

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT
KHOA ĐIỆN - ĐIỆN TỬ

ĐỒ ÁN TỐT NGHIỆP

ĐẠI HỌC

NGÀNH: CÔNG NGHỆ KỸ THUẬT ĐIỆN - ĐIỆN TỬ
CHUYÊN NGÀNH : KỸ THUẬT ĐIỆN TỬ

THIẾT KẾ VÀ CHẾ TẠO MÔ HÌNH HỖ TRỢ

THU GOM RÁC THẢI TỰ ĐỘNG SỬ DỤNG

PHƯƠNG PHÁP DÒ ĐƯỜNG

Người hướng dẫn: ThS.Võ Thị Hương

Sinh viên thực hiện: Lê Phước Hải Đăng

MSV: 1811505120309 Lớp 18D4

Trần Thanh Phúc

MSV: 1811505120233 Lớp 18D4

Đà Nẵng, 06/2022

ĐẠI HỌC ĐÀ NẴNG
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT
KHOA ĐIỆN ĐIỆN TỬ

ĐỒ ÁN TỐT NGHIỆP
ĐẠI HỌC

NGÀNH: CÔNG NGHỆ KỸ THUẬT ĐIỆN – ĐIỆN TỬ
CHUYÊN NGÀNH : KỸ THUẬT ĐIỆN TỬ

THIẾT KẾ VÀ CHẾ TẠO MÔ HÌNH HỖ TRỢ
THU GOM RÁC THẢI TỰ ĐỘNG SỬ DỤNG
PHƯƠNG PHÁP DÒ ĐƯỜNG

Người hướng dẫn: ThS.Võ Thị Hương

Sinh viên thực hiện: Lê Phước Hải Đăng

MSV: 1811505120309 Lớp 18D4

Trần Thanh Phúc

MSV: 1811505120233 Lớp 18D4

Đà Nẵng, 06/2022

TÓM TẮT

Tên đề tài: **THIẾT KẾ VÀ CHẾ TẠO MÔ HÌNH HỖ TRỢ THU GOM RÁC THẢI TỰ ĐỘNG SỬ DỤNG PHƯƠNG PHÁP DÒ ĐƯỜNG**

Sinh viên thực hiện : **Trần Thanh Phúc** Mã SV : **1811505120233** Lớp: **18D4**

Lê Phước Hải Đăng Mã SV : **1811505120309** Lớp: **18D4**

Hiện nay, khi xã hội ngày càng phát triển, dân cư ngày càng đông đúc đặc biệt là ở các thành phố lớn, các khu công nghiệp, các cơ sở sản xuất... Từ đó dẫn đến lượng rác thải con người xả ra ngày càng tăng. Do đó, quản lý chất thải thông minh là một trong những vấn đề nóng hổi mà thế giới phải đối mặt. Vấn đề chính là trong việc quản lý chất thải, thùng rác ở nơi công cộng luôn luôn trong tình trạng đầy và quá tải trước khi được xe rác tới lấy và đưa lượng rác đó ra một vị trí tập trung lượng rác lớn khác để xử lý. Việc thùng rác nơi công cộng luôn đầy và tràn ra lần lượt dẫn đến nhiều mối nguy hiểm khác nhau như bốc mùi, ô nhiễm môi trường cảnh quan và đó cũng là gốc rễ, nguyên nhân chính gây ra những căn bệnh lây nhiễm. Xuất phát từ nguyên nhân đó, nhóm chúng em chọn đề tài “Thiết kế và chế tạo mô hình hỗ trợ thu gom rác thải tự động sử dụng ứng dụng dò đường”.

Với mục đích giúp cho môi trường xanh, sạch, đẹp, hiện đại và đáp ứng cho thời đại công nghệ 4.0, theo xu hướng thành phố thông minh trong tương lai, đồng thời chúng em muốn tìm hiểu sâu hơn về mảng lập trình ứng dụng để tạo ra sản phẩm kết hợp công nghệ điều khiển tự động, điển hình là mô hình thu gom rác thải ứng dụng dò đường. Dựa vào kiến thức đã học môn lập trình vi xử lý và các môn chuyên ngành Điện tử, nhóm chúng em đã nghiên cứu, chế tạo mô hình thu gom rác thải ứng dụng dò đường tự động có thể thông báo rác đầy hiển thị màn hình LCD, tự động di chuyển theo đường đưa rác đến vị trí tập kết và nhấn nút sẽ chạy về vị trí ban đầu. Thùng rác được trang bị cảm biến siêu âm phát hiện tránh vật cản trong quá trình di chuyển.

Trong quá trình thực hiện đề tài “Thiết kế và chế tạo mô hình hỗ trợ thu gom rác thải tự động sử dụng phương pháp dò đường”, mặc dù nhóm thực hiện đề tài đã cố gắng hoàn thành nhiệm vụ đặt ra và đúng thời hạn nhưng còn hạn chế về mặt kiến thức và thời gian thực hiện còn ảnh hưởng rất lớn của dịch Covid- 19, nên chắc chắn sẽ không tránh khỏi những thiếu sót, hạn chế, mong các quý Thầy/Cô và các bạn sinh viên thông cảm. Nhóm rất biết ơn và mong nhận được ý kiến đóng góp hết sức quý báu từ các quý Thầy/Cô và các bạn sinh viên về đề tài này.

NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Giảng viên hướng dẫn : **ThS. Võ Thị Hương**

Sinh viên thực hiện : **Trần Thanh Phúc** Mã SV : **1811505120233**

Lê Phước Hải Đăng Mã SV : **1811505120309**

1. Tên đề tài: THIẾT KẾ VÀ CHẾ TẠO MÔ HÌNH HỖ TRỢ THU GOM RÁC THẢI TỰ ĐỘNG SỬ DỤNG PHƯƠNG PHÁP DÒ ĐƯỜNG

2. Các số liệu, tài liệu ban đầu:

Số liệu, tài liệu ban đầu:

- Tìm hiểu thông tin đề tài qua các trang mạng, và tham khảo các tài liệu trước đó
- Tra cứu thông tin tin linh kiện trên mạng
- Tài liệu về công nghệ IOT và robot dò đường
- Cộng đồng arduino Việt Nam
- Phạm Minh Tuấn – Arduino cho người mới bắt đầu
- Các trang web đã tham khảo:
- Robot tránh vật cản :
<http://arduino.vn/result/1306-robot-tranh-vat-can>
- Sơ lược arduino mega 2560:
<https://dientutuonglai.com/gioi-thieu-arduino-mega-2560.html>
- Robot dò đường tránh vật cản:
<http://kenhchetac.com/chi-tiet-san-pham/95-robot-do-duong-tranh-vat-can.html>
<http://robocon.vn/ndetail/huong-dan-lam-robot-tu-tranh-vat-can-su-dung-cam-bien-sieu-am.html>
- Video tham khảo thùng rác thông minh:
https://youtu.be/3xznZK_8pgk

3. Những nội dung chính:

- Sử dụng Board Arduino Mega 2560 để làm khối điều khiển trung tâm.
- Điều khiển được thùng rác chạy theo đường đường ta tạo sẵn.
- Điều khiển được thùng rác chạy về vị trí ban đầu thông qua nút nhấn.
- Tích hợp sát khuẩn tay tự động cho người dùng.
- Hiển thị màn hình LCD thông báo với người dùng.
- Cảm biến phát hiện rác đầy sẽ tự di chuyển đến điểm tập kết.
- Cảm biến phát hiện vật cản sẽ dừng lại và phát ra cảnh báo.
- Cảm biến tự động mở nắp khi có người đến bỏ rác.

4. Các sản phẩm dự kiến:

- Hoàn thành được mô hình hỗ trợ thu gom rác thải ứng dụng dò đường.
- Báo cáo tổng kết đề tài, Slide thuyết trình đề tài.

5. Ngày giao đồ án: 22/02/2022

6. Ngày nộp đồ án: 31/05/2022

Trưởng Bộ môn

Đà Nẵng, ngày 25 tháng 02 năm 2022

Người hướng dẫn

LỜI NÓI ĐẦU

Trong quá trình làm đồ án tốt nghiệp, nhóm chúng em đã nhận được nhiều sự giúp đỡ, đóng góp ý kiến, tạo điều kiện thuận lợi, định hướng và chỉ bảo nhiệt tình của quý Thầy/Cô và bạn bè.

Chúng em xin gửi lời cảm ơn chân thành đến ThS. Võ Thị Hương, giảng viên trường Đại học Sư phạm Kỹ thuật – Đại Học Đà Nẵng, người đã tận tình hướng dẫn, chỉ bảo nhóm chúng em trong suốt quá trình làm đồ án tốt nghiệp. Nhóm chúng em cũng xin chân thành cảm ơn các thầy, cô giáo trong trường Đại học Sư phạm Kỹ thuật nói chung và các thầy cô trong khoa Điện – Điện tử nói riêng đã dạy dỗ cho em kiến thức về các môn đại cương cũng như các môn chuyên ngành, giúp chúng em có được cơ sở lý thuyết vững vàng và tạo điều kiện giúp đỡ nhóm chúng em trong suốt quá trình học tập.

Cuối cùng nhóm chúng em xin cảm ơn gia đình, bạn bè luôn bên cạnh ủng hộ và động viên, cảm ơn tất cả bạn bè đã gắn bó và cùng học tập, giúp đỡ nhóm chúng em trong suốt thời gian qua, cũng như trong quá trình hoàn thành đồ án tốt nghiệp.

Xin chân thành cảm ơn!

Sinh viên thực hiện
(Ký và ghi rõ họ tên)

Sinh viên thực hiện
(Ký và ghi rõ họ tên)

Lê Phước Hải Đăng

Trần Thanh Phúc

CAM ĐOAN

Nhóm chúng em xin cam đoan rằng đề án tốt nghiệp “Thiết kế và chế tạo mô hình hỗ trợ thu gom rác thải tự động sử dụng phương pháp dò đường” là công trình nghiên cứu của nhóm. Những phần có tham khảo tài liệu có trong đề án đã được liệt kê và nêu rõ ra tại phần tài liệu tham khảo. Đồng thời những số liệu hay kết quả trình bày trong đề án đều mang tính chất trung thực, không sao chép, đạo nhái.

Nếu như sai, nhóm chúng em xin chịu hoàn toàn trách nhiệm và chịu tất cả các kỷ luật của bộ môn cũng như nhà trường đề ra.

Đã bổ sung, cập nhật theo yêu cầu của Giảng viên phản biện và Hội đồng chấm Đề án tốt nghiệp họp ngày ngày 17, 18/6/2022.

Giảng viên hướng dẫn
(Ký và ghi rõ họ tên)

Sinh viên thực hiện
(Ký và ghi rõ họ tên)

MỤC LỤC

TÓM TẮT	i
NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP	ii
LỜI NÓI ĐẦU	iv
CAM ĐOAN	v
MỤC LỤC.....	vi
CÁC BẢNG VÀ HÌNH VẼ.....	ix
DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT.....	xii
MỞ ĐẦU	xiii
CHƯƠNG I: TỔNG QUAN	1
1.1. Đặt vấn đề	1
1.2. Tổng quan về robot:	4
1.3. Giới thiệu về robot dò đường:.....	6
1.4. Nguyên tắc hoạt động của robot dò đường:.....	6
1.5. Ứng dụng của robot dò đường:	7
1.5.1. Ứng dụng trong công nghiệp	7
1.5.2. Ứng dụng trong ngành ô tô	8
1.5.3. Ứng dụng trong nhà	9
1.5.4. Ứng dụng trong ngành y tế	9
1.5.5. Ứng dụng trong ngành du lịch	10
1.5.6. Ứng dụng của đề tài hướng đến trong thực tế.....	10
1.6. Thuật toán PID và ứng dụng trong robot dò đường:.....	10
1.6.1 Định nghĩa thuật toán PID	10
1.6.2 Giải thuật điều khiển vi tích phân tỉ lệ (PID).....	11
1.6.3. Ứng dụng giải thuật trên dò đường:	12

1.6.4. Giải thuật PID	13
Chương 2: THIẾT KẾ HỆ THỐNG	15
2.1. Giới thiệu:	15
2.2. Thiết kế sơ đồ khối:.....	15
2.3. Thiết kế các khối:	16
2.3.1. Khối xử lý trung tâm	16
a. Giới thiệu về Arduino:.....	16
b. Arduino Mega 2560	18
2.3.2. Thiết kế khối động cơ	21
a. Động cơ DC giảm tốc.....	21
b. Động cơ Servo.....	22
c. Động cơ DC bơm nước	24
2.3.3. Thiết kế khối điều khiển động cơ.....	24
2.3.4. Thiết kế khối dò đường:	25
2.3.5. Thiết kế khối cảm biến khoảng cách.....	27
a. Giới thiệu.....	27
b. Hoạt động của HC-SR05.....	28
c. Sơ đồ kết nối.....	28
2.3.6. Thiết kế khối cảm biến tiệm cận	29
2.3.7. Thiết kế khối khử khuẩn	29
2.3.8. Thiết kế khối hiển thị và âm thanh.....	30
a. Khối hiển thị	30
b. Khối âm thanh	31
2.3.9. Thiết kế khối nguồn	32
a. Pin, mạch sạc cân bằng pin	32
b. Mạch sạc cân bằng pin	33

c. Module giảm áp DC-DC Buck LM2596 3A.....	33
2.3.10. Sơ đồ nguyên lý toàn mạch.....	34
Chương 3: THI CÔNG HỆ THỐNG	35
3.1. Giới thiệu:	35
3.2. Mạch in:	35
3.2.1. Giới thiệu phần mềm Altium Designer.....	35
3.2.2. Mạch in khối dò đường:	36
3.2.3. Mạch in khối xử lí trung tâm:	36
3.3. Thi công mô hình:	37
3.4. Lập trình cho mô hình:.....	38
3.4.1. Giới thiệu phần mềm lập trình Arduino IDE	38
3.4.2. Sơ đồ thuật toán.....	40
3.4.3. Giải thích lưu đồ thuật toán.....	40
3.5. Hướng dẫn sử dụng:	41
Chương 4: KẾT QUẢ - ĐÁNH GIÁ	42
4.1. Giới thiệu:	42
4.2. Kết quả yêu cầu đạt được:.....	42
4.3. Kết quả thực nghiệm:	43
4.4. Kết quả thu được thực tế:	43
4.4.1 Chạy thử mô hình thử nghiệm thực tế	43
4.5. Nhận xét- Đánh giá:	44
4.5.1. Nhận xét	44
4.5.2. Đánh giá	45
KẾT LUẬN	46
TÀI LIỆU THAM KHẢO	47
PHỤ LỤC	49

CÁC BẢNG VÀ HÌNH VẼ

Bảng 2.1: Thông số kỹ thuật Arduino Mega 2560 [11].....	19
Bảng 2.2: Các cổng serial của Arduino Mega 2560.....	21
Bảng 3.1: Chức năng các nhóm lệnh Trong Arduino IDE.....	39
.....	
Hình 1.1: Thực trạng quản lý rác thải ở nước ta hiện nay	1
Hình 1.2: Mô hình thu gom rác thải ở biển [4].....	2
Hình 1.3: Mô hình phân loại rác thải [5].....	2
Hình 1.4: Sản phẩm thùng rác thông minh hiện nay	3
Hình 1.5: Lịch sử ra đời của robot.....	4
Hình 1.6: Robot gấp vật	5
Hình 1.7: Robot tự hành trong nhà kho	5
Hình 1.8: Robot cộng tác	6
Hình 1.9: Robot tự động hoá Automatic-Production-Line	6
Hình 1.10: Cấu trúc robot dò đường.....	7
Hình 1.11: Robot vận chuyển hàng hoá	8
Hình 1.12: Công nghệ ô tô tự lái.....	8
Hình 1.13: Robot lau sàn	9
Hình 1.14: Robot tự động hỗ trợ đội ngũ y tế	9
Hình 1.15: Sơ đồ khối giải thuật PID [9].....	11
Hình 1.16: Sơ đồ khối giải thuật PID ứng dụng vào robot dò đường [9].....	12
Hình 2.1: Sơ đồ khối hệ thống.....	15
Hình 2.2: Những người tạo ra Arduino.....	17
Hình 2.3: Bo mạch Arduino Mega 2560	18
Hình 2.4: Sơ đồ linh kiện trên Arduino Mega 2560	19
Hình 2.5: Sơ đồ chân Board Mega 2560.....	20

Hình 2.6: Động cơ giảm tốc JGB37-520 200RPM	22
Hình 2.7: Động cơ Servo MG996	23
Hình 2.8: Sơ đồ kết nối Servo với Arduino Mega 2560	23
Hình 2.9: Động cơ bơm nước 385 12VDC.....	24
Hình 2.10: Hình ảnh và sơ đồ chân module L298N.....	25
Hình 2.11: Kết nối Arduino với LN298N điều khiển động cơ.....	25
Hình 2.12: Nguyên lý hoạt động của cảm biến quang điện trở	26
Hình 2.13: Sơ đồ nguyên lý khối dò đường	26
Hình 2.14: Các mức lệch khỏi quỹ đạo của robot theo cảm biến dò đường.....	27
Hình 2.15: Module cảm biến siêu âm HC-SR05	27
Hình 2.16: Hoạt động của cảm biến siêu âm	28
Hình 2.17: Sơ đồ kết nối khối cảm biến khoảng cách với Arduino.....	28
Hình 2.18: Module cảm biến hồng ngoại LM385	29
Hình 2.19: Sơ đồ nguyên lý của cảm biến hồng ngoại	29
Hình 2.20: Sơ đồ nguyên lý khối khử khuẩn	30
Hình 2.21: Module LCD 1602.....	30
Hình 2.22: Buzzer 12V DC.....	32
Hình 2.23: Sơ đồ nguyên lý khối âm thanh kết nối với Arduino	32
Hình 2.24: Pin 18650	32
Hình 2.25: Sơ đồ kết nối mạch sạc pin.....	33
Hình 2.26: Module Buck DC-DC LM2596 3A	33
Hình 2.27: Sơ đồ kết nối module giảm áp với Arduino	34
Hình 2.28: Sơ đồ nguyên lý toàn mạch	34
Hình 3.1: Giao diện thiết kế Altium Design	35
Hình 3.2: Mạch in khối dò đường	36
Hình 3.3: Mạch 3D khối dò đường	36

Hình 3.4: Mạch in tổng xử lý hệ thống	36
Hình 3.5: Mạch 3D tổng xử lý hệ thống	37
Hình 3.6: Hình ảnh mô hình thùng rác mô phỏng	37
Hình 3.7: Hình ảnh nhóm lắp ráp mô hình	37
Hình 3.8: Hình ảnh thực tế thiết kế mô hình	38
Hình 3.9: Lắp ráp các khối với nhau	38
Hình 3.10: Giao diện Arduino IDE	39
Hình 3.11: Chọn Port để kết nối	39
Hình 3.12: Lưu đồ thuật toán chương trình chính	40

DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT

KÝ HIỆU:

- < : Dấu bé hơn và ngược lại.
- %: Phần trăm.
- ...: Còn nữa.
- V: Volt.
- ° : Độ.

CHỮ VIẾT TẮT:

- PID: Proportional Integral Derivative.
- DxRxC: Chiều dài x Chiều rộng x Chiều Cao.
- DC: Dòng một chiều.
- AGV: Autonomous Guided Vehicles.
- ADC: Analog to Digital Converter. (bộ chuyển đổi analog sang kỹ thuật số)
- Max: tối đa.
- Min: tối thiểu.
- PWM: Pulse Width Modulation (phương pháp điều xung, điều chỉnh áp ra tải)
- LINE: đường đi.
- DIY: Viết tắt từ “Do It Yourself” nghĩa là tự làm, thuật ngữ này để diễn đạt cho việc bạn tự làm một món đồ, một sản phẩm,.. nhằm mục đích sử dụng.

MỞ ĐẦU

Hiện nay vấn đề ô nhiễm môi trường luôn là một vấn đề nóng của nước ta, bên cạnh đó, việc xử lý rác thải sau khi sử dụng còn nhiều hạn chế, bất cập, chưa được xử lý một cách thông minh, có hệ thống. Những căn bệnh lây nhiễm từ nguồn rác thải qua sử dụng nếu không được xử lý đúng cách thì sẽ gây hại rất lớn đến sức khoẻ con người. Đồng thời trong thời đại công nghiệp hoá hiện đại hoá hiện nay, đặc biệt là cuộc cách mạng công nghệ IOT 4.0, yêu cầu về một cuộc sống thông minh, hiện đại và sang trọng luôn được đặt lên hàng đầu. Vì lí do đó, nhóm chúng em quyết định chọn đề tài “Thiết kế và chế tạo mô hình hỗ trợ thu gom rác thải tự động sử dụng phương pháp dò đường” để làm đề tài tốt nghiệp.

Mục tiêu của đề tài:

- Sử dụng module dò đường bằng quang trở để định hướng di chuyển theo đường đặt sẵn.
- Thiết kế được một chiếc thùng rác có khả năng nhận biết được rác đầy trong thùng.
- Thiết kế thùng rác mở nắp tự động khi ta tới gần và đóng lại khi rời đi.
- Khi phát hiện rác trong thùng đầy, thùng rác có thể tự di chuyển đến vị trí thu gom và có thể di chuyển về vị trí cũ trên đường đường định sẵn, điều khiển bằng một nút nhấn.
- Trên đường đi, nếu có vật cản thì mô hình sẽ dừng lại và phát tín hiệu cảnh báo.
- Tích hợp sát khuẩn tay tự động cho thùng rác.
- Hiện thị ra màn hình LCD thông báo cho người dùng.

Giới hạn của đề tài:

Khối lượng, tải trọng, tỉ lệ ngoài của mô hình:

- Kích thước mô hình: 30 x 30 x 37 cm (DxRxH)
- Khối lượng mô hình khi không tải 1- 2kg
- Khối lượng mô hình khi có tải 4-5kg

Phạm vi hoạt động của đề tài:

- Hoạt động trong điều kiện thời tiết ổn định, thoáng đãng, bằng phẳng, không gồ gề.

- Vị trí đặt và đường đi của thùng rác phải thông thoáng, ít bị che khuất, dán đường chắc chắn, không bong tróc đường.
- Nên sử dụng trong văn phòng, trường học, cơ quan thoáng đãng, sạch sẽ.

Các module, linh kiện điện tử cần có trong đề tài:

- Sử dụng board Arduino Mega 2560 làm khối điều khiển trung tâm.
- Sử dụng board cảm biến quang trở để dò đường tự động.
- Sử dụng module L298N để điều khiển động cơ DC 12V: áp điều khiển 5V – 12V, dòng điều khiển 2A. Áp của tín hiệu điều khiển: 5V-7V, dòng của tín hiệu điều khiển: 0-36mA.
- Sử dụng module giảm áp DC LM2596 3A để giảm áp vào: áp hoạt động 3V-30V, dòng hoạt động: 3A.

Số lượng cảm biến, động cơ... cần có trong đề tài:

- 2 Cảm biến siêu âm HC-SR05: điện áp hoạt động 5V, dòng hoạt động 15mA.
- 1 Cảm biến hồng ngoại: điện áp hoạt động 3.3V-5V.
- Có sử dụng buzzer để báo hiệu.
- Sử dụng động cơ Servo MG996R để điều khiển mở nắp tự động: điện áp điều khiển 4.8V-7.2V.
- Sử dụng màn hình LCD hiển thị thông báo: điện áp hoạt động 5V.
- Dùng 2 động cơ DC giảm tốc 250 vòng/phút khi không tải.
- Dùng 1 bơm 12V để bơm nước sát khuẩn.
- Mô hình xây dựng dựa trên thùng rác nhựa có sẵn và đế thùng rác làm bằng mica cắt CNC.

Đề tài gồm có 04 chương chính:

Chương 1: Tổng quan

Chương này đề cập đến vấn đề thực trạng rác thải hiện nay, lý do chọn đề tài, nội dung nghiên cứu, lịch sử phát triển robot và ứng dụng của robot hiện nay, trình bày hướng ứng dụng của đề tài trong thực tế. Giới thiệu trình bày sơ lược về PID, ứng dụng thuật toán PID vào robot dò đường.

Chương 2: Thiết kế hệ thống

Chương này sẽ giới thiệu tổng quan về các yêu cầu của đề tài mà mình thiết kế thiết kế gồm những phần nào từ đó thiết kế sơ đồ khối hệ thống, sơ đồ nguyên lý từng khối, chọn linh kiện và cách kết nối cho các khối chức năng của hệ thống.

Chương 3: Thi công hệ thống

Chương này sẽ giới thiệu các phần mềm sử dụng, làm mạch in, quá trình thực hiện, lắp ráp hệ thống, vẽ lưu đồ thuật toán, lập trình và thi công cho đề tài.

Chương 4: Nhận xét – Đánh giá

Chương này trình bày kết quả yêu cầu đạt được, kết quả thực nghiệm, đánh giá mức độ hoàn thiện sản phẩm, đánh giá ưu điểm, nhược điểm của sản phẩm khi chạy thực tế, và hướng phát triển trong tương lai.

CHƯƠNG I: TỔNG QUAN

1.1. Đặt vấn đề:

Thực trạng quản lý rác thải hiện nay ở nước ta:

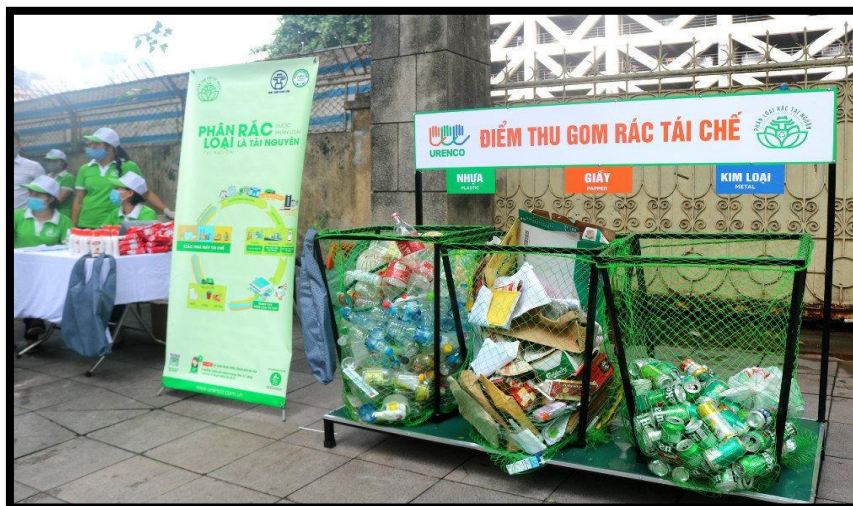


Hình 1.1: *Thực trạng quản lý rác thải ở nước ta hiện nay*

Quản lý chất thải là một trong những vấn đề nóng hổi mà thế giới phải đối mặt không phân biệt đó là nước phát triển hay đang phát triển. Vấn đề chính là trong việc quản lý chất thải, thùng rác ở nơi công cộng được luôn luôn trong tình trạng đầy và quá tải trước khi được xe rác tới lấy và đưa lượng rác đó ra một vị trí tập trung lượng rác lớn khác để xử lý. Việc thùng rác nơi công cộng luôn đầy và tràn ra lần lượt dẫn đến nhiều mối nguy hiểm khác nhau như bốc mùi, ô nhiễm môi trường cảnh quan và đó cũng là gốc rễ, nguyên nhân chính gây ra những căn bệnh lây nhiễm. Điều này cũng đang dần xuất hiện trong môi trường học đường khi xuất phát từ chính trong ý thức của mỗi sinh viên chúng ta, thùng rác đã đầy nhưng chúng ta vẫn bỏ rác vào gây ra tình trạng rác văng ra ngoài. Để tránh tất cả nguy hiểm và duy trì sự sạch sẽ công cộng và sức khỏe cộng đồng trong trường học cũng như là ngoài xã hội, chúng ta cần phải thiết kế ra được một mô hình thu gom rác tự động mà nó có thể kiểm soát được lượng rác và đồng thời đưa ra những xử lý cần thiết nhằm giảm thiểu hiện trạng trên. [3]



Hình 1.2: Mô hình thu gom rác thải ở biển [4]



Hình 1.3: Mô hình phân loại rác thải [5]

Nhìn chung, mô hình hỗ trợ thu gom rác thải ứng dụng dò đường đã khá phổ biến. Đã có rất nhiều công ty start-up và những công ty thiết bị đã và đang đầu tư vào lĩnh vực mô hình thu gom rác thải thông minh này. Điều đó là hoàn toàn hợp lý và có cơ sở, bởi lẽ mô hình thu gom rác thải tự động là một sản phẩm không thể thiếu được trong các văn phòng công ty hiện đại, trong thành phố hiện đại 4.0. Không chỉ làm nhiệm vụ đựng rác, nó còn thể hiện sự văn minh, sang trọng.



Hình 1.4: Sản phẩm thùng rác thông minh hiện nay

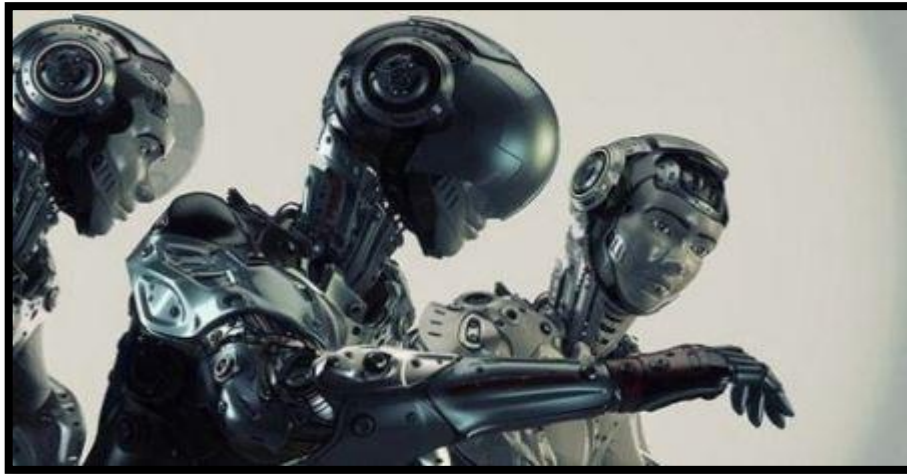
Chính vì vậy, sự ra đời của mô hình thu gom rác thông minh sẽ là sự lựa chọn hiện đại hàng đầu cho các văn phòng công ty, trường học, những nơi công cộng.

Mô hình thu gom rác thông minh hoàn toàn khác so với những mô hình thu gom rác cổ điển mà chúng ta sử dụng ở những nơi công cộng. Đó chính là về khả năng tự động và khả năng di động hơn, sẽ có nhiều không gian và thời gian để xử lý những tác vụ tự động, chẳng hạn như: khi rác đầy, có thể tự động nhận biết và di chuyển tới vị trí nơi chứa rác tổng qua việc định sẵn đường đi, hay hiện đại hơn là sự liên kết nhiều thùng rác lại với nhau để người lao công dễ dàng quản lý... là những tính năng mà mô hình thu gom rác cổ điển không thể làm được.

Do đó, nhận thấy những tính năng và công dụng tuyệt vời của mô hình thu gom rác thải thông minh như vậy, kết hợp với những vấn đề nhức nhối trong việc quản lý rác thải ở những nơi công cộng ở nước ta hiện nay. Do đó, việc hướng đến chọn đề tài “Thiết kế và chế tạo mô hình hỗ trợ thu gom rác thải tự động sử dụng phương pháp dò đường” ứng dụng ở văn phòng công ty, trường học, cơ quan, bệnh viện... và được áp dụng công nghệ dò đường tự động để di chuyển là một nhu cầu hết sức cần thiết và đây chính là lý do mà nhóm nghiên cứu quyết định chọn đề tài này. Đề tài này không những là một thực tại khách quan mà nó còn đóng vai trò quan trọng thực sự trong tương lai sau này, đặc biệt là có thể ứng dụng rất tốt trong môi trường trường học và những nơi công cộng.

1.2. Tổng quan về robot:

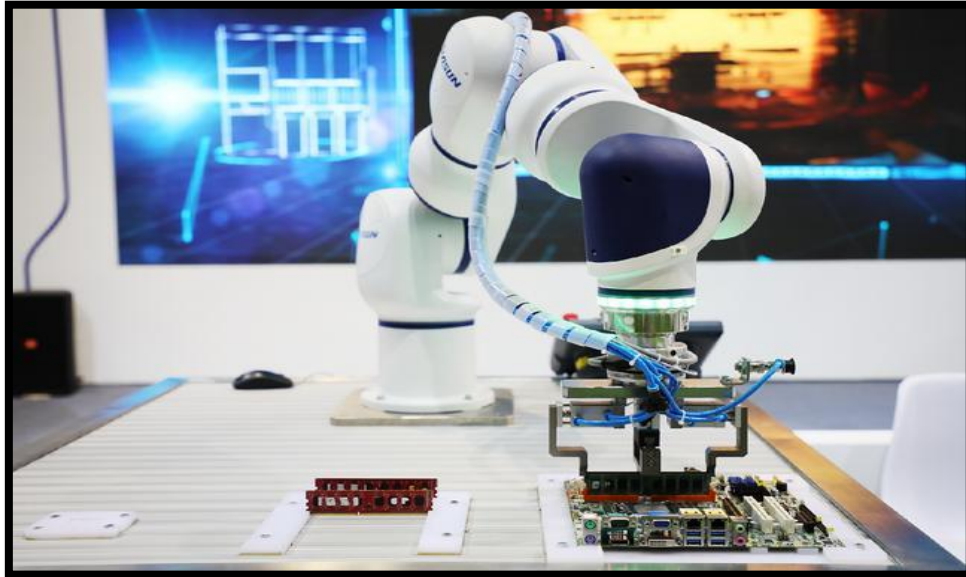
Robot đã và đang xuất hiện trong cuộc sống của chúng ta từ lâu và ngày càng trở thành một phần không thể thiếu trong cuộc sống hiện đại. Chúng đã góp phần mình vào công cuộc lao động, chính robot đang làm nên một cuộc cách mạng về lao động, khoa học, và đang phục vụ đắc lực cho các ngành khoa học như: Khoa học quân sự, khoa học giáo dục, các ngành dịch vụ, giải trí... [6]



Hình 1.5: Lịch sử ra đời của robot

Vậy robot xuất hiện từ khi nào? Đó là vào năm 1921 nhà soạn kịch Karel Capek người Tiệp Khắc đã đưa lên sân khấu vở kịch có tiêu đề “Romands Univesal Robot”. Theo tiếng Séc “Robot” nghĩa là “Người tạp dịch”. Có thể nói đây là một gợi ý, một ý tưởng ban đầu về những cỗ máy có khả năng thao tác như con người. Đến trước chiến tranh thế giới lần thứ hai nhu cầu sử dụng những máy móc có khả năng thay thế con người ở những môi trường làm việc độc hại đã trở thành một nhu cầu cấp thiết. Ban đầu cơ cấu máy này hoạt động giống như tay máy của người vận hành. Cấu tạo của cơ cấu này bao gồm các thanh và các khớp và hệ thống giầy chằng. Người vận hành điều khiển tay máy thông qua một cơ cấu khuyếch đại cơ khí. Trong chiến tranh thế giới lần thứ hai (năm 1945), xuất hiện cơ cấu máy được điều khiển từ xa để cầm nắm chất phóng xạ. Cho đến những năm 1950 cùng với sự ra đời của kỹ thuật điều khiển chương trình số NC (Number Control) kỹ thuật tay máy lúc này đã kết hợp được cả kỹ thuật điều khiển xa và điều khiển chương trình số. Sự kết hợp này đã tạo ra những thể hệ máy điều khiển từ xa có khả năng mềm dẻo, khả năng tự động hoá cao gọi tên là robot. Năm 1949, máy phay điều khiển số ra đời phục vụ sản xuất ở Mỹ. Đến năm

1960 George Devol đưa ra mẫu Robot đầu tiên. Năm 1961 cũng tại Mỹ Robot công nghiệp. (IR: Industrial Robot) đầu tiên đưa ra thị trường: Robot Unimat 1990 (Do trường đại học MIT chế tạo) đây là Robot phản hồi lực nó được ứng dụng vào công nghiệp sản xuất ô tô. Theo con số thống kê thì đến năm 1990 toàn thế giới đã triển khai và ứng dụng khoảng 300.000 IR. Do sự phát triển mạnh mẽ của kỹ thuật Vi xử lý và Tin học mà số lượng IR tăng nhanh chóng và tính năng cũng có nhiều bước phá, giá thành trên một đơn vị IR giảm dần.



Hình 1.6: Robot gắp vật



Hình 1.7: Robot tự hành trong nhà kho



Hình 1.8: Robot cộng tác



Hình 1.9: Robot tự động hoá Automatic-Production-Line

1.3. Giới thiệu về robot dò đường:

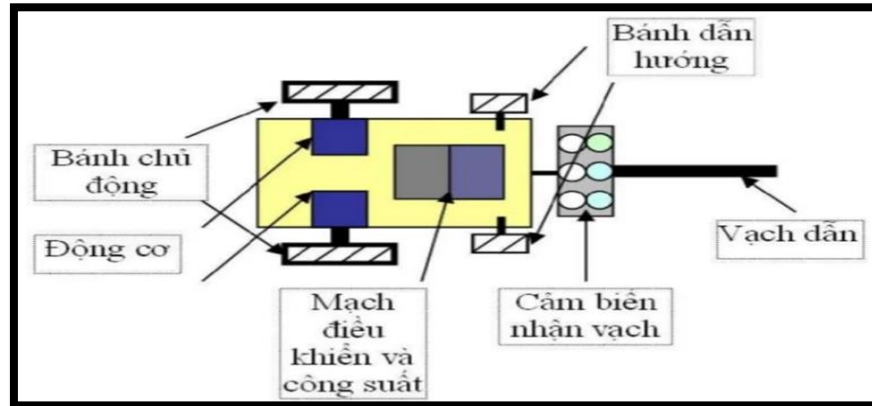
Là loại robot mà việc di động theo một quỹ đạo định sẵn được gọi là Robot bám đường. Robot bám đường có thể di chuyển theo một đường, đường đi có thể được nhìn nhận như một dòng màu đen trên một bề mặt trắng (hoặc ngược lại) hoặc nó có thể là đường vô hình như một từ trường.

1.4. Nguyên tắc hoạt động của robot dò đường:

Nguyên tắc hoạt động: robot chuyển động theo một quỹ đạo mà chúng ta đã mong muốn dự tính trước thông qua việc dán đường đường. Với hệ thống hai bánh xe được dẫn động bởi hai động cơ điện một chiều thông qua mạch điều khiển trung tâm và mạch công suất điều khiển động cơ. Thường thì các vạch đường sẽ có màu khác với màu nền của quỹ đạo chuyển động. Để robot chuyển động đúng quỹ đạo định trước, cần phải có bộ phận cảm biến, bộ phận này có nhiệm vụ phân biệt màu của vạch đường và màu nền của quỹ đạo chuyển động, từ đó đưa tín hiệu giải mã tương

ứng về mạch điều khiển trung tâm. Mạch điều khiển trung tâm có nhiệm vụ thu nhận thông tin phản hồi từ bộ phận cảm biến trả về, từ đó điều khiển tốc độ và chiều quay của hai động cơ điện một chiều sao cho cả hai bánh xe luôn chuyển động tương đối bám được đường vạch đường định sẵn.

Ta có thể hình dung cơ bản cấu trúc của robot dò đường như hình sau:



Hình 1.10: Cấu trúc robot dò đường

Nguyên lý hoạt động của cảm biến dò đường:

Cảm biến hoạt động dựa trên nguyên lý thu – phát ánh sáng. Các mắt phát sẽ phát ra tia hồng ngoại có bước sóng hồng ngoại. Mắt thu ở bề mặt chứa vùng sáng sẽ hấp thụ ánh sáng hồng ngoại đó. ở trạng thái bình thường, mắt thu có nội trở rất lớn (hàng trăm Kilo - Ohm). Khi mắt thu nhận được ánh sáng hồng ngoại chiếu vào thì nội trở của nó giảm xuống (khoảng vài chục Ohm). Do đó robot chỉ hoạt động trong môi trường có tính chất phản xạ ánh sáng cao.

1.5. Ứng dụng của robot dò đường:

1.5.1. Ứng dụng trong công nghiệp

Robot vận chuyển hàng hoá: [7]

Robot vận chuyển hàng hóa hay được biết đến là xe tự hành AGV (Autonomous Guided Vehicles). Đây là Robot được vận hành rất nhiều trong các khu công nghiệp lớn. Với nhiệm vụ phân phối và vận chuyển hàng hóa một cách tự động. Với sự kết hợp của các Sensor quang điện giúp định hướng đường đi. Và Sensor siêu âm tránh va chạm gây ảnh hưởng đến hàng hóa và nhân viên kỹ thuật trong nhà máy. Do vận hành 1 cách tự động, có khả năng tự sạc pin, dễ dàng điều phối và quản lý lượng hàng hóa qua hệ thống. Cộng với ưu điểm hoạt động được lâu dài, vận chuyển lượng hàng

hóa có khối lượng lớn. AGV đang ngày càng được lựa chọn để thay thế hoàn toàn con người để giảm chi phí thuê nhân công, cũng như tăng độ chính xác trong công việc.



Hình 1.11: Robot vận chuyển hàng hoá

1.5.2. Ứng dụng trong ngành ô tô

Công nghệ ô tô tự lái:

Trước đây việc nghĩ đến có 1 chiếc ô tô có thể tự di chuyển mà không cần tài xế, là một suy nghĩ thật viễn vông. Nhưng thực tế ở hiện tại có rất nhiều dòng xe được thử nghiệm và phát triển để có thể thực hiện được điều đó một cách hoàn thiện. Ở Việt Nam cũng đã có những dòng xe được nhập khẩu về có thể tự động tìm và di chuyển vào vị trí đậu xe: các dòng Mercedes, BMW,... mới. Dù hiện tại chưa có dòng xe hoàn toàn có thể được di chuyển trên đường 1 cách tự động. Nhưng trong tương lai gần việc những xe không người lái đi trên đường sẽ không còn xa với đâu, hãy tin tôi đi. Và đây cũng là 1 ứng dụng được phát triển nên từ Robot tự động tìm đường.



Hình 1.12: Công nghệ ô tô tự lái

1.5.3. Ứng dụng trong nhà

Công nghệ robot lau sàn: Gần gũi hơn với chúng ta, Robot lau nhà không còn quá xa lạ với mỗi người. Dù không sử dụng đường đường có sẵn và cảm biến quang điện. Nhưng Robot lau nhà lại áp dụng công nghệ cảm biến siêu âm để tránh va chạm.



Hình 1.13: Robot lau sàn

1.5.4 Ứng dụng trong ngành y tế

Robot vận chuyển hàng hoá, thuốc men tự động trong khu cách ly, bệnh viện dã chiến [8]

Sứ mệnh của Robot này là thay thế con người, hỗ trợ đội ngũ tình nguyện viên trong hoạt động phân phối, phát nhu yếu phẩm cho các khu vực phong tỏa cách ly và bệnh viện điều trị Covid-19, tránh tiếp xúc, lây lan dịch bệnh. Bên cạnh đó, Robot còn có thể thực hiện phun khử khuẩn tầng thấp, hỗ trợ phòng, chống dịch.

Mỗi robot có thể thay thế được 3-5 nhân viên y tế. Ngoài việc giảm rủi ro lây nhiễm, việc sử dụng robot còn tạo điều kiện để nhân viên y tế tập trung thời gian, công sức phục vụ, chăm sóc và điều trị bệnh nhân nặng được tốt hơn.



Hình 1.14: Robot tự động hỗ trợ đội ngũ y tế

1.5.5. Ứng dụng trong ngành du lịch

Robot có thể được sử dụng ở những nơi công cộng để cung cấp bản đồ hướng dẫn đường đi, cung cấp thông tin các điểm tham quan một cách tự động.

1.5.6. Ứng dụng của đề tài hướng đến trong thực tế

Mô hình của đề tài sẽ hoạt động một cách tự động từ khâu chứa rác đến nhận biết rác đầy trong thùng, tự động di chuyển đến điểm tập kết rác đúng quy định và tự động quay về vị trí ban đầu. Với mô hình này, nhóm chúng em hi vọng sẽ có nhiều cơ hội được cải tiến, nâng cấp và ứng dụng vào thực tế cuộc sống hằng ngày, giúp kiểm soát được rác hiệu quả hơn, tránh các bệnh lây nhiễm nguy hiểm và hướng đến một thành phố thông minh, hiện đại trong tương lai.

1.6. Thuật toán PID và ứng dụng trong robot dò đường:

Giới thiệu:

Robot dò đường hay còn gọi là dò line là một trong những phương tiện di chuyển tự động đầu tiên được con người chế tạo, vì môi trường nó yêu cầu khá đơn giản, một nền bằng phẳng và một vạch màu tương phản mạnh (hoặc có các đặc tính khác) khác biệt mạnh so với môi trường xung quanh. Ta sẽ dùng cảm biến để đo đạc, phát hiện sự khác biệt này, sau đó nhúng một thuật toán điều khiển nào đó vào để điều khiển xe bám theo vạch đường đó. Để robot thật sự hoạt động một cách cân bằng và ổn định có thể kể đến rất nhiều thuật toán như PID, pole placement, H infinity, LQR,... trong đó PID là thuật toán chính để điều khiển động cơ cho đề tài này.

1.6.1 Định nghĩa thuật toán PID

PID là bộ điều khiển vi tích phân tỉ lệ được viết tắt từ: Proportional Integral Derivative) là một cơ chế phản hồi vòng điều khiển (bộ điều khiển) tổng quát được sử dụng rộng rãi trong các hệ thống điều khiển công nghiệp – bộ điều khiển PID là bộ điều khiển được sử dụng nhiều nhất trong các bộ điều khiển phản hồi.

Bộ điều khiển PID sẽ tính toán giá trị "sai số" là hiệu số giữa giá trị đo thông số biến đổi và giá trị đặt mong muốn. Bộ điều khiển sẽ thực hiện giảm tối đa sai số bằng cách điều chỉnh giá trị điều khiển đầu vào. Trong trường hợp không có kiến thức cơ bản về hệ thống điều khiển thì bộ điều khiển PID là sẽ bộ điều khiển tốt nhất. Tuy nhiên, để đạt được kết quả tốt nhất, các thông số PID sử dụng trong tính toán phải điều chỉnh theo tính chất của hệ thống, trong khi kiểu điều khiển là giống nhau, các thông số phải phụ thuộc vào đặc thù của hệ thống.

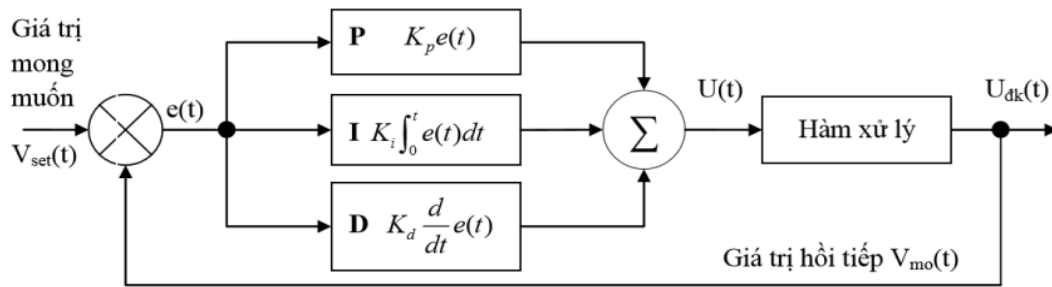
Giải thuật tính toán bộ điều khiển PID bao gồm 3 thông số riêng biệt, do đó đôi khi nó còn được gọi là điều khiển ba khâu, các giá trị tỉ lệ, tích phân và đạo hàm, viết tắt là P, I, và D. Giá trị tỉ lệ xác định tác động của sai số hiện tại, giá trị tích phân xác định tác động của tổng các sai số quá khứ, và giá trị vi phân xác định tác động của tốc độ biến đổi sai số.

Tổng chập của ba tác động này dùng để điều chỉnh quá trình thông qua một phần tử điều khiển như vị trí của van điều khiển hay bộ nguồn của phần tử gia nhiệt. Nhờ vậy, những giá trị này có thể làm sáng tỏ về quan hệ thời gian: (P) phụ thuộc vào sai số hiện tại, (I) phụ thuộc vào tích lũy các sai số quá khứ, và (D) dự đoán các sai số tương lai, dựa vào tốc độ thay đổi hiện tại. [9]

1.6.2 Giải thuật điều khiển vi tích phân tỉ lệ (PID)

Giải thuật điều khiển vi tích phân tỉ lệ hiệu chỉnh sai số giữa giá trị biến đo được và giá trị mong muốn đạt đến bằng cách tính toán và xuất ra một "hành động điều chỉnh" nhanh chóng để giữ cho lỗi ở mức nhỏ nhất có thể được. Bộ điều khiển này bao gồm 3 thông số: Proportional (P), Integral (I) và Derivative (D). [9]

Sơ đồ khối PID:



Hình 1.15: Sơ đồ khối giải thuật PID [9]

- Phương trình PID tổng quát:

$$U(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{d}{dt} e(t) \quad (1.1)$$

- Hàm lỗi $e(t)$ tại thời điểm t : độ lệch giữa giá trị mong muốn và giá trị đo được:

$$e(t) = V_{set} - V_{mo}(t) \quad (1.2)$$

Trong đó:

V_{set} : Setpoint Value (giá trị điểm đặt điện áp mong muốn).

$V_{mo}(t)$: Measurable Output Value (giá trị điện áp đo được tại thời điểm t).

Proportional P(t) (hàm tỉ lệ): điều khiển tỉ lệ với giá trị lỗi hiện tại.

$$P(t) = K_p \cdot e(t) \tag{1.3}$$

Trong đó:

K_p : hằng số tỉ lệ, tham số dùng để tinh chỉnh.

Integral $I(t)$ (hàm tích phân): điều khiển tương ứng với mức lỗi được tích lũy theo thời gian.

$$I(t) = K_i \cdot \int_0^t e(t) dt \tag{1.4}$$

Trong đó:

K_i : hằng số điều chỉnh hàm tích phân.

Derivative $D(t)$ (hàm vi phân): điều khiển tương ứng với tốc độ thay đổi của lỗi, hay độ dốc lỗi theo thời gian.

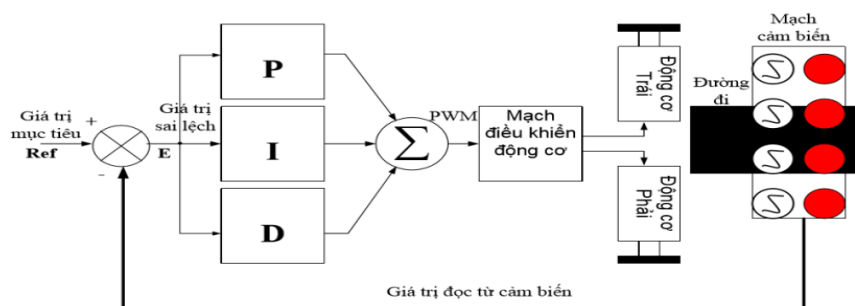
$$D(t) = K_d \cdot \frac{d}{dt} e(t) \tag{1.5}$$

Trong đó: K_d : hằng số điều chỉnh hàm vi phân.

1.6.3. Ứng dụng giải thuật trên dò đường:

Mô hình PID áp dụng dò đường:

Giải thuật PID được tích hợp vào trong giải thuật điều khiển robot dò đường để tính toán và xuất ra các giá trị PWM điều khiển 2 động cơ gắn với 2 bánh sau của robot dựa vào độ sai lệch giữa giá trị đo đạc ngõ vào và giá trị mong muốn. Độ sai lệch, hay còn gọi là giá trị lỗi, được tính toán trong giải thuật PID là độ lệch giữa trạng thái hiện tại của robot so với đường đi. Có nghĩa là, độ sai lệch giữa giá trị hiện tại của bộ cảm biến so với giá trị của cảm biến trong trường hợp robot chạy thẳng về phía trước. [7]



Hình 1.16: Sơ đồ khối giải thuật PID ứng dụng vào robot dò đường [9]

- Trước khi đề cập ứng dụng giải thuật PID vào robot dò đường, tác giả đưa ra các định nghĩa của các thuật ngữ liên quan.

- Vị trí mục tiêu V_{set} : trung tâm của đường đi, vị trí mà ta muốn robot hướng đến.
- Vị trí thực tại $V_{mo}(t)$: vị trí thực của robot, có thể lệch bên trái hoặc lệch bên phải so với đường đi của robot.
- Lỗi $e(t)$: độ sai lệch giữa vị trí mục tiêu và vị trí thực của robot, có thể là giá trị âm hoặc dương hoặc bằng 0, được tính theo công thức (1.2)
- Hệ số K_p , K_i , K_d là các hằng số tương ứng được sử dụng để điều chỉnh ảnh hưởng của các khâu tỉ lệ, khâu tích phân và khâu vi phân.

1.6.4. Giải thuật PID

Giải thuật PID là một chuỗi các phép toán để xác định các giá trị của khâu điều khiển tỉ lệ, khâu điều khiển vi phân và khâu điều khiển tích phân. Sau đó, kết hợp các giá trị lại với nhau và xuất ra giá trị điều rộng xung PWM để điều khiển 2 động cơ bánh sau của robot. Khâu điều khiển tỉ lệ tính toán độ lệch của robot, xác định được robot đang lệch phải hay lệch trái đường và xuất ra các giá trị PWM để điều khiển hai động cơ giúp robot quay về vị trí mục tiêu. [9]

- Độ lệch: $e(n) = V_{mo}(n) - V_{set}$
- Đáp ứng khâu tỉ lệ: $P(n) = K_p \cdot e(n)$
- Trong đó: n là số lần lấy mẫu
- Hằng số K_p càng lớn thì sự điều chỉnh càng lớn khi lỗi càng lớn. Ngược lại, K_p càng nhỏ thì sẽ đáp ứng điều chỉnh nhỏ đối với lỗi.

Khâu điều khiển tích phân tính toán dựa trên sai số tích lũy theo thời gian. Robot càng lệch xa đường thì sai số tích lũy càng lớn. Khâu này thể hiện rằng trong quá khứ robot có bám đường đi tốt hay không. [11]

- Sai số tích lũy:

$$E(n) = \sum e(n) \quad n \geq 1 \quad (1.6)$$

- Đáp ứng khâu tích phân: $I(n) = K_i \cdot E(n)$
- Trong đó: n là số lần lấy mẫu

Khâu vi phân tính toán dựa trên sai số hiện tại. Khâu này cho biết tốc độ robot dao động qua lại của đường đi:

$$\text{Tốc độ dao động: } \Delta e(n) = e(n) - e(n-1)$$

$$\text{Đáp ứng khâu vi phân: } D(n) = K_d \cdot \Delta e(n)$$

Cuối cùng, giá trị PWM để xuất ra điều khiển cho các động cơ điều khiển hoạt động của robot sẽ là:

$$\begin{aligned} \text{PWM} &= P(n) + I(n) + D(n) = K_p \cdot e(n) + K_i \cdot I(n) + K_d \cdot D(n) \\ \Rightarrow \text{PWM} &= K_p \cdot [V_{mo}(n) - V_{set}] + K_i \cdot \sum e(n) + K_d \cdot [e(n) - e(n-1)] \\ & \qquad \qquad \qquad n \geq 1 \end{aligned}$$

1.6.5. Hiệu chỉnh thuật toán PID

Khi sử dụng giải thuật PID để điều khiển robot dò đường, kết quả ban đầu luôn cho thấy robot hoạt động không như mong đợi. Trường hợp này được lý giải là do các giá trị của các hệ số K_p , K_i và K_d chưa phù hợp. Mỗi robot có một đặc tính vật lý khác nhau do đó các hệ số này cũng phải thay đổi để phù hợp.

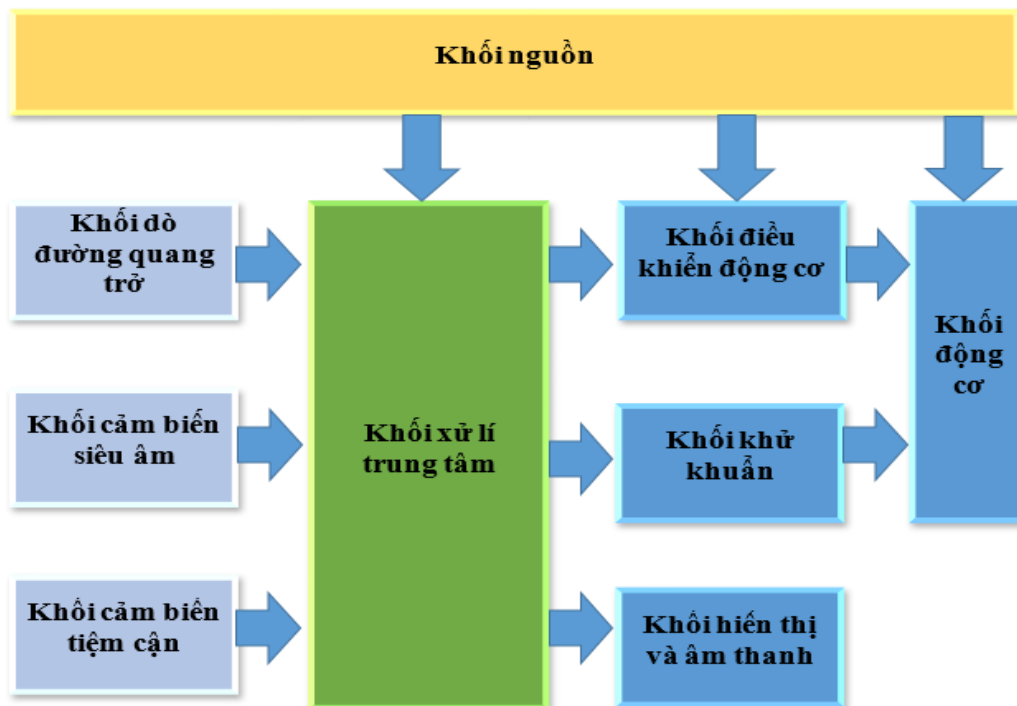
Chương 2: THIẾT KẾ HỆ THỐNG

2.1. Giới thiệu:

Trong chương này, trình bày về sơ đồ khối, lựa chọn linh kiện phù hợp với đề tài, sơ đồ nguyên lý của các board mạch, cách kết nối giữa các module với nhau.

2.2. Thiết kế sơ đồ khối:

Với các yêu cầu đã đưa ra, nhóm đã thiết kế và hình thành sơ đồ khối cho thùng rác như sau:



Hình 2.1: Sơ đồ khối hệ thống

Chức năng của từng khối:

- **Khối nguồn:** có chức năng cung cấp nguồn cho từng hệ thống.
- **Khối xử lý trung tâm:** có chức năng nhận và xử lý các tín hiệu các cảm biến và nút nhấn đầu vào. Sau đó truyền các tín hiệu điều khiển cho khối chấp hành để làm các động cơ hoạt động.
- **Khối cảm biến dò đường:** Có chức năng nhận tín hiệu từ các quang trở với sự hỗ trợ của led hồng ngoại thông qua đọc ADC trên arduino để xử lý và điều khiển động cơ di chuyển theo đường.

- **Khối cảm biến siêu âm:** Có chức năng cung cấp cho hệ thống khả năng đọc khoảng cách đến vật cản.
- **Khối cảm biến tiệm cận:** Có chức năng cung cấp hệ thống khả năng quét phát hiện vật cản.
- **Khối điều khiển động cơ:** Giúp cho dễ dàng trong việc xử lý các thao tác điều khiển độc lập các motor, servo đồng thời cho phép khối xử lý trung tâm điều khiển được nhiều động cơ hơn, với công suất lớn hơn.
- **Khối động cơ:** Bao gồm 2 motor DC để thùng rác di chuyển cũng như 1 Servo để đóng mở nắp thùng rác .
- **Khối khử khuẩn:** Bao gồm một mạch điện tử sử dụng mosfet để bật tắt động cơ bơm khi nhận tín hiệu từ khối cảm biến tiệm cận được xử lí qua khối xử lí trung tâm, có chức năng bơm cồn sát khuẩn tay.
- **Khối hiển thị và âm thanh:** Có chức năng hiển thị và thông báo qua LCD và buzz để giám sát dung lượng rác trong thùng và báo động khi có vật cản trong quá trình di chuyển của hệ thống.

2.3. Thiết kế các khối:

2.3.1. Khối xử lý trung tâm

Có chức năng nhận vào và xử lý các tín hiệu từ các khối dò đường, cảm biến khoảng cách, cảm biến tiệm cận. Sau đó truyền các tín hiệu điều khiển ra Module điều khiển động cơ để làm cho động cơ hoạt động theo cách người sử dụng lập trình.

Về việc đi thiết kế khối xử lý trung tâm thì thị trường hiện nay đáp ứng cho chúng ta rất nhiều các phương án khác như: vi điều khiển hãng Microchip (tiêu biểu PIC 16F887, 18F4550, vi điều khiển hãng ATMEL (AT89C52), RASPBERRY PI, ARDUINO. Mỗi hãng đều có ưu nhược điểm riêng nhưng trong quá trình tìm hiểu thì Board Arduino Mega 2560 chính là vi điều khiển được sử dụng trong đề tài này.

a. Giới thiệu về Arduino:

Arduino thực sự đã gây sóng gió trên thị trường người dùng DIY trên toàn thế giới trong vài năm gần đây, gần giống với những gì Apple đã làm được trên thị trường thiết bị di động. Số lượng người dùng cực lớn và đa dạng với trình độ trải rộng từ bậc phổ thông lên đến đại học đã làm cho ngay cả những người tạo ra chúng phải ngạc nhiên về mức độ phổ biến.



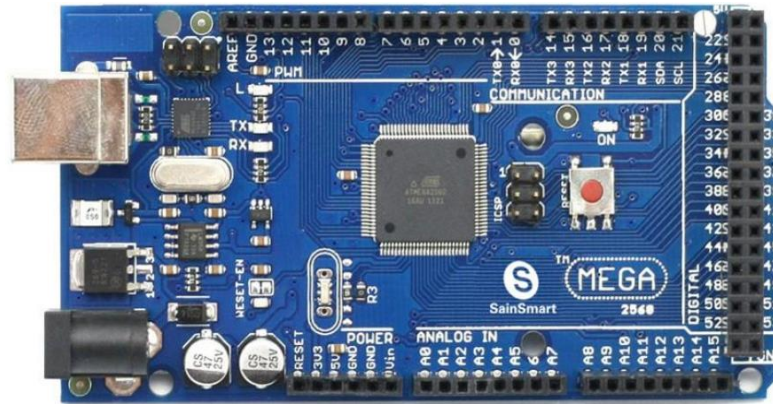
Hình 2.2: Những người tạo ra Arduino

Arduino là gì mà có thể khiến ngay cả những sinh viên và nhà nghiên cứu tại các trường đại học danh tiếng như: Stanford, Carnegie Mellon phải sử dụng, hoặc ngay cả Google cũng muốn hỗ trợ khi cho ra đời bộ kit Arduino Mega ADK dùng để phát triển các ứng dụng Android tương tác với cảm biến và các thiết bị khác.

Arduino thật ra là một bo mạch vi xử lý được dùng để lập trình tương tác với các thiết bị phần cứng như cảm biến, động cơ, đèn hoặc các thiết bị khác. Đặc điểm nổi bật của Arduino là môi trường phát triển ứng dụng cực kỳ dễ sử dụng, với một ngôn ngữ lập trình có thể học một cách nhanh chóng ngay cả với người ít am hiểu về điện tử và lập trình. Và điều làm nên hiện tượng Arduino chính là mức giá rất thấp và tính chất nguồn mở từ phần cứng tới phần mềm. Chỉ với khoảng \$30, người dùng đã có thể sở hữu một board Arduino có 20 ngõ I/O có thể tương tác và điều khiển chùng ấy thiết bị.

Arduino ra đời tại thị trấn Ivrea thuộc nước Ý và được đặt theo tên một vị vua vào thế kỷ thứ 9 là King Arduino. Arduino chính thức được đưa ra giới thiệu vào năm 2005 như là một công cụ khiêm tốn dành cho các sinh viên của giáo sư Massimo Banzi, là một trong những người phát triển Arduino, tại trường Interaction Design Institute Ivrea(IDII). Mặc dù hầu như không được tiếp thị gì cả, tin tức về Arduino vẫn lan truyền với tốc độ chóng mặt nhờ những lời truyền miệng tốt đẹp của những người dùng đầu tiên. Hiện nay Arduino nổi tiếng tới nỗi có người tìm đến thị trấn Ivrea chỉ để tham quan nơi đã sản sinh ra Arduino.

b. Arduino Mega 2560

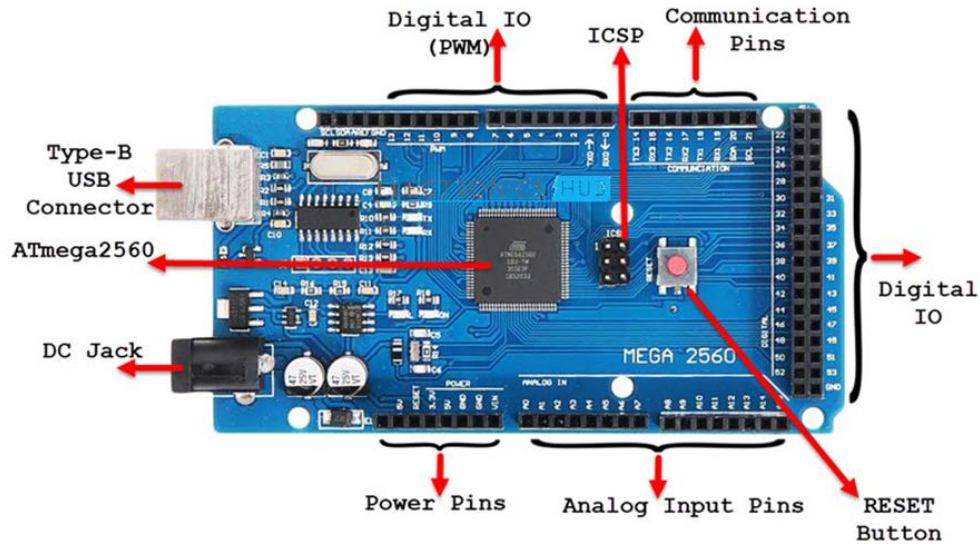


Hình 2.3: *Bo mạch Arduino Mega 2560*

Trong những năm gần đây, sự phát triển mạnh mẽ của Arduino đã cho thấy tầm ảnh hưởng rất nhiều tới thế giới hiện tại, cộng đồng sử dụng lớn cùng với nguồn tài liệu phong phú trên mạng, thư viện hỗ trợ hầu hết các Module, các sản phẩm thiết bị ngoại vi khác, thao tác dễ xử lý và bo mạch gọn nhẹ là ưu điểm sử dụng với đề tài như thế này.

Với những yêu cầu của đề tài này là thực hiện các công việc điều khiển đơn giản và dung lượng chương trình không quá lớn nên ở đây lựa chọn sử dụng Arduino là đủ để đáp ứng cho hệ thống hoạt động hiệu quả ngoài ra các dòng Arduino trên thị trường hiện nay còn có độ thông dụng cao, giá thành rẻ hơn so với các bộ Kit raspberry hay ARM rất phù hợp với sinh viên. Cụ thể là lựa chọn dòng Arduino Mega để đi thiết kế vì so với Arduino Uno thì Arduino Mega hỗ trợ nhiều chân hơn, nhờ thế chúng ta không cần phải lo lắng về vấn đề thiếu chân nối. Vì thế nhóm chọn Arduino Mega để lập trình, điều khiển. Arduino được sử dụng trong đề tài này là Arduino Mega 2560 với thiết kế phần cứng bao gồm 54 chân digital (15 chân có thể sử dụng như các chân PWM, 16 đầu vào analog, 4 UARTs (Cổng nối tiếp phần cứng, 1 thạch anh 16 MHz, 1 cổng kết nối USB, 1 jack cắm điện, 1 đầu ICSP, 1 nút reset. Ngoài ra Arduino Mega 2560 cơ bản là giống với Arduino Uno R3 mà ta vẫn thường hay sử dụng, chỉ khác về số chân và tính năng, do đó ta có thể lập trình tương tự như lập trình cho Arduino Uno R3.

Sơ đồ linh kiện và chân của Arduino Mega 2560:

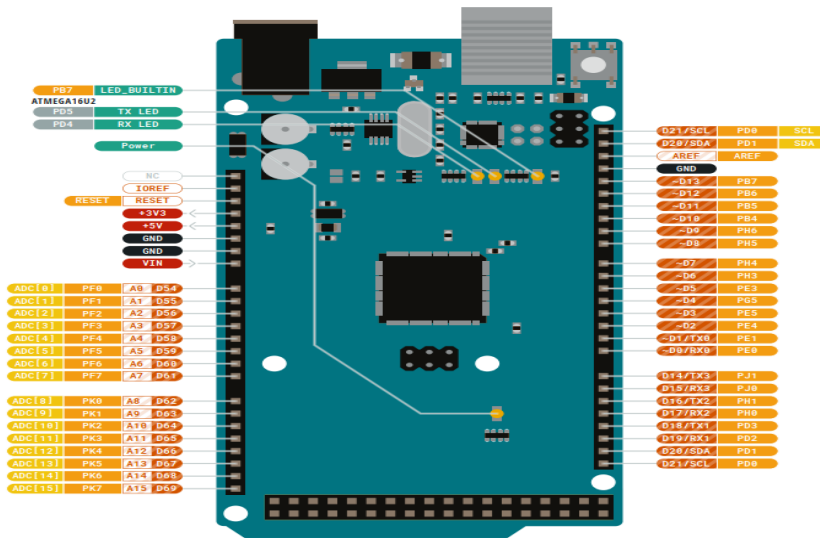


Hình 2.4: Sơ đồ linh kiện trên Arduino Mega 2560

Thông số kỹ thuật Board Arduino Mega 2560:

Bảng 2.1: Thông số kỹ thuật Arduino Mega 2560 [11]

Vi điều khiển	ATmega2560
Điện áp hoạt động	5V
Điện áp đầu vào(được đề nghị)	7-12V
Điện áp đầu vào (giới hạn)	6-20V
Số lượng chân I / O	54(trong đó có 15 cung cấp sản lượng PWM)
Số lượng chân Input Analog	16
Dòng điện DC mỗi I / O	20mA
Dòng điện DC với chân 3.3V	50mA
Dòng điện DC với chân 5V	500mA
Bộ nhớ flash	256 KB trong đó có 8 KB sử dụng bởi bộ nạp khởi động
SRAM	8KB
EEPROM	4 KB



Hình 2.5: Sơ đồ chân Board Mega 2560

Chức năng từng chân:

- **GND (Ground):** cực âm của nguồn điện cấp cho Arduino Mega 2560. Khi dùng các thiết bị sử dụng những nguồn điện riêng biệt thì những chân này phải được nối với nhau.
- **5V:** cấp điện áp 5V đầu ra. Dòng tối đa cho phép ở chân này là 500mA.
- **3.3V:** cấp điện áp 3.3V đầu ra. Dòng tối đa cho phép ở chân này là 50mA.
- **Vin (Voltage Input):** để cấp nguồn ngoài cho Arduino Mega 2560, bạn nối cực dương của nguồn với chân này và cực âm của nguồn với chân GND.
- **IOREF:** điện áp hoạt động của vi điều khiển trên Arduino Mega 2560 có thể được đo ở chân này. Và dĩ nhiên nó luôn là 5V. Mặc dù vậy không được lấy nguồn 5V từ chân này để sử dụng bởi chức năng của nó không phải là cấp nguồn.
- **RESET:** việc nhấn nút Reset trên board để reset vi điều khiển tương đương với việc chân RESET được nối với GND qua 1 điện trở 10KΩ.
- **Các cổng vào/ra:** Arduino Mega 2560 có 54 chân digital dùng để đọc hoặc xuất tín hiệu. Chúng chỉ có 2 mức điện áp là 0V và 5V với dòng vào/ra tối đa trên mỗi chân là 40mA. Ở mỗi chân đều có các điện trở pull-up từ được cài đặt ngay trong vi điều khiển Atmega2560(mặc định thì các điện trở này không được kết nối). Một số chân digital có các chức năng đặc biệt như sau:
- **4 Cổng Serial:**

Bảng 2.2: Các cổng serial của Arduino Mega 2560

Cổng Serial	Chân RX	Chân TX
Cổng 0	0	1
Cổng 1	19	18
Cổng 2	17	16
Cổng 3	15	14

- **RX và TX:** dùng để gửi (transmit – TX) và nhận (receive – RX) dữ liệu TTL Serial. Arduino Mega 2560 có thể giao tiếp với thiết bị khác thông qua 2 chân này. Kết nối bluetooth thường thấy nói nôm na chính là kết nối Serial không dây. Nếu không cần giao tiếp Serial, không nên sử dụng 4 chân này nếu không cần thiết.
- **Chân PWM (~):** 15 từ chân 2 - 13 và chân 44 45 46: cho phép bạn xuất ra xung PWM với độ phân giải 8bit. (giá trị từ 0 - 255 tương ứng với 0V - 5V) bằng hàm analogWrite(). Nói một cách đơn giản, có thể điều chỉnh được điện áp ra ở chân này từ mức 0V đến 5V thay vì chỉ cố định ở mức 0V và 5V như những chân khác.
- **Chân giao tiếp SPI:** 22 (SS), 23 (SCK), 24(MOSI), 25(MISO) Ngoài các chức năng thông thường, 4 chân này còn dùng để truyền phát dữ liệu bằng giao thức SPI với các thiết bị khác.
- **LED 13:** trên Arduino Mega 2560 có 1 đèn led màu cam (kí hiệu chữ L). Khi bấm nút Reset, sẽ thấy đèn này nhấp nháy để báo hiệu. Nó được nối với chân số 13. Khi chân này được người dùng sử dụng, LED sẽ sáng.
- **Arduino Mega 2560:** có 16 chân analog (A0 - A15) cung cấp độ phân giải tín hiệu 10bit (0 - 1023) để đọc giá trị điện áp trong khoảng 0V - 5V. Với chân AREF trên board, có thể để đưa vào điện áp tham chiếu khi sử dụng các chân analog. Tức là nếu cấp điện áp 2.5V vào chân này thì có thể dùng các chân analog để đo điện áp trong khoảng từ 0V - 2.5V với độ phân giải vẫn là 10bit.

2.3.2. Thiết kế khối động cơ

a. Động cơ DC giảm tốc

Trên thị trường hiện nay có rất nhiều loại động cơ giảm tốc 12V DC như: JGA25-370 DC, JGB37-520 DC, SK11S Plastic Geared TT,... Với yêu cầu đề tài là động cơ chạy ở mức trung bình và cần lực kéo ổn định nên động cơ giảm tốc JGB37-520 DC

200RPM là lựa chọn phù hợp, với giá thành tương đối rẻ, chất lượng tốt, lõi đồng và chổi than to, hộp giảm tốc chắc chắn bằng kim loại, cho lực kéo mạnh, moment lớn, thích hợp ứng dụng cho đề tài.



Hình 2.6: Động cơ giảm tốc JGB37-520 200RPM

Thông số kỹ thuật:

- Điện áp làm việc: 6-15VDC
- Điện áp định mức: 12VDC
- Dòng điện không tải: 0.1A
- Tốc độ: 200 RPM
- Lực kéo moment: 0.45kg.cm
- Công suất: 7-15W
- Trục 6mm khuyết D

b. Động cơ Servo

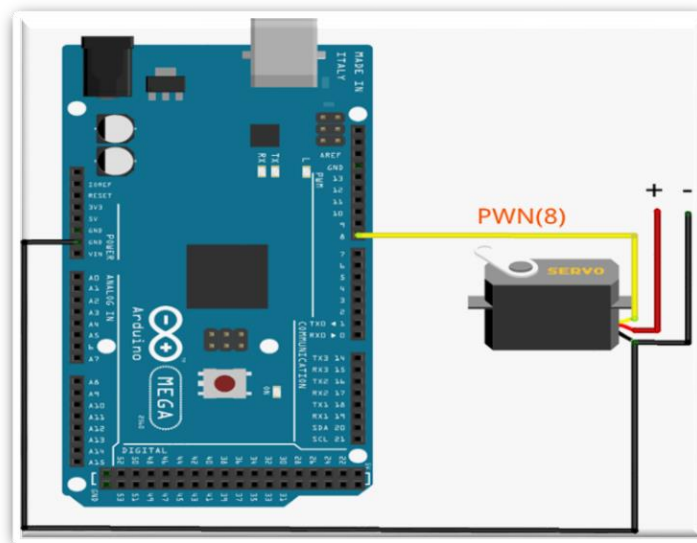
Với yêu cầu đề tài là đóng mở nắp thùng không yêu cầu cao về lực kéo, nên động cơ Động cơ RC Servo MG996 hoàn toàn đáp ứng yêu cầu đề ra so với các loại khác trên thị trường hiện nay như: Động cơ servo MG90S bánh răng full kim loại, Động cơ servo 360 độ, Servo TD8120MG,.. Trong tất cả, động cơ RC Servo MG996 là loại thường được sử dụng nhiều nhất trong các thiết kế Robot hoặc dẫn hướng xe, MG996 giá thành rẻ, có lực kéo mạnh, các khớp và bánh răng được làm hoàn toàn bằng kim loại nên có độ bền cao, động cơ được tích hợp sẵn driver điều khiển động cơ bên trong theo cơ chế phát xung - quay góc nên rất dễ sử dụng, phù hợp với yêu cầu đề tài.



Hình 2.7: Động cơ Servo MG996

Thông số kỹ thuật:

- **Chủng loại:** Analog RC Servo.
- **Điện áp hoạt động:** 4.8~6.6VDC
- **Lực kéo:**
 - + 3.5 kg-cm (180.5 ozin) at 4.8V-1.5A
 - + 5.5 kg-cm (208.3 ozin) at 6V-1.5A
- **Tốc độ quay:**
 - + 0.17sec / 60 degrees (4.8V no load)
 - + 0.13sec / 60 degrees (6.0V no load)
- **Kích thước:** 40mm x 20mm x 43mm
- **Trọng lượng:** 55g



Hình 2.8: Sơ đồ kết nối Servo với Arduino Mega 2560

c. Động cơ DC bơm nước

Hiện nay có rất nhiều loại bơm 12V DC trên thị trường ứng với nhiều mục đích sử dụng khác nhau và giá thành khác nhau, có thể kể đến như: Bơm ZUKUI 5002 (7.0 Bar), Bơm nhu động Kamoer Peristaltic Pump NKP-DCL-S10B 12VDC, Bơm 365 12V DC,... Với yêu cầu đề tài là bơm dung dịch sát khuẩn tay thì động cơ bơm nước 385 12V DC là lựa chọn hợp lý. Với kích thước nhỏ gọn áp lực mạnh, giá rẻ, đáp ứng yêu cầu của đề tài.



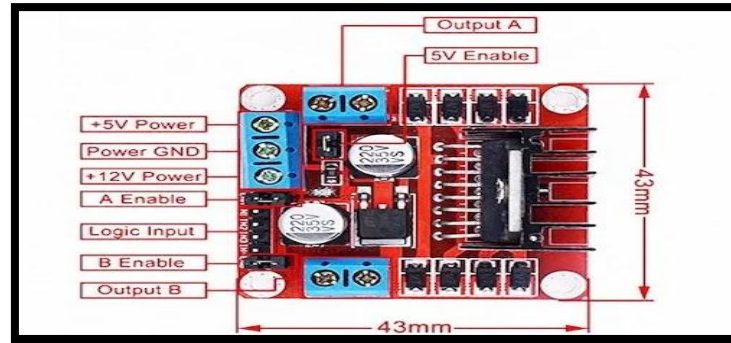
Hình 2.9: Động cơ bơm nước 385 12VDC

Thông số kỹ thuật:

- Điện áp: 12V DC
- Công suất: 3W
- Dòng tiêu thụ: 200mA
- Lưu lượng nước bơm: 1,8L/phút
- Áp suất nước 0.3Mpa

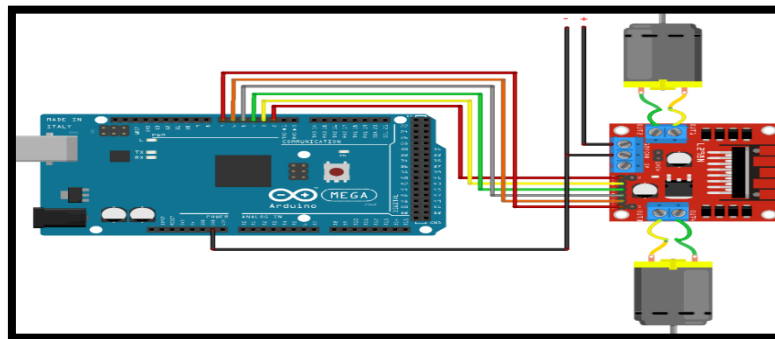
2.3.3. Thiết kế khối điều khiển động cơ

Với yêu cầu của đề tài là điều khiển nhiều động cơ DC 12V và không cần thiết đảo chiều quay (để làm cho xe chạy lùi) thì ở đây có thể lựa chọn giải pháp là sử dụng các module mạch cầu H để điều khiển. Trên thị trường hiện nay thì có rất nhiều các mạch điều khiển động cơ DC với nhiều mẫu mã như: Easydriver - điều khiển động cơ bước A3967, module điều khiển động cơ cầu H DC 30A VNH2SP30,... Tuy nhiên để có thể giao tiếp dễ dàng với board Arduino Mega 2560 và điều khiển đồng thời 2 động cơ DC theo từng cặp theo yêu cầu ở đề tài này thì việc lựa chọn sử dụng mạch điều khiển động cơ L298N là một giải pháp thích hợp. Module điều khiển động cơ L298N còn có độ thông dụng cao, dễ lắp đặt, dễ mua trên thị trường với giá thành rẻ.



Hình 2.10: Hình ảnh và sơ đồ chân module L298N

Sơ đồ kết nối Arduino:

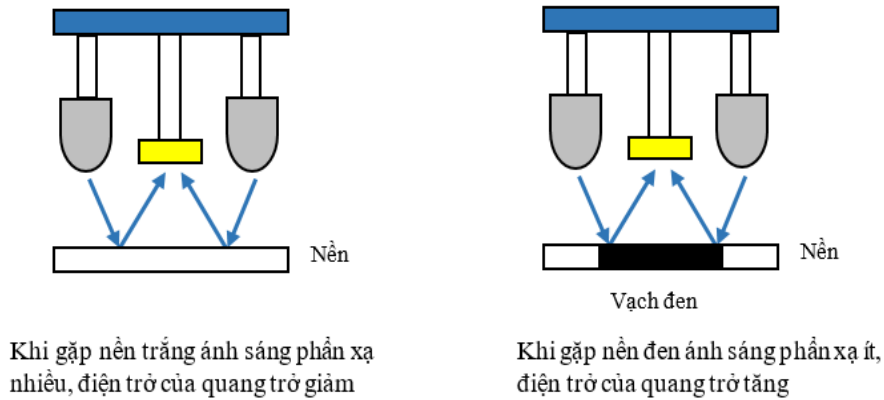


Hình 2.11: Kết nối Arduino với LN298N điều khiển động cơ

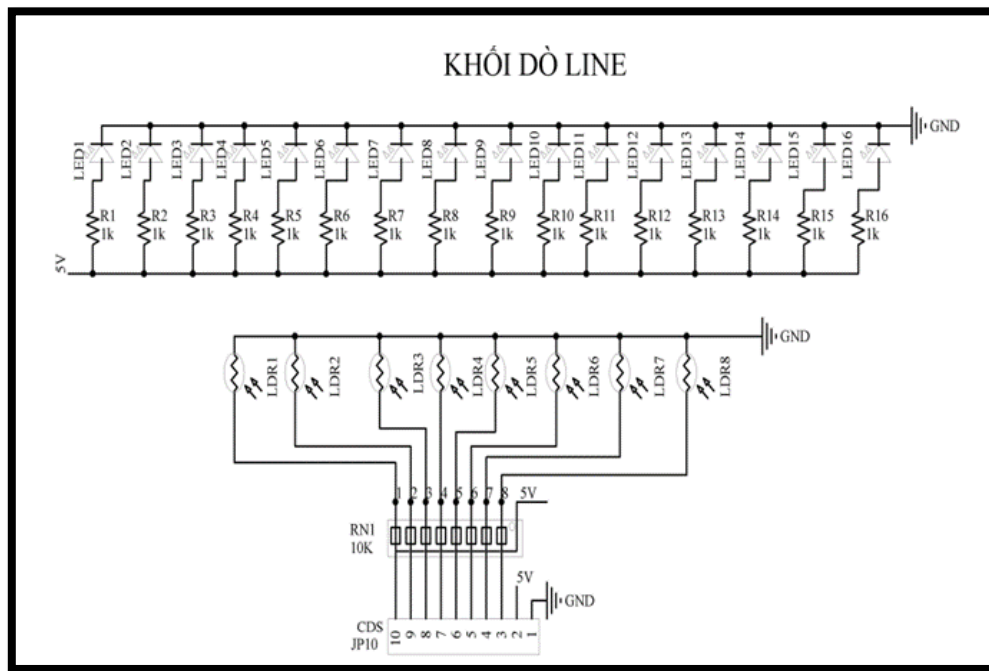
2.3.4. Thiết kế khối dò đường:

Module dò đường quang trở:

Hiện nay trên thị trường có rất nhiều cảm biến và module dò đường khác nhau với nhiều mẫu mã và phổ biến nhất là ứng dụng thu phát hồng ngoại như: TCR5000, XD-201,... nhược điểm của các cảm biến, module này tốc độ phản hồi kém, do nhiễu từ môi trường bên ngoài. Để giải quyết vấn đề trên nhóm em đã tìm hiểu và thiết kế một module dò đường gồm 9 cảm biến quang trở và 13 cặp led trợ sáng để dàng tinh chỉnh lập trình. Với module này ta chỉ lấy tín hiệu cường độ của quang trở thông qua bộ ADC của Arduino để đưa vào khối xử lý trung tâm xử lý. Dưới đây là nguyên lý thu và nhận tín hiệu quang trở.



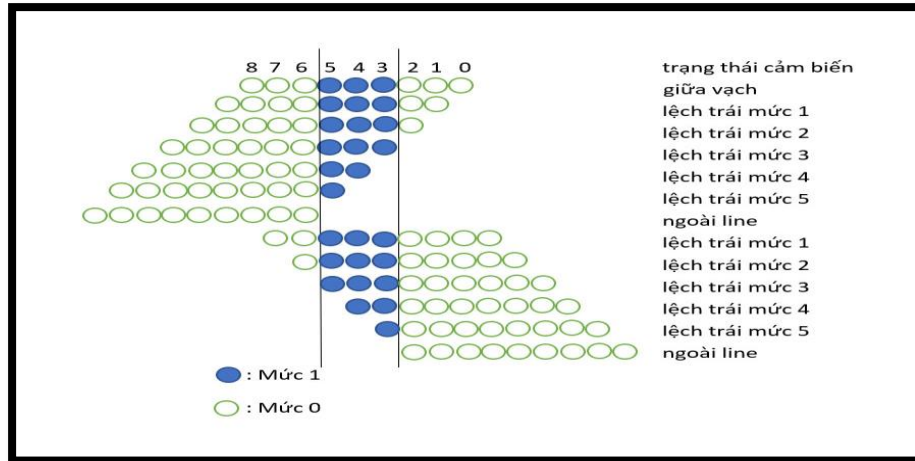
Hình 2.12: Nguyên lý hoạt động của cảm biến quang điện trở



Hình 2.13: Sơ đồ nguyên lý khối dò đường

Giải thuật:

Khi xe hoạt động, sensor (cảm biến quang trở) thu thập thông tin qua bộ chuyển đổi ADC chuyển thành tín hiệu điện đưa về vi điều khiển. Giá trị nhận về sẽ được so sánh với một giá trị trước (ngưỡng). Nếu giá trị $ADC[i] < \text{giá trị ngưỡng}$ thì led thứ i đang ở phần sẫm, ngược lại led thứ i đang ở phần vạch. Dựa trên 8 giá trị thu về, xác định độ lệch tương đối của giữa quỹ đạo của robot và quỹ đạo mong muốn, sau đó so sánh độ lệch thành các mức.



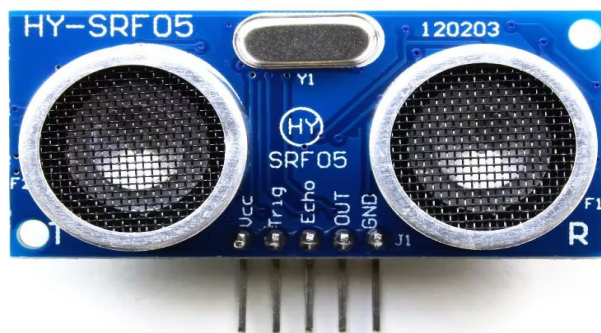
Hình 2.14: Các mức lệch khỏi quỹ đạo của robot theo cảm biến dò đường

Dựa trên các mức lệch, điều chỉnh tốc độ 2 bánh trái – phải để điều khiển robot theo mong muốn. Cụ thể, để rẽ trái tốc độ bánh xe bên phải robot phải nhanh hơn bên trái một giá trị tương ứng với các mức lệch (cần phải thử nghiệm nhiều lần), và ngược lại. Phương pháp này đơn giản nhưng robot chạy không ổn định, lúc rất nhanh lúc rất chậm, tính ổn định phụ thuộc vào động cơ và kết cấu robot cơ khí.

2.3.5. Thiết kế khối cảm biến khoảng cách

a. Giới thiệu

Khi tạo ra một robot tự hành, một trong những yếu tố chủ yếu là phải phát hiện và tránh được vật cản. Một cảm biến có thể phát hiện đối tượng, để robot tránh và di chuyển về hướng định sẵn. Để được như vậy, cảm biến phải nhỏ, tổn hao ít và dễ dàng chế tạo và phải sử dụng nhiều ứng dụng. một cảm biến có độ tin cậy cho những yêu cầu trên là cảm biến siêu âm. Có rất nhiều loại cảm biến siêu âm trên thị trường như: cảm biến HC-SR04, cảm biến siêu âm US-015, cảm biến vật cản NPN E18-D80NK,...



Hình 2.15: Module cảm biến siêu âm HC-SR05

Cảm biến siêu âm HC-SR05 sử dụng rất phổ biến để xác định khoảng cách. HC-SR05 sử dụng sóng siêu âm và có thể đo khoảng cách trong khoảng từ 2 đến 300 cm, với độ chính xác gần như chỉ phụ thuộc vào cách lập trình. Nhóm quyết định chọn cảm biến này dùng trong đề tài.

Nguyên lý làm việc:

- Tín hiệu mức cao được gửi trong vòng 10us sử dụng chân Trigger
- Module gửi tín hiệu tự động 40kHz. Và sau đó dò xung nhận hay không
- Nếu tín hiệu được nhận, sau đó có dạng mức cao. Thời gian ở mức cao là khe giữa việc truyền và nhận tín hiệu

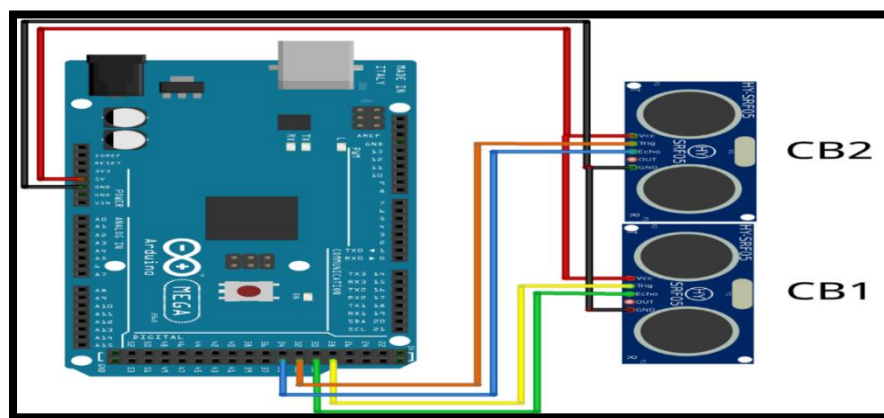
b. Hoạt động của HC-SR05



Hình 2.16: *Hoạt động của cảm biến siêu âm*

Để đo khoảng cách, ta sẽ phát 1 xung rất ngắn (5 microSeconds) từ chân Trigger. Sau đó, cảm biến sẽ tạo ra xung HIGH ở chân Echo cho đến khi nhận lại được sóng phản xạ. Chiều rộng của xung sẽ bằng với thời gian sóng siêu âm được phát ra từ cảm biến và quay trở lại. Tốc độ âm thanh trong không khí là 340m/s (hằng số vật lý), tương đương 29,412 microSeconds/cm ($10^6/(340*100)$). Khi đã tính được thời gian, ta sẽ chia cho 29,412 để nhận được khoảng cách.

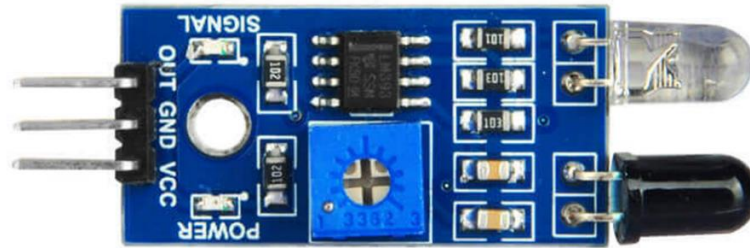
c. Sơ đồ kết nối



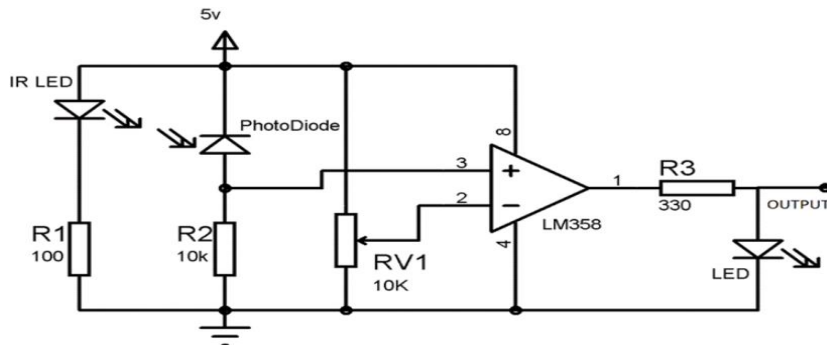
Hình 2.17: *Sơ đồ kết nối khối cảm biến khoảng cách với Arduino*

2.3.6. Thiết kế khối cảm biến tiệm cận

Hiện nay trên thị trường có rất nhiều cảm biến tiệm cận khác nhau như cảm biến tiệm cận điện cảm (Proximity), Cảm biến tiệm cận điện dung, ... Với yêu cầu đề tài là 1 cảm biến tiệm cận cung khả năng phát hiện bàn tay cho khối khử khuẩn hoạt động thì module cảm biến hồng ngoại LM385 là lựa chọn hợp lý. Với giá thành rẻ và cảm biến này giúp phát hiện vật bằng hồng ngoại thích nghi với điều kiện ánh sáng của môi trường xung quanh.



Hình 2.18: Module cảm biến hồng ngoại LM385



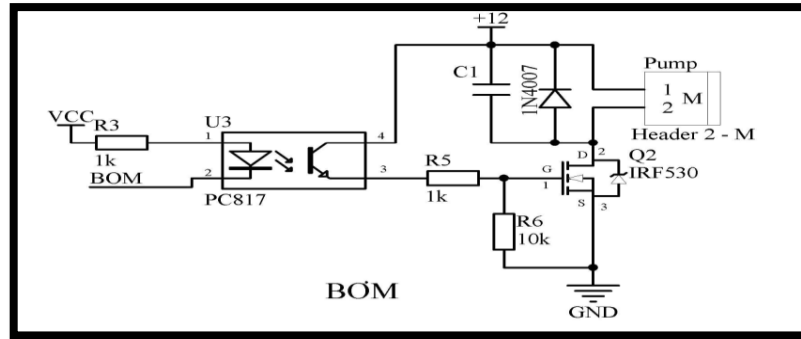
Hình 2.19: Sơ đồ nguyên lý của cảm biến hồng ngoại

Thông số kỹ thuật:

- Bộ so sánh sử dụng LM393, làm việc ổn định
- Điện áp làm việc: 3.3V – 5V DC.
- Khi bật nguồn, đèn báo nguồn màu đỏ sáng.
- Lỗ vít 3 mm, dễ dàng cố định, lắp đặt.
- Kích thước: 3.2cm * 1.4cm.

2.3.7. Thiết kế khối khử khuẩn

Với yêu cầu đề tài là thùng rác sẽ bơm khử khuẩn khi ta chạm tay vào khối cảm biến tiệm cận đầu vào. Vì vậy khối khử khuẩn ra đời.



Hình 2.20: Sơ đồ nguyên lý khối khởi khuẩn

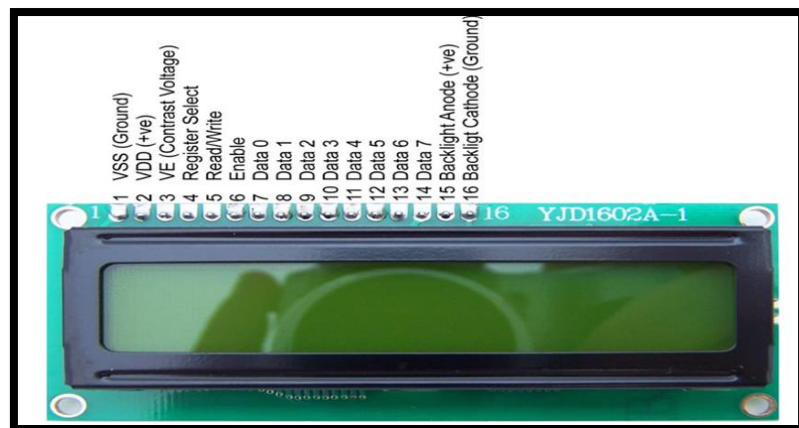
Nguyên lý hoạt động:

Khi có tín hiệu vật cản từ khối cảm biến tiệm cận sẽ xuất tín hiệu vào khối xử lý trung tâm Arduino Mega 2560, khối xử lý trung tâm sẽ kích cho PC817 dẫn dòng qua mosfet IRF530 hoạt động làm cho bơm bật.

2.3.8. Thiết kế khối hiển thị và âm thanh

a. Khối hiển thị

Với yêu cầu đề tài là hiển thị lưu lượng phần trăm rác trong thùng và cảnh báo rác đầy. Vì vậy module hiển thị LCD 1602 là lựa chọn tốt nhất để ứng dụng với khả năng hiển thị kí tự đa dạng (chữ, số, kí tự), dễ dàng đưa vào mạch ứng dụng theo nhiều giao thức giao tiếp khác nhau, tiêu tốn rất ít tài nguyên hệ thống, giá thành rẻ,...



Hình 2.21: Module LCD 1602

Thông số kỹ thuật:

- Điện áp hoạt động: 5VDC
- LCD 16×2 có 16 chân trong đó 8 chân dữ liệu (D0 – D7) và 3 chân điều khiển (RS, RW, EN).

- 5 chân còn lại dùng để cấp nguồn và đèn nền cho LCD 16×2.
- Các chân điều khiển giúp ta dễ dàng cấu hình LCD ở chế độ lệnh hoặc chế độ dữ liệu.

Chức năng của từng chân LCD 1602:

- Chân số 1-VSS : chân nối đất cho LCD được nối với GND của mạch điều khiển
- Chân số 2 - VDD : chân cấp nguồn cho LCD, được nối với VCC=5V của mạch điều khiển
- Chân số 3 - VE : điều chỉnh độ tương phản của LCD
- Chân số 4 - RS : chân chọn thanh ghi, được nối với logic "0" hoặc logic "1":
 - + Logic "0": Bus DB0 - DB7 sẽ nối với thanh ghi lệnh IR của LCD (ở chế độ "ghi" - write) hoặc nối với bộ đếm địa chỉ của LCD (ở chế độ "đọc" -read)
 - + Logic "1": Bus DB0 - DB7 sẽ nối với thanh ghi dữ liệu DR bên trong LCD
- Chân số 5 - R/W : chân chọn chế độ đọc/ghi (Read/Write), được nối với logic "0" để ghi hoặc nối với logic "1" đọc
- Chân số 6 - E : chân cho phép (Enable). Sau khi các tín hiệu được đặt lên bus DB0-DB7, các lệnh chỉ được chấp nhận khi có 1 xung cho phép của chân này như sau:
 - + Ở chế độ ghi: Dữ liệu ở bus sẽ được LCD chuyển vào thanh ghi bên trong khi phát hiện một xung (high-to-low transition) của tín hiệu chân E
 - + Ở chế độ đọc: Dữ liệu sẽ được LCD xuất ra DB0-DB7 khi phát hiện cạnh lên (low-to-high transition) ở chân E và được LCD giữ ở bus đến khi nào chân E xuống mức thấp.

b. Khối âm thanh

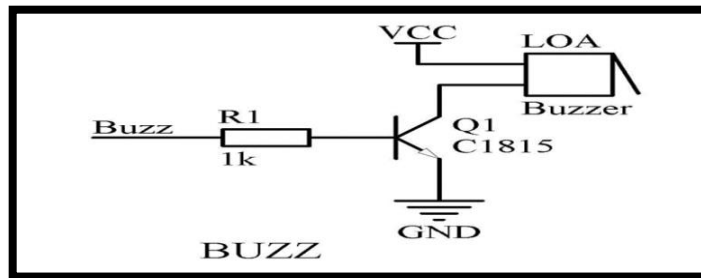
Với yêu cầu đề tài là phát ra âm thanh cho mô hình khi đầy rác và gặp vật cản cho người đi lại biết mà chủ động né tránh. Ta chọn còi chip Buzzer 12V DC vì có hiệu suất ổn định, âm thanh chất lượng tốt, được sản xuất nhỏ gọn, rất phù hợp yêu cầu thiết kế của đề tài.



Hình 2.22: Buzzer 12V DC

Thông số kỹ thuật:

- Nguồn: 5V - 12V
- Dòng điện tiêu thụ: <25mA
- Tần số cộng hưởng: 2300Hz ± 500Hz
- Biên độ âm thanh: >80 dB
- Nhiệt độ hoạt động: -20 °C đến +70 °C



Hình 2.23: Sơ đồ nguyên lý khối âm thanh kết nối với Arduino

2.3.9. Thiết kế khối nguồn

a. Pin, mạch sạc cân bằng pin

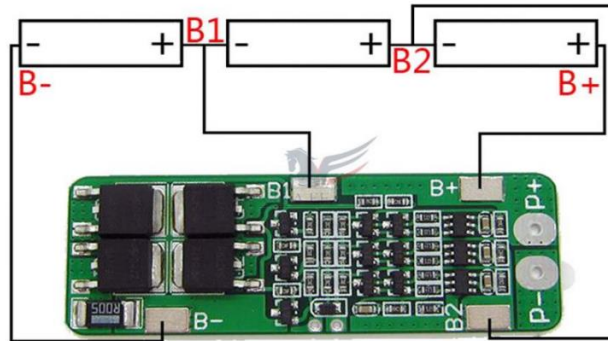
Pin 18650 là loại pin sạc sử dụng công nghệ Li-on, viên pin có điện áp là 3.7V khi chưa sạc và đạt 4.2V khi sạc đầy, ngoài ra khi pin yếu có thể đạt dưới 3V. Tuổi thọ Pin có thể đạt tới 500 lần xả và sạc đầy. Trong đề tài sử dụng 3 pin 18650 có dung lượng 2200mAh mắc nối tiếp với nhau tạo thành mạch pin 3S tạo ra nguồn 12V cung cấp cho toàn hệ thống.



Hình 2.24: Pin 18650

b. Mạch sạc cân bằng pin

Mạch sạc và cân bằng Pin có điện áp 12.6V 3S 20A dùng cho các loại pin ghép lithium bảo vệ pin quá dòng, bảo vệ pin khi sạc giúp ổn định pin tránh hư khi dùng hoặc sạc.



Hình 2.25: Sơ đồ kết nối mạch sạc pin

c. Module giảm áp DC-DC Buck LM2596 3A

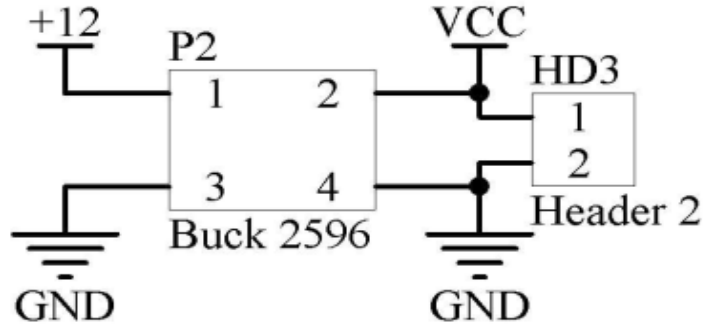


Hình 2.26: Module Buck DC-DC LM2596 3A

Bộ điều chỉnh giảm điện áp hay gọi là Buck converter hay Step-down conveter, là bộ chuyển đổi nguồn điện DC sang DC giúp giảm điện áp đầu vào 12V từ mạch pin 3s đến điện áp đầu ra 5V cho Arduino Mega 2560 hoạt động ổn định và lâu dài.

Thông số kỹ thuật:

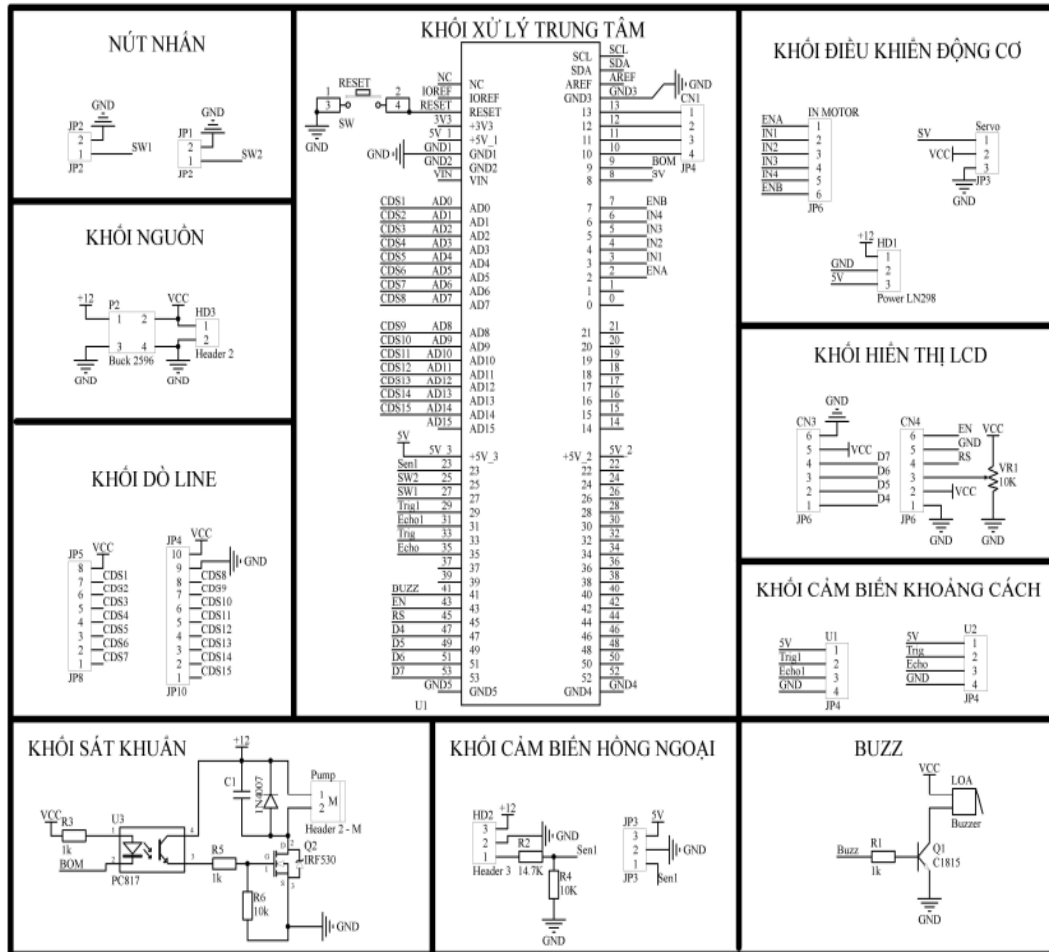
- Điện áp đầu vào: 4V-35V
- Điện áp đầu ra: 1.23V-30V
- Dòng đầu ra: 3A (max)
- Hiệu suất chuyển đổi: 92% (tối đa)
- Tần số hoạt động module hạ áp: 150kHz
- Nhiệt độ hoạt động: -40 °C đến + 85 °C
- Kích thước: 4.3x2.1x1.7 cm



Hình 2.27: Sơ đồ kết nối module giảm áp với Arduino

2.3.10. Sơ đồ nguyên lý toàn mạch

Dưới đây là toàn bộ kết nối giữa các khối với nhau, bao gồm các cảm biến, nút nhấn, các chân kết nối với board xử lý Arduino Mega 2560.



Hình 2.28: Sơ đồ nguyên lý toàn mạch

Chương 3: THI CÔNG HỆ THỐNG

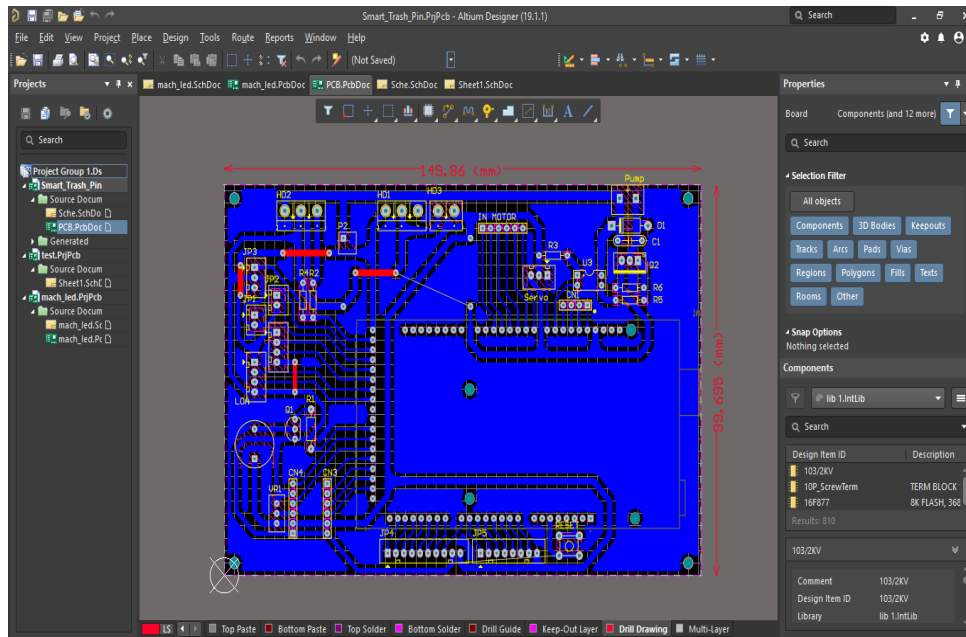
3.1. Giới thiệu:

Trong chương này là quá trình thi công mô hình thùng rác, làm mạch in, lập trình, lắp ráp phần cứng và test các module. Bên cạnh đó là hình vẽ được chụp từ mô hình thực của thùng rác bên ngoài, hình chụp các kết quả chạy mà mô hình có thể có tính tới thời điểm hiện tại

3.2. Mạch in:

3.2.1. Giới thiệu phần mềm Altium Designer

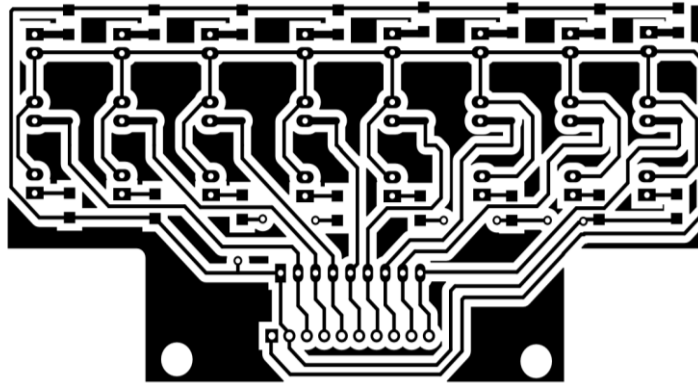
Altium design có lẽ không còn gì xa lạ với sinh viên kỹ thuật chúng ta. Với cộng đồng sử dụng đông đảo trên khắp thế giới. Phần mềm Altium Designer thiết kế PCB của Altium đã giúp một loạt các công ty thành công trong việc thiết kế các sản phẩm và hệ thống điện tử thế hệ tiếp theo. Nó bao gồm tất cả những công cụ cần thiết cho một bản thiết kế điện tử hoàn thiện, như công cụ thiết kế bản vẽ nguyên lý, bản vẽ mạch in, mô phỏng mạch điện, phân tích tín hiệu, môi trường lập trình VHDL, môi trường thiết kế và phát triển hệ thống nhúng FPGA, ... Đặc biệt Altium Design mạnh về khả năng mô phỏng 3D giúp người thiết kế có thể nhìn mạch của mình một cách trực quan và chính xác nhất.



Hình 3.1: *Giao diện thiết kế Altium Design*

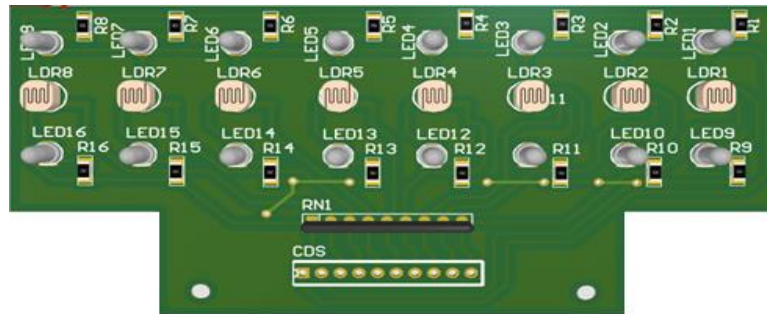
3.2.2. Mạch in khối dò đường:

Với yêu cầu thiết kế một module led quang trở để đọc vị trí trên Đường và gửi giá trị về Board để xử lý. Vì thế nhóm đã thuyết kế sơ đồ nguyên lý và mạch in bằng Altium design.



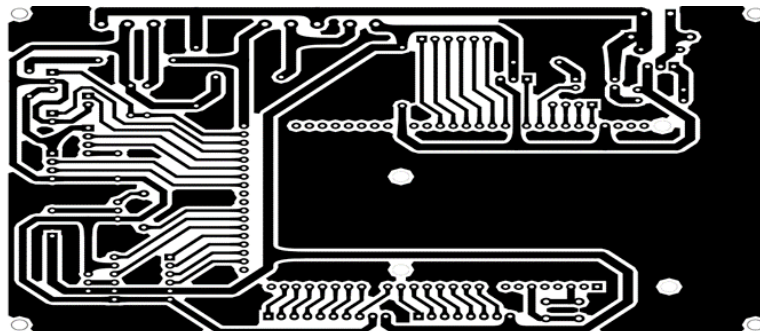
Hình 3.2: Mạch in khối dò đường

Mạch in được thiết kế theo sơ đồ nguyên lý và các linh kiện được bố trí và sắp xếp như hình 3.3



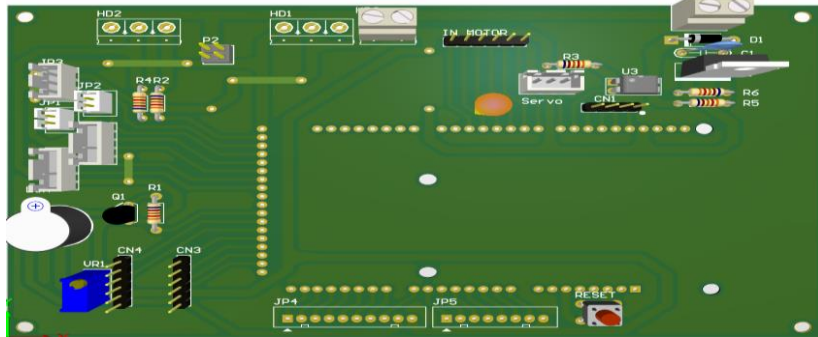
Hình 3.3: Mạch 3D khối dò đường

3.2.3. Mạch in khối xử lý trung tâm:



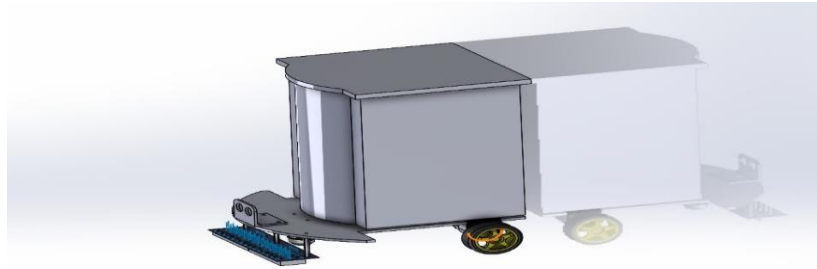
Hình 3.4: Mạch in tổng xử lý hệ thống

Tương tự như mạch in khối dò đường, mạch được thiết kế theo sơ đồ nguyên lý và các linh kiện được bố trí và sắp xếp như hình 3.5



Hình 3.5: Mạch 3D tổng xử lý hệ thống

3.3. Thi công mô hình:



Hình 3.6: Hình ảnh mô hình thùng rác mô phỏng

Chuẩn bị vật liệu: Thước, kéo, mica, khoan, ốc vít, 1 thùng rác có sẵn ở nhà để tiến hành thi công.

Ban đầu chúng ta đo đạc và tiến hành lắp ráp các chi tiết với nhau dựa vào yêu cầu của đề tài.

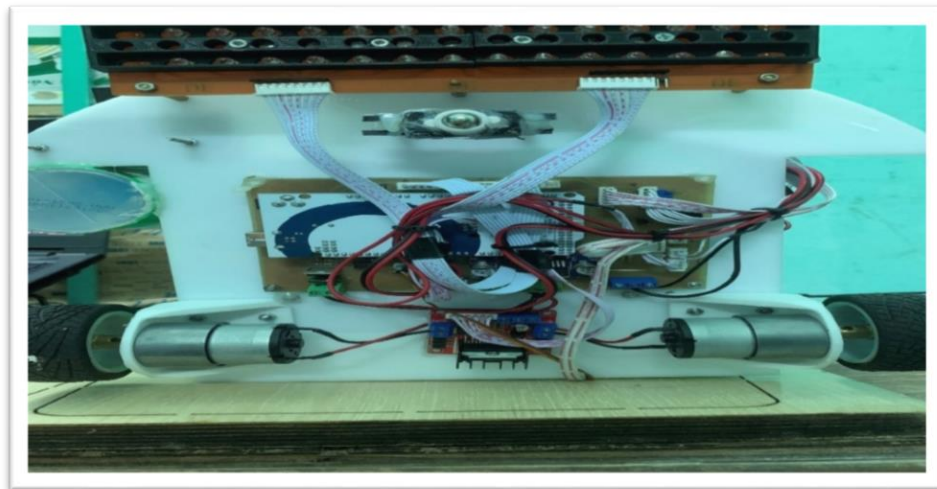


Hình 3.7: Hình ảnh nhóm lắp ráp mô hình



Hình 3.8: Hình ảnh thực tế thiết kế mô hình

Tiến hành lắp ráp các board xử lý Arduino Mega 2560, module dò đường và các cảm biến cho đề tài.

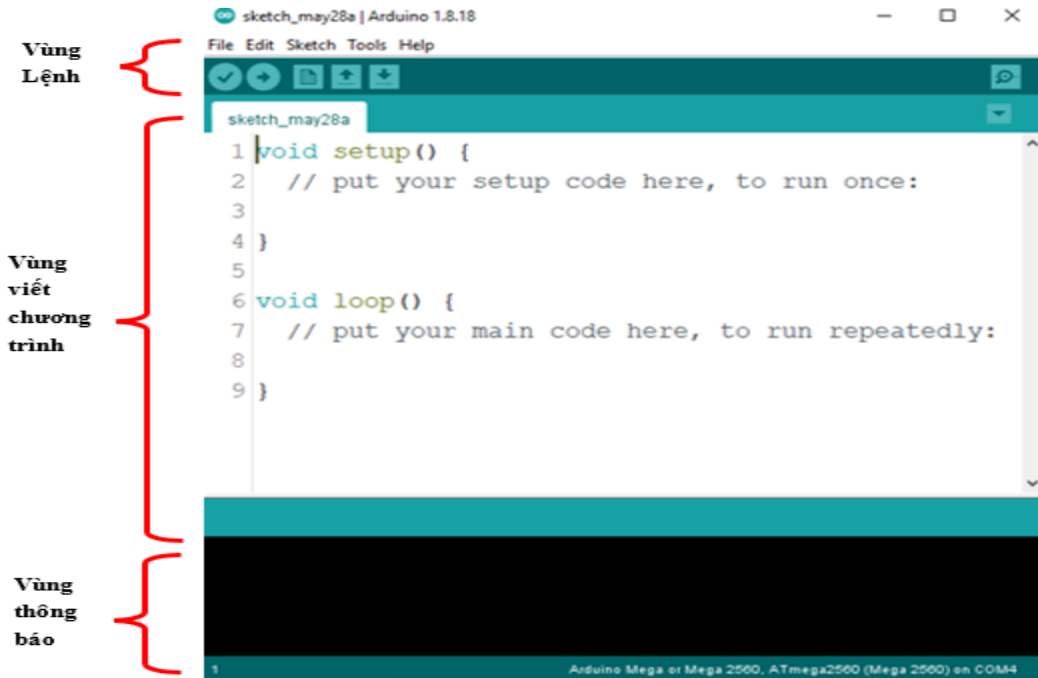


Hình 3.9: Lắp ráp các khối với nhau

3.4. Lập trình cho mô hình:

3.4.1. Giới thiệu phần mềm lập trình Arduino IDE

Arduino IDE không thể thiếu khi chúng ta lập trình Arduino. Là một phần mềm lập trình mã nguồn mở miễn phí, sử dụng Sử dụng ngôn ngữ lập trình C/C++ thân thiện với các lập trình viên và cộng đồng sử dụng đông đảo và chia sẻ thư viện phong phú. Nó bao gồm các phần chính là Editor (trình soạn thảo văn bản, dùng để viết code), Debugger, Compiler hoặc interpreter (công cụ giúp biên dịch code thành ngôn ngữ mà vi điều khiển có thể hiểu được và thực thi code theo yêu cầu người dùng).

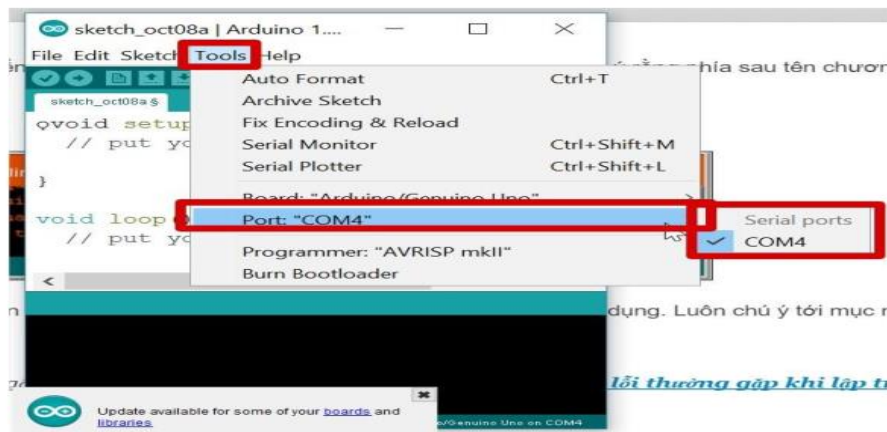


Hình 3.10: *Giao diện Arduino IDE*

Bảng 3.1: *Chức năng các nhóm lệnh Trong Arduino IDE*

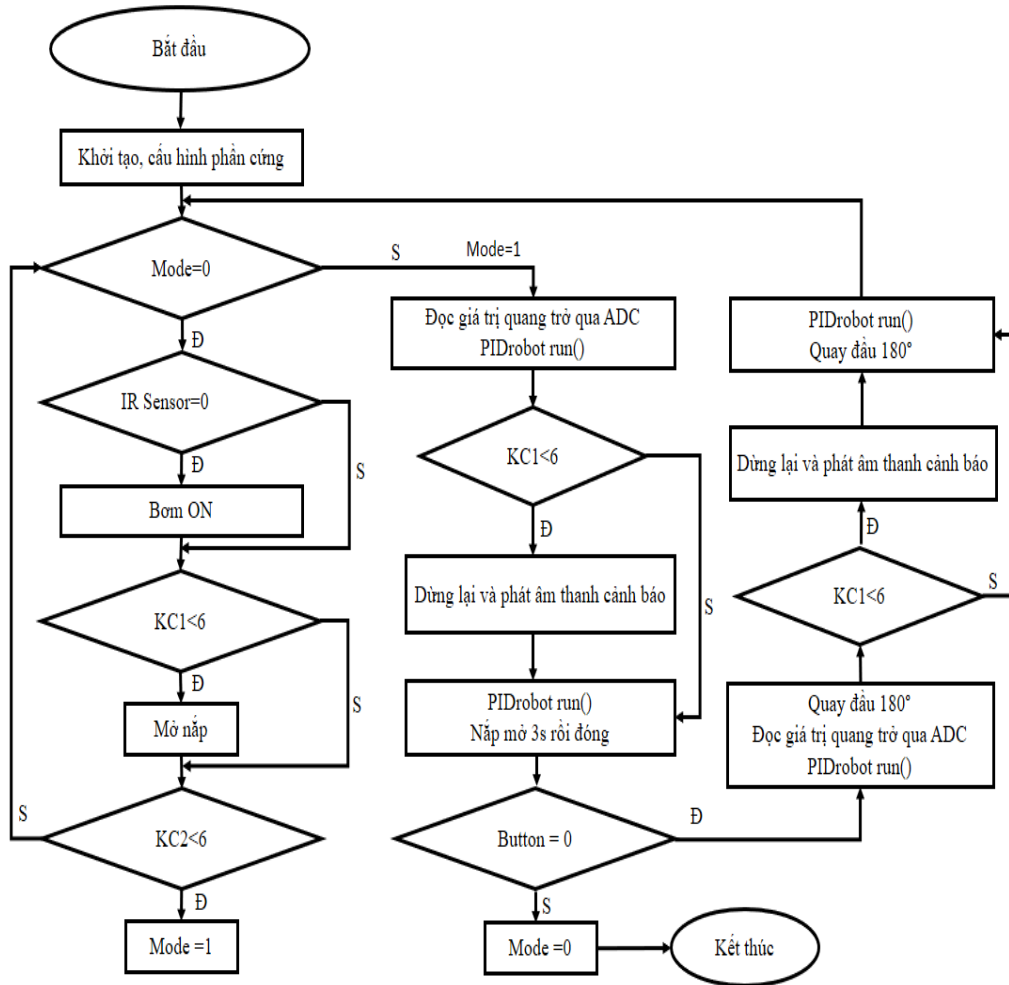
	Verify	Kiểm tra lỗi và biên dịch code
	Upload	Dịch và upload code vào bo mạch đã được cài đặt sẵn
	New	Tạo sketch mới
	Open	Mở một sketch có sẵn
	Save	Lưu sketch
	Serial Monitor	Mở serial monitor.

Lưu ý: Chúng ta phải chọn Port và Board mạch phù hợp mới nạp chương trình được.



Hình 3.11: *Chọn Port để kết nối*

3.4.2. Sơ đồ thuật toán



Hình 3.12: Lưu đồ thuật toán chương trình chính

3.4.3. Giải thích lưu đồ thuật toán

Khi cấp nguồn, thiết bị sẽ khởi tạo, cấu hình định nghĩa các chân, đọc các giá trị cảm biến, thiết lập tốc độ xung, tốc độ cho 2 động cơ khai báo các biến liên quan và trong chế độ chờ (Mode 0): trong chế độ chờ, các cảm biến hồng ngoại và siêu âm liên tục phát và thu tín hiệu để gửi về board xử lý thực hiện các chức năng như đóng mở cửa, bơm sát khuẩn.

Khi thùng rác nhận tín hiệu đầy, thùng rác sẽ chuyển sang chế độ di chuyển đến nơi thu gom (Mode 1). Bằng cách đọc giá trị quang trở thông qua bộ chuyển đổi ADC trên vi điều khiển, để xác định vị trí và tính toán PID điều khiển tốc độ động cơ để

thùng rác có thể bám đường và di chuyển đến nơi tập kết. Khi đến nơi tập kết thùng rác sẽ mở nắp trong 3 giây rồi đóng.

Nếu ta nhấn nút nhấn, thùng rác sẽ quay đầu 180° tiến về điểm bắt đầu tiếp tục quay 180° và trở về trạng thái chờ (Mode 0).

3.5. Hướng dẫn sử dụng:

Ban đầu, ta gạt nút công tắc để cung cấp nguồn, khởi động cho toàn bộ hệ thống, khi gạt công tắc, cảm biến dò đường sẽ phát sáng và phát tín hiệu âm thanh thông báo, sau khoảng 1 giây đến 5 giây, toàn bộ hệ thống đã sẵn sàng trạng thái hoạt động thông qua 3 tiếng âm thanh cảnh báo. Để robot chạy đúng yêu cầu, trước tiên ta phải cho robot học màu ở bên ngoài, tức là phần nền màu trắng trong khoảng 5 giây, sau đó ta sẽ đưa robot vào chế độ sẵn sàng, ở trạng thái chờ, thùng rác có cảm biến phát hiện vật cản để mở nắp tự động, phát hiện vật cản để bơm nước rửa tay sát khuẩn. khi thùng rác phát hiện rác trong thùng đã đầy, thùng rác xuất ra màn hình LCD tỉ lệ phần trăm rác trong thùng là hiện cảnh báo với người dùng, ngay sau đó, thùng rác tự động dò đường sau ba tiếng âm thanh cảnh báo, thùng rác chạy với tốc độ vừa phải để tránh rơi rải rác trên đường đi, đồng thời nếu có vật cản chắn ngang đường đi, thùng rác sẽ dừng lại ngay và phát ra âm thanh cảnh báo, khi đi hết đường đi kẻ sẵn, thùng rác tự động mở nắp và sau đó đóng lại. ta chỉ cần bấm nút thì thùng rác sẽ tự quay ngược 180° để trở về vị trí cũ và chờ ở đó.

Chương 4: KẾT QUẢ - ĐÁNH GIÁ

4.1. Giới thiệu:

Chương này trình bày kết quả của quá trình nghiên cứu làm đề tài trong thời gian vừa qua. Bao gồm nhận xét, đánh giá và hướng đề xuất phát triển của mô hình để hoàn thiện và đi vào thực tế sử dụng.

4.2. Kết quả yêu cầu đạt được:

Biết được chức năng, nguyên lí hoạt động của các loại cảm biến. Các cảm biến sử dụng trong đề tài này đa phần là cảm biến siêu âm, cảm biến tiệm cận. Tuy vậy, chỉ cần nhìn vào mức độ xuất hiện của cảm biến siêu âm, cảm biến tiệm cận cũng như tính đa năng của nó kết hợp được với nhiều yêu cầu đo khoảng cách khác nhau cũng đã chỉ cho chúng ta thấy được tính thông dụng của nó như thế nào trong cuộc sống hiện đại ngày nay. Quá trình nghiên cứu đề tài này đã giúp nhóm thực hiện nâng cao kỹ năng đọc, hiểu cảm biến, trau dồi thêm kiến thức về các loại cảm biến. Đặc biệt là, tăng khả năng vận dụng lý thuyết vào thực tế, có khả năng lựa chọn được loại cảm biến phù hợp với từng yêu cầu của thực tiễn của đề tài.

Biết cách sử dụng Arduino Mega 2560. Arduino Mega 2560 là một trong những board mạch thông minh được sử dụng rộng rãi. Tuy nhiên, để sử dụng thành thạo và đầy đủ các chức năng của nó thì không phải dễ. Và đó là kết quả cả quá trình nghiên cứu đề tài này mà nhóm thực hiện có được. Ngoài khả năng điều khiển các ngõ ra để điều khiển động cơ, tốc độ động cơ theo ý muốn; khả năng đọc giá trị của các loại cảm biến khác nhau. Trong đề tài này, việc ứng dụng Arduino Mega để truy xuất dữ liệu từ cảm biến cũng như dùng để điều khiển động cơ hay nhận dữ liệu từ module cảm biến dò đường phản hồi về đã cho thấy điều đó.

Hiểu về nguyên lý và cách hoạt động của module L298N. Mạch điều khiển động cơ DC L298 có khả năng điều khiển 2 động cơ DC, dòng tối đa 2A mỗi động cơ, mạch tích hợp bảo vệ và IC nguồn 7805 giúp cấp nguồn 5VDC cho các module khác (chỉ sử dụng 5V này nếu nguồn cấp <12VDC). Mạch điều khiển động cơ DC L298 dễ sử dụng, chi phí thấp, dễ lắp đặt, là sự lựa chọn tối ưu trong tầm giá.

Hiểu và vẽ mạch nguyên lý module cảm biến dò đường trên phần mềm Altium 19. Mạch cảm biến dò đường với yêu cầu phải đáp ứng đầy đủ được các chức năng

theo yêu cầu của đề tài như độ nhạy ổn định, hoạt động tốt, có thiết kế bộ phận cách ly tại mỗi mắt thu, mắt nhận để tránh trường hợp nhiễu ánh sáng từ bên ngoài, ngoài ra còn phải có tính thẩm mỹ cao, gọn nhẹ, đầy đủ linh kiện.

Lúc chạy thử nghiệm thực tế, mô hình có chạy sai số, chưa ổn định do nhiễu ánh sáng tự nhiên. Bằng sự cố gắng tìm tòi và tìm cách khắc phục từ nhiều nguồn tài liệu tham khảo, nhóm đã khắc phục một phần sai số và chống nhiễu cho mô hình, hiện tại chạy ổn định trên vạch đường. Đây cũng chính là module quan trọng nhất đề tài vì xác định được vị trí và cho xe chạy đúng vị trí đó trong môi trường ngoài trời thực tế không phải là chuyện đơn giản. Nhóm thực sự cảm thấy rất may mắn vì đây cũng là dịp và cơ hội để nghiên cứu thêm về hệ thống dò đường tự động, từ đó có hướng phát triển trong tương lai.

Tìm hiểu thuật toán PID, nguyên lý và ứng dụng PID vào robot dò đường, cụ thể là ứng dụng vào một số phần điều khiển trong đề tài.

4.3. Kết quả thực nghiệm:

- Cảm biến dò đường nhận biết được đường đường trên quỹ đạo chuyển động.
- Để điều khiển được mô hình thông minh này, chúng ta cần phải biết được những vị trí mà thùng rác đi qua như điểm bắt đầu, điểm tập kết.
- Vì mô hình chủ yếu luôn đặt sẵn tại một vị trí nhất định, nên vị trí và hướng di chuyển sẽ được ổn định, không bị xáo trộn hướng di chuyển phức tạp.
- Ta cho mô hình chạy thử nghiệm trên đoạn đường dài khoảng 2 mét, gồm điểm xuất phát và điểm kết thúc.
- Đây là vạch đường chạy thử nghiệm thực trên giấy A0 nền trắng, vạch đường màu đen.

4.4. Kết quả thu được thực tế:

4.4.1 Chạy thử mô hình thử nghiệm thực tế

- Chạy thử nghiệm lần 1: mô hình bám đường 90%
- Chạy thử nghiệm lần 2: mô hình bám đường 90%
- Chạy thử nghiệm lần 3: mô hình bám đường 90%
- Chạy thử nghiệm lần 4: mô hình bám đường 90%
- Chạy thử nghiệm lần 5: mô hình bám đường 90%

Sau nhiều lần tinh chỉnh hệ số PID, tốc độ động cơ, delay,..., mô hình dần ổn định tín hiệu hơn, kết quả như sau:

- Chạy thử nghiệm lần 6: mô hình bám đường 95%
- Chạy thử nghiệm lần 7: mô hình bám đường 95%
- Chạy thử nghiệm lần 8: mô hình bám đường 95%
- Chạy thử nghiệm lần 9: mô hình bám đường 95%
- Chạy thử nghiệm lần 10: mô hình bám đường 95%

=> Suy ra tỉ lệ bám đúng đường là: 93%

4.5. Nhận xét- Đánh giá:

4.5.1. Nhận xét

Sau gần 15 tuần nghiên cứu và thực hiện, đề tài cơ bản đáp ứng được yêu cầu thiết kế ban đầu. Dưới đây là một số nhận xét:

Ưu điểm:

- Mô hình có thể hoạt động trong môi trường thông thoáng, có bề mặt phản ánh sáng tốt.
- Mô hình tự nhận biết rác đầy thông qua cảm biến siêu âm và đồng thời báo chuông và hiển thị LCD tình trạng rác trong thùng.
- Phát hiện rác đầy, mô hình sẽ tự động di chuyển đến điểm tập kết rác và chờ ở đó.
- Khi xác nhận rác trong thùng đã hết, bằng cách bấm nút, mô hình sẽ tự động di chuyển theo vạch đường về vị trí ban đầu và sẵn sàng di chuyển đến điểm tập kết khi rác trong thùng lại đầy.
- Trên đường đi chuyển, nếu gặp vật cản, thùng rác sẽ dừng lại và phát thông báo ra buzzer để cảnh báo, nếu không có vật cản, mô hình sẽ đi tiếp theo lịch trình.
- Mô hình sẽ phát hiện người đến bỏ rác và sẽ mở nắp tự động.
- Mô hình có sát khuẩn tay tự động để tránh các mầm bệnh lây nhiễm phát sinh.

Nhược điểm:

- Bị nhiễu do ánh sáng mạnh từ bên ngoài dẫn đến nhiều sai số và mô hình chạy sai thông số.
- Vì mô hình được lập trình bằng thuật toán PID nên khó để trở về trạng thái cân bằng chuẩn trên đường sau khi dừng lại
- Cảm biến rửa tay tự động đôi khi cũng còn nhiễu do ánh sáng bên ngoài.

- Di chuyển với độ chính xác tương đối, vẫn còn có thể xảy ra tình trạng không bám đường.
- Chưa có sự hỗ trợ của nhiều module để mô hình có thể hoàn thiện, tự hành độc lập được tốt hơn.
- Chưa có thiết bị giám sát, dễ gây tình trạng bị mất hoặc thùng rác di chuyển lạc đường sai lối.
- Khó có thể dùng trong điều kiện thời tiết mưa bão, hoặc những nơi mặt đường quá gồ ghề, không phản xạ ánh sáng tốt.

4.5.2. Đánh giá

- Sau quá trình vận hành thử chạy mô hình, nhóm thực hiện đánh giá mô hình đã hoàn thành những yêu cầu ban đầu đã đề ra 93%.
- Tuy rằng khi chạy trong môi trường ngoài trời có ánh sáng mạnh hay ánh sáng mặt trời chiếu trực tiếp, thì mô hình vận hành cũng chưa thực sự tối ưu một trăm phần trăm, vẫn còn sai số dẫn đến lệch vạch đường khi di chuyển.
- Đối với mỗi môi trường ánh sáng, đường đường khác nhau, ta phải điều chỉnh lại thông số PID sao cho phù hợp, điều này cũng chưa thật sự tối ưu vì kiến thức còn hạn hẹp, cần nhiều thời gian hơn để tìm hiểu.
- Với mô hình này ta có thể áp dụng ở trường học, văn phòng, các nhà máy, xí nghiệp, khu dân cư thương mại, từ đó mang lại tính thẩm mỹ cao, sang trọng, giúp giảm thiểu sức lao động, giảm phí thuê nhân công.
- Với giá thành rẻ, cấu tạo không quá phức tạp, gọn nhẹ hơn so với các sản phẩm có trên thị trường, nhóm hi vọng sẽ đáp ứng được nhu cầu khách hàng.

KẾT LUẬN

Sau khoảng thời gian nghiên cứu và tìm hiểu, nhóm thực hiện đã hoàn thành báo cáo đồ án và thi công mô hình theo những yêu cầu đã đặt ra ban đầu. Trong quá trình thực hiện, nhóm đã thu được những kết quả nhất định.

- Thiết kế thành công mô hình hỗ trợ thu gom rác thải ứng dụng dò đường khá cân đối, gọn gàng tuy nhiên chưa thực sự thẩm mỹ cao, độ chính xác 90 – 95%.
- Thùng rác có khả năng nhận biết được rác đầy trong thùng. Hiện thị tình trạng rác trong thùng ra màn hình LCD.
- Khi phát hiện rác trong thùng đầy, thùng rác có thể tự di chuyển đến vị trí tập kết rác.
- Khi ta để tay vào khu vực sát khuẩn thì thùng rác sẽ nhận tín hiệu từ cảm biến tiệm cận và điều khiển bơm dung dịch vào tay để sát khuẩn.
- Tới vị trí định vị sẵn thì dừng lại, mở nắp sẵn, chờ nhấn nút thì có thể quay về vị trí cũ.
- Trong lúc di chuyển, nếu gặp vật cản phía trước, thùng rác dừng lại và phát ra tiếng kêu để mọi người né tránh.
- Có kiến thức vững về cách lập trình Arduino Mega, cách sử dụng, lập trình cho các module công suất, cảm biến siêu âm.

Tuy nhiên, đề tài còn một số điểm hạn chế như:

- Mô hình hiện tại sử dụng không phù hợp với môi trường sử dụng ngoài trời mà phù hợp sử dụng trong văn phòng, trong nhà hơn.
- Thỉnh thoảng gặp trục trặc về vấn đề cảm biến siêu âm không do được khoảng cách mặc dù cảm biến siêu âm hoạt động bình thường.
- Sai số vị trí vẫn còn tương đối lớn do module cảm biến dò đường bị nhiễu ánh sáng.
- Thời lượng pin sử dụng chưa được lâu.
- Khả năng bảo quản thùng rác khi ở điều kiện xấu như mưa bão sẽ là rào cản lớn khi thùng rác được sử dụng ở ngoài trời

Hướng phát triển:

- Sử dụng nguồn pin năng lượng mặt trời thay thế pin sạc hiện tại.
- Ứng dụng thuật toán xử lý ảnh thực để nhận biết vật cản và môi trường xung quanh được rõ ràng, chi tiết hơn.
- Kết hợp thêm các cảm biến khác để tối ưu cho mô hình khi hoạt động ngoài trời cũng như trong việc nhận dạng quỹ đạo di chuyển tốt hơn.
- Ứng dụng IOT để quản lý thùng rác trên app điện thoại hoặc webserver.
- Có thể tích hợp robot cho nhiều ứng dụng khác như giao hàng.

TÀI LIỆU THAM KHẢO

1. Tài liệu tham khảo là sách:

- Phạm Minh Tuấn (2020) – Arduino cho người mới bắt đầu.
- Tài liệu tham khảo từ các nguồn khoá luận, bài luận khác:

[1]. Thái Xuân Thiện (2014). *Điều khiển xe robot*. Báo cáo đồ án môn học 2 ngành tự động hoá, Trường Đại học Bách Khoa HCM.

[2]. Hoàng Văn An, Đỗ Tuấn Anh, Trần Minh Chiến, Doãn Hoàng, Nguyễn Minh Hoàng (2017). *Thiết kế robot dò đường*. Đồ án môn học, Trường Đại học Bách Khoa HCM.

2. Tài tham khảo từ Internet:

[3]. Thực trạng quản lý rác thải ở nước ta hiện nay:

<https://kinhthemoitruong.vn/thuc-trang-va-cac-giai-phap-quan-ly-rac-thai-sinh-hoat-bang-cac-cong-cu-kinh-te-o-viet-nam-59229.html>

[4]. Mô hình hỗ trợ thu gom rác thải ở biển:

<https://baotainguyenmoitruong.vn/sinh-vien-bach-khoa-da-nang-voi-sang-che-may-thu-gom-rac-thai-o-bien-242182.html>

[5]. Thùng rác tự động thông minh:

<https://dean1665.vn/svs2021/linh-vuc-cuoc-thi/thung-rac-tu-dong-thong-minh-322.html>

[6]. Lịch sử phát triển robot:

<http://seibu.com.vn/tin-tuc/lich-su-phat-trien-cua-robot-177.html>

[7]. Ứng dụng robot dò đường trong thực tế:

<https://nangluongsachvietnam.vn/d6/vi-VN/news/Mo-hinh-thu-gom-rac-thai-tu-dong-doat-giai-nhat-6-172-4016>

[8]. Robot phục vụ y tế:

<https://www.qdnd.vn/tren-tuyen-dau-chong-dich/tin-tuc/ro-bot-cua-hoc-vien-ky-thuat-quan-su-phuc-vu-benh-nhan-covid-19-661717>

[9]. Thiết kế và thi công robot dò đường áp dụng thuật toán PID:

<http://thuvien.vku.udn.vn/handle/123456789/1049>

[10]. Robot tránh vật cản :

<http://arduino.vn/result/1306-robot-tranh-vat-can>

[11]. Sơ lược arduino mega 2560:

<https://dientutuonglai.com/gioi-thieu-arduino-mega-2560.html>

[12]. Robot dò đường tránh vật cản:

<http://kenhchetac.com/chi-tiet-san-pham/95-robot-do-duong-tranh-vat-can.html>

[13] Cộng đồng Arduino Việt Nam

<http://arduino.vn/>

PHỤ LỤC

```
// config.h
#define adc(A) analogRead(A)
#include <LiquidCrystal.h>
LiquidCrystal lcd(45, 43, 47, 49, 41, 53);
#include <Servo.h>
Servo myservo;

#define trig1 29 // cb vật cản
#define echo1 31
#define trig2 33 // cb nắp
#define echo2 35
#define sw1 23
#define sw2 25 //button
#define pump 9
#define IR1 23
#define buzz 39
#define ena 2 // left motor
#define in1 3
#define in2 4
#define in3 5 // right motor
#define in4 6
#define enb 7

void tien(int x, int y){
    digitalWrite(in1,1);digitalWrite(in2,0);
    digitalWrite(in3,1);digitalWrite(in4,0);
    analogWrite(ena,x);analogWrite(enb,y);
}

void spin(int x, int y){
    digitalWrite(in1,1);digitalWrite(in2,0);
```

```
digitalWrite(in3,0);digitalWrite(in4,1);
analogWrite(ena,x);analogWrite(enb,y);
}

void dung(){
digitalWrite(in1,0);digitalWrite(in2,0);
digitalWrite(in3,0);digitalWrite(in4,0);
analogWrite(ena,0);analogWrite(enb,0);
}
// tick buzz
void tick(int x)
{
for(int j=0;j<x;j++)
{
digitalWrite(buzz,1); delay(100);
digitalWrite(buzz,0); delay(100);
}
}
// đóng nắp servo 180->0
void close_lid(){
for (int pos = 180; pos >= 0; pos -= 1) { // close entry
myservo.write(pos);
delay(15);
}
}
// mở nắp servo từ 0->180
void open_lid(){
for (int pos = 0; pos <= 180; pos += 1) { // goes from 0 degrees to 180 degrees
// in steps of 1 degree
myservo.write(pos); // tell servo to go to position in variable 'pos'
delay(15); // waits 15ms for the servo to reach the position
}
}
// khai báo input
void pin_init(){
```

```
myservo.attach(8);
myservo.write(0);

pinMode(ena,OUTPUT);pinMode(enb,OUTPUT);
pinMode(in1,OUTPUT);pinMode(in2,OUTPUT);
pinMode(in3,OUTPUT);pinMode(in4,OUTPUT);
dung();
pinMode(buzz,OUTPUT);pinMode(pump,OUTPUT);
pinMode(trig1,OUTPUT);pinMode(echo1,INPUT);
pinMode(trig2,OUTPUT);pinMode(echo2,INPUT);
pinMode(IR1,INPUT);
pinMode(sw1,INPUT_PULLUP);
pinMode(sw2,INPUT_PULLUP);
digitalWrite(pump,1);
tick(1);
}
int kcach(int Trig_PIN, int Echo_PIN)
{
  unsigned long duration; // biến đo thời gian
  int distance;          // biến lưu khoảng cách

  /* Phát xung từ chân trig */
  digitalWrite(Trig_PIN,0); // tắt chân tri
  delayMicroseconds(2);
  digitalWrite(Trig_PIN,1); // phát xung từ chân tri
  delayMicroseconds(5); // xung có độ dài 5 microSecond
  digitalWrite(Trig_PIN,0); // tắt chân trig

  /* Tính toán thời gian */
  // Đo độ rộng xung HIGH ở chân echo.
  duration = pulseIn(Echo_PIN,HIGH);
  // Tính khoảng cách đến vật
  distance = int(duration/2/29.412);
  if (distance>100) distance=100;
  return distance;
}
```

```
}

// MAIN:
#include "config.h"
unsigned int a[9]; // mảng a gồm 9 phần tử
unsigned int thresh[9], unthresh[9];

//position variables (biến khoảng cách)
float Err=0, pre_Err=0; // biến tìm lỗi sai
float target=0;

// speed of motor
const uint8_t baseSpeeda = 170; // base speed of left motor
const uint8_t baseSpeedb = 170; // base speed of right motor

// PID constants
float kp = 27;
float kd = 2.4;
float ki = 1.2;

// PID calculation
float eprev = 0;
float eintegral = 0;

// time
long prevT = 0;

//global variables(biến toàn cục)
bool check=0;
int mode=0;
int count=0;

//trash variables(biến thùng rác)
int dis1=0,dis2=0; // khoảng cách 2 cảm biến siêu âm
int trash_percent=0; // phần trăm cho LCD
```

```
bool NapOn=1;
long time2=0;
int bins_height=29.5;

//virus variables
bool pumpOn=1;
long time1=0;

void setup() {
  Serial.begin(9600);
  pin_init();
  lcd.begin(16,2);// lcd 1602
  lcd.setCursor(0,0);lcd.print("Loading.....!");
  for(int i=3;i<12;i++){ // học màu
    unsigned int sum=0;
    for(int i1=0;i1<50;i1++) {sum += adc(i); delay(2);}
    a[i-3]=sum/50;
    Serial.println(String(i-2) + ":" +String(a[i-3]));
  }
  calc_thresh();
  delay(1000);lcd.clear();
  tick(3);
}

void calc_thresh()
{
  unsigned int max_a[9] = {530,445,315,500,600,750,680,610,500}; //new version(giá trị
đọc được khi tất cả nằm trên line)
  for(int i=0;i<9;i++)
  {
    thresh[i] = a[i] + (max_a[i]-a[i])*5/10;
    unthresh[i] = a[i] + (max_a[i]-a[i])*3/10;
  }
  for(int i=0;i<9;i++){
    Serial.print(thresh[i]);Serial.print(" ");
```



```
}
Serial.println();
for(int i=0;i<9;i++){
  Serial.print(unthresh[i]);Serial.print(" ");

}
Serial.println();
}

unsigned int line()
{
  unsigned int temp=0;
  for(int i=3;i<12;i++)
  {
    if(adc(i) > thresh[i-3] ) temp|=(0x0001<<(i-3));
    if(adc(i) < unthresh[i-3]) temp&=~(0x0001<<(i-3));
    delayMicroseconds(50);
  }
  //Serial.println(temp,BIN);
  return temp;
}

void findErr()
{
  /*
  * err > 0 : robot is in the leftside
  * err > 0 : turn right, v of left motor > right one
  * and other wise
  */
  unsigned int lineT=line();
  // Serial.print(lineT,BIN); Serial.print(" ");
  // Serial.println(Err);
  if (lineT==0b000000000) {Err=10; return;} // robot out of line
```

```
else if (lineT==0b000010000) {Err=0; return;}
else if (lineT==0b000110000) {Err=0; return;}
else if (lineT==0b000011000) {Err=0; return;}
else if (lineT==0b000111000) {Err=0; return;}

else if (lineT==0b000001000) {Err=0.85; return;}
else if (lineT==0b000001100) {Err=0.85; return;}
else if (lineT==0b000011100) {Err=0.85; return;}

else if (lineT==0b000000100) {Err=1.5; return;}
else if (lineT==0b000000110) {Err=1.5; return;}
else if (lineT==0b000001110) {Err=1.5; return;}

else if (lineT==0b000000010) {Err=3; return;}
else if (lineT==0b000000011) {Err=3; return;}
else if (lineT==0b000000111) {Err=3; return;}

else if (lineT==0b000000001) {Err=5; return;}

else if (lineT==0b000100000) {Err=-0.85; return;}
else if (lineT==0b001100000) {Err=-0.85; return;}
else if (lineT==0b001110000) {Err=-0.85; return;}

else if (lineT==0b001000000) {Err=-1.5; return;}
else if (lineT==0b011000000) {Err=-1.5; return;}
else if (lineT==0b011100000) {Err=-1.5; return;}

else if (lineT==0b010000000) {Err=-3; return;}
else if (lineT==0b110000000) {Err=-3; return;}
else if (lineT==0b111000000) {Err=-3; return;}

else if (lineT==0b100000000) {Err=-5; return;}
else {
```

```
}  
void PID_robotRun(float ln) // Err as input for Pid void  
{  
  if(ln!=10){  
    long currT = micros();  
    float deltaT = ((float) (currT - prevT))/( 1.0e6 );  
    prevT = currT;  
  
    //error  
    float e=ln-target;  
    //derivative  
    float dedt = (e-eprev)/(deltaT);  
    // integral  
    eintegral = eintegral + e*deltaT;  
  
    //control pwm signal  
    float u = kp*e + kd*dedt + ki*egral;// công thức PID  
  
    int Motorspeeda = baseSpeeda + u;  
    int Motorspeedb = baseSpeedb - u;  
  
    // motor power  
    constrain(Motorspeeda,0,255);  
    constrain(Motorspeedb,0,255);  
  
    Serial.print(e);Serial.print(" ");  
    Serial.print(Motorspeeda);Serial.print(" ");  
    Serial.println(Motorspeedb);  
  
    // robot run in1 and in3  
    digitalWrite(in1,1); digitalWrite(in2,0);  
    digitalWrite(in3,1); digitalWrite(in4,0);  
    analogWrite(ena,Motorspeeda);analogWrite(enb,Motorspeedb);
```

```
// store previous error(luu tru loi lan truoc)
eprev = e;
}
else {
  dung();
  count++;
  if((count%2)==1){
    open_lid();      // open entry
    delay(3000);
    close_lid();     // close entry
    lcd.clear();
    mode=0;
    Serial.println("robot stop!!");
    return;
  }
  else{
    turnAround();
    mode=0;
    Serial.println("robot stop!!");
    return;
  }
}
}

void find_line(){
  while(line()==0) tien(200,200);
  dung();
  tick(2);
}

void turnAround(){      // robot turn around into line then stop
  spin(170,170);
  delay(2500);
  while(line()<8) Serial.println(line(),BIN);
  dung();delay(500);
```

```
}

void loop() {
// char c=Serial.read();
// if(c=='q'){
//   find_line();
//   check=0;
// }
// else if(c=='r'){
//   check=1;
// }
// else if(c=='s'){
//   dung();
//   check=0;
// }
// else if(c=='t'){
//   turnAround();
// }
// else if(c=='l'){
//   findErr();
// }
// if(check==1){
//   findErr();
//   PID_robotRun(Err);
//   //if(Err==10) {dung();check=0;}
// }

while(mode==0){
  lcd.setCursor(0,0);lcd.print("Rac: ");lcd.print(trash_percent); lcd.print("% ");
  if(digitalRead(IR1)==0){
    time1=millis();
    while(millis()-time1<3000&&umpOn==true){ // bom nuoc trong 3s
      digitalWrite(pump,0);
      if(digitalRead(IR1)==1){
```

```
    digitalWrite(pump,1);
    delay(500);
    break;
  }
}
pumpOn=false;
digitalWrite(pump,1);
}
else {
  pumpOn=true;
  digitalWrite(pump,1);
}

dis1=kcach(trig1,echo1);    // do kcach phía trước

if(dis1<6){
  time2=millis();
  NapOn=0;
  open_lid();    // mở nắp trong 4s
  while(millis()-time2<4000){
  }
  close_lid(); //
}
else {
  NapOn=1;
  myservo.write(0);
}

if(NapOn==1){    // nắp closed
  dis2=kcach(trig2,echo2);    // do kcach tới rác
  trash_percent=dis2*100/bins_height;
  trash_percent=100-trash_percent;
  if(trash_percent>80){
    lcd.setCursor(0,1);lcd.print("Trash full!!!");
    tick(5);
  }
}
```

```
    find_line();
    mode=1;
    goto Go;
  }
}

if(digitalRead(sw2)==0){
  delay(30);
  if(digitalRead(sw2)==0){
    Serial.print("button is pressed!");
    while(digitalRead(sw2)==0);
    tien(150,150);delay(1000);
    turnAround();
    mode=1;
    goto Go;
  }
}
delay(100);
}

Go:
while(mode==1){
  dis1=kcach(trig1,echo1);
  if(dis1<10) {dung();digitalWrite(buzz,1);}
  else {findErr();PID_robotRun(Err);digitalWrite(buzz,0);}
}
}
```