bảo mật nếu có), thay vì nhúng trực tiếp vào mã nguồn.
- Mã hóa dữ liệu: Mã hóa dữ liệu trước khi gửi đi và giải mã ở phía server để tăng cường bảo mật thông tin.

Bằng cách đi theo những hướng phát triển này, hệ thống có thể từ một minh chứng khái niệm đơn giản trở thành một giải pháp định vị trong nhà mạnh mẽ, chính xác và có khả năng mở rộng cho nhiều ứng dụng thực tế.

TÀI LIỆU THAM KHẢO

[1] UnderstandingRSSI. [Online].

Xem tại: <https://www.metageek.com/training/resources/understanding-rssi/> [Ngày truy cập: 25/04/2025]

[2] Path Loss Model, RF and Digital Signal Processing for Software-Defined Radio, Tony J. Roupheal, 2009

[3] Rappaport, T. S. (2002). *Wireless Communications: Principles and Practice* (2nd ed.). Prentice Hall.

[4] Mishra, A. (2019). IoT Based Indoor Positioning System using RSSI and Trilateration. In *Advances in Signal Processing and Communication* (pp. 53-61). Springer.

[5] Mazrouei, S. K., & Mohammad, A. (2018). *Indoor Positioning System Using Wi-Fi Technology*. In *Wireless Communications and Mobile Computing*. IntechOpen.

[6] Yassin, M., P. S. Ho, S. S. A. Ghafar, and Z. A. Mohamad. (2019). *Performance Analysis of RSSI Based Indoor Localization Using ESP32 Wi-Fi*. In *International Conference on Wireless and Satellite Communications (WSC)*, pp. 1-6. IEEE.

[7] Li, X., Wang, J., Liu, L., & Zhang, Y. (2016). *An RSSI-based Indoor Localization Algorithm with Dynamic Reference Points*. In *International Conference on Information and Communications Technology (ICT)*, pp. 1-5. IEEE.

[8] Espressif Systems.(n.d.).*ESP32-WROOM-32 Module*. Truy cập từ

<https://www.sapo.vn/blog/kich-thuoc-thiet-ke-website>.

[9] Bhattacharjee, A., & Das, S. K. (2015). *An RSSI Based Indoor Localization System Using Weighted k-Nearest Neighbor and Channel Smoothing*. In *International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pp. 182-187. IEEE.

[10] Liu, H., Darabi, H., Liu, P., & Greenberg, J. (2007). *Survey of Wireless Indoor Positioning Techniques and Systems*. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 37(6), 1067-1080.

[11] Thư viện ESP_Google_Sheet_Client. [Online].

Xem tại <https://github.com/mobizt/ESP-Google-Sheet-Client>

[Ngày truy cập: 20/05/2025] K. Suwatchai (Mobizt), 2024

PHỤ LỤC

CODE TOÀN BỘ HỆ THỐNG

```

1  /*
2  B1: Thay đổi name, pass Wifi
3  B2: Nạp code, đặt thiết bị ở vị trí 1m so với nguồn phát và nhập "1" vào Serial
4  B3: Thay đổi đến vị trí khác, sau đó nhập khoảng cách khi đó, lúc đó sẽ chuyển sang chế độ đo khoảng cách
5  (B4): Nếu muốn set lại các giá trị thì nhấn 0 để làm lại
6  */
7  #include <Arduino.h>
8  #include <WiFi.h>
9  #include "time.h"
10 #include <ESP_Google_Sheet_Client.h>
11 #include <GS_SoHelper.h>
12 /* THAY ĐỔI NAME VÀ PASS WIFI */
13 // #define WIFI_NAME "Kiên"
14 // #define WIFI_PASS "11052002"
15 #define WIFI_NAME "vngalaxy2g"
16 #define WIFI_PASS "vngalaxy2024"
17
18 // Google Project ID
19 #define PROJECT_ID "rssi-data"
20
21 // Service Account's client email
22 #define CLIENT_EMAIL "rssi-data@rssi-data.iam.gserviceaccount.com"
23 // Service Account's private key
24 const char PRIVATE_KEY[] PROGMEM = "-----BEGIN PRIVATE KE
25                                     Loading...
26                                     rQIBADANBgkqhkiG9w0BAQEFAASCBCwggSjAgEAAoIBAQC4Kj7L7iF8nGVR\nnbqiy7C1RNJMah3kUSIBAx1Nw5Z/dAnnPW6KF/odTRVnvz1mwST5
27 // The ID of the spreadsheet where you'll publish the data
28 const char spreadsheetId[] = "1f71n3D2-1jqdruKPMTy1m5CIa19Ah_HhCBGexPqay80";
29
30 float measuredPower = 0; // RSSI ở 1m
31 float n = 0; // Giá trị suy hao môi trường
32 int mode = 1; // Bắt đầu với mode 1
33 float rssiAvg = 0; // Giá trị RSSI trung bình
34
35 float distanceAvg = 0; // Khoảng cách trung bình
36 int number_row = 1;
37
38 const char *ntpServer = "pool.ntp.org";
39
40 void setup()
41 {
42   Serial.begin(115200);
43   Serial.println();
44   Serial.println();
45
46   configTime(0, 0, ntpServer);
47   GSheet.printf("ESP Google Sheet Client v%s\n\n", ESP_GOOGLE_SHEET_CLIENT_VERSION);
48
49   // Connect to Wi-Fi
50   WiFi.setAutoReconnect(true);
51   WiFi.begin(WIFI_NAME, WIFI_PASS);
52
53   Serial.print("connecting to Wi-Fi");
54   while (WiFi.status() != WL_CONNECTED)
55   {
56     Serial.print(".");
57     delay(1000);
58   }
59   Serial.println();
60   Serial.print("connected with IP: ");
61   Serial.println(WiFi.localIP());
62   Serial.println();
63
64   GSheet.setTokenCallback(tokenStatusCallback);
65   GSheet.setPrereshSeconds(10 * 60);
66   GSheet.begin(CLIENT_EMAIL, PROJECT_ID, PRIVATE_KEY);

```

Ứng dụng ESP32 trong hệ thống định vị vật thể

```
68 void loop()
69 {
70   rssiAvg = measureRSSI();
71   bool ready = GSheet.ready();
72
73   if (mode == 1)
74   {
75     Serial.println("Mode 1 (Read RSSI): " + String(rssiAvg) + " dBm");
76     Serial.println("Put '1' to get measuredPower value");
77   }
78   else if (mode == 2)
79   {
80     Serial.println("Mode 2 (Read RSSI): " + String(rssiAvg) + " dBm");
81     Serial.println("Put other distance to get n value");
82   }
83   else if (mode == 3)
84   {
85     distanceAvg = calculateDistance(measuredPower, rssiAvg, n);
86     Serial.println("Mode 3 (Read distance): " + String(distanceAvg) + " m");
87     Serial.println("Put '0' to change to Mode 1");
88
89     if (ready)
90     {
91       FirebaseJson response;
92       FirebaseJson valueRange;
93
94       valueRange.add("majorDimension", "COLUMNS");
95       valueRange.set("values/[0]/[0]", number_row);
96       valueRange.set("values/[1]/[0]", rssiAvg);
97       valueRange.set("values/[2]/[0]", distanceAvg);
98
99       number_row++;
100
101       bool success = GSheet.values.append(&response, spreadsheetId, "Sheet1!A1", &valueRange);
102       if (success)
103       {
104         response.toString(Serial, true);
105         valueRange.clear();
106       }
107       else
108       {
109         Serial.println(GSheet.errorReason());
110       }
111     }
112   }
113
114   if (Serial.available())
115   {
116     String input = Serial.readStringUntil('\n');
117     input.trim();
118
119     if (mode == 1 && input == "1")
120     {
121       measuredPower = rssiAvg;
122       Serial.print("=> Save measuredPower = ");
123       Serial.println(measuredPower);
124       mode = 2;
125     }
126     else if (mode == 2)
127     {
128       float d = input.toFloat();
129       if (d > 0 && d != 1)
130       {
131         n = calculateN(measuredPower, rssiAvg, d);
132         Serial.print("=> Save n = ");
133         Serial.println(n);
134
135         mode = 3;
136       }
137       else
138       {
139         Serial.println("Put again !!!");
140       }
141     }
142     else if (mode == 3 && input == "0")
143     {
144       measuredPower = 0;
145       n = 0;
146       mode = 1;
147       Serial.println("=> Reset to Mode 1");
148     }
149     Serial.println("-----");
150     delay(1000);
151   }
152
153   void tokenStatusCallback(TokenInfo info)
154   {
155     if (info.status == token_status_error)
156     {
157       GSheet.printf("Token info: type = %s, status = %s\n", GSheet.getTokenType(info).c_str(), GSheet.getTokenStatus(info).c_str());
158       GSheet.printf("Token error: %s\n", GSheet.getTokenError(info).c_str());
159     }
160     else
161     {
162       GSheet.printf("Token info: type = %s, status = %s\n", GSheet.getTokenType(info).c_str(), GSheet.getTokenStatus(info).c_str());
163     }
164   }
165
166   // Hàm đo RSSI trung bình
```

Ứng dụng ESP32 trong hệ thống định vị vật thể

```
167 float measureRSSI()
168 {
169     long rssiSum = 0;
170
171     for (int i = 0; i < SAMPLES; i++)
172     {
173         rssiSum += WiFi.RSSI();
174         delay(10);
175     }
176
177     return (float)rssiSum / SAMPLES;
178 }
179
180 // Hàm tính n từ measuredPower và RSSI tại khoảng cách d
181 float calculateN(float measuredPower, float rssiAtD, float d)
182 {
183     return (measuredPower - rssiAtD) / (10 * log10(d));
184 }
185
186 // Hàm tính khoảng cách từ RSSI
187 float calculateDistance(float measuredPower, float rssi, float n)
188 {
189     return pow(10.0, (measuredPower - rssi) / (10 * n));
190 }
```