

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**



**BÁO CÁO TỔNG KẾT
ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN**

**THIẾT KẾ VÀ THI CÔNG ROBOT NHẬN DI CHUYỂN TỰ ĐỘNG HOẶC
ĐIỀU KHIỂN BẰNG ĐIỆN THOẠI**

MÃ SỐ: SV2024-161

CHỦ NHIỆM ĐỀ TÀI: NGÔ TRỌNG NGHĨA

TP Hồ Chí Minh, tháng 11 năm 2024

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**



**BÁO CÁO TỔNG KẾT
ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN**

**THIẾT KẾ VÀ THI CÔNG ROBOT NHẬN DI CHUYỂN TỰ ĐỘNG HOẶC
ĐIỀU KHIỂN BẰNG ĐIỆN THOẠI**

Mã số: SV2024-161

Thuộc nhóm ngành khoa học: Kỹ Thuật

SV thực hiện: Ngô Trọng Nghĩa

Nam, nữ: Nam

Dân tộc: Kinh

Lớp, Khoa: 21119CL1 – Khoa Điện-Điện Tử Năm thứ: 4/4

Ngành học: : CNKT Máy Tính

Giảng viên hướng dẫn: ThS. Huỳnh Hoàng Hà

TP Hồ Chí Minh, tháng 11 năm 2024

MỤC LỤC

CHƯƠNG 1	1
TỔNG QUAN VỀ ĐỀ TÀI	1
1.1. LÝ DO CHỌN ĐỀ TÀI	1
1.2. MỤC TIÊU ĐỀ TÀI.....	1
1.3. NỘI DUNG NGHIÊN CỨU	1
1.4. GIỚI HẠN.....	1
1.5. BỐ CỤC	2
CHƯƠNG 2	3
CƠ SỞ LÝ THUYẾT	3
2.1. TỔNG QUAN VỀ PHƯƠNG PHÁP DI CHUYỂN	3
2.1.1. Giới thiệu về tọa độ Descartes.....	3
2.1.2. Giới thiệu về tọa độ cực.....	3
2.1.3. Khoảng cách Euclid.	5
2.1.4. Phương pháp quy đổi mô hình toán học sang góc quay servo.	5
2.2. GIỚI THIỆU VỀ GIAO THỨC BLUETOOTH	7
2.3. TỔNG QUAN VỀ ĐIỀU CHẾ ĐỘ RỘNG XUNG.....	8
2.4. GIỚI THIỆU PHẦN CỨNG SỬ DỤNG	9
2.4.1. Module ESP32 NodeMCU LuaNode32 38 Chân.....	9
2.4.2. Servo SG90 180	12
2.4.3. Pin lithium 18650 Ion 3.7V	14
2.4.4. Mạch sạc pin lithium 2A typeC	15
2.5. GIỚI THIỆU PHẦN MỀM SỬ DỤNG.....	16
2.5.1. Arduino IDE.....	16
2.5.2. Altium Designer	17
2.5.3. MIT App Inventor	18
CHƯƠNG 3	20
THIẾT KẾ HỆ THỐNG	20
3.1. GIỚI THIỆU	20
3.2. SƠ ĐỒ KHỐI HỆ THỐNG.....	20
3.2.1 Sơ đồ khối	20

3.2.2. Chức năng từng khối.....	21
3.3. SƠ ĐỒ NGUYÊN LÝ MẠCH.....	22
3.4. THIẾT KẾ PHẦN MỀM.....	25
3.4.1. Hoạt động hệ thống.....	25
3.4.2. Chức năng đi thẳng và đi lùi.....	28
3.4.3. Chức năng xoay trái và xoay phải	30
3.4.4. Chức năng vẫy tay chào.....	33
3.4.5. Chức năng đứng và ngồi.....	34
3.4.6. Giao diện ứng dụng điều khiển trên Android	35
CHƯƠNG 4.....	37
THI CÔNG HỆ THỐNG.....	37
4.1. GIỚI THIỆU	37
4.2. MÔ HÌNH PHẦN CỨNG.....	37
4.3. GIAO DIỆN PHẦN MỀM.....	39
4.4. HƯỚNG DẪN SỬ DỤNG	40
4.4.1. Trên mô hình.....	40
4.4.2. Trên ứng dụng di động.....	40
CHƯƠNG 5	42
KẾT QUẢ.....	42
5.1. KẾT QUẢ ĐẠT ĐƯỢC.....	42
5.1.1. Kết quả sau thi công mạch điều khiển	42
5.1.2. Kết quả sau thi công mô hình robot.....	44
5.2. KIỂM CHỨNG BẰNG THỰC NGHIỆM.....	45
CHƯƠNG 6	47
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN.....	47
6.1. KẾT LUẬN	47
6.2. HƯỚNG PHÁT TRIỂN	48
TÀI LIỆU THAM KHẢO	49

DANH MỤC HÌNH ẢNH

Hình 2.1: Biểu diễn tọa độ trên tọa độ Descartes.	3
Hình 2.2: Biểu diễn tọa độ trên tọa độ cực.	4
Hình 2.3: Công thức định lý cosin trong tam giác.	4
Hình 2.4: Khoảng cách Euclid trong tọa độ Descartes.	5
Hình 2.5: Mô tả góc quay của Servo.	6
Hình 2.6: Mô tả góc quay 150 độ của Servo.	6
Hình 2.7: Mô tả góc quay đối xứng của Servo.	7
Hình 2.8: Sơ đồ mô tả giao thức Bluetooth.	8
Hình 2.9: Ví dụ về điều chế độ rộng xung.	9
Hình 2.10: Module ESP32 NodeMCU LuaNode32 38 Chân	9
Hình 2.11: Sơ đồ chân của Module ESP32 NodeMCU LuaNode32 38 Chân	10
Hình 2.12: Servo S90 180	13
Hình 2.13. Pin Lithium Ion 3.7 V 6800mAh.	14
Hình 2.14: Mạch sạc xả pin 18650 typeC	15
Hình 2.15: Logo Arduino IDE.	16
Hình 2.16: Giao diện bắt đầu của Altium Designer.	18
Hình 2.17: Logo MIT App Inventor.	19
Hình 3.1: Sơ đồ khối hệ thống.	20
Hình 3.2: Sơ đồ khối mạch điện.	22
Hình 3.3: Sơ đồ nguyên lý mạch điều khiển.	23
Hình 3.4: Lưu đồ hoạt động của robot nhện.	25
Hình 3.5: Lưu đồ cho hàm set_site.	26
Hình 3.6: Lưu đồ cho hàm chuyển đổi tọa độ cực sang góc quay Servo.	27
Hình 3.7: Lưu đồ cho chức năng đi thẳng.	28
Hình 3.8: Lưu đồ cho chức năng xoay trái.	30
Hình 3.9: Lưu đồ cho chức năng vẫy tay chào.	33
Hình 3.10: Lưu đồ cho ứng dụng điện thoại Android.	35
Hình 4.1: Mạch in PCB trong giai đoạn thiết kế bằng phần mềm Altium.	37
Hình 4.2: Mạch in PCB ở dạng 3D.	38
Hình 4.3: Mô hình robot nhện.	38
Hình 4.4: Giao diện ứng dụng điều khiển trên Android.	39
Hình 5.1: Mô hình mạch điều khiển bằng ESP32.	42
Hình 5.2: Mô hình robot nhện 4 chân.	44
Hình 5.3: Thực nghiệm mô hình robot nhện 4 chân trên môi trường bằng phẳng.	45
Hình 5.4: Thực nghiệm mô hình robot nhện 4 chân trên đoạn đường dốc.	45
Hình 5.5: Thực nghiệm mô hình robot nhện 4 chân trên môi trường đá không bằng phẳng.	46

DANH MỤC BẢNG BIỂU

Bảng 2.1: Bảng thông số kỹ thuật của Module Module ESP32 NodeMCU LuaNode32 38 Chân.....	10
Bảng 2.2: Bảng thông số kỹ thuật của servo SG90 180.....	13
Bảng 2.3: Bảng thông số kỹ thuật của Pin Lithium Ion 3.7V 6800mAh.	14
Bảng 2.4: Bảng thông số kỹ thuật của mạch sạc xả pin 18650 typeC.	15

DANH MỤC NHỮNG TỪ VIẾT TẮT

ADC	Analog-to-Digital Converter
BLE	Bluetooth Low Energy
BMS	Battery Management System
DAC	Digital-to-Analog Converter
DC	Direct Current
FPGA	Field-Programmable Gate Array
GPIO	General-Purpose Input/Output
I2C	Inter-Integrated Circuit
IDE	Integrated Development Environment
IoT	Internet of Things
LED	Light-Emitting Diode
MAC	Media Access Control
MCU	Microcontroller Unit
PCB	Printed Circuit Board
PWM	Pulse-Width Modulation
SoC	System on Chip
SPI	Serial Peripheral Interface
UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐH SƯ PHẠM KỸ THUẬT TP.HCM

THÔNG TIN KẾT QUẢ NGHIÊN CỨU CỦA ĐỀ TÀI

1. Thông tin chung:

- Tên đề tài: Thiết kế và thi công robot di chuyển tự động hoặc điều khiển bằng điện thoại.

- Chủ nhiệm đề tài: Ngô Trọng Nghĩa

Mã số SV: 21161155

- Lớp: 21119CL1

Khoa: Điện-Điện tử

- Thành viên đề tài:

STT	Họ và tên	MSSV	Lớp	Khoa
1	Đàm Thuận An	21119325	21119CL1	Điện-Điện tử
2	Nguyễn Phước Dư	21119327	21119CL1	Điện-Điện tử
3	Ngô Trọng Nghĩa	21161155	21119CL1	Điện-Điện tử
4	Hồng Lý Trung Nhân	21119109	21119CL1	Điện-Điện tử
5	Đỗ Trần Tuấn Vũ	21119162	21119CL1	Điện-Điện tử

- Người hướng dẫn: ThS. Huỳnh Hoàng Hà

2. Mục tiêu đề tài:

Thiết kế và thi công robot nhện có thể được điều khiển bằng điện thoại và phát triển phần mềm trên hệ điều hành Android với mục đích điều khiển robot. Mô hình này sẽ dựa trên các mô hình toán học để có thể di chuyển được cũng như là kết hợp với giao thức bluetooth để điều khiển robot một cách hiệu quả. Mô hình này sẽ được ứng dụng nhiều trong các môi trường chật hẹp cũng như là các môi trường có không khí không tốt nhằm thay thế con người làm những việc tại môi trường đó.

3. Tính mới và sáng tạo:

Về mong muốn điểm mới của đề tài sẽ là kết hợp cả hai tính năng di chuyển tự động và điều khiển bằng điện thoại và mong muốn chuyển đổi giữa các chế độ này sang chế độ các một cách linh hoạt, nhưng hiện tại nhóm chưa thể thực hiện được chức năng có thể di chuyển tự động và mong muốn nó sẽ được hiện thực hóa trong tương lai. Về các chức năng cơ bản là di chuyển cũng như có thể điều khiển bằng điện thoại thông qua giao thức bluetooth và ứng dụng có thể chạy trên hệ điều hành Android.

4. Kết quả nghiên cứu:

Nghiên cứu và phát triển robot nhện 4 chân:

- + Nghiên cứu hoạt động của servo để điều khiển các khớp chân với các chức năng di chuyển cơ bản.

- + Nghiên cứu phát triển mobile app để điều khiển robot nhện từ xa thông qua giao thức Bluetooth.

5. Đóng góp về mặt giáo dục và đào tạo, kinh tế - xã hội, an ninh, quốc phòng và khả năng áp dụng của đề tài:

Kết quả nghiên cứu của đề tài này sẽ góp phần cho các công cuộc thám hiểm cũng như về công nghiệp robot ngày càng có cơ hội được sử dụng nhiều hơn trong tương lai.

Sản phẩm của đề tài này được nghiên cứu là để thay thế con người có thể tham quan các môi trường khó khăn mà con người khó đáp ứng được như môi trường chật hẹp, khí độc,...

6. Công bố khoa học của sinh viên từ kết quả nghiên cứu của đề tài:

Không có công bố khoa học của sinh viên từ kết quả nghiên cứu của đề tài.

Ngày tháng năm 2024

SV chịu trách nhiệm chính thực hiện đề tài

(kí, họ và tên)

Nhận xét của người hướng dẫn về những đóng góp khoa học của SV thực hiện đề tài:

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Ngày tháng năm 2024

Giảng viên hướng dẫn

(kí, họ và tên)

CHƯƠNG 1

TỔNG QUAN VỀ ĐỀ TÀI

1.1. LÝ DO CHỌN ĐỀ TÀI

Trong xã hội ngày nay, robot đang ngày càng được phổ biến trong hầu hết các lĩnh vực như là trong công nghiệp, nông nghiệp, cũng như là công nghệ,... Robot đều có thể đáp ứng được những công việc mà con người muốn. Với một thời đại tự động hóa đang phát triển rất mạnh, thì robot cũng vẫn là một chủ đề được rất nhiều người quan tâm với mục đích tự động hóa mọi việc, cũng như việc đó thì có thể có hoặc không thể đáp ứng từ con người, chẳng hạn như là thám hiểm một môi trường khí độc, môi trường chật hẹp, hoặc là môi trường khó khăn khác,... Vì thế, để đáp ứng được điều đó, tác giả chọn đề tài “Thiết kế và thi công robot di chuyển tự động hoặc điều khiển bằng điện thoại” với mục đích ứng dụng trong các môi trường mà con người khó đáp ứng được điều kiện môi trường đó.

1.2. MỤC TIÊU ĐỀ TÀI

Thiết kế và thi công robot nhện có khả năng di chuyển ở nhiều loại địa hình khác nhau, những môi trường không khí khác nhau.

Phát triển ứng dụng di động trên Android để có thể điều khiển robot thông qua Bluetooth.

1.3. NỘI DUNG NGHIÊN CỨU

- Tìm kiếm tài liệu, phân tích chức năng hệ thống.
- Xây dựng phần cứng hệ thống.
- Viết chương trình điều khiển phần mềm.
- Kiểm thử chức năng và điều chỉnh phù hợp.
- Viết báo cáo tổng kết.

1.4. GIỚI HẠN

Ở đề tài này, tác giả đã thu hẹp phạm vi sử dụng cũng như các chức năng của mô hình như:

- Chỉ sử dụng mô hình này ở các địa hình trên cạn với quy mô phù hợp với kích thước thực tế của mô hình. Không sử dụng mô hình này ở các địa hình không phù hợp như là dưới nước, các địa hình gồ ghề với mức độ không thể đáp ứng của mô hình,...
- Chỉ tập trung nghiên cứu các chức năng cơ bản mà một robot nên có chẳng hạn như là di chuyển thẳng, lùi, trái, phải,...

1.5. BỐ CỤC

Bố cục được chia làm 6 chương:

Chương 1: Tổng quan về đề tài

Giới thiệu vấn đề thực tế mà đề tài giải quyết được, nêu lý do chọn chủ đề, đặt mục tiêu nghiên cứu, trình bày nội dung chi tiết, xác định giới hạn nghiên cứu và mô tả cấu trúc của báo cáo.

Chương 2: Cơ sở lý thuyết

Tập trung vào các kiến thức lý thuyết có liên quan ảnh hưởng đến chức năng phần cứng và phần mềm được sử dụng trong đề tài.

Chương 3: Thiết kế hệ thống

Dựa trên các kiến thức lý thuyết đã được trình bày, chương này mô tả quá trình thiết kế hệ thống thông qua sơ đồ khối, mô tả lưu đồ của từng chức năng trong hệ thống và hiện thực hóa trên mô hình.

Chương 4: Thi công hệ thống

Chương này bao gồm việc thực hiện thiết kế mạch PCB, quy trình lắp đặt mô hình, sử dụng phần mềm cùng với mô tả chi tiết hướng dẫn sử dụng cả phần cứng lẫn phần mềm.

Chương 5: Kết quả

Tổng hợp kết quả và kiểm chức chức năng để đánh giá độ chính xác và khả năng ứng dụng của thiết bị.

Chương 6: Kết luận và Hướng phát triển

Đưa ra kết luận về đề tài và đề xuất các hướng phát triển cho đề tài trong tương lai.

CHƯƠNG 2

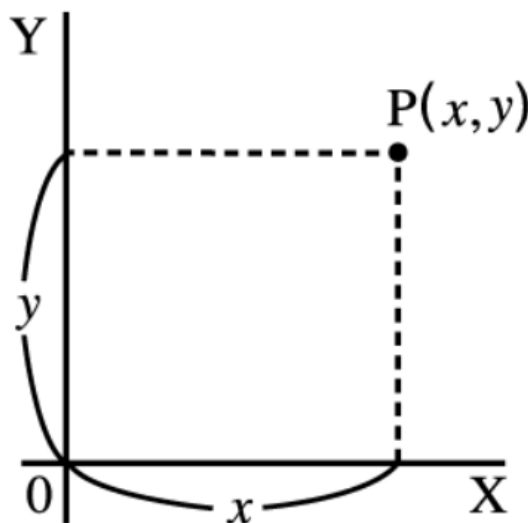
CƠ SỞ LÝ THUYẾT

2.1. TỔNG QUAN VỀ PHƯƠNG PHÁP DI CHUYỂN

2.1.1. Giới thiệu về tọa độ Descartes

Là vị trí của một điểm trên mặt phẳng được xác định dựa trên cặp số tọa độ x và y . Trong đó, giá trị của x và y lần lượt nằm trên 2 đường thẳng vuông góc với nhau. Điểm giao nhau của hai đường thẳng là gốc tọa độ. Tọa độ của điểm trên mặt phẳng được viết $P(x, y)$, trong đó x là hoành độ, y là tung độ của điểm.

Trong không gian 3 chiều, tọa độ **Descartes** dựa trên 3 trục vuông góc với nhau từng đôi một. Được xác định bởi $P(x, y, z)$

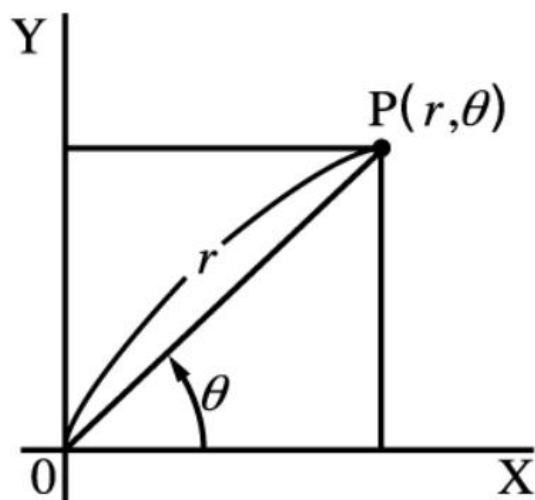


Hình 2.1: Biểu diễn tọa độ trên tọa độ Descartes.

2.1.2. Giới thiệu về tọa độ cực

Biểu diễn vị trí của một điểm trong không gian hai chiều bằng cách sử dụng hai thông số chính: bán kính (radius) và góc (angle).

Bán kính(r) là khoảng cách từ điểm đó đến gốc O, góc (θ) tạo bởi đường thẳng từ O đến điểm đó và hướng gốc cho trước.



Hình 2.2: Biểu diễn tọa độ trên tọa độ cực.

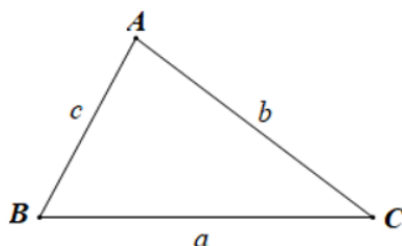
Chuyển đổi giữa tọa độ Descartes sang tọa độ Cực:

Quy đổi từ tọa độ vuông góc(x , y) sang tọa độ cực(r , θ) bằng cách sử dụng các công thức chuyển đổi:

$$r = \sqrt{x^2 + y^2}$$

$$\theta = \tan^{-1} \left(\frac{y}{x} \right)$$

Định lý cosin trong tam giác:



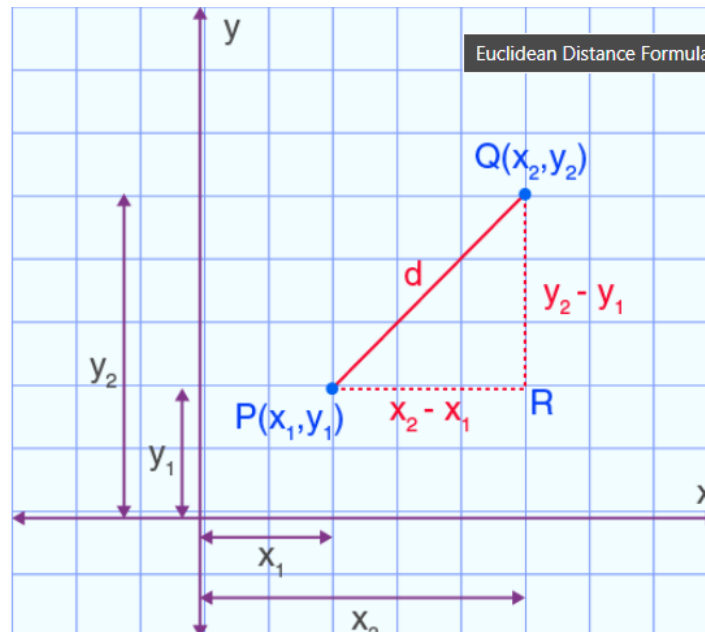
$$\cos A = \frac{b^2 + c^2 - a^2}{2bc}$$

Hình 2.3: Công thức định lý cosin trong tam giác.

$$\text{Atan2}(x, y) = \text{góc của arctan } y/x$$

2.1.3. Khoảng cách Euclid.

Trong đại số, khoảng cách Euclid là một phương pháp để có thể tính toán khoảng cách giữa hai điểm trong không gian và kết quả là độ dài đoạn thẳng giữa hai điểm đó và nó được sử dụng nhiều trong tọa độ Descartes bằng cách sử dụng định lý Py-ta-go, ở một số tài liệu khoảng cách Euclid có thể được gọi bằng khoảng cách Py-ta-go.



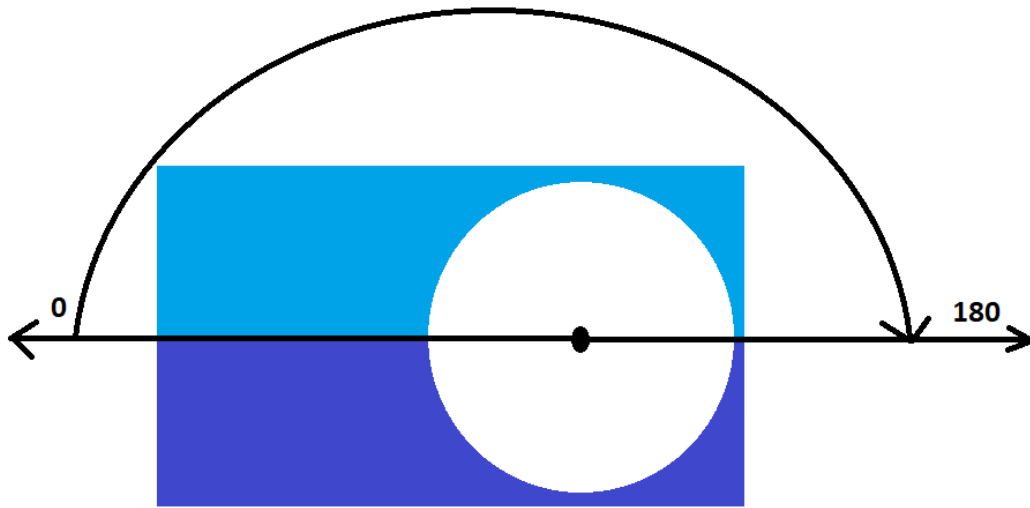
Hình 2. 4: Khoảng cách Euclid trong tọa độ Descartes.

2.1.4. Phương pháp quy đổi mô hình toán học sang góc quay servo.

Servo motor 180 có thể điều chỉnh góc quay dựa trên việc điều chỉnh độ rộng xung. Trong mã nguồn, để có thể điều khiển độ quay của servo theo ý muốn, chúng ta cần truyền vào tham số tọa độ ta muốn tương ứng.

Vì servo 180 chỉ có giới hạn quay từ 0 đến 180 (khác với servo 360 có thể quay tròn) vì vậy cần hiểu nguyên lý để có thể truyền tọa độ cho servo hoạt động theo ý muốn.

Nguyên lý truyền tọa độ cho điều khiển servo để điều khiển di chuyển của chân:



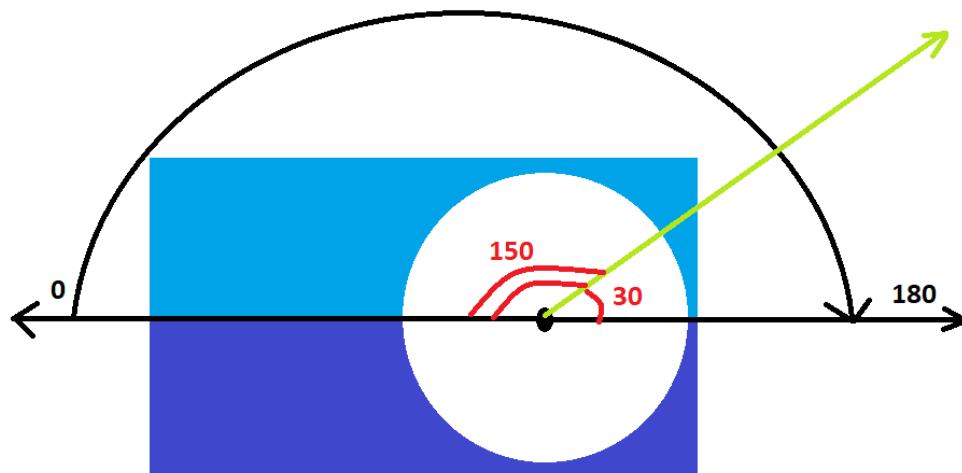
Hình 2. 5: Mô tả góc quay của Servo.

Với servo như hình:

- + Servo 180 có thể quay góc từ 0 đến 180 độ.
- + Trục 0 độ và 180 độ, chiều dương của trục quay.

Ta có các trường hợp sau:

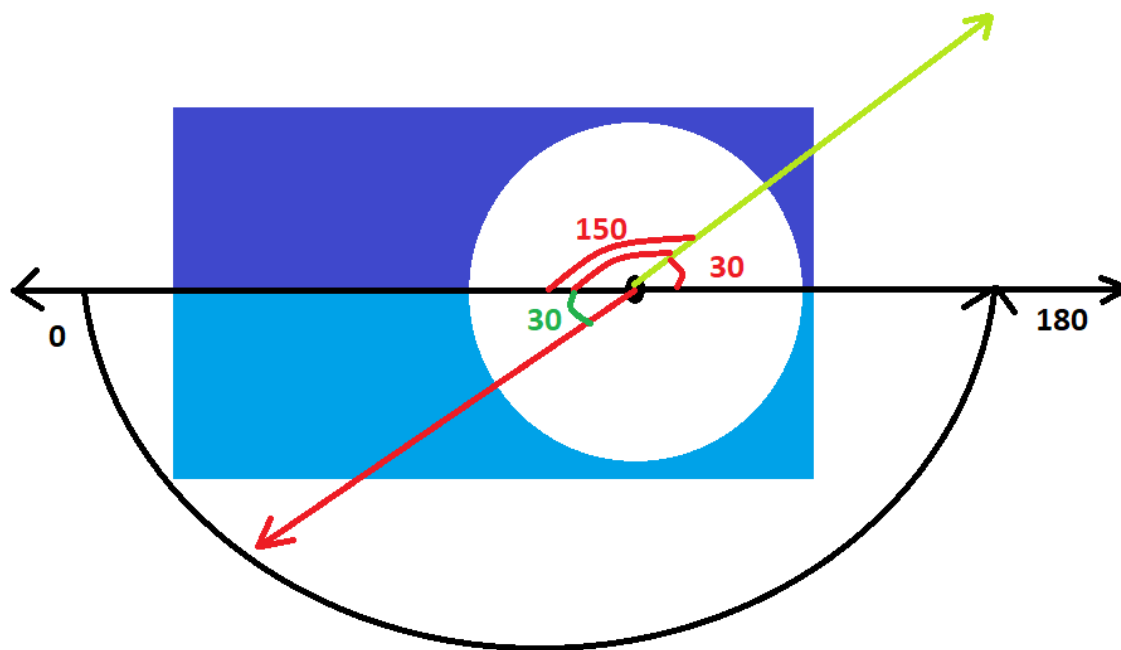
Trường hợp 1: Ta có đoạn chân muốn di chuyển với các thông số như hình:



Hình 2. 6: Mô tả góc quay 150 độ của Servo.

Với trường hợp này, để đoạn chân di chuyển đến vị trí đó thì thông số cần truyền cho servo là 150.

Trường hợp 2: Cũng với đoạn chân đó và ta muốn di chuyển đoạn chân như trường hợp 1, tuy nhiên đoạn chân này lại gắn bên nửa còn lại của servo:



Hình 2. 7: Mô tả góc quay đối xứng của Servo.

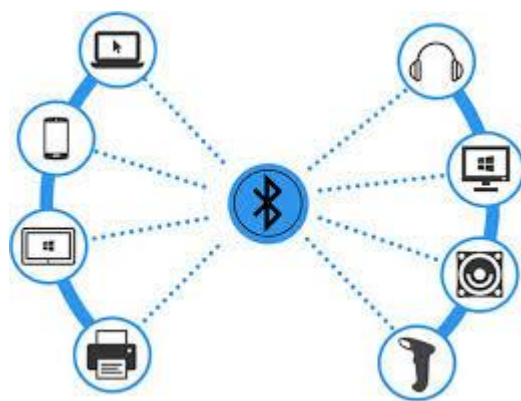
Với trường hợp này, để đoạn chân di chuyển với vị trí như trường hợp 1, ta không thể truyền cho servo thông số là $180+30 = 210$ (vì là servo chỉ quay được 180). Bằng cách lấy đối xứng qua điểm, thông số ta phải truyền vào cho servo để hoạt động là 30.

Điều này cho thấy để có thể điều khiển nhiều servo 180 cùng 1 lúc ta cần đồng bộ hệ trục cho các servo và truyền thông số cho servo cho phù hợp.

2.2. GIỚI THIỆU VỀ GIAO THỨC BLUETOOTH

Bluetooth là chuẩn kết nối không dây trong tầm gần thường là khoảng 10m ở băng tần 2.4GHz và cũng khá gần với chuẩn Wifi 2.4GHz hiện nay.

Về nguyên lý thì bluetooth sử dụng sóng radio trong băng tần 2.4GHz để truyền nhận dữ liệu không dây. Mỗi thiết bị sử dụng bluetooth luôn có một địa chỉ MAC riêng biệt để phân biệt với nhau giúp nhận diện cũng như để kết nối các thiết bị khác nhau.



Hình 2. 8: Sơ đồ mô tả giao thức Bluetooth.

Ưu điểm của bluetooth: bảo mật tốt cùng với các phương pháp mã hóa dữ liệu và dễ dàng kết nối.

Nhược điểm của bluetooth: Bị giới hạn phạm vi hoạt động trong khoảng 10m, cũng như là tốc độ truyền tải dữ liệu sẽ không cao như Wifi hoặc các chuẩn không dây khác. Và có một hạn chế khác thì bluetooth cũng đang có vấn đề về mặt tối ưu hóa năng lượng. Nhưng hiện nay thì cũng đã có các chuẩn BLE được ra đời với mục đích sử dụng ít năng lượng hơn với Bluetooth.

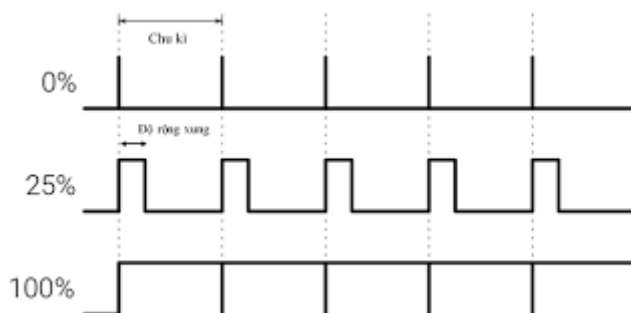
Ứng dụng của bluetooth: Thường sử dụng để kết nối các thiết bị không dây với các thiết bị cá nhân như là điện thoại và tai nghe, hay điện thoại và các thiết bị khác để truyền dữ liệu.

2.3. TỔNG QUAN VỀ ĐIỀU CHẾ ĐỘ RỘNG XUNG

Điều chế độ rộng xung là một kỹ thuật điều chế được điều khiển điện áp ra dựa trên sự thay đổi về độ rộng của xung hay có thể gọi là thay đổi thời gian điện áp mức cao của tín hiệu.

Điều chế độ rộng xung được điều chế dựa trên nguyên tắc đóng ngắt điện áp theo chu kỳ và điều chỉnh thời gian đóng ngắt. Linh kiện phổ biến nhất để sử dụng là các linh kiện bán dẫn. Hoạt động bằng cách tạo ra một xung điện một chiều có 2 mức bao gồm mức cao và mức thấp, đồng thời thay đổi thời gian đóng ngắt ở mỗi chu kỳ xung. Chẳng hạn như là tạo ra một xung có chu kỳ xung là 1ms, nếu như muốn điều

chế độ rộng xung với mức 70% thì thời gian đóng phải là 0.7ms và thời gian ngắt phải là 30% còn lại.



Hình 2. 9: Ví dụ về điều chế độ rộng xung.

Các ứng dụng của điều chế độ rộng xung như:

- Điều khiển động cơ (DC, Servo,...).
- Được ứng dụng trong các hệ thống thông tin vô tuyến.
- Ứng dụng trong chuyển mạch điều khiển tín hiệu bằng tín hiệu số.
- Điều chế mức điện áp khác nhau.

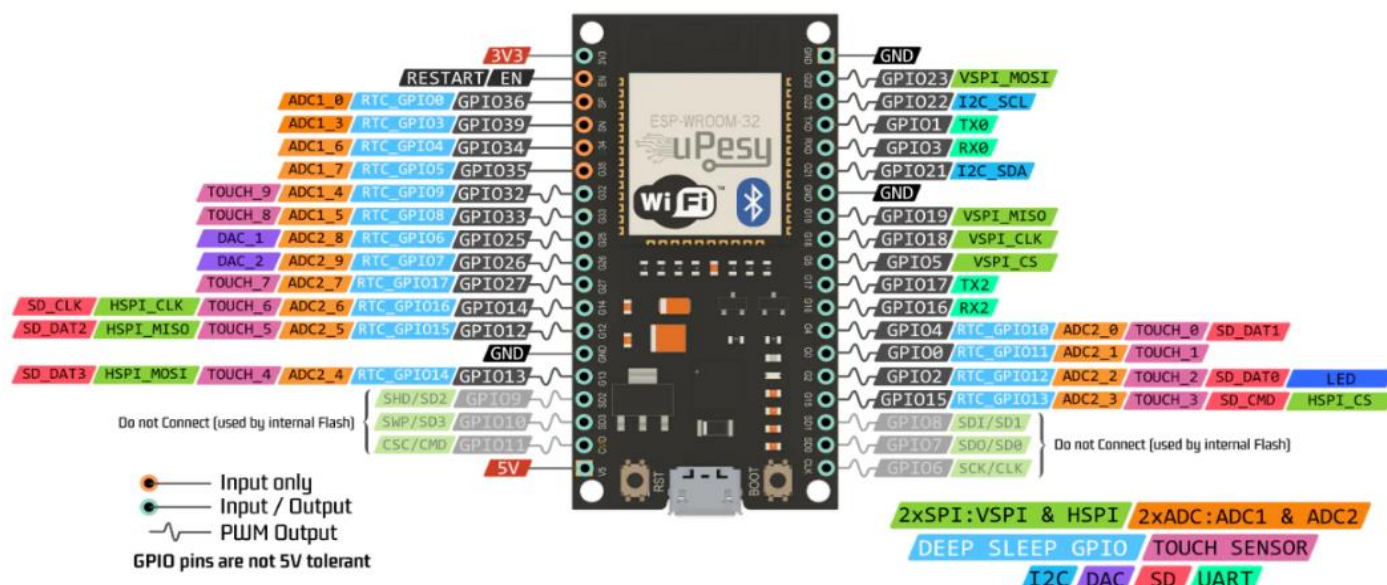
2.4. GIỚI THIỆU PHẦN CỨNG SỬ DỤNG

2.4.1. Module ESP32 NodeMCU LuaNode32 38 Chân

Module ESP32 NodeMCU LuaNode32 38 Chân là kit thu phát WiFi, Bluetooth dựa trên nền chip WiFi SoC ESP32. Được dùng cho các ứng dụng cần kết nối, thu thập dữ liệu và điều khiển qua sóng Wifi, Bluetooth đặc biệt là các ứng dụng liên quan đến IoT. Có thể dễ dàng sử dụng trực tiếp trình biên dịch của Arduino IDE để lập trình và nạp mã.



Hình 2.10: Module ESP32 NodeMCU LuaNode32 38 Chân



Thông số kỹ thuật

Vi xử lý	LX6 32-bit lõi đơn hoặc lõi kép
Điện áp nguồn (USB)	5V
Điện áp cấp (hoạt động tốt)	3.3V
Điện áp cấp (giới hạn)	3.0V – 3.6V
Chân I/O digital	38 (có 5 chân chỉ làm ngõ vào)
Chân Input digital	5 (GPIO34, GPIO35, GPIO36, GPIO39, EN)
Dao động thạch anh	40MHz
Flash	SPI Flash lên đến 32Mbit
SRAM	512 kBytes
ROM	448 kBytes
Công suất tiêu thụ	5μA (hệ thống treo chế độ)
Tốc độ xung nhịp	80 MHz - 240 MHz
Kích thước	2.8 x 5.5 cm
Nhiệt độ hoạt động	-40°C – 85°C

I/O Pins

- **GPIO:** Các chân GPIO có thể được cấu hình là đầu vào để đọc các tín hiệu từ nút nhấn, cảm biến, hoặc các tín hiệu từ các thiết bị khác. Và các chân GPIO (trừ chân GPIO34, GPIO35, GPIO36, GPIO39) cũng cho phép sử dụng như các chân đầu ra để điều khiển các thiết bị ngoại vi như LED, động cơ servo, báo động,... thông qua các hàm chính: `pinMode()`, `digitalWrite()`, `digitalRead()`.

- **PWM:** Hầu hết các chân GPIO (trừ GPIO6 – GPIO11, GPIO34, GPIO35, GPIO36, GPIO39) có thể được sử dụng để xuất tín hiệu PWM (Điều chế độ rộng xung) thông qua hàm `analogWrite()`.

- **ADC:** Các chân GPIO (0, 2, 4, 12, 13, 14, 15, 25, 26, 27, 32, 33, 34, 35, 36, 39) cho phép đo các tín hiệu Analog từ các cảm biến và chuyển đổi chúng thành giá trị số để xử lý trong các ứng dụng nhúng.

- **DAC:** Các chân GPIO25 và GPIO26 còn có thể được sử dụng để tạo ra các tín hiệu Analog từ các giá trị số, phù hợp cho việc điều khiển các mạch điều tần số, các loại cảm biến, ...

- **UART:** ESP32 cho phép truyền dữ liệu thông qua các chân GPIO3, 16(RX) và chân GPIO1, 17 (TX).

- **I2C:** ESP32 sử dụng các chân I2C (SDA: GPIO21, SCL: GPIO22) để giao tiếp với các thiết bị ngoại vi như cảm biến, màn hình, EEPROM,...thông qua giao thức I2C (giao thức truyền thông nối tiếp sử dụng hai dây để truyền dữ liệu giữa các thiết bị).

- **Deep Sleep:** Một số chân GPIO (0, 2, 4, 12, 13, 14, 15, 25, 26, 27, 32, 33, 34, 35, 36, 39) được sử dụng trong chế độ tiết kiệm năng lượng để đánh thức ESP32 từ giấc ngủ sâu (Deep Sleep).

- **Touch Sensor:** Các chân GPIO4, GPIO0, GPIO2, GPIO15, GPIO13, GPIO12, GPIO14, GPIO27, GPIO33, GPIO32 có thể được sử dụng cho các ứng dụng cảm ứng điện dung.

- **SPI:** Các chân GPIO18, GPIO23, GPIO19, GPIO5 được sử dụng để giao tiếp với các thiết bị ngoại vi như màn hình, cảm biến, thẻ nhớ SD,... thông qua giao thức SPI (giao thức

truyền thông nối tiếp đồng bộ, sử dụng nhiều dây để truyền dữ liệu giữa các thiết bị với tốc độ cao).

Nguồn

Có ba cách cấp nguồn chính cho ESP32: cổng USB, chân 5V và chân 3V3. Các chân nguồn trên ESP32:

- **5V:** Chấp nhận điện áp ngõ vào từ 5-12V. Tuy nhiên, điện áp này sẽ được bộ điều chỉnh điện áp trên bo mạch chuyển đổi xuống 3.3V để cung cấp cho ESP32.

- **3V3:** Nguồn 3V3 cấp cho chân này phải ổn định và chính xác. Nếu điện áp sai lệch quá mức ($\pm 5\%$) có thể gây hỏng bo mạch hoặc hoạt động không ổn định.

- **USB:** Kết nối ESP32 với máy tính hoặc cấp nguồn thông qua cổng USB. Điều này cho phép nạp chương trình và cấp nguồn cho ESP32.

2.4.2. Servo SG90 180

Động cơ Servo là một dạng động cơ điện có nguyên lý vô cùng đặc biệt, không giống như các động cơ DC khác, có thể cắm nguồn vào và hoạt động, nhưng Servo thì khác, đây là dạng động cơ chỉ có thể điều khiển bằng kỹ thuật điều chế độ rộng xung (PWM) với góc quay trong khoảng 0 đến 180 độ. Trên thị trường hiện nay có nhiều loại servo có nhiều kích cỡ và trọng lượng khác nhau.

Động cơ Servo được thiết kế bằng những hệ thống hồi tiếp vòng kín. Tín hiệu ra của động cơ được kết nối với một mạch điều khiển nhỏ. Khi động cơ quay, vận tốc và vị trí sẽ được hồi tiếp về mạch điều khiển này. Trong trường hợp nào đó mà có sự ngăn cản chuyển động quay của động cơ, cơ cấu hồi tiếp sẽ nhận ra rằng tín hiệu chưa đạt được vị trí như đã mong muốn. Mạch này có nhiệm vụ điều chỉnh sai lệch cho động cơ đạt được điểm mong muốn.



Hình 2. 12: Servo S90 180

Thông số kỹ thuật của Servo SG90 180

Bảng 2. 2: Bảng thông số kỹ thuật của servo SG90 180.

Điện áp hoạt động	4.8-5VDC
Lực kéo	1.6 Kg.cm
Trọng lượng	9g.
Tốc độ	0.12 sec/ 60 deg (4.8VDC)
Kích thước	21x12x22mm

2.4.3. Pin lithium 18650 Ion 3.7V



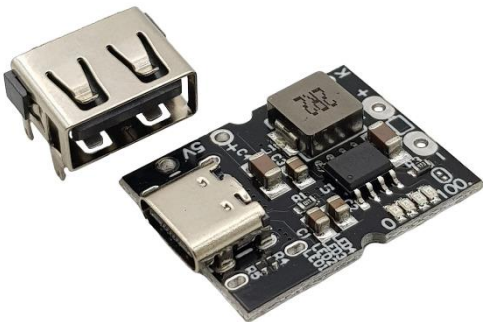
Hình 2. 13. Pin Lithium Ion 3.7 V 6800mAh

Thông số kỹ thuật:

Bảng 2. 3: Bảng thông số kỹ thuật của Pin Lithium Ion 3.7V 6800mAh.

Mẫu	18650
Loại	Pin Li-ion có thể sạc lại
Dung lượng	6800mAh
Điện áp định mức	3.7V
Điện áp sạc đầy	4.2V
Số lần sạc xả	1000 lần
Kích thước	18mm X 65mm

2.4.4. Mạch sạc pin lithium 2A typeC



Hình 2.14: Mạch sạc xả pin 18650 typeC

Thông số kỹ thuật:

Bảng 2.4: Bảng thông số kỹ thuật của mạch sạc xả pin 18650 typeC.

Cổng đầu vào	USB Type-C
Điện áp ngõ vào	3.7V
Điện áp sạc pin	4.2VDC (có thể lên tối đa 4.35V)
Dòng sạc	2.4A ±5%
Đầu ra	Tiếp điểm 5V hoặc đầu USB A cái
Điện áp ngõ ra	~5V
Dòng đầu ra	Tối đa 2A
Hiệu Suất	92.5% (đối với đầu vào 3.6V, đầu ra 5V 2A)
Tăng áp ngõ ra	Ổn định ở mức 5V±0,3V

Nguyên lý hoạt động:

- Chế độ sạc: Khi nguồn điện được kết nối, mạch điều khiển sạc sẽ kiểm tra mức điện áp của pin. Nếu điện áp pin thấp hơn điện áp định mức, mạch sẽ bắt đầu quá trình sạc. Có 2 chế độ sạc:
 - + Dòng điện không đổi: Pin sẽ nhận dòng sạc không đổi cho đến khi điện áp pin đạt đến một mức nhất định (thường khoảng 4.2V).
 - + Điện áp không đổi: Sau khi đạt đến điện áp không đổi, mạch sẽ chuyển sang chế độ CV. Lúc này, điện áp được giữ ổn định ở 4.2V và dòng sạc sẽ giảm dần cho đến khi pin đạt được trạng thái đầy (thường khi dòng sạc giảm xuống dưới 10% dòng sạc định mức).

- Chế độ xả:

- + Khi thiết bị sử dụng pin, mạch sẽ cho phép dòng điện chạy từ pin ra tải.
- + Mạch sẽ giám sát điện áp pin và ngắt xả khi điện áp xuống dưới mức an toàn (thường khoảng 2.5V-3.0V) để tránh làm hỏng pin.

2.5. GIỚI THIỆU PHẦN MỀM SỬ DỤNG

2.5.1. Arduino IDE

Arduino IDE là một ứng dụng máy tính hỗ trợ nền tảng về lập trình Arduino hoặc các dòng vi điều khiển khác bằng cách sử dụng framework Arduino để lập trình.

Ngôn ngữ chính của nó được dựa trên ngôn ngữ C/C++ và có thể dễ dàng học và sử dụng, hiện nay đây là một trong những nền tảng có cộng đồng hỗ trợ mạnh mẽ nhất trong lĩnh vực điện tử cũng như là vi điều khiển, có thể dễ dàng tìm thấy các video cũng như là các website hướng dẫn sử dụng cũng như là lập trình cho Arduino từ cơ bản đến nâng cao.



Hình 2.15: Logo Arduino IDE

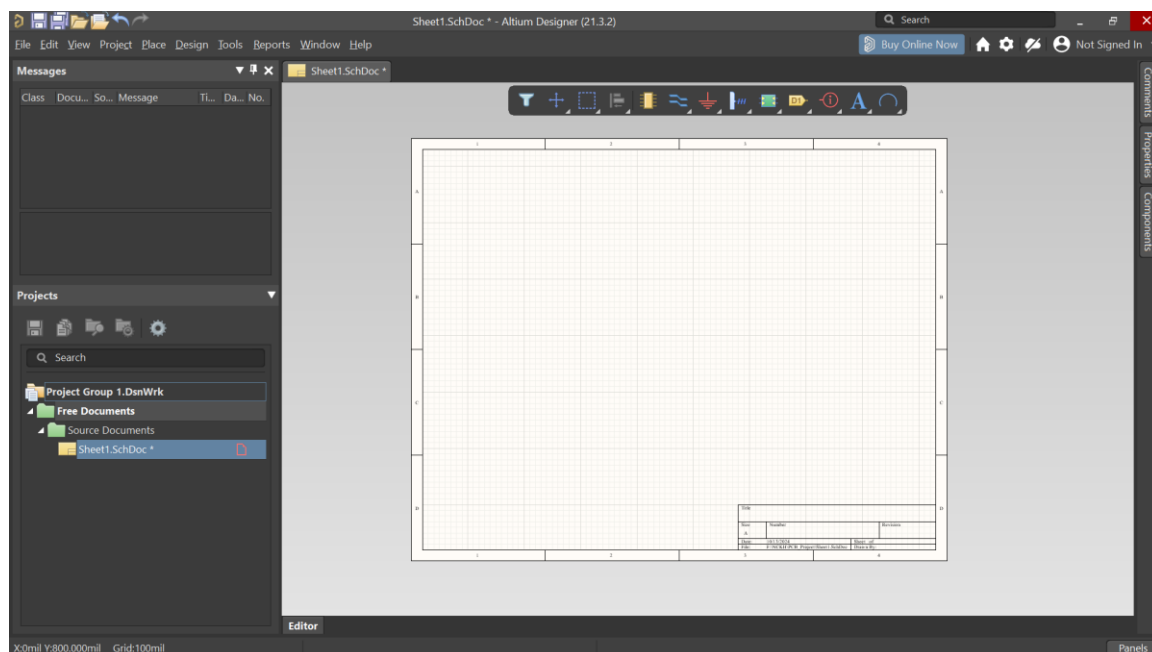
Arduino IDE được tích hợp cả tính năng biên dịch chương trình và hỗ trợ nạp file hex vào vi điều khiển để sử dụng, đây là một phần mềm mã nguồn mở cho nên có thể dùng tất cả tính năng của nó một cách miễn phí. Hiển nhiên là có cũng hỗ trợ cho các hệ điều hành phổ biến hiện nay như là Window, Linux, MacOS mà dung lượng không quá nhiều.

2.5.2. Altium Designer

Altium Designer là phần mềm thiết kế PCB hỗ trợ chụp sơ đồ, mô hình 3D, bản vẽ lắp ráp và mô phỏng. Nó còn cho phép các nhà thiết kế PCB chia sẻ thiết kế và nhận phản hồi theo thời gian thực. Với công nghệ tiên tiến thì Altium Designer đã trở nên phổ biến trong ngành công nghiệp bo mạch điện tử.

Altium Designer có các tính năng:

- Giao diện thân thiện, hỗ trợ quản lý, chỉnh sửa, biên dịch, quản lý file và phiên bản của tài liệu thiết kế hiệu quả.
- Hỗ trợ thiết kế, đi dây tự động một cách tối ưu, phân tích lắp ráp linh kiện và tìm giải pháp chỉnh sửa mạch dựa trên các tham số mới.
- Dễ dàng mở, xem và in file thiết kế mạch với đầy đủ thông tin về linh kiện, netlist, bản vẽ, kích thước và số lượng.
- Thư viện linh kiện phong phú, hỗ trợ đầy đủ các linh kiện số, tương tự và nhúng.
- Tùy chỉnh các lớp mạch in, đặt và chỉnh sửa đối tượng, định nghĩa luật thiết kế và chuyển từ schematic sang PCB dễ dàng.
- Mô phỏng mạch PCB 3D, hỗ trợ MCAD-ECAD, liên kết với mô hình STEP.
- Hỗ trợ thiết kế từ PCB sang FPGA và ngược lại.



Hình 2.16: Giao diện bắt đầu của Altium Designer.

Ưu điểm:

- Được thiết kế chuyên dụng để vẽ Schematic và PCB với bộ các tính năng mạnh mẽ có thể đáp ứng hầu hết các yêu cầu thị trường.
- Hỗ trợ mô phỏng 3D một cách trực quan và cho phép sử dụng file 3D từ các phần mềm khác.
- Có cộng đồng lớn và thư viện phong phú do Altium là phần mềm hàng đầu trong lĩnh vực thiết kế PCB.

Nhược điểm:

- Phần mềm nặng gây khó khăn khi sử dụng trên máy có cấu hình yếu.
- Giá thành khá cao so với các phần mềm thiết kế PCB khác.

2.5.3. MIT App Inventor

MIT App Inventor là mô hình nền tảng giáo dục với mục đích tạo ra các ứng dụng di động mà không bắt buộc phải có kiến thức và kỹ năng chuyên sâu về lập trình.

Ưu điểm:

- Dành cho những người mới tiếp cận lập trình, giao diện vô cùng trực quan, dễ học và sử dụng cho người mới.

- Cho phép người học tập trung vào logic và ý tưởng của phần mềm không cần biết nhiều về lập trình.
- Có thể tạo ra ứng dụng đơn giản mà không tốn nhiều thời gian. Khuyến khích người học có thể sáng tạo và giúp cho người dùng cảm thấy hứng thú khi có thể tạo ra được sản phẩm cho riêng mình.
- Có thể kiểm tra trực tiếp trên chính điện thoại di động của người dùng. Giúp cho người dùng có thể tự kiểm tra và trải nghiệm trên chính sản phẩm của họ.
- Có thể dùng để tạo ra ứng dụng, trò chơi cũng như là các ứng dụng sách, báo và các ứng dụng khác.
- Tăng cường khả năng tư duy logic cho người sử dụng nó bằng cách tự tư duy xây dựng các chức năng cũng như các ý tưởng cho riêng mình.
- Có cộng đồng hỗ trợ rất lớn, có thể tìm tài liệu, video hướng dẫn sử dụng và học hỏi từ cộng đồng này.

Nhược điểm:

- Không có quá nhiều tính năng để cho người dùng có thể khai thác được.
- Có một số giới hạn về chất lượng đồ họa. Đây có thể là rào cản của ứng dụng này và người dùng.
- Tất nhiên đây là ứng dụng phục vụ cho việc học tập thì chắc chắn không thể nào tạo ra được một ứng dụng có hiệu suất cao. Điều này bắt buộc phải thông qua lập trình.



Hình 2.17: Logo MIT App Inventor.

CHƯƠNG 3

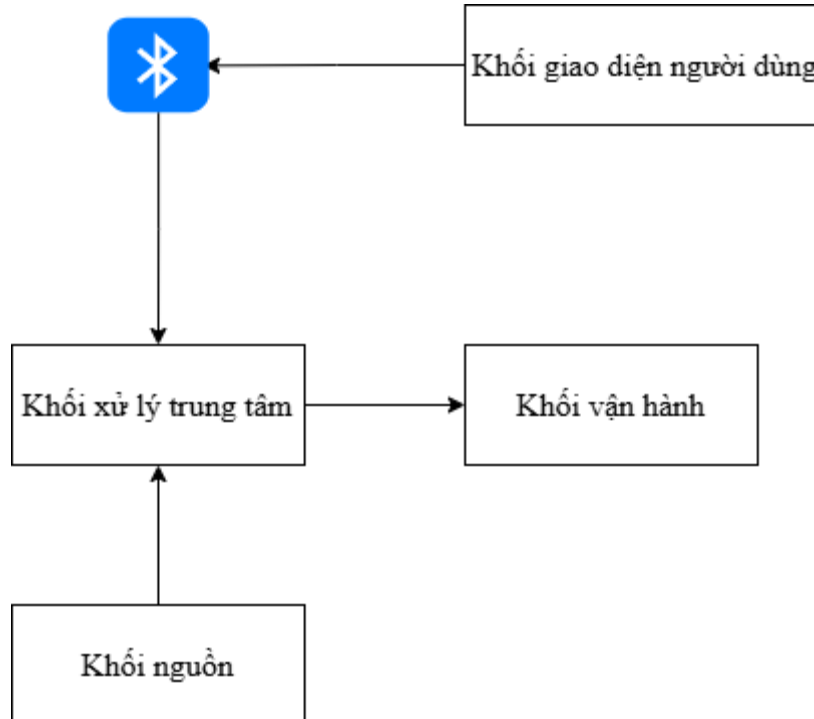
THIẾT KẾ HỆ THỐNG

3.1. GIỚI THIỆU

Đề tài tập trung vào việc thiết kế và phát triển một hệ thống robot nhện với khả năng di chuyển linh hoạt, hoạt động hiệu quả trong các môi trường không khí khác nhau. Robot được trang bị cấu trúc chân đa khớp, giúp nó duy trì sự ổn định và thích ứng tốt với những bề mặt phức tạp hay không bằng phẳng. Đặc biệt, hệ thống còn cho phép người dùng điều khiển robot từ xa thông qua một ứng dụng di động được phát triển trên nền tảng Android, sử dụng kết nối Bluetooth. Ứng dụng này mang đến sự tiện lợi, giúp người dùng dễ dàng kiểm soát hướng di chuyển và các thao tác của robot.

3.2. SƠ ĐỒ KHỐI HỆ THỐNG

3.2.1 Sơ đồ khối



Hình 3.1: Sơ đồ khối hệ thống.

3.2.2. Chức năng từng khối

Khối xử lý trung tâm:

Chức năng: Nhận tín hiệu điều khiển từ người dùng thông qua bluetooth, đồng thời xử lý tín hiệu điều khiển để tạo ra tín hiệu điều khiển khối vận hành thông qua các chân đầu ra (GPIO).

Lựa chọn linh kiện: Sau khi tìm hiểu, cân nhắc và chọn lọc thì nhóm đã quyết định chọn ESP32 NodeMCU LuaNode32 Module Thu Phát Wifi 38 Chân. Dòng này cho phép kết nối bluetooth và wifi, giúp việc nhận và xử lý tín hiệu từ người dùng trở nên dễ dàng hơn. Bên cạnh đó, ESP32 NodeMCU LuaNode32 Module Thu Phát Wifi 38 Chân có vi xử lý hai nhân với xung nhịp tối đa lên đến 240Mhz, đáp ứng tốt cho nhu cầu xử lý dữ liệu và điều khiển nhiều ngoại vi. Với 38 chân GPIO, vi điều khiển này có thể kết nối được nhiều thiết bị ngoại vi và điều khiển nhiều ngoại vi. Mặc dù esp32 có nhiều tính năng và được hỗ trợ tốt từ cộng đồng, giá thành của nó lại vô cùng thân thiện với sinh viên.

Khối vận hành:

Chức năng: Nhận tín hiệu từ khối trung tâm, thực hiện các thao tác cơ học giúp cho robot di chuyển.

Lựa chọn linh kiện: Động cơ servo SG90 180 độ trở thành lựa chọn của nhóm vì nhiều điểm nổi bật. Đầu tiên, nó có kích thước nhỏ gọn và trọng lượng nhẹ giúp dễ dàng gắn vào các phần nhỏ của robot nhện, không làm tăng trọng lượng đáng kể cũng như gây di chuyển khó khăn cho robot nhện. Giá cả phải chăng nhưng chất lượng lại được bảo đảm cũng là một điểm mạnh của linh kiện này. Ngoài ra, với góc quay 180 độ cũng giúp cho việc di chuyển của robot nhện trở nên linh hoạt hơn.

Khối giao diện người dùng:

Chức năng: Giúp người dùng có thể dễ dàng điều khiển robot nhện trên thiết bị cá nhân thông qua giao diện phần mềm. Từ đó các tín hiệu điều khiển sẽ được gửi thông qua truyền thông vô tuyến đến khối xử lý trung tâm để rồi có thể điều khiển khối vận hành như ý muốn.

Lựa chọn ứng dụng: Việc phát triển ứng dụng để tạo ra giao diện cho người dùng sử dụng một cách đơn giản và dễ hiểu, nhóm đã tìm hiểu và quyết định chọn MIT App Inventor và giao tiếp với khối xử lý trung tâm thông qua bluetooth. MIT App Inventor là một công cụ hỗ trợ người lập trình dễ dàng tạo ra giao diện cũng như các chức năng cho phần mềm trên điện thoại thông qua hoạt động kéo thả giao

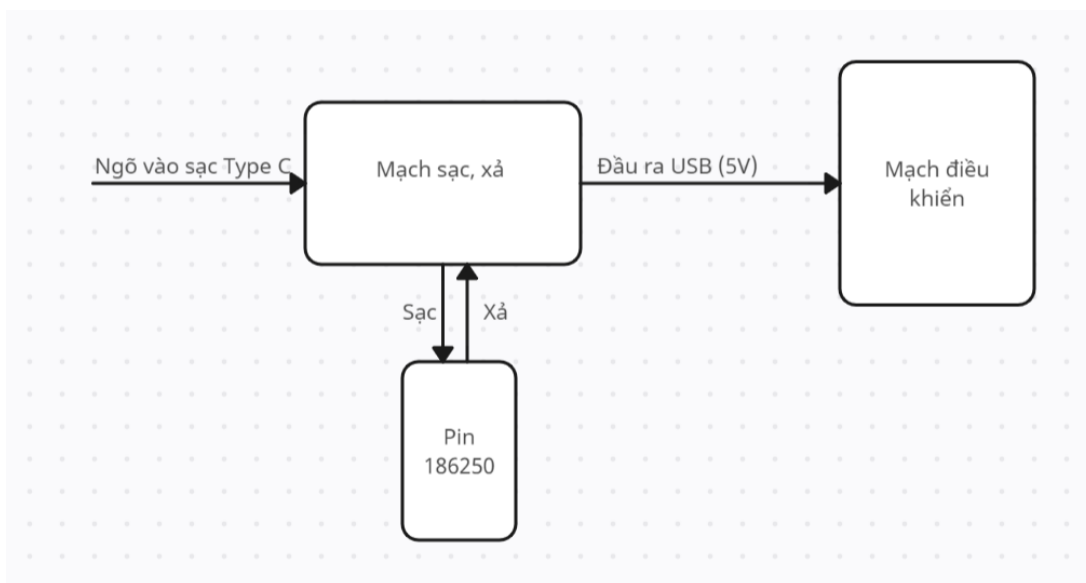
điện. Đồng thời nó cũng hỗ trợ các chuẩn giao tiếp không dây phổ biến, giúp ứng dụng trở nên linh hoạt hơn.

Khối nguồn:

Chức năng: Cung cấp năng lượng cho hệ thống hoạt động.

Lựa chọn linh kiện: sử dụng mạch tăng áp tích hợp sạc xả cổng Type-C 5V-2A cho pin 18650 Li-ion/Lithium 3.7V vì những lợi ích vượt trội mà nó mang lại. Đầu tiên, mạch này giúp tối ưu hóa hiệu suất sạc và xả của pin, đảm bảo rằng thiết bị luôn được cung cấp nguồn điện ổn định và an toàn. Với khả năng sạc nhanh, mạch giúp tiết kiệm thời gian và giảm thiểu rủi ro quá tải, quá áp và quá nhiệt. Kế đến, cùng với thiết kế nhỏ gọn và tích hợp của mạch giúp cho người dùng có thể dễ dàng lắp đặt và sử dụng, phù hợp cho các ứng dụng cần thiết hoặc nâng cấp thiết bị hiện có. Cuối cùng, việc sử dụng cổng Type-C hiện đại mang đến sự tiện lợi trong việc kết nối và tương thích với nhiều thiết bị khác nhau. Tóm lại, mạch này không chỉ cải thiện tuổi thọ của pin mà còn nâng cao hiệu suất hoạt động của toàn bộ hệ thống. Ngoài ra, mạch này còn có một cổng type A output giúp cho người dùng dễ dàng cung cấp điện áp cho các thiết bị thông qua cổng typeA.

3.3. SƠ ĐỒ NGUYÊN LÝ MẠCH

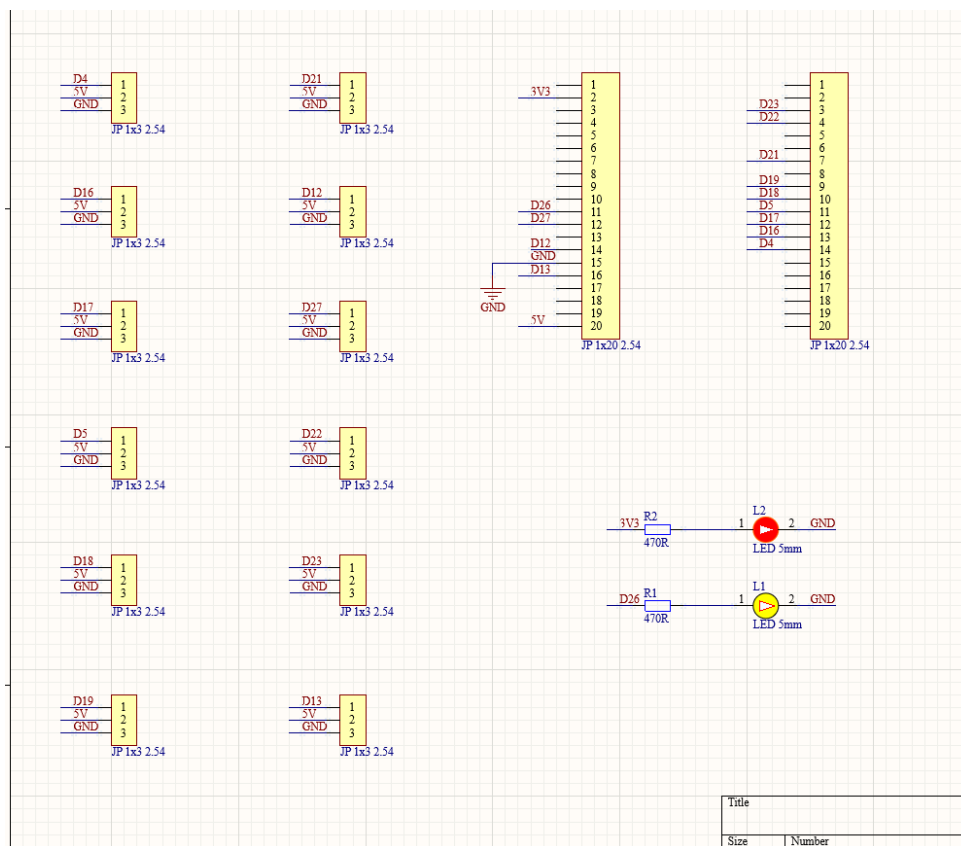


Hình 3.2: Sơ đồ khối mạch điện.

Hình 3.2 trên mô tả một sơ đồ nguyên lý của hệ thống sạc và xả pin được sử dụng trong đề tài. Với ngõ vào sạc Type-C là cổng kết nối nhận dòng điện từ bên ngoài chẳng hạn như từ một bộ sạc hoặc máy tính để cung cấp năng lượng cho hệ thống.

Mạch sạc và xả là bộ phận quản lý việc sạc và xả của pin, khi kết nối nguồn từ ngõ vào sạc thì mạch này sẽ thực hiện quá trình sạc cho pin 18650. Đồng thời, nó cũng có chức năng quản lý quá trình xả pin. Đi cùng với mạch sạc xả là pin 18650, đây là loại pin Li-ion có dung lượng cao được sử dụng rộng rãi. Pin này có khả năng sạc lại và cung cấp điện cho mạch điều khiển khi cần thiết.

Khi pin đã được sạc đầy thì mạch sạc và xả sẽ chuyển điện áp pin sang ngõ ra USB để cung cấp dòng điện ổn định 5V cho mạch điều khiển. Và cuối cùng, mạch điều khiển là nơi điện từ pin thông qua mạch sạc và xả được cấp cho các thiết bị khác. Mạch điều khiển sẽ quản lý việc tiêu thụ điện từ pin, đảm bảo thiết bị hoạt động bình thường.



Hình 3.3: Sơ đồ nguyên lý mạch điều khiển.

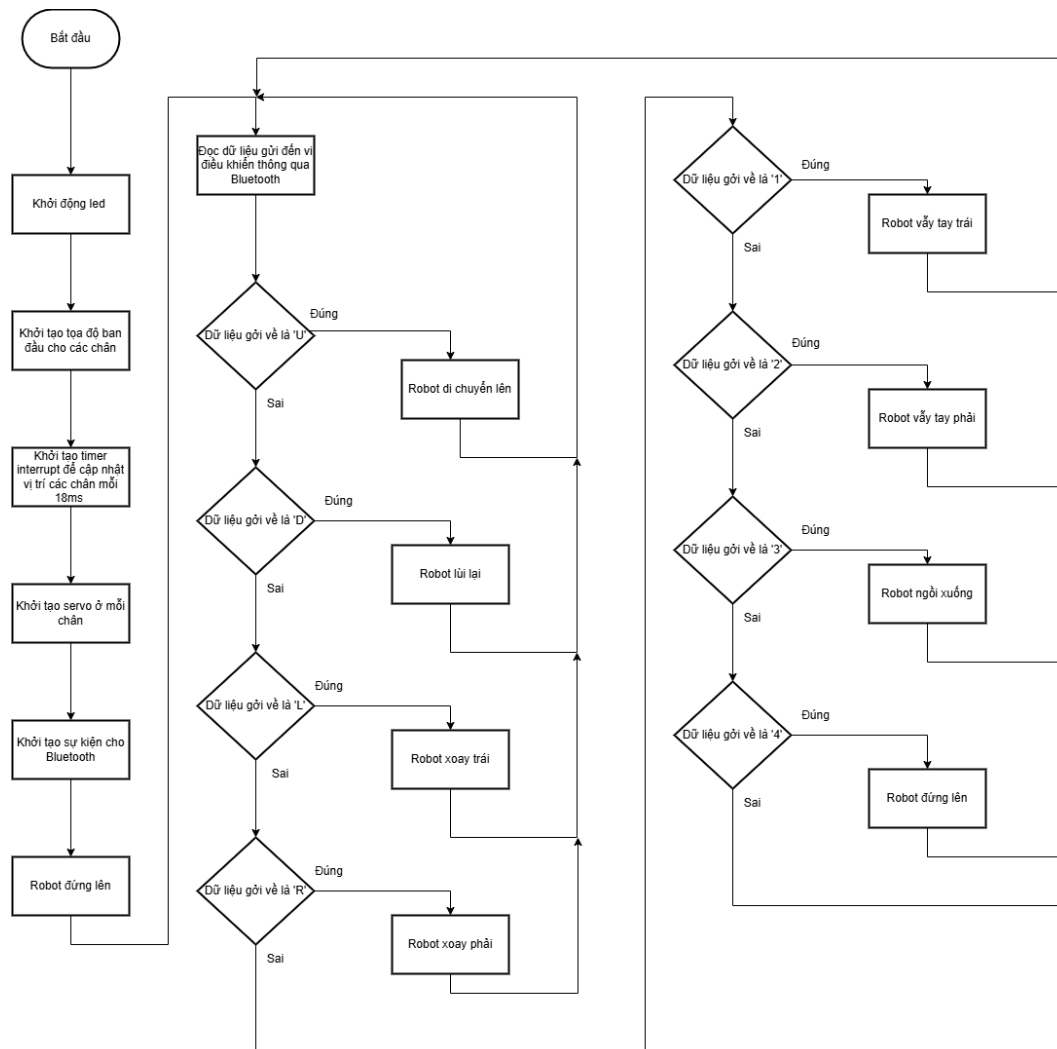
Các chân header trong mạch đóng vai trò trung gian quan trọng, kết nối giữa các thành phần khác nhau và cho phép sự tương tác linh hoạt với nhiều module, thiết bị ngoại vi. Ví dụ điển hình là kết nối với module Bluetooth hoặc vi điều khiển. Những chân header này không chỉ đảm nhận chức năng kết nối mà còn cung cấp nguồn điện và các tín hiệu điều khiển cần thiết cho toàn bộ hệ thống mạch. Điều này giúp hệ thống dễ dàng mở rộng và tích hợp thêm các thành phần khác, đảm bảo sự linh hoạt và dễ sử dụng.

Ngoài ra, mạch còn được trang bị hai đèn LED để cung cấp các thông báo trực quan về trạng thái hoạt động của hệ thống. LED nguồn (L1) có nhiệm vụ luôn sáng khi hệ thống được cấp điện, giúp người dùng nhận biết rằng mạch đã được kích hoạt và đang hoạt động ổn định. Điều này đặc biệt hữu ích trong việc đảm bảo rằng hệ thống đã sẵn sàng để hoạt động mà không cần đến các phương pháp kiểm tra phức tạp khác.

Trong khi đó, LED báo trạng thái kết nối Bluetooth (L2) được thiết kế để cung cấp thông tin về trạng thái kết nối của module Bluetooth. Khi không có kết nối Bluetooth, đèn sẽ hoàn toàn tắt để báo hiệu cho người dùng biết rằng kết nối chưa được thiết lập. Tuy nhiên, khi kết nối Bluetooth thành công, đèn sẽ nhấp hai lần như một thông báo, sau đó đèn sẽ sáng liên tục để cho biết kết nối đã được thiết lập và đang hoạt động ổn định. Chức năng này giúp người dùng dễ dàng giám sát tình trạng kết nối Bluetooth mà không cần phải kiểm tra qua phần mềm hay giao diện phức tạp khác.

3.4. THIẾT KẾ PHẦN MỀM

3.4.1. Hoạt động hệ thống

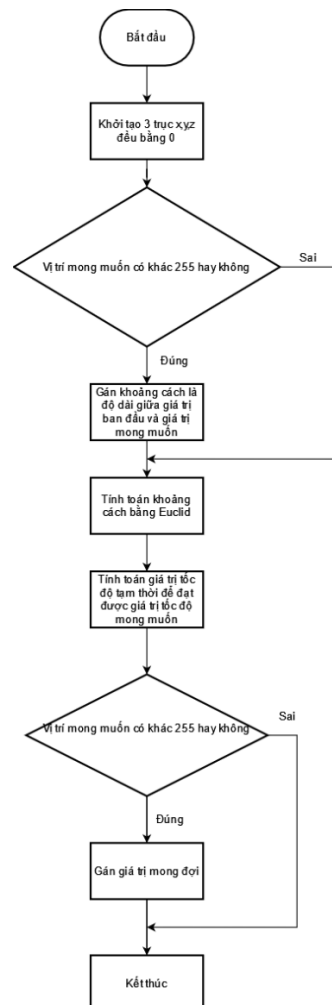


Hình 3.4: Lưu đồ hoạt động của robot nhện.

Giải thích lưu đồ: Hệ thống được bắt đầu khi được cấp nguồn, khi đó LED đỏ sẽ sáng để báo hiệu rằng hệ thống này đã được cấp nguồn và chờ hoạt động tiếp theo. Kế đến, với mong muốn là hệ thống này sau khi cấp nguồn thì sẽ đứng chờ kết nối Bluetooth cũng như là chờ nhận lệnh điều khiển thông qua ứng dụng điện thoại trên Android bằng cách xác định tọa độ ban đầu để có nó thể đứng được cũng như là tạo ra một đối tượng Interrupt timer với mục đích sau một khoảng thời gian nhỏ thì sẽ kiểm tra xem tọa độ đặt ban đầu đúng không, nếu không sẽ cố gắng điều chỉnh vị trí lại cho phù hợp với tọa độ đầu vào. Sau khi khởi tạo các điều kiện đầu vào cho

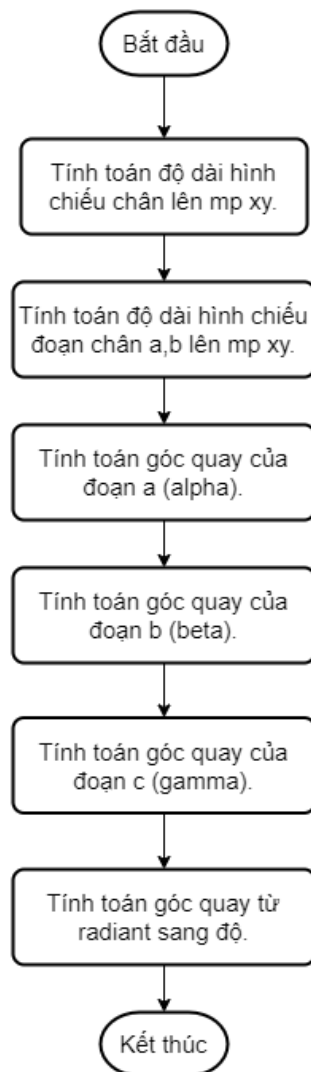
việc đứng thì sẽ khởi tạo ban đầu cho các servos, đầy đủ các yếu tố như trên thì nhện sẽ có thể đứng lên và chờ kết nối Bluetooth, nếu có thiết bị kết nối với Bluetooth của robot này thì đèn xanh lá sẽ nháy hai lần và sáng để báo hiệu đã ghép đôi robot với điện thoại di động Android. Kế đến, robot sẽ chờ nhận lệnh để có thể làm các chức năng đã được hiện thực hóa trên nó thông qua ứng dụng di động. Chẳng hạn như là đi thẳng thì sẽ chờ lệnh ‘U’, hoặc lệnh ngò là ‘3’ và tương tự với các lệnh khác, sau khi thực thi lệnh thì robot sẽ tiếp tục chờ lệnh mới.

Để có thể kiểm tra cũng như là điều chỉnh lại vị trí tọa độ theo mong muốn với sau khoảng thời gian ngắn thì hàm `set_site()` dưới đây sẽ có thể làm được và đáp ứng tốt công việc đó.



Hình 3.5: Lưu đồ cho hàm `set_site`.

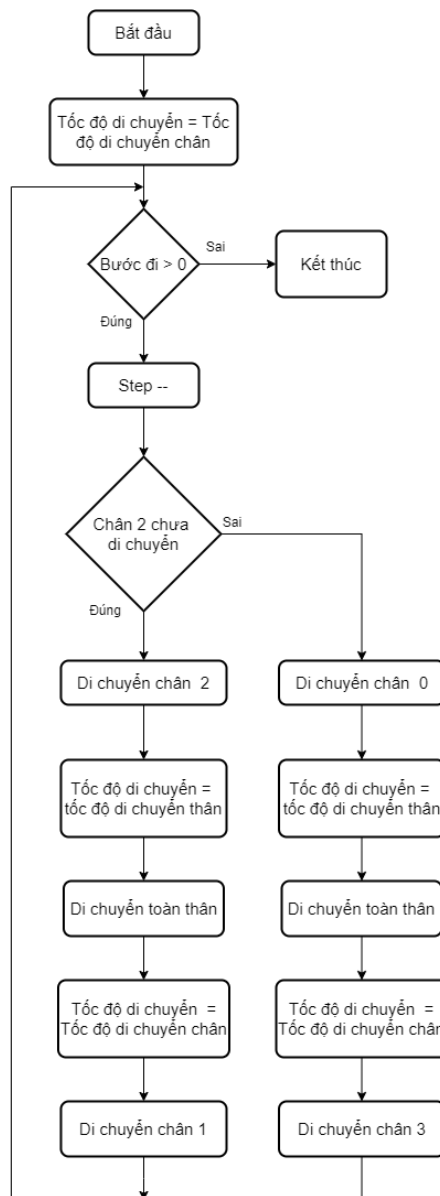
Giải thích lưu đồ: ban đầu khi vào hàm này thì sẽ đặt các giá trị khoảng cách đều bằng 0 để thuận để tính toán sau đó, sau đó sẽ lần lượt kiểm tra các giá trị x , y , z có khác 255 hay không, nếu có thì tính toán lần lượt vị trí của khoảng cách giữa độ dài mong muốn từ là đầu vào so với ban đầu là bằng 0. Kế đến sẽ tính khoảng cách bằng công thức Euclid và tính toàn giá trị tốc độ di chuyển thông qua các biến đã được tùy chỉnh trước đó. Cuối cùng, kiểm tra một lần nữa xem x , y , z có khác 255 hay không, nếu có thì gán các biến x , y , z vào một mảng giá trị mong đợi cho việc giữ vị trí cũng như là các tính năng khác cho robot nhện.



Hình 3.6: Lưu đồ cho hàm chuyển đổi tọa độ cực sang góc quay Servo.

Giải thích lưu đồ: Khi bắt đầu hàm này thì tính toán trước độ dài hình chiếu của chân robot lên mặt phẳng xOy . Kế đến, tiếp tục tính toán độ dài hình chiếu của hai đoạn chân alpha và beta, từ đó có thể tính toán được góc quay của servo bằng tọa độ cực với các góc quay alpha, beta, gamma. Cuối cùng sẽ quy đổi các góc alpha, beta, gamma thành góc quay mong muốn của servo.

3.4.2. Chức năng đi thẳng và đi lùi



Hình 3.7: Lưu đồ cho chức năng đi thẳng.

```
set_site(2, x_default + x_offset, y_start, z_up);  
wait_all_reach();
```

Gọi hàm thực hiện việc nhắc chân lên.

```
set_site(2, x_default + x_offset, y_start + 2 * y_step, z_up);  
wait_all_reach();
```

Gọi hàm thực hiện đưa chân về phía trước.

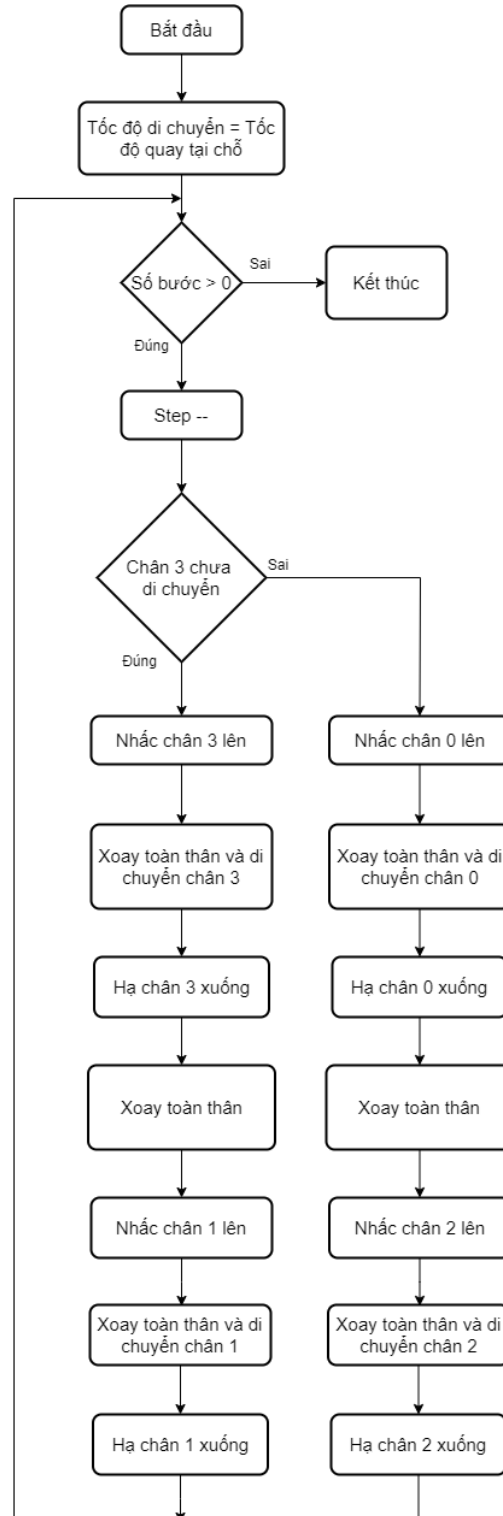
```
set_site(2, x_default + x_offset, y_start + 2 * y_step, z_default);  
wait_all_reach();
```

Gọi hàm thực hiện việc hạ chân xuống.

```
set_site(0, x_default + x_offset, y_start, z_default);  
set_site(1, x_default + x_offset, y_start + 2 * y_step, z_default);  
set_site(2, x_default - x_offset, y_start + y_step, z_default);  
set_site(3, x_default - x_offset, y_start + y_step, z_default);
```

Giải thích lưu đồ: Đây là lưu đồ thực hiện chức năng đi thẳng của robot nhện, đối với chức năng đi lùi của robot thì cũng sẽ thực hiện tương tự như vậy chỉ khác là sẽ xoay servo theo hướng ngược lại so với đi thẳng. Khi bắt đầu hàm này thì sẽ lấy tốc độ điều chỉnh chân, sau đó sẽ kiểm tra xem số bước đi còn lại có lớn hơn không hay không, nếu không thì sẽ đứng yên và kết thúc hàm, ngược lại thì giảm một lần với số bước chân, kể đến bắt đầu quá trình di chuyển bằng cách kiểm tra chân số 2 xem đã di chuyển hay chưa, nếu chưa thì bắt đầu di chuyển chân số 2, và điều chỉnh lại tốc độ di chuyển của chân sau đó di chuyển chân số 1, và quay lại kiểm tra xem số bước còn lại có lớn hơn 0, trong trường hợp đã di chuyển chân số 2 hoàn tất thì sẽ di chuyển chân số 0 và điều chỉnh tốc độ di chuyển của chân rồi của thân robot và di chuyển tiếp chân số 3 và quay lại kiểm tra số bước còn lại, và lặp lại cho đến khi số bước còn lại bằng 0. Thì sẽ thoát khỏi hàm và đợi lệnh tiếp theo.

3.4.3. Chức năng xoay trái và xoay phải



Hình 3. 8: Lưu đồ cho chức năng xoay trái.


```
move_speed = spot_turn_speed;
```

Thực hiện việc gán tốc độ di chuyển bằng tốc độ quay tại chỗ.

```
while (step-- > 0)
{
    if (site_now[3][1] == y_start)
    {
        set_site(3, x_default + x_offset, y_start, z_up);
    }
}
```

Hàm thực hiện việc nhấc chân số 3 lên.

```
wait_all_reach();
set_site(0, turn_x1 - x_offset, turn_y1, z_default);
set_site(1, turn_x0 - x_offset, turn_y0, z_default);
set_site(2, turn_x1 + x_offset, turn_y1, z_default);
set_site(3, turn_x0 + x_offset, turn_y0, z_up);
wait_all_reach();
```

Xoay toàn thân nhện đồng thời cả chân 3 theo(chân 3 vẫn đang nhấc lên).

```
set_site(3, turn_x0 + x_offset, turn_y0, z_default);
wait_all_reach();
```

Hạ chân số 3 xuống.

```
set_site(0, turn_x1 + x_offset, turn_y1, z_default);
set_site(1, turn_x0 + x_offset, turn_y0, z_default);
set_site(2, turn_x1 - x_offset, turn_y1, z_default);
set_site(3, turn_x0 - x_offset, turn_y0, z_default);
wait_all_reach();
```

Thực hiện việc xoay toàn thân một lần nữa.

```
set_site(1, turn_x0 + x_offset, turn_y0, z_up);  
wait_all_reach();
```

Thực hiện việc nhấc chân 1 lên.

```
set_site(0, x_default + x_offset, y_start, z_default);  
set_site(1, x_default + x_offset, y_start, z_up);  
set_site(2, x_default - x_offset, y_start + y_step, z_default);  
set_site(3, x_default - x_offset, y_start + y_step, z_default);  
wait_all_reach();
```

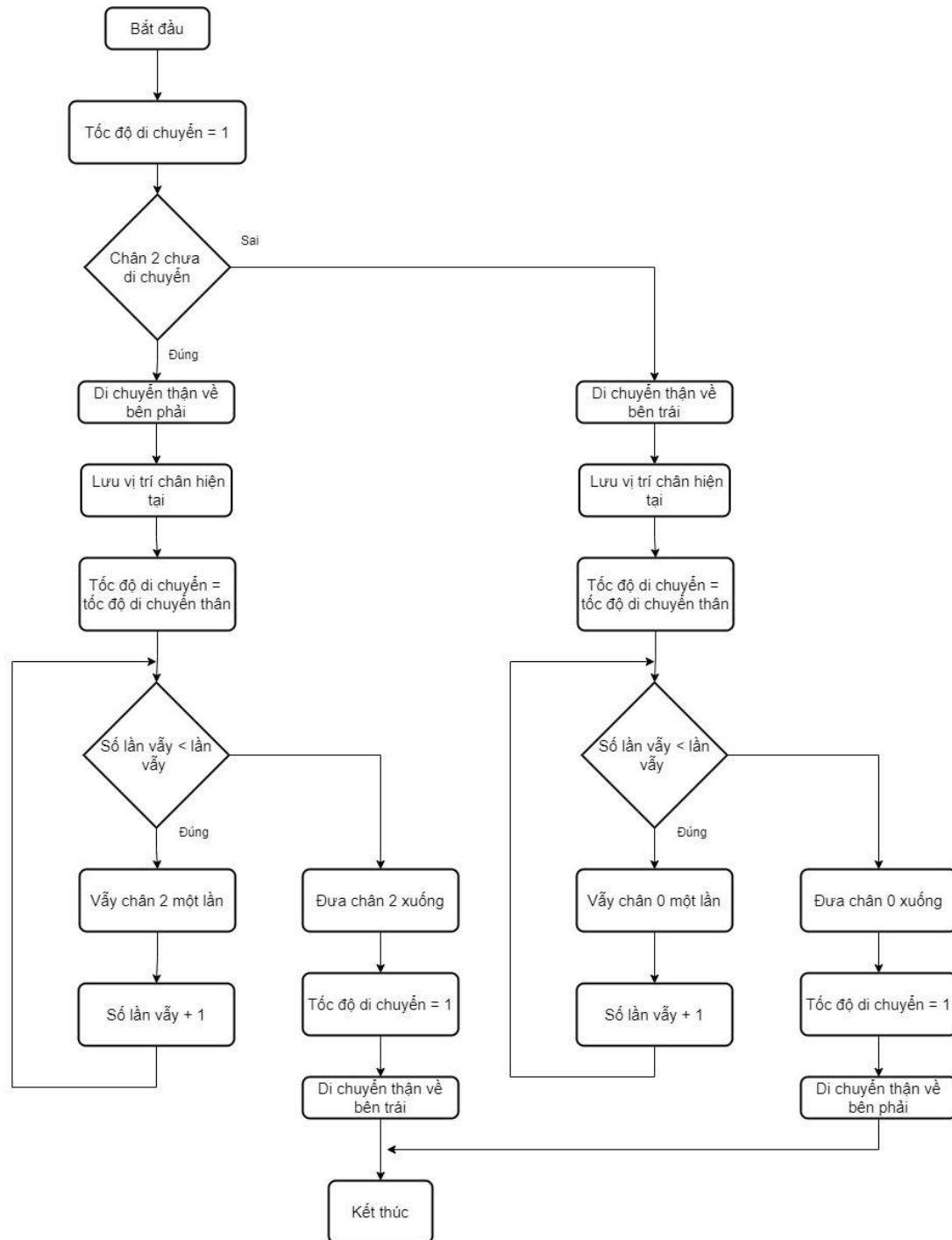
Thực hiện việc xoay toàn thân và chân 1.

```
set_site(1, x_default + x_offset, y_start, z_default);  
wait_all_reach();
```

Hạ chân 1 xuống.

Giải thích lưu đồ: Đây là lưu đồ mô tả chức năng xoay trái, cũng tương tự đối với chức năng xoay phải, thì điểm khác nhau giữa chức năng xoay phải và xoay trái là ở điểm xoay của servo sẽ ngược với nhau. Khi bắt đầu với hàm này, điều đầu tiên sẽ lấy giá trị tốc độ của chân khi xoay tại chỗ, sau đó sẽ kiểm tra số bước còn lại là bao nhiêu cụ thể có lớn hơn không hay không, nếu có sẽ tiếp tục trừ đi số bước còn lại đi 1 vì sẽ dùng 1 bước đó để di chuyển, sau đó sẽ kiểm tra chân số 3 đã di chuyển hay chưa nếu chưa thì nhấc chân 3 lên và kết hợp xoay toàn thân về hướng chân số 3 và sau khi xoay xong thì hạ chân 3 xuống, sau đó sẽ điều chỉnh lại tốc độ xoay toàn thân, và di chuyển tiếp chân số 1, trong trường hợp ngược lại thì sẽ xoay chân số 0 thay vì chân số 3, cũng sẽ xoay toàn thân và hạ chân số 0 xuống cuối cùng sẽ xoay tiếp chân số 2 và hạ xuống. Kết thúc một trong hai quá trình xoay như trên thì sẽ quay về kiểm tra tiếp số bước còn lại và thực hiện cho đến khi số bước còn lại bằng 0 và thoát khỏi hàm và chờ thêm lệnh tiếp theo.

3.4.4. Chức năng vẩy tay chào



Hình 3.9: Lưu đồ cho chức năng vẩy tay chào.

Giải thích lưu đồ: Hình 3.9 là lưu đồ cho chức năng vẫy tay chào của robot nhện, hàm này bắt đầu với gán lại tốc độ di chuyển bằng một, sau đó sẽ kiểm tra xem chân số 2 có di chuyển hai chưa, nếu chưa sẽ đưa thân robot về hướng bên phải và giữ vị trí tại đó, kể đến lấy tốc độ bằng tốc độ của thân robot, sau đó sẽ kiểm tra số lần vẫy tay còn lại có còn lớn hơn không hay chưa, nếu còn sẽ đưa chân số 2 lên vẫy 1 lần và hạ xuống, ngược lại nếu số lần còn lại không còn nữa thì sẽ kết thúc hàm và đợi lệnh tiếp theo, trong trường hợp mà chân số 2 đã di chuyển rồi thì sẽ đưa thân mình về bên trái và sẽ làm quy trình như trên nhưng sẽ thay thế chân số 2 bằng chân số 0. Cuối cùng kiểm tra xem số lần còn lại còn bao nhiêu nếu không còn thì sẽ thoát khỏi hàm và đợi lệnh tiếp theo.

3.4.5. Chức năng đứng và ngồi

```
void sit(void){
    move_speed = stand_seat_speed;
    for (int leg = 0; leg < 4; leg++){
        set_site(leg, KEEP, KEEP, z_boot);
    }
    wait_all_reach();
}
```

Giải thích:

```
stand_seat_speed = 1
z_boot = z_absolute = -28
length_a = 55
length_b = 77.5
length_c = 27.5
```

Các thông số trên thay vào hàm cartesian_to_polar():

Tính toán góc alpha:

$$\text{atan2}(-28, -27.5) + \text{acos}((\text{pow}(55, 2) - \text{pow}(77.5, 2) + \text{pow}(27.5, 2) + \text{pow}(28, 2)) / 2 / 55 / \sqrt{\text{pow}(27.5, 2) + \text{pow}(28, 2)}))$$

Kết quả của biểu thức khi chuyển đổi sang độ là khoảng -25.2°

Tính toán góc beta:

$$\text{acos}((\text{pow}(55, 2) + \text{pow}(77, 2) - \text{pow}(27.5, 2) - \text{pow}(28, 2)) / 2 / 55 / 77)$$

Kết quả của biểu thức khi chuyển đổi sang độ là khoảng 28.95°

```
void stand(void){
```

```

move_speed = stand_seat_speed;

for (int leg = 0; leg < 4; leg++) {

    set_site(leg, KEEP, KEEP, z_default); }

wait_all_reach();}

```

Các thông số trên thay vào hàm cartesian_to_polar():

Tính toán góc alpha:

$$\text{atan2}(-50, -27.5) + \text{acos}((\text{pow}(55, 2) - \text{pow}(77.5, 2) + \text{pow}(27.5, 2) + \text{pow}(50, 2)) / 2 / 55 / \sqrt{\text{pow}(27.5, 2) + \text{pow}(50, 2)}))$$

Kết quả của biểu thức khi chuyển đổi sang độ là khoảng -26.77°

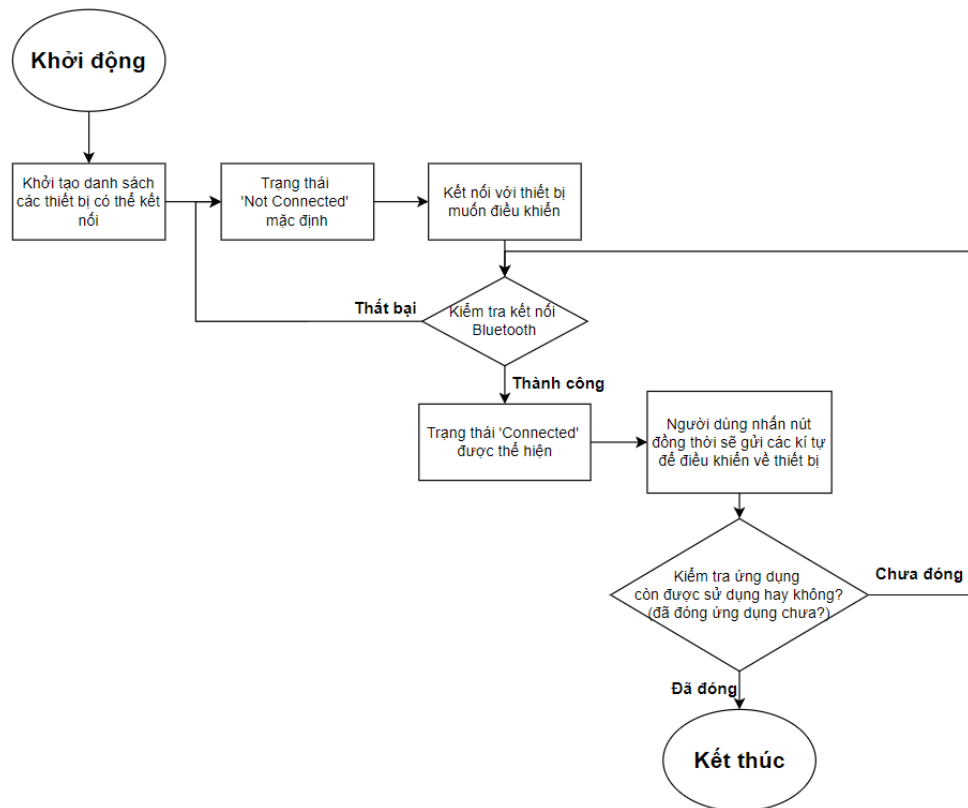
Tính toán góc beta:

$$\text{cos}((\text{pow}(55, 2) + \text{pow}(77, 2) - \text{pow}(27.5, 2) - \text{pow}(50, 2)) / 2 / 55 / 77).$$

Kết quả của biểu thức khi chuyển đổi sang độ là khoảng 47.58°

3.4.6. Giao diện ứng dụng điều khiển trên Android

*Lưu đồ hoạt động:



Hình 3.10: Lưu đồ cho ứng dụng điện thoại Android.

***Giải thích lưu đồ:**

Khởi động: Ứng dụng sẽ khởi tạo danh sách các thiết bị có thể kết nối bằng Bluetooth. Trạng thái hiển thị ban đầu là “Not Connected”.

Kết nối với thiết bị: Người dùng sẽ chọn thiết bị muốn điều khiển trong danh sách đã được khởi tạo trước đó. Sau khi người dùng chọn thiết bị xong sẽ kiểm tra lại quá trình kết nối của thiết bị và ứng dụng điều khiển, nếu kết nối thành công thì sẽ thể hiện dòng trạng thái “Connected”, nếu kết nối thất bại thì sẽ hiển thị “Not Connected” và người dùng sẽ kiểm tra và kết nối lại một lần nữa.

Điều khiển thiết bị bằng các nút trên ứng dụng: Người dùng sẽ nhấn vào các nút điều khiển có trên màn hình. Các nút “Up”, “Down”, “Left”, “Right”, 2 nút “Shake”, “Sit” và nút “Stand” sau khi được nhấn sẽ gửi một kí tự tương trưng cho các nút đó về thiết bị thông qua kết nối Bluetooth. Các kí tự đã truyền sẽ được thiết bị kiểm tra và hoạt động theo bảng mã điều khiển đã được lập trình.

Kiểm tra ngắt kết nối hoặc đóng thiết bị: Nếu thiết bị vẫn được sử dụng tiếp tục sẽ được quay lại bước 2 kiểm tra kết nối Bluetooth có xảy ra sự cố trong quá trình sử dụng hay không? Nếu thoát ứng dụng thì chương trình kết thúc.

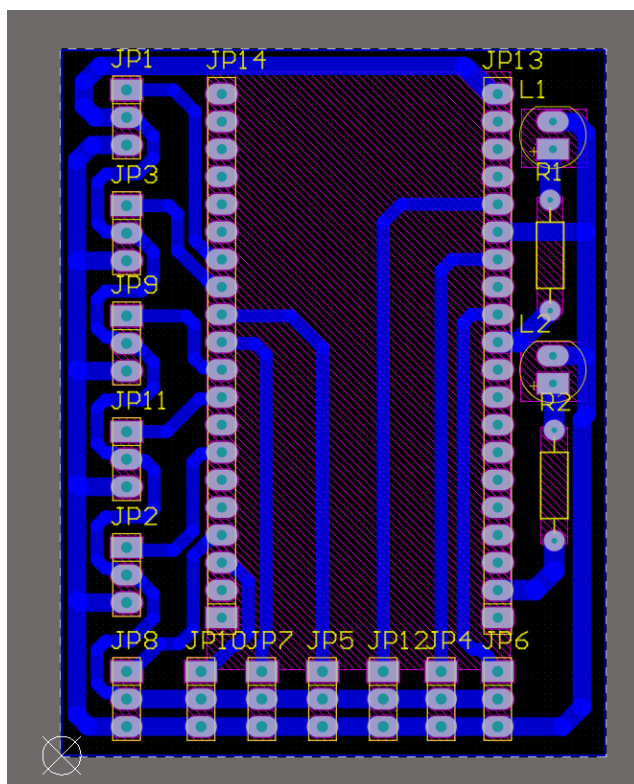
CHƯƠNG 4

THI CÔNG HỆ THỐNG

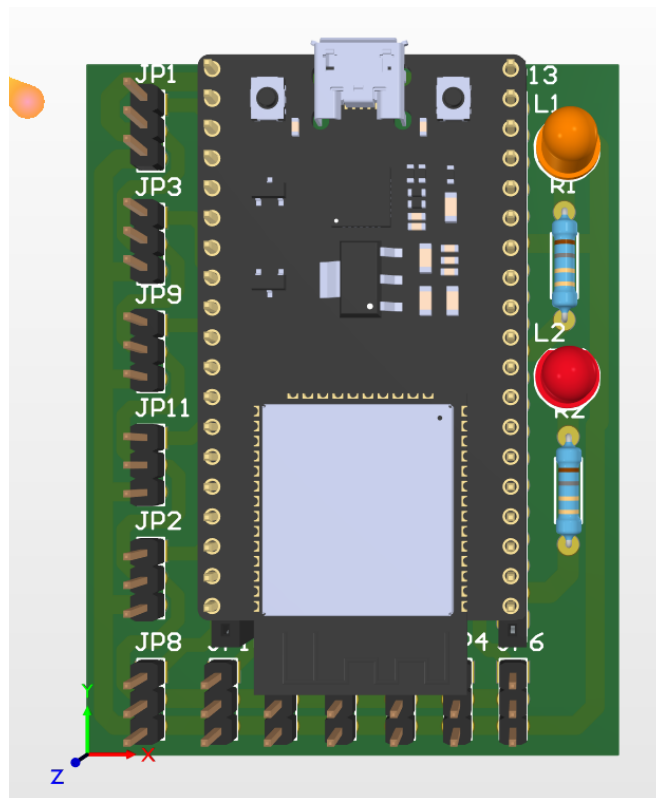
4.1. GIỚI THIỆU

Mạch mô hình được thiết kế bằng cách lắp ráp các module chân lại với nhau tạo thành một con robot nhện hoàn chỉnh với khả năng di chuyển linh hoạt. Các module này được kết nối chặt chẽ, giúp robot có thể thực hiện các thao tác di chuyển và điều khiển một cách chính xác. Trung tâm điều khiển của hệ thống là bộ xử lý ESP32 đóng vai trò quan trọng trong việc xử lý các tín hiệu và điều phối hoạt động từng bộ phận của robot. Robot sẽ được điều khiển từ xa thông qua một ứng dụng di động trên nền tảng Android, sử dụng kết nối Bluetooth để truyền tải các lệnh điều khiển.

4.2. MÔ HÌNH PHẦN CỨNG

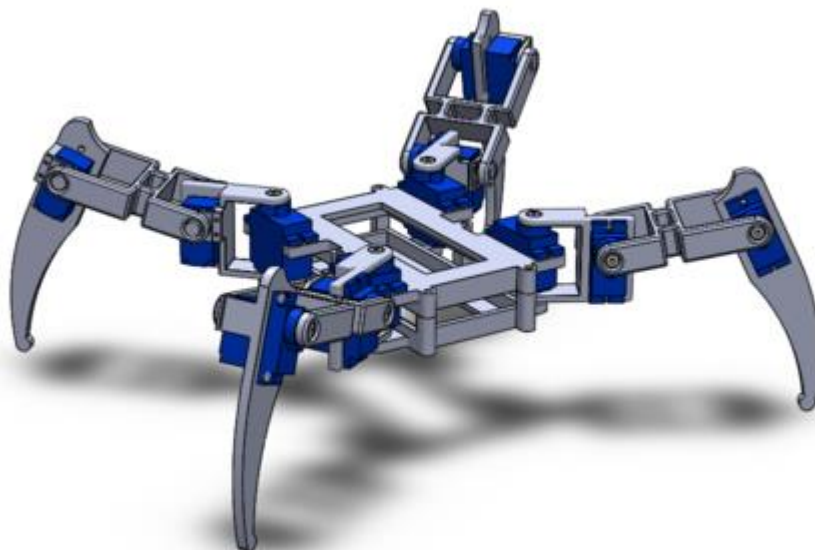


Hình 4.1: Mạch in PCB trong giai đoạn thiết kế bằng phần mềm Altium.



Hình 4.2: Mạch in PCB ở dạng 3D.

Cùng với việc sử dụng phần mềm thiết kế mạch Altium, nhóm đã làm ra được mạch với chức năng ra chân cho 12 servo cũng như các trạng thái khác.



Hình 4.3: Mô hình robot nhện.

Hình 4.3 cho thấy kết quả khi lắp ráp robot nhện 4 chân cùng kết hợp với 12 servo. Mô hình này có các chân trong đó có các khớp với mục đích hướng tới khả năng di chuyển linh hoạt, phù hợp dành cho các ứng dụng thăm dò,...

4.3. GIAO DIỆN PHẦN MỀM



Hình 4.4: Giao diện ứng dụng điều khiển trên Android.

Trạng thái kết nối: Dòng trạng thái hiển thị ngay dưới nút "Show Device", mặc định là '**Not Connected**' khi chưa kết nối với thiết bị Bluetooth nào. Khi kết nối thành công, trạng thái sẽ chuyển thành '**Connected**'. Nếu kết nối bị ngắt, trạng thái sẽ trở lại '**Not Connected**'.

Nút 'Show Device': Hiển thị danh sách các thiết bị Bluetooth khả dụng để kết nối.

Nút 'Up': Khi nhấn, con nhện robot sẽ tiến lên phía trước.

Nút 'Down': Khi nhấn, con nhện robot sẽ lùi lại.

Nút 'Left': Khi nhấn, con nhện robot sẽ quay về phía trái.

Nút 'Right': Khi nhấn, con nhện robot sẽ quay về phía phải.

Nút 'Sit': Khi nhấn, con nhện sẽ ngồi xuống.

Nút 'Stand': Khi nhấn, con nhện sẽ đứng lên.

Nút 'Shake tay trái': Khi nhấn, con nhện sẽ vẫy tay trái chào.

Nút 'Shake tay phải': Khi nhấn, con nhện sẽ vẫy tay phải chào.

4.4. HƯỚNG DẪN SỬ DỤNG

4.4.1. Trên mô hình

Bước 1: Lắp ráp mô hình

- Lắp ráp servo vào mô hình.
- Đảm bảo mô hình không bị ngược trong quá trình lắp.

Bước 2: Kết nối thiết bị và mô hình

- Kết nối dây của các servo vào mạch
- Để có thể khởi động mô hình thì kết nối dây Micro USB vào ESP32 để khởi động, hoặc có thể thay bằng sạc dự phòng có kích thước tương đối.

Bước 3: Kết nối thiết bị và điện thoại di động(android)

- Kiểm tra thiết bị có kết nối được bluetooth hay không. Nếu có đèn LED vàng sẽ chớp tắt 2 lần và sau đó sáng, nếu không có kết nối hoặc mất kết nối thì đèn sẽ ở trạng thái tắt.
- Kiểm chứng hoạt động bằng các nút điều khiển trên điện thoại.

4.4.2. Trên ứng dụng di động

Bước 1: Khởi động ứng dụng

- Mở bluetooth trên điện thoại trước sau đó mở ứng dụng trên điện thoại.
- Khi ứng dụng khởi động, trạng thái kết nối sẽ được hiển thị là "**Not Connected**" – nhấn dán dưới nút "**Show Device**".

Bước 2: Kết nối với thiết bị Bluetooth

- Nhấn nút **“Show Device”** để xem danh sách các thiết bị Bluetooth khả dụng.
- Chọn thiết bị bạn muốn kết nối từ danh sách.
- Ứng dụng sẽ cố gắng kết nối với thiết bị.
 - Nếu kết nối thành công, trạng thái sẽ chuyển thành **“Connected”**.
 - Nếu kết nối thất bại, trạng thái vẫn là **“Not Connected”**.

Bước 3: Điều khiển robot nhện

- Khi đã kết nối, bạn có thể sử dụng các nút điều khiển dưới đây:
 - **Nút ‘Up’**: Robot nhện sẽ tiến lên phía trước.
 - **Nút ‘Down’**: Robot nhện sẽ lùi lại.
 - **Nút ‘Left’**: Robot nhện sẽ quay trái.
 - **Nút ‘Right’**: Robot nhện sẽ quay phải.
 - **Nút ‘Sit’**: Robot nhện sẽ ngồi xuống.
 - **Nút ‘Stand’**: Robot nhện sẽ đứng lên.
 - **Nút ‘Shake’ bên trái**: Robot nhện sẽ vẫy tay trái.
 - **Nút ‘Shake’ bên phải**: Robot nhện sẽ vẫy tay phải nhưng hiện tại vẫn chưa hiện thực hóa tính năng này.

Bước 4: Kiểm tra trạng thái kết nối

- Trong quá trình điều khiển, ứng dụng sẽ kiểm tra kết nối Bluetooth liên tục.
- Nếu kết nối bị mất hoặc xảy ra lỗi, trạng thái sẽ trở lại **“Not Connected”**.
- Bạn có thể chọn lại thiết bị từ danh sách để kết nối lại.

Bước 5: Ngắt kết nối

- Khi bạn không còn sử dụng robot hoặc muốn kết nối với thiết bị khác, hãy đảm bảo ngắt kết nối và quay lại bước chọn thiết bị.

CHƯƠNG 5

KẾT QUẢ

5.1. KẾT QUẢ ĐẠT ĐƯỢC

5.1.1. Kết quả sau thi công mạch điều khiển



Hình 5.1: Mô hình mạch điều khiển bằng ESP32.

Mạch điện này bao gồm một số thành phần chính và hoạt động với nhiều mục đích cụ thể như:

- ESP32: Đây là bộ vi điều khiển trung tâm đóng vai trò quan trọng trong việc kết nối không dây thông qua Bluetooth và xử lý toàn bộ các tín hiệu của hệ thống. ESP32 không chỉ đảm nhiệm việc giao tiếp giữa các thiết bị mà còn điều khiển các thành phần khác trong mạch. Nhờ có ESP32 mà mạch có thể nhận và gửi dữ liệu tới điện thoại thông minh hoặc các thiết bị ngoại vi khác một cách linh hoạt.

- LED: Hệ thống mạch có hai đèn LED (một LED đỏ và một LED xanh), mỗi đèn phục vụ một chức năng cụ thể.

+ LED đỏ: Đèn này dùng để báo trạng thái nguồn. Khi nguồn điện được cấp cho mạch thì đèn đỏ sẽ sáng, báo hiệu rằng hệ thống đã có nguồn và đang hoạt động bình thường. Nếu không có nguồn cấp, đèn sẽ tắt. Điều này giúp người sử dụng biết ngay lập tức tình trạng nguồn của mạch mà không cần kiểm tra sâu hơn.

+ LED xanh: Đèn xanh được sử dụng để thông báo trạng thái kết nối Bluetooth. Nếu mạch được kết nối thành công với thiết bị khác, đèn xanh sẽ nhấp hai lần để báo hiệu quá trình kết nối và sau đó sẽ chuyển sang sáng liên tục khi kết nối đã ổn định. Ngược lại, nếu không có kết nối thì đèn xanh sẽ tắt giúp người dùng dễ dàng nhận biết tình trạng kết nối Bluetooth mà không cần phải truy cập vào giao diện điều khiển trên thiết bị.

- Hàng rào ra chân: Đây là các chân kết nối được thiết kế đặc biệt để sử dụng với các servo điều khiển các khớp chân. Các servo này sẽ nhận lệnh từ ESP32 để thực hiện các chuyển động cơ học cần thiết. Việc bố trí hàng rào ra chân giúp người dùng dễ dàng kết nối và điều khiển các servo một cách hiệu quả.

- Các điện trở: Trong mạch còn có các điện trở được sử dụng nhằm bảo vệ và đảm bảo độ bền cho các đèn LED cũng như các thành phần khác. Điện trở giúp giới hạn dòng điện đi qua LED, ngăn không cho LED bị hư hỏng do quá dòng, đồng thời đảm bảo tuổi thọ lâu dài của hệ thống.

5.1.2. Kết quả sau thi công mô hình robot



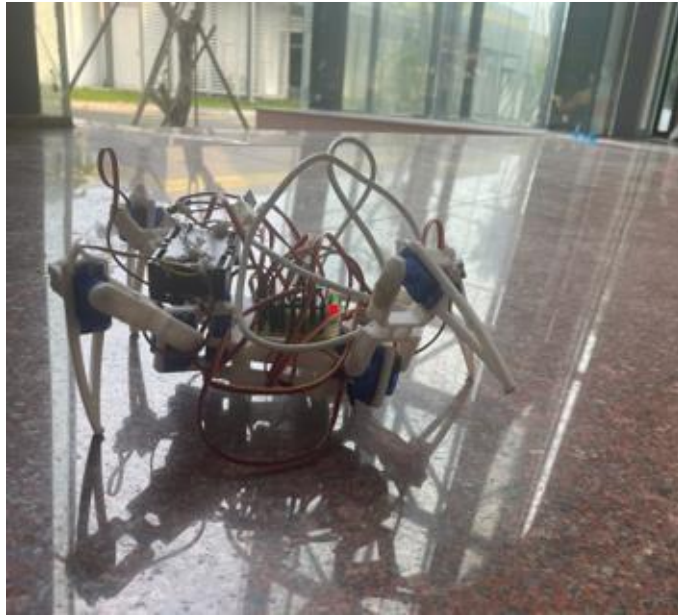
Hình 5.2: Mô hình robot nhện 4 chân.

Mô hình robot nhện trên đây đã được lắp ráp hoàn chỉnh bao gồm 12 động cơ servo được bố trí để điều khiển bốn chân, mỗi chân có ba khớp chuyển động riêng biệt. Hệ thống này còn giúp robot có khả năng di chuyển linh hoạt với các chuyển động mô phỏng chân thật giống như chuyển động của một con nhện. Việc điều khiển các khớp chân thông qua các servo mang lại khả năng thực hiện nhiều dạng chuyển động phức tạp, từ việc đi tới, lùi, đến chuyển hướng.

Cùng với các thành phần cơ khí và điện tử khác, mô hình robot nhện này gần như đã hoàn thiện. Tất cả các chi tiết đã được lắp đặt đúng vị trí giúp robot có thể vận hành theo thiết kế ban đầu. Một điểm đáng chú ý là các dây dẫn của 12 servo như thể hiện trong hình trên sẽ được sử dụng để kết nối với mạch điều khiển đã đề cập trước đó. Mạch điều khiển với bộ vi xử lý trung tâm ESP32 không chỉ đảm nhận vai trò điều khiển toàn bộ hoạt động của các servo mà còn thực hiện việc kết nối không dây với các thiết bị ngoại vi thông qua Bluetooth.

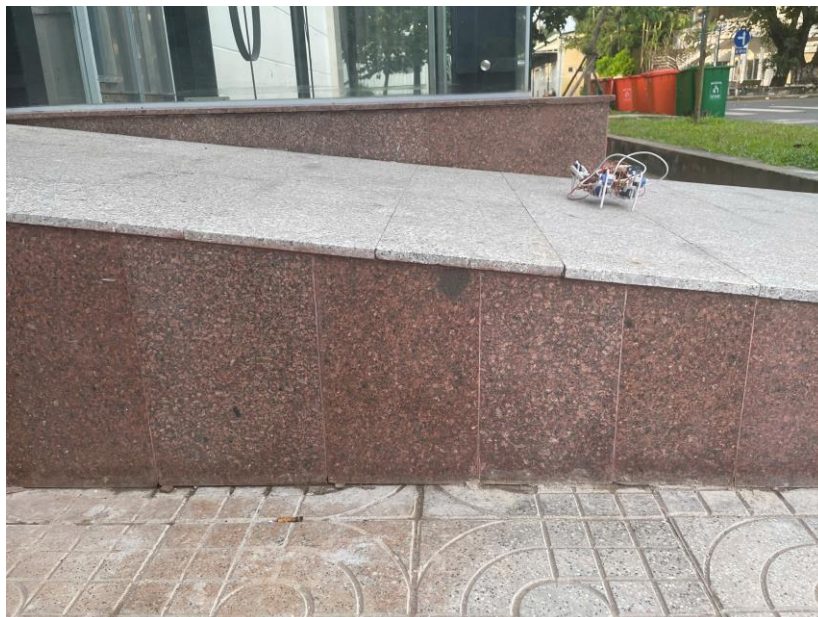
Kết hợp với các linh kiện như đèn LED, điện trở và các cổng kết nối, hệ thống này đã đạt đến giai đoạn hoàn thiện cao, sẵn sàng cho các bước thử nghiệm và tinh chỉnh cuối cùng trước khi đưa vào sử dụng thực tế.

5.2. KIỂM CHỨNG BẰNG THỰC NGHIỆM



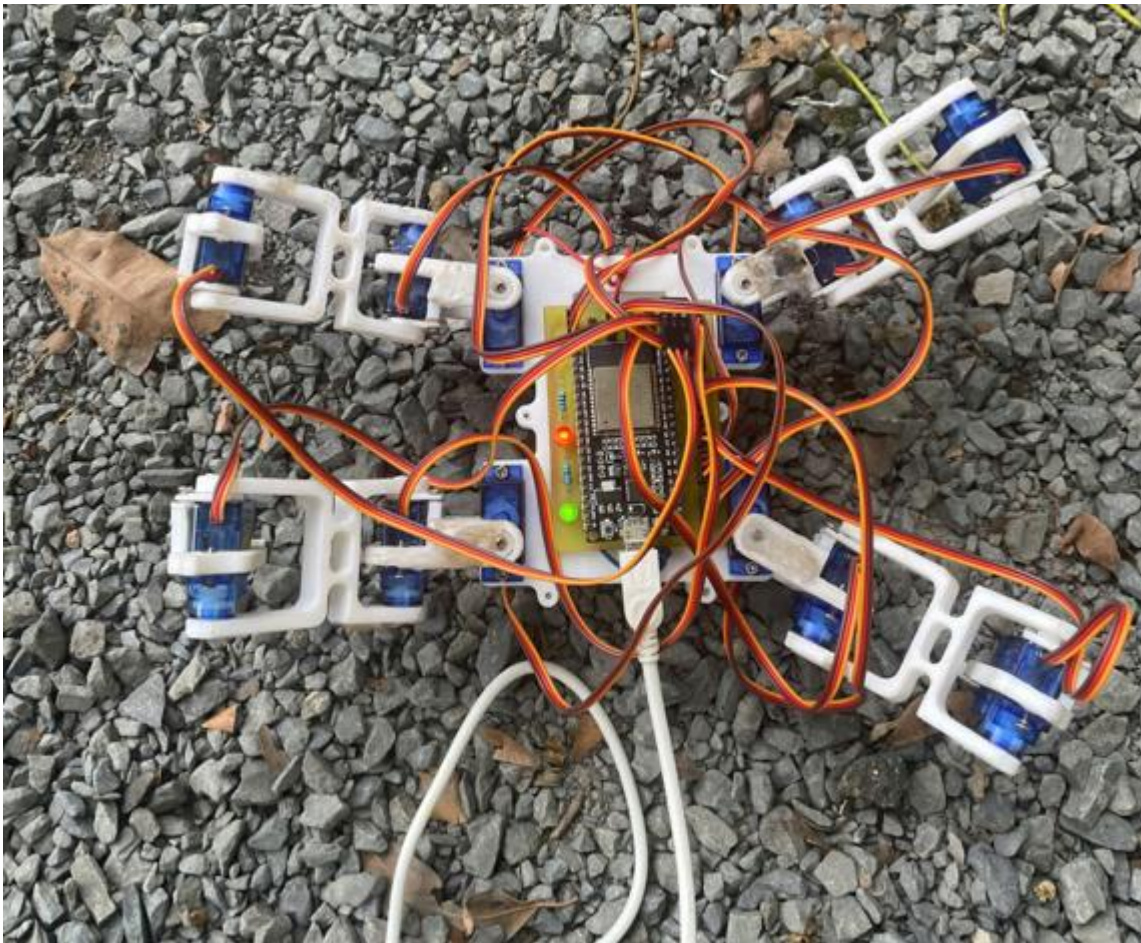
Hình 5.3: Thực nghiệm mô hình robot nhện 4 chân trên môi trường bằng phẳng.

Từ hình 5.3 cho thấy robot đang hoạt động trong một không gian có ánh sáng tự nhiên, phản chiếu rõ ràng xuống bề mặt sàn bóng cho thấy sự ổn định và tính chính xác của mô hình khi di chuyển trên bề mặt này.



Hình 5.4: Thực nghiệm mô hình robot nhện 4 chân trên đoạn đường dốc.

Hình trên ghi lại quá trình thực nghiệm mô hình robot nhện 4 chân trên một đoạn đường dốc. Robot được đặt trên bề mặt nghiêng để kiểm tra khả năng di chuyển và độ linh hoạt của nó khi hoạt động trong điều kiện không bằng phẳng (dốc). Các khớp chân của robot được điều khiển bởi 12 servo phải hoạt động đồng bộ để đảm bảo robot có thể duy trì thăng bằng và vượt qua độ dốc một cách hiệu quả. Thực nghiệm này nhằm đánh giá khả năng thích nghi của robot với môi trường địa hình phức tạp hơn so với mặt phẳng.



Hình 5.5: Thực nghiệm mô hình robot nhện 4 chân trên môi trường đá không bằng phẳng.

Hình trên là quá trình thực nghiệm mô hình robot nhện 4 chân trên một bề mặt đá gồ ghề. Robot được đặt trên địa hình không bằng phẳng hơn cả đoạn đường dốc nhằm kiểm tra khả năng thích nghi của nó với môi trường khó khăn hơn. Thử nghiệm này giúp đánh giá hiệu suất của robot trong việc duy trì cân bằng và di chuyển ổn định trên các bề mặt gồ ghề, khắc nghiệt.

CHƯƠNG 6

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

6.1. KẾT LUẬN

Sau khoảng thời gian dài tìm hiểu cũng như thực hiện cũng như vượt qua những khó khăn trong việc thực hiện đề tài này. Tham khảo qua các nguồn tài liệu khác nhau trên các website, bài báo khoa học,... Song cũng tham khảo các ý kiến đến từ thầy Huỳnh Hoàng Hà để có thể hoàn thiện đề tài này nhiều nhất có thể, cuối cùng nhóm cũng đã thực hiện đề tài “Thiết kế và thi công robot nhện di chuyển tự động hoặc điều khiển bằng điện thoại” cùng với khả năng điều khiển thông qua điện thoại cùng với các chức năng có thể điều khiển cơ bản và đã hoàn thành được các mục tiêu cơ bản:

- Thiết kế và thi công robot nhện có các chức năng cơ bản như đi thẳng, đi lùi, xoay trái, xoay phải, chào, đứng, ngồi thông qua điều khiển.
- Thiết kế ứng dụng di động điều khiển robot thông qua bluetooth.

Có thể được kể ra những điểm mạnh của mô hình này như:

- Có thể di chuyển cơ bản trên các địa hình hẹp.
- Sử dụng trong cái môi trường khí độc.

Bên cạnh đó thì đề tài này vẫn còn một nhược điểm như sau:

- Không thể điều khiển từ xa trên 10m vì đây là giới hạn của Bluetooth.
- Chưa tối ưu cũng như sử dụng Pin 1 cách tối ưu.
- Chưa thể di chuyển trên các địa hình lún cũng như đầm lầy hoặc môi trường có nước vì không thể kháng được nước.

Còn khả năng có thể tự di chuyển nhóm chưa thể hoàn thành được tính năng này cho nên tính năng này sẽ được đưa vào hướng phát triển trong tương lai cũng như có thể bổ sung thêm một vài tính năng nếu có thể.

6.2. HƯỚNG PHÁT TRIỂN

Dựa vào các nhược điểm của robot thì có thể rút ra một vài hướng phát triển tiếp theo trong tương lai như sau:

- Tích hợp thêm các cảm biến để robot có thể tự di chuyển và tránh vật cản.
- Dùng thêm camera hồng ngoại để có thể gửi video về điện để có thể thay con người quan sát những nơi chật hẹp, tối tăm.
- Nâng cấp kích thước cũng như là vật liệu của robot để robot có thể cứng cáp hơn và chống chịu hơn những nơi có nhiệt độ cao.
- Có thể thêm các tính năng định vị cũng như quan sát trạng thái pin từ xa bằng hệ thống BMS.

TÀI LIỆU THAM KHẢO

- [1]. Robot điều khiển bằng Bluetooth sử dụng Arduino Uno và MIT APP Inventor, 10/05/2021, Youtube, <https://www.youtube.com/watch?v=QC6TDIduhfg>
- [2]. Bluetooth LED Control App with MIT App Inventor - STEP By STEP, 06/02/2023, Youtube, <https://www.youtube.com/watch?v=w5LgLSCumFI>
- [3]. Polar co-ordinates, 2009, mathcentre, <https://www.mathcentre.ac.uk/resources/uploaded/mc-ty-polar-2009-1.pdf>
- [4]. Lý thuyết Hệ thức lượng trong tam giác, 2021, Loigiaihay, <https://loigiaihay.com/ly-thuyet-he-thuc-luong-trong-tam-giac-a109890.html#:~:text=%C4%90%E1%BB%8Bnh%20l%C3%BD%20cosin,c%E1%BB%A7a%20g%C3%B3c%20xen%20gi%E1%BB%AFa%20ch%C3%BAng>.
- [5]. Hướng dẫn sử dụng động cơ Servo SG90 với Arduino, 2023, arduinokit, <https://arduinokit.vn/huong-dan-su-dung-dong-co-servo-sg90-voi-arduino/>.
- [6]. Điều chế độ rộng xung, 17/08/2021, Wikipedia, https://vi.wikipedia.org/wiki/%C4%90i%E1%BB%8lu_ch%E1%BA%BF_%C4%91%E1%BB%99_r%E1%BB%99ng_xung.
- [7]. Khoảng cách Euclid, 3/10/2024, Wikipedia, https://vi.wikipedia.org/wiki/Kho%E1%BA%A3ng_c%C3%A1ch_Euclid.
- [8]. Understanding Bluetooth Technology, 01/02/2021, CISA, <https://www.cisa.gov/news-events/news/understanding-bluetooth-technology>.
- [9]. ESP32 NodeMCU LuaNode32 Module Thu Phát Wifi 38 Chân, 21/08/2024, Dientutuonglai, <https://dientutuonglai.com/so-do-chan-esp32.html>.
- [10]. Arduino IDE là gì? , 21/12/2020, arduinokit, https://arduinokit.vn/arduino-ide-la-gi/#google_vignette.
- [11]. Giới thiệu phần mềm Altium Designer, 21/08/2024, jywsoft, <https://jywsoft.com/gioi-thieu-phan-mem-altium-designer.htm>.
- [12]. Bản thiết kế robot nhện 4 chân, 04/05/2023, GrabCAD, <https://grabcad.com/library/quadruped-spider-2>.
- [13]. Phung Tu Minh, Nguyen Hoang Huynh (2020), “Design, implementation and control of hexapod robot combining image processing on android platform”, Thư viện số Đại học Sư Phạm Kỹ Thuật Tp. Hồ Chí Minh.

[14]. Nguyễn Quốc Thành, Nguyễn Nhật Nam, Trần Khả Anh Tùng, Nguyễn Việt Long (2020), “Lập trình thi công và phát triển robot nhện IoT”, Thư viện số Đại học Sư Phạm Kỹ Thuật Tp. Hồ Chí Minh.

