

BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP. HỒ CHÍ MINH
KHOA CÔNG NGHỆ ĐIỆN – ĐIỆN TỬ



ĐỒ ÁN 2

NGÀNH CÔNG NGHỆ KỸ THUẬT
ĐIỀU KHIỂN VÀ TỰ ĐỘNG HÓA

ĐỀ TÀI:
THIẾT KẾ MÔ HÌNH ROBOT TỰ HÀNH THẢ VẬT

GVHD: ThS. Trần Hoàn
SVTH: Lương Tuyền Vinh
MSSV: 2032225871
LỚP: 13DHTDH01

THÀNH PHỐ HỒ CHÍ MINH, THÁNG 6 NĂM 2025

BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG THƯƠNG TP. HỒ CHÍ MINH
KHOA CÔNG NGHỆ ĐIỆN – ĐIỆN TỬ



ĐỒ ÁN 2

NGÀNH CÔNG NGHỆ KỸ THUẬT
ĐIỀU KHIỂN VÀ TỰ ĐỘNG HÓA

ĐỀ TÀI:
THIẾT KẾ MÔ HÌNH ROBOT TỰ HÀNH THẢ VẬT

GVHD: ThS. Trần Hoàn

TÊN SINH VIÊN: Lương Tuyên Vinh

MSSV: 2032225871

LỚP: 13DHTDH01

THÀNH PHỐ HỒ CHÍ MINH, THÁNG 6 NĂM 2025

NHIỆM VỤ ĐỒ ÁN 2

1. Họ và tên sinh viên được giao đề tài (Số lượng sinh viên: 3)

- (1) Ngô Quốc Việt MSSV: 2032225853 Lớp: 13DHTDH01 ...
(2) Trần Đăng Quốc Anh..... MSSV: 2032220146.....Lớp: 13DHTDH02 ...
(3) Lương Tuyền Vinh..... MSSV: 2032225871Lớp: 13DHTDH01 ...

2. Tên đề tài: THIẾT KẾ VÀ THI CÔNG MÔ HÌNH ROBOT TỰ HÀNH DÒ LINE VÀ GẮP VẬT

3. Nhiệm vụ của đề tài:

Thiết kế và thi công một mô hình robot có khả năng di chuyển tự động theo line bằng cảm biến hồng ngoại, đồng thời điều khiển cơ cấu gấp để thực hiện thao tác lấy và di chuyển vật thể. Đề tài tập trung vào việc thiết kế phần cứng và phát triển chương trình điều khiển nhằm đảm bảo robot hoạt động chính xác, ổn định và hiệu quả trong môi trường thực tế, từ đó góp phần ứng dụng vào lĩnh vực tự động hóa và công nghệ thông minh.....

4. Ngày giao nhiệm vụ: ...27/02/2025.....

5. Ngày hoàn thành và nộp về khoa: 14/06/2025.....

Tp.HCM, ngày 14 tháng 6 năm 2025

Giảng viên hướng dẫn

LỜI CAM ĐOAN

Nhóm em xin cam đoan đây là công trình nghiên cứu của riêng nhóm em và được sự hướng dẫn của thầy Trần Hoàn. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong Báo cáo đề án 2 còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào nhóm em xin hoàn toàn chịu trách nhiệm về nội dung Báo cáo đề án 2 của mình. Trường Đại học Công Thương Thành Phố Hồ Chí Minh không liên quan đến những vi phạm tác quyền, bản quyền do nhóm em gây ra trong quá trình thực hiện (nếu có).

TP. Hồ Chí Minh, ngày 14 tháng 6 năm 2025

Đại diện nhóm

Việt

Ngô Quốc Việt

LỜI CẢM ƠN

Nhóm sinh viên thực hiện xin chân thành cảm ơn thầy Trần Hoàn đã luôn theo sát, giúp đỡ tận tình, và có những lời khuyên bổ ích giúp nhóm em hoàn thành đồ án này.

Trong suốt quá trình thực hiện đồ án “THIẾT KẾ VÀ THI CÔNG MÔ HÌNH ROBOT TỰ HÀNH DÒ LINE VÀ GẮP VẬT” nhóm em cũng đã nhận được rất nhiều đóng góp và giúp đỡ của các thầy cô trong trường và các bạn cùng lớp. Nhóm em xin chân thành gửi lời cảm ơn tới tất cả các thầy cô, và các bạn đã giúp đỡ chúng em trong quá trình làm đồ án.

Đặc biệt nhóm em xin chân thành gửi lời cảm ơn tới Ban Giám Hiệu Trường Đại Học Công Thương TP. Hồ Chí Minh, và các thầy cô trong trường đã truyền đạt cho chúng em những kiến thức hay và bổ ích.

Nhóm em xin chân thành cảm ơn!

TP. Hồ Chí Minh, ngày 14 tháng 6 năm 2025

Đại diện nhóm

Viết

Ngô Quốc Việt

TÓM TẮT

1. Một số thông tin nền về đề tài

Lĩnh vực: Kỹ thuật điều khiển và tự động hóa

Cấp học: Đại học

Mục tiêu: Rèn luyện kỹ năng thiết kế và chế tạo robot tự hành cho sinh viên

2. Mục đích của nghiên cứu và phạm vi đề tài

Mục đích của nghiên cứu là thiết kế và thi công một mô hình robot có thể tự động dò đường thông qua các cảm biến, đồng thời thực hiện gấp vật bằng cơ cấu điều khiển. Phạm vi của đề tài giới hạn trong việc xây dựng một mô hình robot ở quy mô nhỏ, hoạt động ổn định trên mặt phẳng có đường line cố định, với chức năng nhận diện và xử lý vật thể đơn giản. Đề tài tập trung vào nền tảng điều khiển cơ bản và ứng dụng thực tế trong môi trường được xác định trước.

3. Phương pháp được dùng trong nghiên cứu

Trong quá trình nghiên cứu, nhóm sử dụng phương pháp thiết kế hệ thống tích hợp bao gồm cả phần cứng và phần mềm. Phần cứng gồm các cảm biến dò line, động cơ cho cơ cấu gấp và di chuyển, Arduino làm trung tâm xử lý và khung sườn xe. Phần mềm điều khiển được lập trình trên vi điều khiển nhằm xử lý dữ liệu từ cảm biến và điều khiển hoạt động của robot theo chương trình nhóm đã xây dựng. Phương pháp thử nghiệm – hiệu chỉnh cũng được áp dụng thường xuyên để đảm bảo độ chính xác và hiệu quả trong vận hành.

4. Kết quả nghiên cứu quan trọng nhất

Kết quả đạt được là một mô hình robot hoàn chỉnh có khả năng tự động di chuyển theo đường line cố định, nhận biết vật thể ở vị trí định sẵn và thực hiện thao tác gấp và di chuyển vật đến vị trí mới một cách chính xác. Robot hoạt động ổn định, phản hồi nhanh, và có thể thực hiện lặp lại nhiều lần mà không sai lệch đáng kể. Mô hình này không chỉ đáp ứng tốt mục tiêu đề ra mà còn tạo nền tảng cho các nghiên cứu nâng cao trong lĩnh vực robot tự hành và hệ thống tự động hóa.

MỤC LỤC

LỜI CẢM ƠN.....	iv
TÓM TẮT.....	v
MỤC LỤC.....	vi
DANH MỤC CÁC THUẬT NGỮ, CHỮ VIẾT TẮT	ix
DANH SÁCH BẢNG.....	x
DANH SÁCH HÌNH.....	xi
CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI.....	1
1.1. Lý do chọn đề tài.....	1
1.2. Mục đích.....	1
1.3. Đối tượng và phạm vi nghiên cứu.....	1
1.4. Phương pháp nghiên cứu.....	2
1.5. Sơ lược các bước thực hiện đề tài	2
CHƯƠNG 2. CƠ SỞ LÝ THUYẾT.....	4
2.1. Các thiết bị sử dụng.....	4
2.2. Arduino Mega 2560	4
2.2.1. Giới thiệu chung.....	4
2.2.2. Các đặc tính kỹ thuật của Board mạch Arduino Mega 2560.....	6
2.2.3. Sơ đồ chân của Arduino Mega 2560	6
2.3. Module cảm biến dò line 1 kênh TCRT5000	8
2.3.1. Giới thiệu.....	8
2.3.2. Cấu tạo.....	9
2.3.3. Nguyên lý hoạt động	9
2.4. Module cảm biến vật cản hồng ngoại.....	10
2.4.1. Giới thiệu.....	10
2.4.2. Cấu tạo.....	10
2.4.3. Nguyên lý hoạt động	11
2.5. Module điều khiển động cơ L298N.....	12
2.5.1. Giới thiệu.....	12

2.5.2. Thông số kỹ thuật và sơ đồ chân	13
2.5.3. Nguyên lý hoạt động	15
2.6. Động cơ giảm tốc V1.....	15
2.6.1. Giới thiệu.....	15
2.6.2. Thông số kỹ thuật.....	16
2.6.3. Nguyên lý hoạt động	16
2.7. Module hạ áp DC LM2596.....	17
2.7.1. Giới thiệu.....	17
2.7.2. Thông số kỹ thuật.....	18
2.7.3. Nguyên lý hoạt động	18
2.8. Pin Li-ion 18650.....	18
2.8.1. Giới thiệu.....	18
2.8.2. Cấu tạo.....	19
2.8.3. Nguyên lý hoạt động	19
2.9. Bánh xe	19
2.9.1. Giới thiệu.....	19
2.9.2. Thông số kỹ thuật.....	20
2.10. Các linh kiện khác	20
2.10.1. Công tắc.....	20
2.10.2. Công tắc hành trình	21
2.10.3. Giắc đực-cái DC.....	21
2.10.4. Nút nhấn tròn.....	22
2.10.5. Vôn kế điện tử	22
2.10.6. Mạch bảo vệ và sạc pin 3S	23
2.11. Phần mềm Arduino IDE	23
2.11.1. Giới thiệu.....	23
2.11.2. Giao diện phần mềm.....	24
2.11.3. Các lệnh thường dùng.....	25
2.12. Phần mềm Easy EDA	26
2.12.1 Giới thiệu.....	26

2.12.2	Giao diện phần mềm.....	26
2.13.	Phần mềm AutoCAD 2007.....	27
2.13.1	Giới thiệu.....	28
2.13.2	Giao diện phần mềm.....	28
CHƯƠNG 3. THIẾT KẾ VÀ THI CÔNG		30
3.1.	Quy trình công nghệ.....	30
3.1.1.	Sơ đồ công nghệ	30
3.1.2.	Yêu cầu bài toán	30
3.2.	Sơ đồ khối	30
3.2.1.	Khối nguồn (Power source).....	31
3.2.2.	Khối ngõ vào (Input)	31
3.2.3.	Khối xử lý (Operation)	31
3.2.4.	Khối ngõ ra (Output)	32
3.3.	Lưu đồ giải thuật	33
3.3.1.	Giải thích lưu đồ giải thuật.....	36
3.4.	Sơ đồ đấu dây	37
3.5.	Mô hình phần cứng.....	39
CHƯƠNG 4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN		41
4.1.	Kết quả đạt được	41
4.1.1.	Ưu điểm.....	41
4.1.2.	Nhược điểm.....	41
4.2.	Hướng phát triển đề tài.....	42
4.3.	Kết luận	42
TÀI LIỆU THAM KHẢO		43
PHỤ LỤC.....		44

DANH MỤC CÁC THUẬT NGỮ, CHỮ VIẾT TẮT

Viết tắt	Tiếng anh	Tiếng việt
UART	Universal Asynchronous Receiver/Transmitter	Giao tiếp nối tiếp không đồng bộ
SPI	Serial Peripheral Interface	Giao tiếp ngoại vi nối tiếp
I2C	Inter-Integrated Circuit	Giao tiếp liên tích hợp
JTAG	Joint Test Action Group	Giao tiếp JTAG
PWM	Pulse Width Modulation	Điều chế độ rộng xung

DANH SÁCH BẢNG

Bảng 2.1. Các linh kiện sử dụng cho robot dò line và gấp vật.....	4
Bảng 2.2: Các đặc tính kỹ thuật của Arduino Mega 2560	6
Bảng 2.3: Chân nguồn của mạch Arduino Mega 2560	8
Bảng 2.4. Bảng kết nối các chân cảm biến SCRT5000.....	9
Bảng 2.5. Các lệnh thường dùng trong Arduino IDE.....	25
Bảng 3.1. Chân nối với Arduino.....	37

DANH SÁCH HÌNH

Hình 2.1. Arduino Mega 2560.....	4
Hình 2.2. Sơ đồ chân Arduino Mega 2560.....	7
Hình 2.3. Module TCRT5000	8
Hình 2.4. Nguyên lý hoạt động của cảm biến	9
Hình 2.5. Cảm biến vật cản hồng ngoại HW-488	10
Hình 2.6. Sơ đồ chân và chức năng.....	11
Hình 2.7. Nguyên lý hoạt động cảm biến hồng ngoại.....	12
Hình 2.8. Module điều khiển động cơ L298N.....	12
Hình 2.9. Sơ đồ chân Module L298N	14
Hình 2.10. Mạch cầu H.....	15
Hình 2.11. Động cơ giảm tốc	15
Hình 2.12. Module hạ áp DC LM2596.....	17
Hình 2.13. Pin 18650.....	18
Hình 2.14. Nguyên lý hoạt động pin 18650	19
Hình 2.15. Bánh xe.....	20
Hình 2.16. Công tắc bấm bẻ.....	20
Hình 2.17. Công tắc hành trình	21
Hình 2.18. jack cắm DC	21
Hình 2.19. Nút nhấn DC.....	22
Hình 2.20. Vôn kế điện tử 0.28Inch	22
Hình 2.21. Mạch bảo vệ và sạc pin 3S	23
Hình 2.22. Phần mềm arduino IDE	24
Hình 2.23. Giao diện phần mềm Arduino IDE.....	24
Hình 2.24: Vùng lệnh	25
Hình 2.25. Phần mềm Easy EDA	26
Hình 2.26. Giao diện phần mềm Easy EDA.....	27
Hình 2.27. Phần mềm Auto CAD 2007.....	28
Hình 2.28. Giao diện phần mềm Auto CAD 2007	28
Hình 3.1. Sơ đồ công nghệ	30
Hình 3.2. Sơ đồ khối xe robot dò line và gắp vật.....	31
Hình 3.3. Lưu đồ giải thuật	35

Hình 3.4. Sơ đồ đầu dây	37
Hình 3.5. Thiết kế phần cứng trên Autocad	39
Hình 3.6. Mặt trước và sau xe robot.....	39
Hình 3.7. Mặt trái và phải của xe robot.....	40
Hình 3.8. Mặt trên của xe robot.....	40

CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI

1.1. Lý do chọn đề tài

Ngành công nghiệp robot toàn cầu đã đạt được những bước tiến quan trọng và đem lại nhiều sản phẩm hiện đại phục vụ mọi hoạt động đời sống của con người cũng như là trong sản xuất. Ngày nay, Việt Nam hay ở các nước trên thế giới, việc ứng dụng robot trong các dây chuyền sản xuất cũng ngày càng được sử dụng và ứng dụng rộng rãi.

Trước những thách thức này và với sự sáng tạo từ môi trường học tập tại khoa Công nghệ Điện - Điện tử, Trường Đại học Công Thương TP. Hồ Chí Minh, nhóm em đã lựa chọn đề tài “Thiết kế và thi công mô hình robot tự hành dò line và gấp vật”. Đề tài giúp rèn luyện kỹ năng thực hành và nắm vững kiến thức chuyên môn, mà còn khuyến khích sự tư duy sáng tạo và phát triển kỹ năng làm việc nhóm cũng như là hành trang kinh nghiệm cho con đường học tập và làm việc sau này.

1.2. Mục đích

Robot dò line không chỉ có nhiều ứng dụng trong thực tế mà còn là công cụ học tập lý tưởng cho sinh viên áp dụng kiến thức từ giảng đường vào thực hành. Với cấu trúc cơ khí đơn giản nhưng linh hoạt, kết hợp với các linh kiện điện tử như cảm biến dò line và cảm biến hồng ngoại phát hiện vật cản, robot dò line trở thành một sản phẩm học thuật để học tập và nghiên cứu rất phù hợp cho sinh viên ngành Tự động hóa. Những robot này không chỉ giúp sinh viên thực hành và nắm vững kiến thức về điện tử, cơ khí và lập trình mà còn giúp phát triển kỹ năng làm việc nhóm và tư duy sáng tạo. Việc thiết kế và lập trình robot dò line đòi hỏi sự kết hợp giữa các kiến thức lý thuyết và kỹ năng thực hành, từ đó giúp sinh viên rèn luyện khả năng phân tích và giải quyết vấn đề một cách hiệu quả nhất. Ngoài ra, việc thực hiện các dự án robot cũng giúp sinh viên hiểu rõ hơn về quá trình sản xuất và ứng dụng của robot trong các ngành công nghiệp khác nhau, từ đó chuẩn bị tốt hơn cho sự nghiệp của mình trong lĩnh vực tự động hóa và trang bị những kỹ năng cần thiết để tham gia vào các dự án công nghệ trong tương lai.

1.3. Đối tượng và phạm vi nghiên cứu

Đối tượng: Mô hình robot tự hành dò line và gấp vật

Là một mô hình xe dò line, được thiết kế để di chuyển theo một đường line cố định. Mục tiêu chính là đảm bảo rằng mô hình này có thể hoạt động ổn định, tức là có thể nhận biết được đường line và điều khiển động cơ sao cho xe di chuyển theo đường line đó một cách chính xác tới vật để cơ cấu có thể gấp vật.

Bao gồm 4 phần chính:

Cảm biến dò line

Bộ điều khiển

Cơ cấu gấp vật

Động cơ và hệ thống điều khiển động cơ

Phạm vi nghiên cứu:

Tìm hiểu về nguyên lý hoạt động của một số module, linh kiện: Arduino Mega 2560, cảm biến dò line, cảm biến hồng ngoại phát hiện vật cản, mạch điều khiển động cơ L298N và một số linh kiện khác phục vụ cho đề tài. Cụ thể, chúng ta sẽ tập trung vào việc thiết kế và chế tạo phần cứng và phần mềm để xe dò line có thể tự động di chuyển theo đường line và đồng thời có khả năng gấp, thả và di chuyển các vật thể đến khu vực quy định.

Kiểm tra và điều chỉnh:

Tiến hành các thử nghiệm và điều chỉnh hệ thống để đảm bảo rằng xe có thể hoạt động ổn định và chính xác trong nhiều điều kiện khác nhau.

Qua quá trình tìm hiểu này, sinh viên sẽ có thể hiểu rõ hơn về cách hoạt động của một hệ thống robot phức tạp và cũng như nắm vững kiến thức về cơ khí, điện tử và lập trình. Thời gian và kiến thức có hạn, nên phạm vi đề án chỉ tập trung vào tìm hiểu và thiết kế xe đơn giản trên cơ sở hoạt động của bộ phận cảm biến và điều chỉnh động cơ.

1.4. Phương pháp nghiên cứu

Tham khảo các tài liệu khoa học, sách báo, bài báo liên quan đến đề tài để thu thập thông tin và kiến thức nền tảng.

Phương pháp nghiên cứu thực nghiệm:

Thiết kế, chế tạo và thử nghiệm robot để kiểm tra tính chính xác, hiệu quả và độ ổn định.

1.5. Sơ lược các bước thực hiện đề tài

Thiết kế khung xe của robot:

Xác định kích thước và hình dạng của khung xe dựa trên yêu cầu cụ thể của dự án, bao gồm chiều dài, chiều rộng và chiều cao của robot.

Sử dụng phần mềm CAD để vẽ thiết kế khung xe, bao gồm cả các chi tiết như lỗ để gắn bánh xe và động cơ, khe hở để lắp các linh kiện điện tử và cơ khí.

Chế tạo khung xe:

Sử dụng các công cụ thủ công và máy in 3D để chế tạo các phần của khung xe từ vật liệu mica. Lắp ráp và gắn kết các phần của khung xe theo thiết kế đã vẽ trên phần mềm CAD, sử dụng các phụ kiện và vật liệu kết nối như ốc vít, bulong và keo dán để đảm bảo sự chắc chắn và ổn định của cấu trúc.

Lập trình và kiểm tra:

Phát triển mã lập trình cho robot bằng ngôn ngữ lập trình Arduino IDE. Lập trình các chức năng như điều khiển động cơ, xử lý dữ liệu từ các cảm biến và điều khiển cơ cấu gấp vật. Tiến hành các thử nghiệm thực tế và kiểm tra để đảm bảo rằng robot hoạt động đúng như mong đợi và đáp ứng các yêu cầu của đề tài.

Tinh chỉnh và cải tiến:

Dựa trên kết quả kiểm tra và phản hồi từ quá trình thử nghiệm, tiến hành các điều chỉnh và cải tiến để cải thiện hiệu suất và độ chính xác của robot.

Lặp lại quy trình lập trình và kiểm tra nếu cần thiết để đảm bảo robot hoạt động ổn định và hiệu quả.

CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

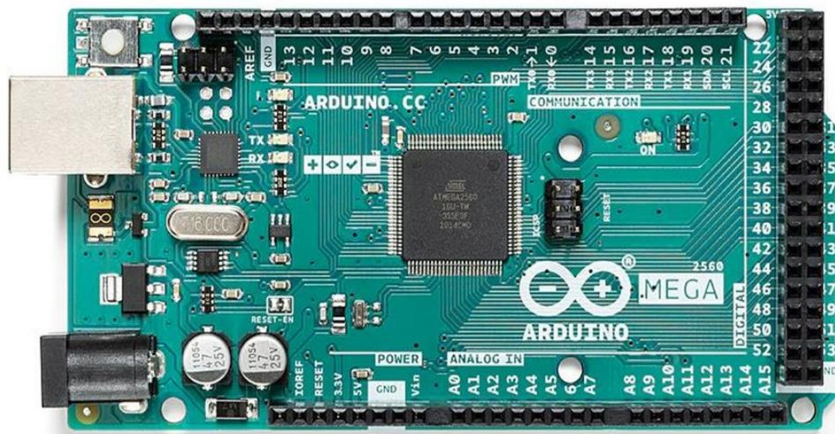
2.1. Các thiết bị sử dụng

Bảng 2.1. Các linh kiện sử dụng cho robot dò line và gấp vật

Tên linh kiện	Số lượng
Arduino Mega 2560	01
Module cảm biến dò line 1 kênh TCRT5000	03
Module cảm biến vật cản hồng ngoại HW-488	01
Động cơ DC giảm tốc V1	06
Module hạ áp DC LM2596	02
Module điều khiển động cơ L298N	02
Pin 18650	03
Mạch cân bằng và sạc pin 3S 20A	01
Công tắc hành trình	04
Nút nhấn	02
Công tắc bật/bênh	01
Jack cắm tròn DC cái	01
Vôn kế điện tử 0.28 Inch	01

2.2. Arduino Mega 2560

2.2.1. Giới thiệu chung



Hình 2.1. Arduino Mega 2560

Arduino Mega 2560 là một trong những bo mạch vi điều khiển tiên tiến và mạnh mẽ nhất trong dòng sản phẩm Arduino, được thiết kế nhằm thay thế và nâng cấp các phiên bản trước đó, đồng thời đáp ứng nhu cầu của các dự án kỹ thuật phức tạp hơn. Với khả năng tích hợp của nhiều tính năng vượt trội, bo mạch này cho phép triển khai các hệ thống điều khiển nhiều đầu vào/đầu ra (I/O) với quy mô lớn, mang lại sự linh hoạt và hiệu quả trong thiết kế. Cụ thể, Arduino Mega 2560 được trang bị 54 chân I/O digital và 16 chân đầu vào analog, hỗ trợ đầy đủ các giao thức giao tiếp như UART (Universal Asynchronous Receiver/Transmitter), SPI (Serial Peripheral Interface) và I2C (Inter-Integrated Circuit), từ đó tạo điều kiện thuận lợi cho việc kết nối với nhiều thiết bị ngoại vi và cảm biến khác nhau.

Về mặt kỹ thuật, Arduino Mega 2560 được tích hợp nhiều tính năng hiện đại nhằm đáp ứng các yêu cầu khắt khe của các dự án phức tạp. Bo mạch này hỗ trợ giao thức JTAG (Joint Test Action Group), cho phép lập trình, gỡ lỗi và xử lý sự cố một cách hiệu quả. Ngoài ra, Arduino Mega 2560 còn được trang bị các chức năng như bộ định thời (Timer), so sánh tín hiệu, quản lý năng lượng và các ngắt (Interrupt), giúp tối ưu hóa khả năng kiểm soát hệ thống. Với xung nhịp 16 MHz, đảm bảo tốc độ xử lý nhanh, cho phép thực thi các chương trình phức tạp một cách mượt mà và chính xác. Về dung lượng lưu trữ, Arduino Mega 2560 sử dụng bộ xử lý ATmega2560 của Microchip Technology, với bộ nhớ flash 256 KB, SRAM 8 KB và EEPROM 4 KB, cung cấp không gian lưu trữ rộng rãi cho dữ liệu, biến và chương trình trong quá trình lập trình.

Arduino Mega 2560 sở hữu tổng cộng 54 chân I/O digital, trong đó bao gồm 15 chân hỗ trợ PWM (Pulse Width Modulation), 16 chân đầu vào analog và 4 cổng UART, tạo điều kiện thuận lợi cho việc giao tiếp với nhiều thiết bị ngoại vi. Các chân I/O này không chỉ đa dạng về số lượng mà còn linh hoạt trong ứng dụng, cho phép mạch đáp ứng các yêu cầu điều khiển phức tạp trong các dự án kỹ thuật. Bên cạnh đó, bo mạch có thể được cấp nguồn từ 5V đến 12V thông qua cổng nguồn hoặc kết nối USB, đồng thời hỗ trợ truyền tải chương trình từ máy tính qua cổng USB một cách đơn giản, hiệu quả và tiện lợi.

Về mặt lập trình, Arduino Mega 2560 được phát triển để hoạt động trên phần mềm Arduino IDE, sử dụng ngôn ngữ lập trình C/C++. Phần mềm này cung cấp một hệ sinh thái phong phú với nhiều thư viện và ví dụ mẫu, giúp đơn giản hóa quá trình phát triển ứng dụng và lập trình, ngay cả đối với những người mới bắt đầu. Điểm nổi bật của bo

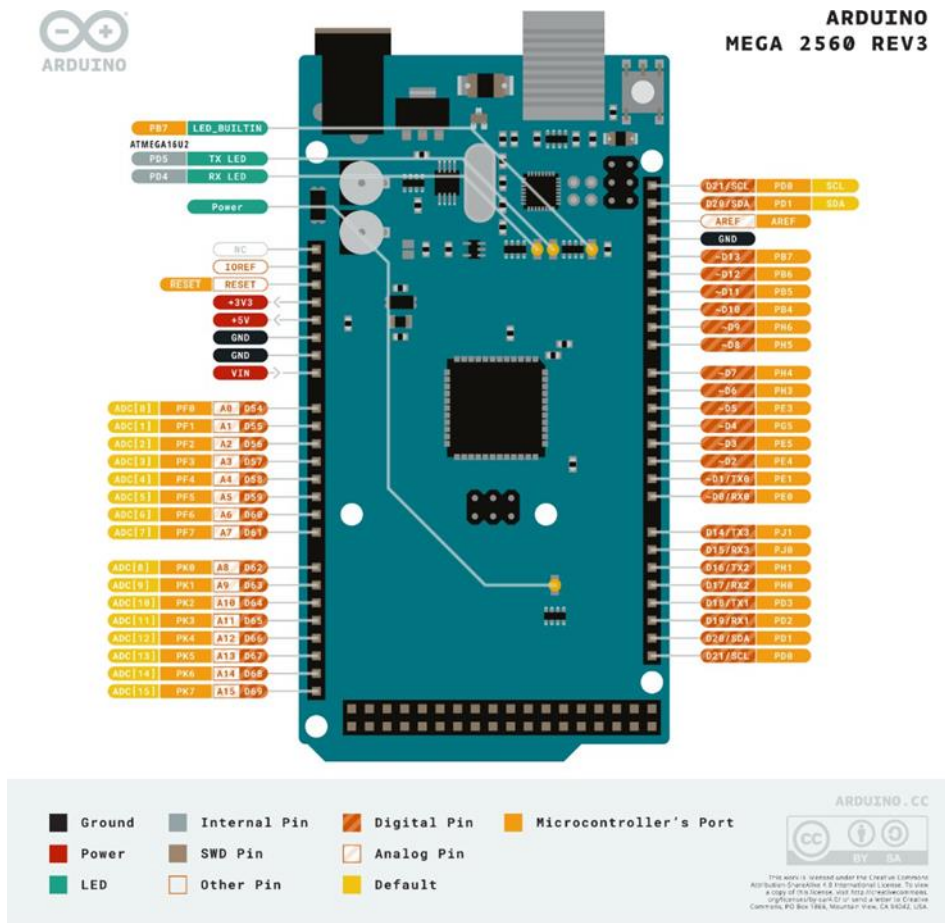
mạch là khả năng xử lý các chương trình hệ thống lớn nhờ vào bộ nhớ flash và SRAM rộng rãi, mang lại sự linh hoạt trong việc triển khai các dự án đòi hỏi dung lượng lưu trữ cao. Ngoài ra, với kích thước 101,6 mm x 53,3 mm, Arduino Mega 2560 có kích thước lớn hơn so với các bo mạch Arduino khác, nhưng điều này là cần thiết để đáp ứng số lượng chân I/O và các tính năng kỹ thuật vượt trội, từ đó phù hợp với các dự án phức tạp yêu cầu nhiều kết nối và khả năng điều khiển đa dạng.

2.2.2. Các đặc tính kỹ thuật của Board mạch Arduino Mega 2560

Bảng 2.2: Các đặc tính kỹ thuật của Arduino Mega 2560

Arduino Mega 2560	Tính năng, đặc điểm
Vi điều khiển	AVR Atmega 2560 (8bit)
Nguồn cung cấp	5-12V
Số chân I/O digital	54
Số chân I/O analog	16
Xung clock	16 MHz
Bộ nhớ flash	128 KB
SRAM	8 KB
Giao tiếp	USB ICSP SPI I2C
Bộ Timer	2 (8bit) + 4 (16bit) = 6 Timer
PWM	12 (2-16 bit)
ADC	16 (10 bit)
USART	4
Kích thước	101.52mm x 53.3mm

2.2.3. Sơ đồ chân của Arduino Mega 2560



Hình 2.2. Sơ đồ chân Arduino Mega 2560

Chân RESET: Với nút ấn có thể tác động để reset lại chương trình hệ thống. Chân này dùng khi thiết bị bị lỗi, ngoài ra chân RESET còn có thể sử dụng để kết nối với thiết bị bên ngoài nhằm điều khiển việc thiết lập lại vi điều khiển

Chân XTAL1, XTAL2: Thạch anh (16MHz) được kết nối với xung clock để cấp xung nhịp cho hệ thống.

Chân AREF: Dùng để chuyển đổi tín hiệu điện áp tham chiếu từ bên ngoài khi sử dụng ADC. Từ đó không cần sử dụng điện áp nội bộ 1.1V và 5V.

Chân Digital: Các chân từ 00-53. Các chân này có thể sử dụng làm đầu vào hoặc đầu ra. Được thiết lập bằng các hàm pinMode(), digitalRead(), digitalWrite().

Chân Analog: Các chân từ A0-A15. Có thể sử dụng làm đầu vào analog cho bộ ADC hoặc đầu ra digital. Đọc trạng thái chân bằng hàm analogRead().

Chân SPI: MISO (25), SS (22), SCK (23), MOSI (24).

Chân I2C: SDA (20), SCK (21) với tốc độ 400kHz.

Chân PWM: 2 - 13, dùng để tạo ra xung PWM với giá trị từ 0 đến 255 bằng hàm analogWrite().

Chân USART: RXD0 (0), TXD0 (1), RXD1 (19), TXD1 (18), RXD2 (17), TXD2 (16), RXD3 (15), TXD3 (14). Dùng để truyền và nhận dữ liệu từ bo mạch đến máy tính và các thiết bị khác, có thể cài đặt tốc độ truyền dữ liệu bằng hàm Serial.begin().

Chân ngắt:

Các chân digital: 0, 10, 11, 12, 13, 14, 15, 22, 23, 24, 25.

Các chân analog: 6, 7, 8, 9, 10, 11, 12, 13, 14, 15.

Dùng để ngắt thiết bị khi cần thiết.

Chân ngắt phần cứng: Chân 18 - 21, 2, 3.

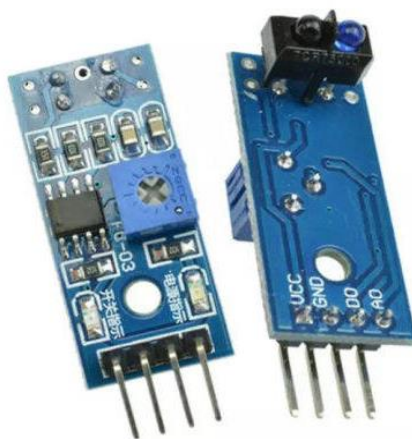
Dùng để ngắt phần cứng từ các thiết bị khác.

Bảng 2.3: Chân nguồn của mạch Arduino Mega 2560

VIN	Cung cấp điện áp (7-12V)
GND	Chân nối đất
Nguồn 5V	Cung cấp cho thiết bị phần cứng bên ngoài
Nguồn 3.3V	Cung cấp cho thiết bị phần cứng có điện áp thấp bên ngoài

2.3. Module cảm biến dò line 1 kênh TCRT5000

2.3.1. Giới thiệu



Hình 2.3. Module TCRT5000

Cảm biến TCRT5000 là một loại cảm biến quang phản xạ, được sử dụng phổ biến trong các dự án robot dò line, cảm biến vật cản, ... TCRT5000 hoạt động dựa trên nguyên lý phát hiện phản xạ hồng ngoại từ bề mặt. Nhờ thiết kế nhỏ gọn, độ nhạy cao và giá thành thấp, TCRT5000 trở thành một lựa chọn lý tưởng cho các hệ thống tự động.

Module cảm biến dò line đơn TCRT5000 tích hợp cả phần tử phát (IR LED) và thu (phototransistor) trên cùng một bo mạch, giúp dễ dàng tích hợp với các vi điều khiển như Arduino, ESP32, STM32, hay 8051.

2.3.2. Cấu tạo

IR LED (Diode hồng ngoại): phát ánh sáng hồng ngoại ra phía trước.

Phototransistor: thu nhận ánh sáng phản xạ từ vật thể phía trước.

Điện trở hạn dòng: giới hạn dòng điện qua LED để bảo vệ mạch.

Op-amp hoặc bộ so sánh điện áp: khuếch đại tín hiệu và đưa ra mức logic phù hợp (HIGH/LOW).

Chiết áp (potentiometer): dùng để điều chỉnh độ nhạy của cảm biến (khoảng cách phản xạ).

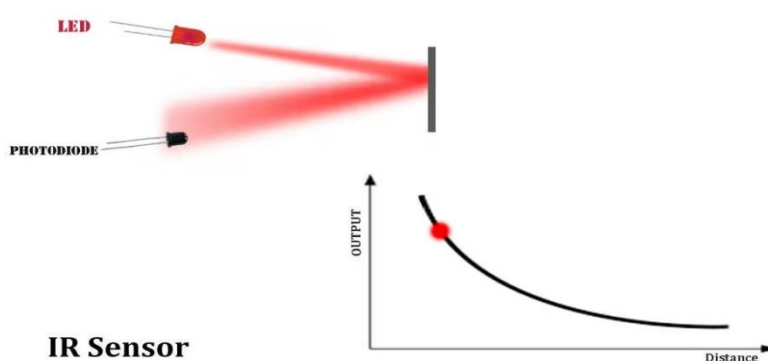
LED: hiển thị trạng thái hoạt động của cảm biến (sáng khi phát hiện vật thể).

Các chân kết nối: gồm VCC, GND, D0, A0.

Bảng 2.4. Bảng kết nối các chân cảm biến SCRT5000

VCC	Cấp nguồn +5VDC
GND	Cấp nguồn -5VDC
D0	Ngõ ra Digital
A0	Ngõ ra Analog

2.3.3. Nguyên lý hoạt động



Hình 2.4. Nguyên lý hoạt động của cảm biến

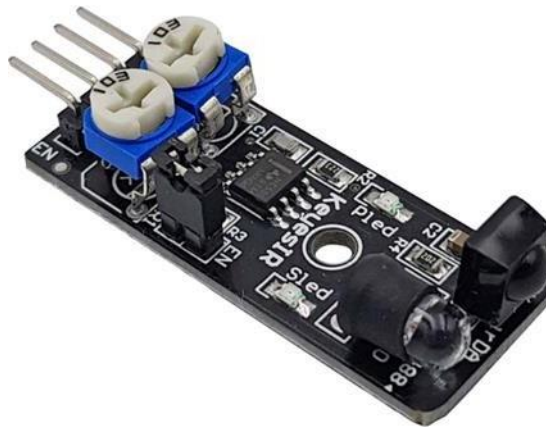
Cảm biến phát ra tia hồng ngoại (IR) nhờ vào một diode phát sáng hồng ngoại (IR LED). Nếu tia hồng ngoại gặp bề mặt sáng (như nền trắng), nó sẽ bị phản xạ trở lại và được thu bởi phototransistor.

Nếu tia hồng ngoại chiếu vào bề mặt tối (như vạch đen), ánh sáng bị hấp thụ, và phototransistor không nhận được tín hiệu phản xạ.

Từ sự khác biệt này, cảm biến có thể xác định được khi nào robot đang chạy trên vạch đen hay nền trắng, qua đó điều chỉnh hướng đi sao cho phù hợp...

2.4. Module cảm biến vật cản hồng ngoại

2.4.1. Giới thiệu



Hình 2.5. Cảm biến vật cản hồng ngoại HW-488

Cảm biến vật cản hồng ngoại là một thiết bị dùng để phát hiện vật cản dựa trên nguyên lý phản xạ tia hồng ngoại. Nó được sử dụng rộng rãi trong các hệ thống tự động hóa, robot, giám sát và nhiều ứng dụng khác.

2.4.2. Cấu tạo



Hình 2.6. Sơ đồ chân và chức năng

Tổng quan về mạch sẽ bao gồm:

Bộ phát (LED hồng ngoại): Phát ra tia hồng ngoại với tần số nhất định (thường khoảng 38kHz để giảm nhiễu từ ánh sáng môi trường).

Bộ thu (Photodiode hoặc Phototransistor): Nhận tia hồng ngoại phản xạ từ vật cản.

Mạch xử lý tín hiệu: Bao gồm IC555 và các linh kiện điều chỉnh như biến trở

Biến trở điều chỉnh: Hai chiết áp trên mạch cho phép chỉnh độ nhạy và khoảng cách phát hiện

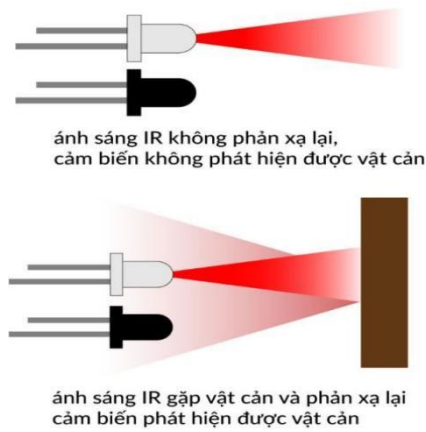
2.4.3. Nguyên lý hoạt động

Phát tia hồng ngoại: LED hồng ngoại phát ra tia hồng ngoại liên tục theo xung với tần số cụ thể.

Phản xạ từ vật cản: Khi có vật cản (bề mặt phản xạ) trong phạm vi phát hiện, tia hồng ngoại sẽ phản xạ lại và được bộ thu hồng ngoại tiếp nhận.

Xử lý tín hiệu: Bộ thu chuyển đổi tia hồng ngoại phản xạ thành tín hiệu điện. Tín hiệu này được đưa vào mạch để khuếch đại và xử lý. Nếu cường độ tín hiệu vượt ngưỡng (do vật cản gần), mạch sẽ xuất ra tín hiệu (mức 1 hoặc 0) tại ngõ ra TTL.

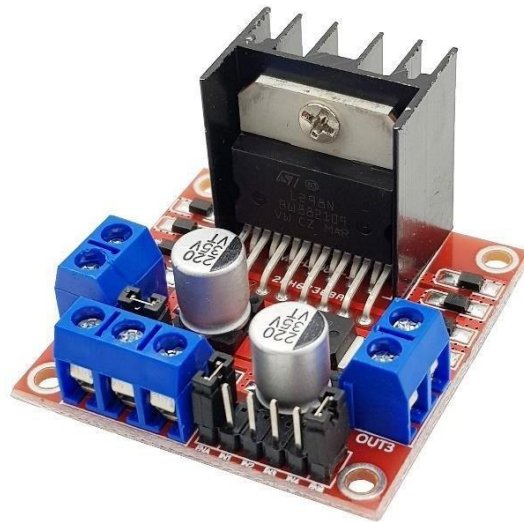
Đầu ra: Khi phát hiện vật cản, đèn báo trên cảm biến sáng lên và ngõ ra digital (thường là mức thấp hoặc cao tùy mạch) được kích hoạt để điều khiển các thiết bị khác (như robot, hệ thống tự động).



Hình 2.7. Nguyên lý hoạt động cảm biến hồng ngoại

2.5. Module điều khiển động cơ L298N

2.5.1. Giới thiệu



Hình 2.8. Module điều khiển động cơ L298N

Mạch điều khiển động cơ L298N là một trong những giải pháp hàng đầu và đa năng nhất cho việc điều khiển động cơ DC trong các ứng dụng điện tử và tự động hóa – từ quy mô nhỏ đến các ứng dụng phức tạp hơn. Với khả năng điều khiển hai động cơ DC độc lập, mạch này cung cấp sự linh hoạt cho các dự án. Nhờ vào khả năng hoạt động ổn định và dễ tích hợp, L298N trở thành giải pháp lý tưởng cho cả người mới bắt đầu lẫn những người phát triển hệ thống chuyên sâu.

Khả năng điều khiển đa dạng: mạch L298N cho phép điều khiển hai động cơ DC độc lập hoặc một động cơ bước đôi với tính năng chuyển đổi dễ dàng.

Điều này cung cấp sự linh hoạt cho nhiều ứng dụng, từ các robot đến hệ thống điều khiển tự động.

Tích hợp bảo vệ: mạch được thiết kế với các tính năng bảo vệ như chống ngược dòng và chống quá nhiệt, giúp bảo vệ cả mạch điều khiển và động cơ khỏi sự cố do điều kiện vận hành không mong muốn

Đơn giản và dễ sử dụng: với thiết đơn giản và dễ hiểu, mạch L298N thích hợp cho đa số người dùng. Việc kết nối và cấu hình mạch được thực hiện một cách dễ dàng, giúp tiết kiệm thời gian và công sức.

Tích hợp điều khiển tốc độ: mạch L298N cho phép điều khiển tốc độ của động cơ DC thông qua PWM (Pulse Width Modulation), qua đó cho phép điều chỉnh mức công suất đầu ra và tốc độ quay của động cơ một cách linh hoạt và chính xác.

Sử dụng rộng rãi: Từ các dự án DIY đơn giản đến các ứng dụng công nghiệp phức tạp, mạch điều khiển động cơ L298N là một sự lựa chọn đáng tin cậy cho việc điều khiển động cơ DC.

2.5.2. Thông số kỹ thuật và sơ đồ chân

Thông số kỹ thuật:

Điện áp điều khiển: 5 V - 12 V

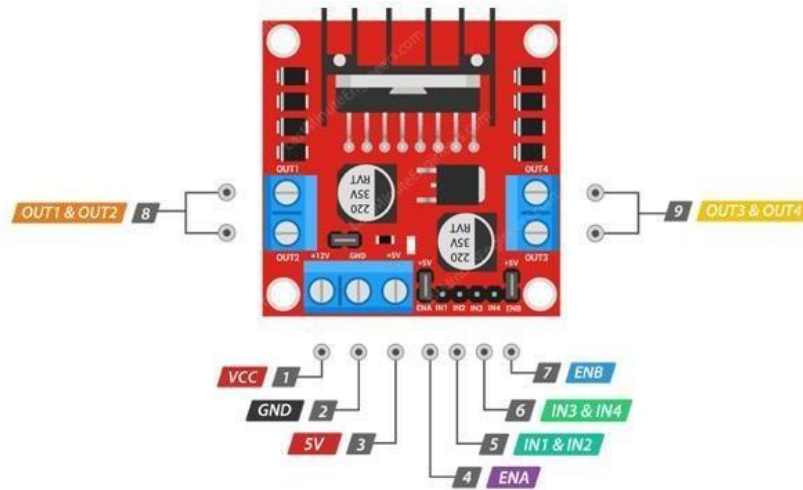
Dòng tối đa cho mỗi cầu H: 2A

Điện áp của tín hiệu điều khiển: 5 V - 7

Dòng của tín hiệu điều khiển: 0 - 36mA

Công suất hao phí: 20W (khi nhiệt độ $T = 75\text{ }^{\circ}\text{C}$)

Sơ đồ chân:



Hình 2.9. Sơ đồ chân Module L298N

VCC:

Chức năng: Cung cấp nguồn cho mạch L298N và là nguồn động lực cho động cơ.

Kết nối: Kết nối với nguồn điện dương (+).

5V:

Chức năng: Cung cấp nguồn 5V cho vi điều khiển Arduino khi Jumper 5V Enable.

Kết nối: Kết nối với nguồn điện dương (+) khi sử dụng.

GND:

Chức năng: Chân cung cấp điện âm (-) cho mạch.

Kết nối: Kết nối với nguồn điện âm (-).

ENA, ENB:

Chức năng: Cấp xung PWM để điều khiển tốc độ của động cơ.

Kết nối: Khi muốn điều khiển tốc độ, rút jumper ra và cắm chân PWM của Arduino vào.

Khi không sử dụng điều khiển tốc độ, không kết nối.

IN1, IN2:

Chức năng: Nhận tín hiệu từ vi điều khiển để điều khiển chiều và tốc độ của động cơ 1.

Kết nối: IN1: Điều khiển chiều quay của động cơ 1.

IN2: Điều khiển tốc độ của động cơ 1.

IN3, IN4:

Chức năng: Nhận tín hiệu từ vi điều khiển để điều khiển chiều và tốc độ của động cơ 2.

Kết nối: IN3: Điều khiển chiều quay của động cơ 2.

IN4: Điều khiển tốc độ của động cơ 2.

OUT1, OUT2, OUT3, OUT4:

Chức năng: Chân ngõ ra điều khiển động cơ.

Kết nối: OUT1, OUT2: Kết nối với động cơ 1.

OUT3, OUT4: Kết nối với động cơ 2.

2.5.3. Nguyên lý hoạt động

Điều khiển chiều quay thuận:

Đóng công tắc S1 và S4, mở công tắc S2 và S3.

Dòng điện sẽ chạy từ nguồn VCC qua công tắc S1, đi qua động cơ, qua công tắc S4 và trở về mát (MASS).

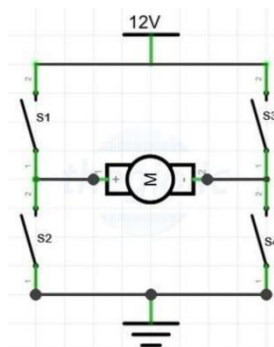
Động cơ sẽ quay theo chiều thuận.

Điều khiển chiều quay ngược:

Đóng công tắc S2 và S3, mở công tắc S1 và S4.

Dòng điện sẽ chạy từ nguồn VCC qua công tắc S3, đi qua động cơ, qua công tắc S2 và trở về mát (MASS).

Động cơ sẽ quay theo chiều ngược lại



Hình 2.10. Mạch cầu H

2.6. Động cơ giảm tốc V1

2.6.1. Giới thiệu



Hình 2.11. Động cơ giảm tốc

Động cơ DC giảm tốc màu vàng là loại động cơ phổ biến, thường được ứng dụng trong các dự án chế tạo robot và mô hình DIY. Sản phẩm đặc biệt phù hợp với học sinh, sinh viên trong quá trình học tập và nghiên cứu kỹ thuật. Động cơ hoạt động ổn định trong dải điện áp từ 3V đến 9V, được thiết kế với hai trục truyền động. Tốc độ quay của động cơ đạt khoảng 125 vòng/phút ở mức điện áp 3VDC và khoảng 208 vòng/phút khi cấp điện 5VDC

2.6.2. Thông số kỹ thuật

Điện áp: Sử dụng điện áp và dòng điện khuyến nghị từ nhà sản xuất, 3- 12VDC, nên ở dải điện áp (6-8VDC)

Dòng: 70mA (250mA Max) ở 3V

Mô men xoắn: Thường thì mô men xoắn được cung cấp sẵn từ thông số kỹ thuật của động cơ, $\text{Moment} = \text{lực} \times \text{bán kính} = 0,08 \text{ Nm}$.

Tỷ số truyền: 1:48

Tốc độ: Bảng số vòng quay/ thời gian

125 rpm 3VDC (bánh 66mm: 26rpm)

208 rpm 5VDC (bánh 66mm: 44rpm)

Kích thước: 64 * 19 * 22.6mm (L * W * H)

Trọng lượng: 27g

2.6.3. Nguyên lý hoạt động

Động cơ DC: Tạo ra chuyển động quay tốc độ cao nhưng mô-men xoắn thấp dựa trên nguyên lý giữa từ trường và dòng điện.

Hộp giảm tốc: Một hệ thống các bánh răng (thường là bánh răng trụ hoặc bánh răng xoắn) được gắn vào trục động cơ DC. Hệ thống này làm giảm tốc độ quay của trục đầu ra và đồng thời tăng mô-men xoắn theo tỷ lệ tương ứng. Tỷ số truyền (gear ratio) của hộp giảm tốc quyết định mức độ giảm tốc và tăng mô-men xoắn này (ví dụ: tỷ số 1:48 có nghĩa là tốc độ đầu ra sẽ chậm hơn 48 lần so với tốc độ động cơ, nhưng mô-men xoắn đầu ra sẽ lớn hơn 48 lần, bỏ qua hiệu suất).

Ưu điểm:

Mô-men xoắn cao: Khả năng tạo ra lực xoắn lớn ở tốc độ thấp, phù hợp để di chuyển tải nặng hoặc thực hiện các tác vụ đòi hỏi lực kéo.

Kiểm soát tốc độ tốt: Dễ dàng điều khiển tốc độ quay bằng cách điều chỉnh điện áp cấp cho động cơ DC.

Kích thước nhỏ gọn: Tích hợp cả động cơ và hộp giảm tốc giúp tiết kiệm không gian trong thiết kế.

Đa dạng tỷ số truyền: Có nhiều loại động cơ giảm tốc với các tỷ số truyền khác nhau, cho phép lựa chọn phù hợp với yêu cầu cụ thể của ứng dụng.

Ứng dụng rộng rãi: Được sử dụng trong nhiều lĩnh vực như robot, tự động hóa, đồ chơi, thiết bị gia dụng và công nghiệp

2.7. Module hạ áp DC LM2596

2.7.1. Giới thiệu



Hình 2.12. Module hạ áp DC LM2596

Mạch hạ áp DC LM2596 là một thành phần quan trọng trong các hệ thống điện tử, có chức năng chính là chuyển đổi và điều chỉnh điện áp từ nguồn có điện áp cao xuống mức thấp hơn, phù hợp và ổn định để cấp cho các thiết bị hoặc linh kiện khác trong mạch.

Các chức năng chính của LM2596 bao gồm:

Chuyển đổi điện áp: LM2596 có khả năng hạ điện áp đầu vào xuống mức thấp hơn theo yêu cầu, thích hợp để cung cấp nguồn cho các thiết bị yêu cầu điện áp thấp khác nhau.

Bảo vệ quá nhiệt: Một số phiên bản của mạch này được tích hợp IC có tính năng bảo vệ khi nhiệt độ vượt ngưỡng an toàn, giúp đảm bảo độ bền và an toàn cho toàn mạch.

Ổn định điện áp: Mạch giúp duy trì mức điện áp đầu ra ổn định ngay cả khi điện áp đầu vào dao động, nhờ đó bảo vệ các linh kiện nhạy cảm khỏi ảnh hưởng của sự thay đổi nguồn điện.

Hiệu suất cao: Với hiệu suất chuyển đổi tốt, LM2596 giúp tiết kiệm năng lượng, giảm tổn hao nhiệt và đảm bảo cung cấp điện ổn định, đáng tin cậy cho toàn bộ hệ thống.

2.7.2. Thông số kỹ thuật

Điện áp đầu vào: Từ 3V đến 30V.

Điện áp đầu ra: Điều chỉnh được trong khoảng 1.5V đến 30V.

Dòng đáp ứng tối đa: 3A.

Hiệu suất: 92%

Công suất: 15W

Kích thước: 45 x 20 x 14 mm

2.7.3. Nguyên lý hoạt động

Mạch hạ áp DC sử dụng IC LM2596 hoạt động theo nguyên lý chuyển mạch xung (buck converter), cho phép chuyển đổi điện áp DC cao xuống một mức điện áp thấp hơn một cách linh hoạt. Khi hoạt động, LM2596 bật tắt nhanh chóng một transistor bên trong với tần số khoảng 150kHz. Trong chu kỳ bật, dòng điện đi qua cuộn cảm và tích năng lượng dưới dạng từ trường. Khi transistor tắt, cuộn cảm tiếp tục duy trì dòng điện bằng cách giải phóng năng lượng đã tích, cung cấp điện áp ổn định cho tải. Diode Schottky trong mạch cho phép dòng điện tiếp tục lưu thông khi transistor tắt, trong khi các tụ điện đầu vào và đầu ra giúp lọc nhiễu và làm mượt điện áp.

IC này có thể hoạt động với điện áp đầu vào từ 3V đến 30V, cho điện áp đầu ra điều chỉnh được trong khoảng từ 1.5V đến 30V, với dòng tối đa khoảng 3A. Nhờ hiệu suất cao đến 92%, mạch LM2596 được ứng dụng rộng rãi trong các dự án điện tử như cung cấp nguồn cho vi điều khiển, thiết bị nhúng, robot và các mô hình DIY.

2.8. Pin Li-ion 18650

2.8.1. Giới thiệu



Hình 2.13. Pin 18650

Pin 18650 là một loại pin lithium-ion thông dụng phổ biến trong tất cả thiết bị điện tử và các ứng dụng công nghiệp. Cùng với thiết kế nhỏ gọn cùng khả năng sạc và

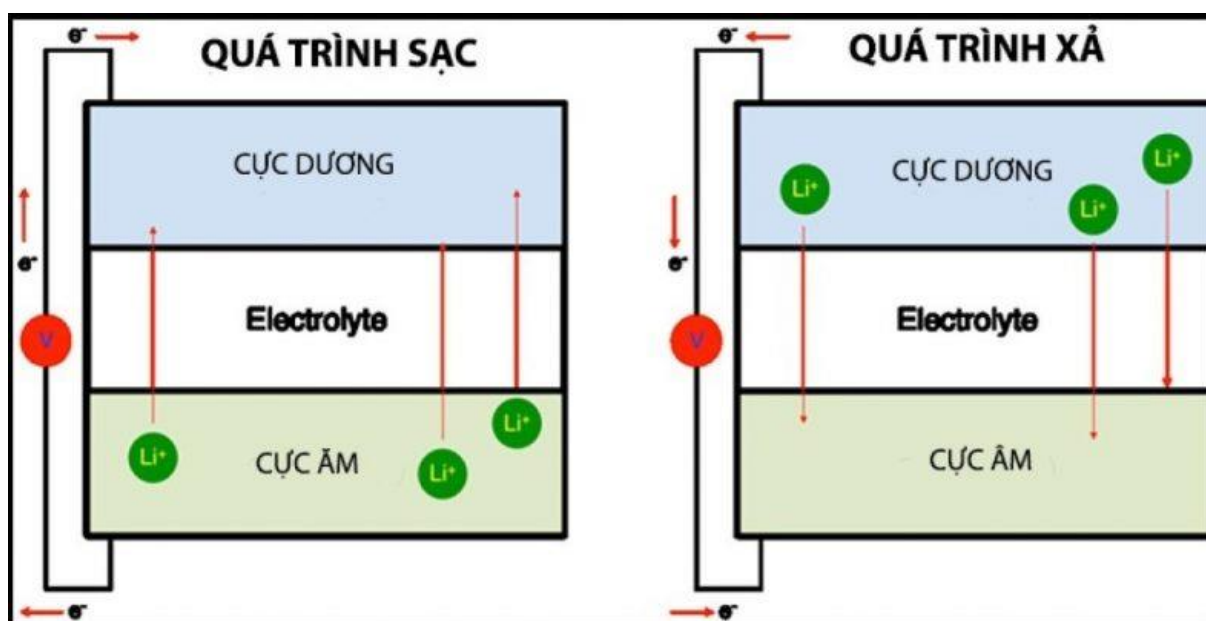
xả hiệu quả, loại pin này đã trở thành lựa chọn đáng tin cậy cho nhiều thiết bị di động cũng như các hệ thống lưu trữ năng lượng.

2.8.2. Cấu tạo

Pin 18650 có hình dáng hình trụ với kích thước chuẩn là đường kính 18mm và chiều cao 65mm, do đó được gọi là 18650. Bên trong, pin bao gồm các tế bào lithium-ion được xếp chồng lên nhau trong một ống kim loại nhỏ. Mỗi tế bào lithium-ion bao gồm một điện cực dương (anode) là graphite, một điện cực âm (cathode) là oxide kim loại và một chất điện phân (electrolyte) dẻo như gel polymer.

2.8.3. Nguyên lý hoạt động

Trong pin Li-ion, cực dương và cực âm đóng vai trò như các vật liệu phản ứng trong quá trình điện hóa. Dung dịch điện phân nằm giữa hai điện cực tạo thành môi trường dẫn, cho phép các ion lithium (Li^+) di chuyển qua lại giữa cực dương và cực âm. Khi pin hoạt động, dòng điện được sinh ra và di chuyển qua mạch ngoài, trong khi các ion lithium di chuyển bên trong pin thông qua chất điện phân. Quá trình này được thể hiện rõ trong hai giai đoạn: sạc và xả. Cụ thể:



Hình 2.14. Nguyên lý hoạt động pin 18650

2.9. Bánh xe

2.9.1. Giới thiệu



Hình 2.15. Bánh xe

Các rãnh vân trên lốp được thiết kế tối ưu nhằm tăng độ bám và ma sát với mặt đường. Lốp được làm từ chất liệu có độ đàn hồi cao, giúp bánh giữ được hình dạng ổn định khi chịu tải và tăng khả năng tiếp xúc hiệu quả với mặt đường.

2.9.2. Thông số kỹ thuật

Chất liệu: Nhựa.

Đường kính: 66mm.

Đồ dày: 6.5mm.

Chiều rộng: 27mm.

Trục phi: 5mm, độ rộng khuyết là 3.66mm.

Màu sắc: đen và vàng.

2.10. Các linh kiện khác

2.10.1. Công tắc



Hình 2.16. Công tắc bấm bẻ

Công tắc 15x20mm (đỏ) là sản phẩm bằng nhựa, có vỏ màu đen, dùng để đóng hoặc cắt dòng điện – tín hiệu sử dụng trong các thiết bị điện tử, mạch điện công suất nhỏ.

Thông số kỹ thuật:

Điện áp: 6A/250V hoặc 10A/125 V

Chất liệu: đồng, nhựa

Kích thước: 21*15*18.1 (mm)

Tiếp điểm: NO

2.10.2. Công tắc hành trình



Hình 2.17. Công tắc hành trình

Công tắc hành trình được dùng để giới hạn hành trình các bộ phận chuyển động trong một cơ cấu hay một hệ thống.

Thông số kỹ thuật:

Điện áp tối đa: 125VAC

Tiếp điểm: 1NO, 1NC

Dòng điện tối đa: 1A

Nhiệt độ làm việc: -25 - 85°C

2.10.3. Giắc đực-cái DC



Hình 2.18. jack cắm DC

Dùng để làm jack cấp nguồn DC cho các mạch điện tử hoặc các ứng dụng tương đương

Thông số kỹ thuật:

Kích thước: 5.5x2.1mm

Điện áp: DC

Số chân: 3

2.10.4. Nút nhấn tròn



Hình 2.19. Nút nhấn DC

Nút nhấn thường được sử dụng để đóng cắt mạch điện hoặc để điều khiển tín hiệu, ...

Thông số kỹ thuật:

Kích thước lỗ lắp: 12mm

Dòng định mức: 3A/125V

2.10.5. Vôn kế điện tử



Hình 2.20. Vôn kế điện tử 0.28Inch

Volt kế này có kích thước khá nhỏ và được trang bị ba dây để dễ dàng đo, hiển thị điện áp acquy, pin, và các sản phẩm khác....

Với chức năng bảo vệ kết nối ngược, dây sẽ không bị cháy nếu bị đảo ngược.

Thông số kỹ thuật:

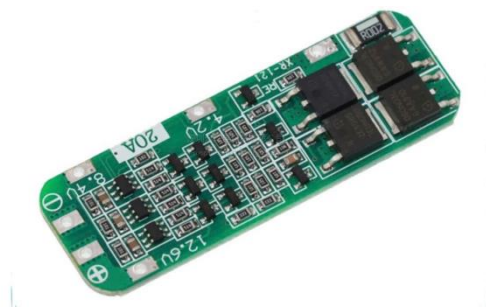
Điện áp: 4,5-30V

Phạm vi đo :4,5V đến 30V

Kích thước: 35x23x20mm

Thời gian lấy mẫu: 200ms

2.10.6. Mạch bảo vệ và sạc pin 3S



Hình 2.21. Mạch bảo vệ và sạc pin 3S

Mạch sạc và bảo vệ pin 3S 20A (có chức năng phục hồi tự động – Auto Recovery) được thiết kế để sạc và xả cho các loại pin lithium như 18650, 26650 hoặc pin lithium polymer được kết nối nối tiếp 3 cell. Mạch cho phép dòng xả tối đa lên đến 20A, phù hợp với các ứng dụng cần công suất cao.

Thiết bị tích hợp các chức năng bảo vệ an toàn như: tự động ngắt khi pin đầy, chống chập mạch, bảo vệ quá áp và ngăn ngừa hiện tượng xả sâu khi điện áp xuống dưới mức giới hạn cho phép. Tính năng phục hồi tự động giúp mạch hoạt động ổn định trở lại sau khi lỗi.

Thông số kỹ thuật:

Loại pin sử dụng: 18650, 26650, pin lithium polymer

Điện áp Sạc: 12.6V – 13.0V

Dòng xả đỉnh: 20A

Dòng xả liên tục: 10A

Kích thước: 59 * 20 * 3.4mm

2.11. Phần mềm Arduino IDE

2.11.1. Giới thiệu

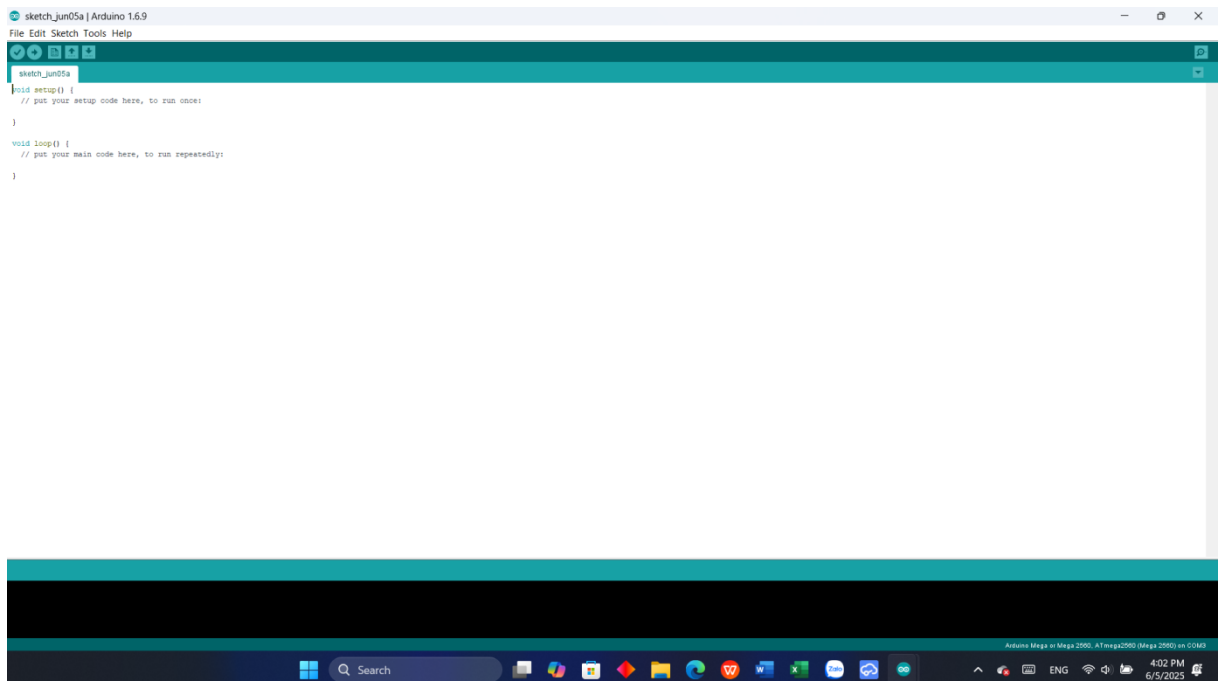
Arduino đã trở thành một trong những nền tảng phổ biến nhất cho việc phát triển các dự án IoT, robot và nhiều ứng dụng khác nhau. Sự tiện lợi ngày càng được chứng tỏ theo thời gian với vô vàn các ứng dụng mở (open source) độc đáo được chia sẻ rộng rãi trong cộng đồng người dùng.

Ngôn ngữ lập trình Arduino là một ngôn ngữ đơn giản và cơ bản để lập trình với các vi điều khiển Arduino. Ngôn ngữ lập trình Arduino dựa trên ngôn ngữ C/C++, nhưng đã đơn giản hóa để phù hợp với vi điều khiển Arduino và các tác vụ điều khiển phần cứng. Để lập trình cũng như gửi lệnh và nhập tín hiệu từ mạch Arduino, nhóm phát triển dự án này đã cung cấp đến cho người dùng một môi trường lập trình Arduino được gọi là Arduino IDE (Intergrated Development Environment)



Hình 2.22. Phần mềm arduino IDE

2.11.2. Giao diện phần mềm








Hình 2.23. Giao diện phần mềm Arduino IDE

Về cơ bản giao diện của Arduino IDE được chia thành 3 phần:

Vùng lệnh (thanh menu):

Bao gồm các nút lệnh menu (File, Edit, Sketch, Tools, Help). Phía dưới là các icon cho phép sử dụng nhanh các chức năng thường dùng của IDE được miêu tả như sau:

Icon	Chức năng
	Biên dịch chương trình đang soạn thảo để kiểm tra các lỗi lập trình.
	Biên dịch và upload chương trình đang soạn thảo.
	Mở một trang soạn thảo mới.
	Mở các chương trình đã lưu.
	Lưu chương trình đang soạn.
	Mở cửa sổ Serial Monitor để gửi và nhận dữ liệu giữa máy tính và board Arduino.

Hình 2.24: Vùng lệnh

Vùng viết chương trình:

Nơi chương trình lập trình chính được viết ở đây.

Vùng thông báo:

Những thông báo từ IDE sẽ được hiển thị tại đây. Để ý rằng góc dưới cùng bên phải hiển thị loại board Arduino và cổng COM được sử dụng. Luôn chú ý tới mục này bởi nếu chọn sai loại board hoặc cổng COM sẽ không thể upload được code.

2.11.3. Các lệnh thường dùng

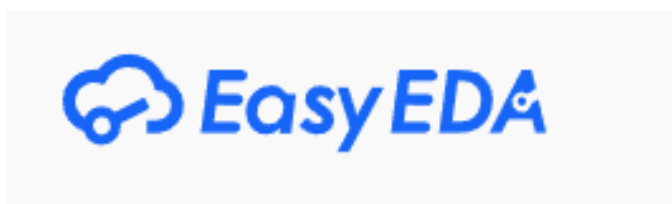
Bảng 2.5. Các lệnh thường dùng trong Arduino IDE

Nhóm lệnh	Lệnh	Chức năng
Cấu trúc chương trình	setup()	Hàm khởi tạo, chạy một lần khi bắt đầu
	loop()	Hàm chính, lặp vô hạn sau setup()
I/O (Vào/Ra)	pinMode(pin, mode)	Thiết lập chân là INPUT, OUTPUT hoặc INPUT_PULLUP
	digitalWrite(pin, value)	Ghi giá trị HIGH hoặc LOW ra chân số
	digitalRead(pin)	Đọc giá trị HIGH/LOW từ chân số
	analogWrite(pin, value)	Ghi tín hiệu PWM (0–255) ra chân hỗ trợ PWM
	analogRead(pin)	Đọc tín hiệu analog (0–1023) từ chân analog

Điều khiển chương trình	if, else	Cấu trúc điều kiện
	for, while	Vòng lặp
	break, continue	Thoát vòng lặp hoặc bỏ qua vòng lặp hiện tại
	return	Trả về giá trị và kết thúc hàm
Thời gian	delay(ms)	Dừng chương trình trong ms mili giây
	delayMicroseconds(us)	Dừng chương trình trong us micro giây
Serial (Giao tiếp)	Serial.begin(baud_rate)	Khởi động giao tiếp Serial
	Serial.print(data)	In dữ liệu ra Serial không xuống dòng
	Serial.println(data)	In dữ liệu ra Serial và xuống dòng
Thư viện	Servo.attach(pin)	Gắn servo vào chân
	Servo.write(angle)	Gửi góc tới servo (0–180 độ)
	lcd.begin(cols, rows)	Khởi tạo LCD
	lcd.print("text")	In chữ lên LCD
	Wire.begin(), Wire.write()	Giao tiếp I2C

2.12. Phần mềm Easy EDA

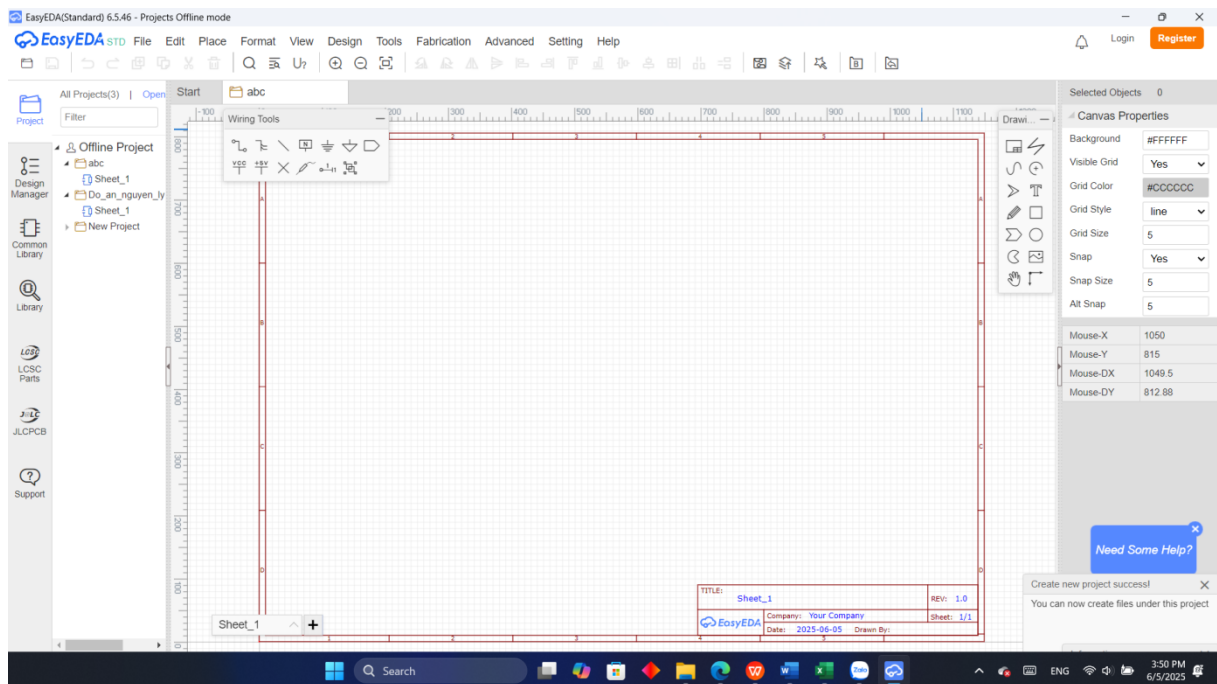
2.12.1 Giới thiệu



Hình 2.25. Phần mềm Easy EDA

EasyEDA là một nền tảng thiết kế mạch tiện lợi và thân thiện với người dùng, được phát triển nhằm hỗ trợ các kỹ sư điện tử, nhà giáo dục, sinh viên, nhà sáng chế và cộng đồng yêu thích điện tử trong việc xây dựng và chia sẻ các dự án của họ. Công cụ này tích hợp trực tiếp với thư viện linh kiện từ LCSC và dịch vụ sản xuất PCB của JLCPCB, giúp rút ngắn thời gian thiết kế, đồng thời tối ưu hóa quá trình hiện thực hóa ý tưởng thành sản phẩm hoàn chỉnh.

2.12.2 Giao diện phần mềm



Hình 2.26. Giao diện phần mềm Easy EDA

Về cơ bản giao diện của Easy EDA được chia thành 3 phần:

Thanh công cụ (Top Toolbar):

Thanh công cụ này nằm ở phía trên cùng của giao diện và cung cấp quyền truy cập nhanh đến các chức năng chính như tạo mới, lưu dự án, nhập/xuất tệp, thêm linh kiện, kiểm tra thiết kế và xuất file Gerber để sản xuất PCB.

Thanh điều hướng (Left Sidebar):

Project: Quản lý danh sách các dự án và sheet thiết kế hiện tại.

Design Manager: Cho phép theo dõi toàn bộ thành phần mạch đang sử dụng.

Library: Cung cấp các linh kiện cơ bản và tùy chỉnh.

LCSC Parts: Kết nối trực tiếp đến kho linh kiện trực tuyến của LCSC, hỗ trợ tìm kiếm linh kiện thực tế.

JLCPCB: Hỗ trợ đặt mạch in ngay sau khi thiết kế hoàn tất thông qua đối tác sản xuất PCB.

Khu vực thiết kế chính (Main Canvas):

Đây là khu vực thực hiện việc kéo thả linh kiện, nối dây và tạo sơ đồ nguyên lý hoặc bố trí mạch PCB. Khu vực này có lưới giúp căn chỉnh chính xác, viền đỏ bao quanh biểu thị khung bản vẽ kỹ thuật, tương tự như bản vẽ công nghiệp.

2.13. Phần mềm AutoCAD 2007

2.13.1 Giới thiệu

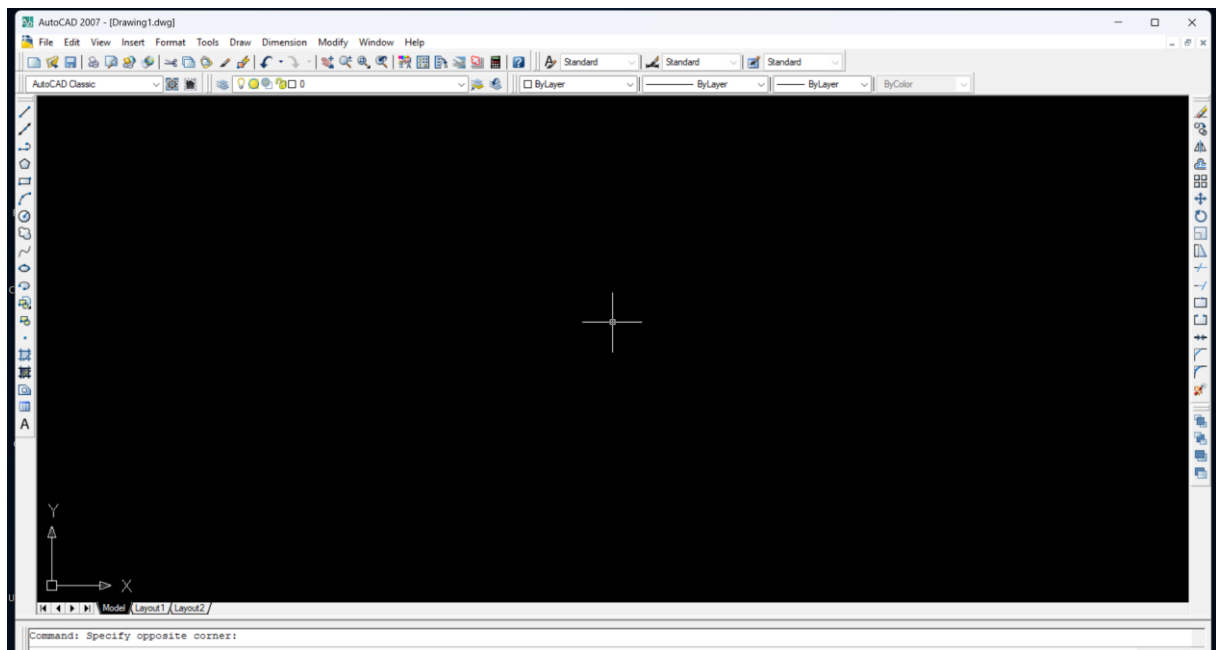


Hình 2.27. Phần mềm Auto CAD 2007

AutoCAD 2007 là phần mềm thiết kế đồ họa 2D và 3D do hãng Autodesk phát triển, được phát hành vào năm 2006. Đây được xem là một trong những phiên bản nổi bật và phổ biến nhất nhờ vào tính ổn định, giao diện thân thiện cùng hệ thống tính năng hỗ trợ hiệu quả cho công việc thiết kế kỹ thuật.

Phần mềm có giao diện trực quan, dễ làm việc, đặc biệt phù hợp với cả người mới bắt đầu và người dùng chuyên nghiệp. Các thanh công cụ và menu được bố trí hợp lý, giúp thao tác nhanh và tiết kiệm thời gian. Ngoài ra, AutoCAD 2007 hỗ trợ nhiều định dạng tệp khác nhau, tạo điều kiện thuận lợi cho việc chia sẻ và tương thích với các phần mềm kỹ thuật khác.

2.13.2 Giao diện phần mềm



Hình 2.28. Giao diện phần mềm Auto CAD 2007

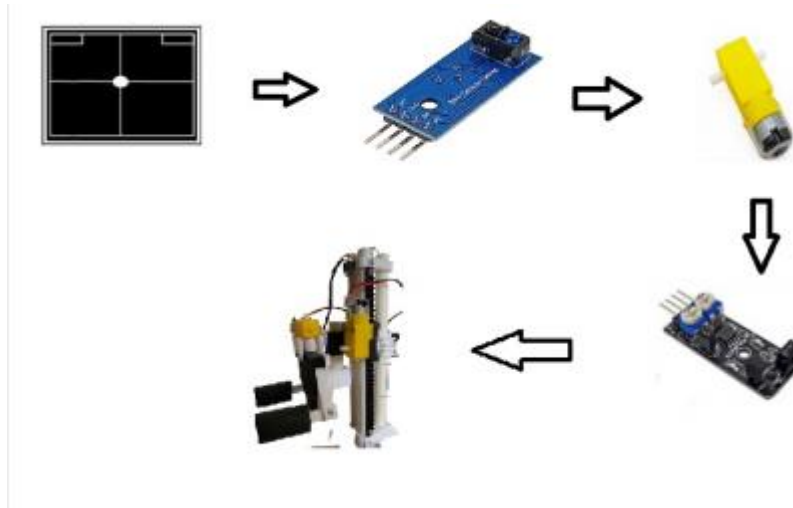
Giao diện phần mềm AutoCAD 2007 gồm nhiều thành phần chính hỗ trợ người dùng trong việc thiết kế kỹ thuật. Phía trên cùng là thanh tiêu đề, hiển thị tên phần mềm và tên bản vẽ đang làm việc. Bên dưới là thanh menu, chứa các mục lệnh chính như File, Edit, View, Draw, Dimension, ... giúp truy cập đầy đủ tính năng của phần mềm. Tiếp theo là các thanh công cụ (toolbar) như Standard, Draw, Modify, ... với các biểu tượng cho phép người dùng thao tác nhanh các lệnh cơ bản. Trung tâm màn hình là vùng bản vẽ với nền màu đen, nơi người dùng thực hiện các thao tác thiết kế, được hỗ trợ bởi con trỏ hình chữ thập và hệ trục tọa độ X, Y (UCS Icon) ở góc dưới trái.

Phía dưới vùng vẽ là dòng lệnh (Command Line) – nơi người dùng nhập các lệnh vẽ và chỉnh sửa bằng bàn phím, giúp kiểm soát chính xác quá trình thiết kế. Gần cuối giao diện là các tab chế độ làm việc gồm Model, Layout1 và Layout2, cho phép chuyển đổi giữa không gian thiết kế và không gian bố trí bản vẽ để in ấn. Ngoài ra, hai bên màn hình còn có thanh công cụ dọc, hỗ trợ vẽ hình, điều chỉnh kích thước, thu phóng bản vẽ và các thao tác phụ trợ khác. Nhờ giao diện trực quan và bố cục hợp lý, AutoCAD 2007 vẫn là công cụ hữu ích cho sinh viên, kỹ sư và kỹ thuật viên trong lĩnh vực thiết kế kỹ thuật.

CHƯƠNG 3. THIẾT KẾ VÀ THI CÔNG

3.1. Quy trình công nghệ

3.1.1. Sơ đồ công nghệ



Hình 3.1. Sơ đồ công nghệ

3.1.2. Yêu cầu bài toán

Xe robot dò line và gấp vật tự hành hoạt động như sau:

Khi các cảm biến dò line TCRT5000 phát hiện ra line sẽ phát ra các tín hiệu về Arduino Mega 2560 để xử lý và xuất ra các tín hiệu điện áp để xe có thể đi đúng line và tinh chỉnh để xe có thể chạy bám theo line.

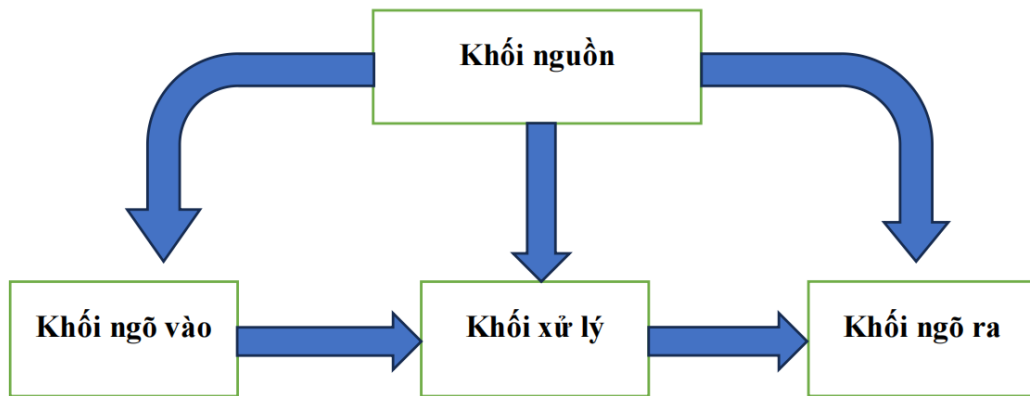
Khi cảm biến vật cản hồng ngoại HW488 phát hiện vật sẽ xuất ra tín hiệu về Arduino Mega 2560 để xử lý và đưa ra các tín hiệu điện áp cho cơ cấu trượt xuống sau đó có thể mở và gấp vật.

Khi thao tác gấp vật được hoàn thành, board Arduino sẽ tiếp tục chuyển chương trình về chương trình bám theo line để xe có thể về khu vực thả vật.

Khi cảm biến vật cản hồng ngoại phát hiện được khu vực trụ thả vật sẽ đưa ra tín hiệu về Arduino để thực hiện thả vật vào khu vực thả vật theo quy định.

Sau khi thả vật sẽ tiến hành hạ thanh trượt xuống mức nhất định và đóng kẹp lại để tránh đối thủ thả vật vào, xe dừng và chương trình kết thúc.

3.2. Sơ đồ khối



Hình 3.2. Sơ đồ khối xe robot dò line và gấp vật

3.2.1. Khối nguồn (Power source)

Robot sử dụng 3 pin 18650 mắc nối tiếp có điện áp 12.6V để cấp nguồn cho hệ thống với công suất xả 20C. Trong hệ thống nguồn pin cung cấp điện cho các phần bao gồm: board Arduino Mega 2560, 3 cảm biến dò line TCRT500, cảm biến vật cản hồng ngoại HW488 và các động cơ giảm tốc vàng ở phần bánh xe và ở phần cơ cấu gấp thông qua mạch hạ áp DC LM 2596 và mạch điều khiển động cơ L298N.

Phần điện áp của xe còn được theo dõi linh hoạt thông qua vôn kế điện tử đặt ở phía trước xe.

3.2.2. Khối ngõ vào (Input)

Các cảm biến dò line TCRT500 phát hiện line và đưa ra các tín hiệu digital, so sánh với ngưỡng nhất định để xuất ra mức điện áp tương đương cho các động cơ giảm tốc vàng thông qua board Arduino Mega 2560.

Cảm biến vật cản hồng ngoại phát hiện có vật hay không và đưa ra các tín hiệu tại chân OUT, so sánh với ngưỡng nhất định để xuất ra mức điện áp tương đương điều khiển động cơ giảm tốc vàng để tiến hành trượt lên xuống và gấp thả vật theo lệnh của Arduino Mega 2560

Các công tắc hành trình được chia ra làm:

Hành trình lên, hành trình xuống, hành trình đóng và hành trình mở kẹp sẽ giúp Arduino biết vị trí chính xác của cơ cấu gấp và đưa ra lệnh phù hợp cho cơ cấu gấp.

3.2.3. Khối xử lý (Operation)

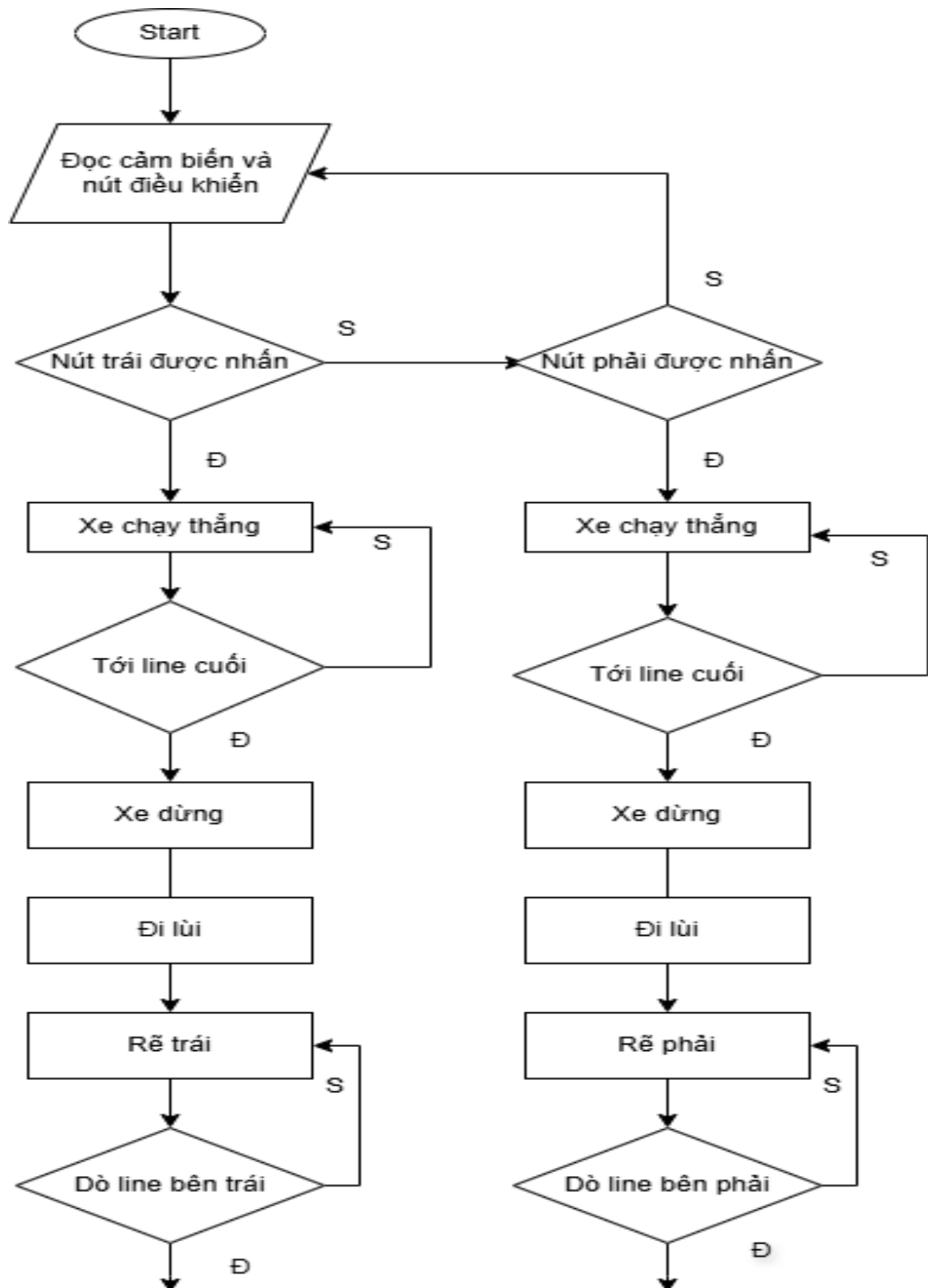
Arduino Mega 2560 sau khi nhận tín hiệu tương ứng từ các cảm biến dò line, cảm biến vật cản hồng ngoại và công tắc hành trình sẽ thực hiện xử lý và điều khiển bằng cách đưa ra các mức điện áp tương ứng để điều khiển các động cơ.

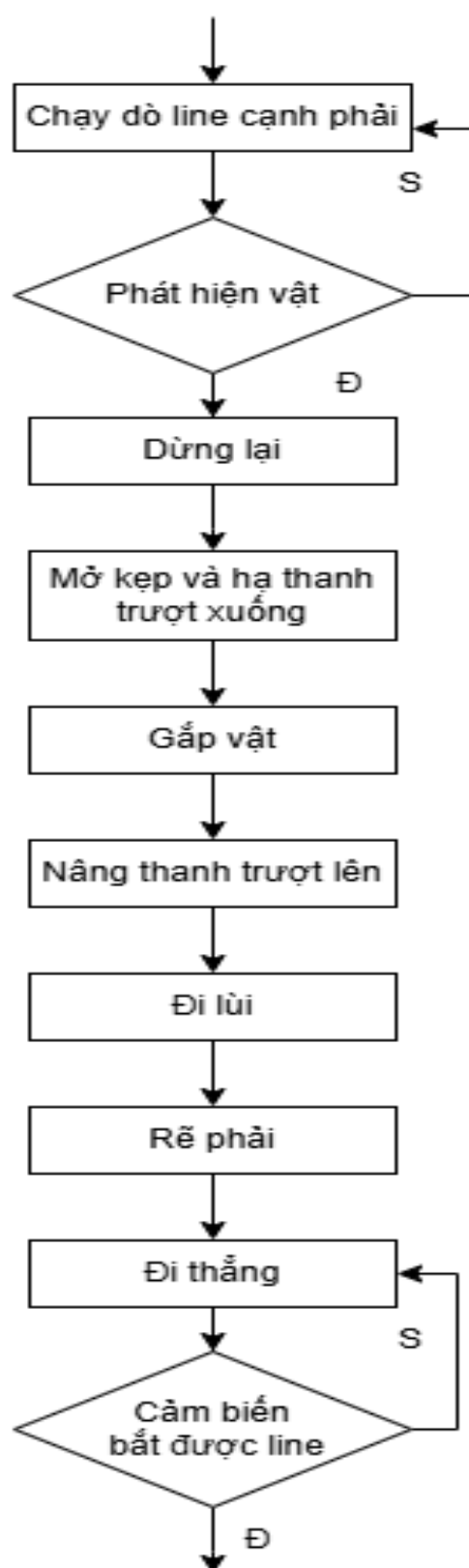
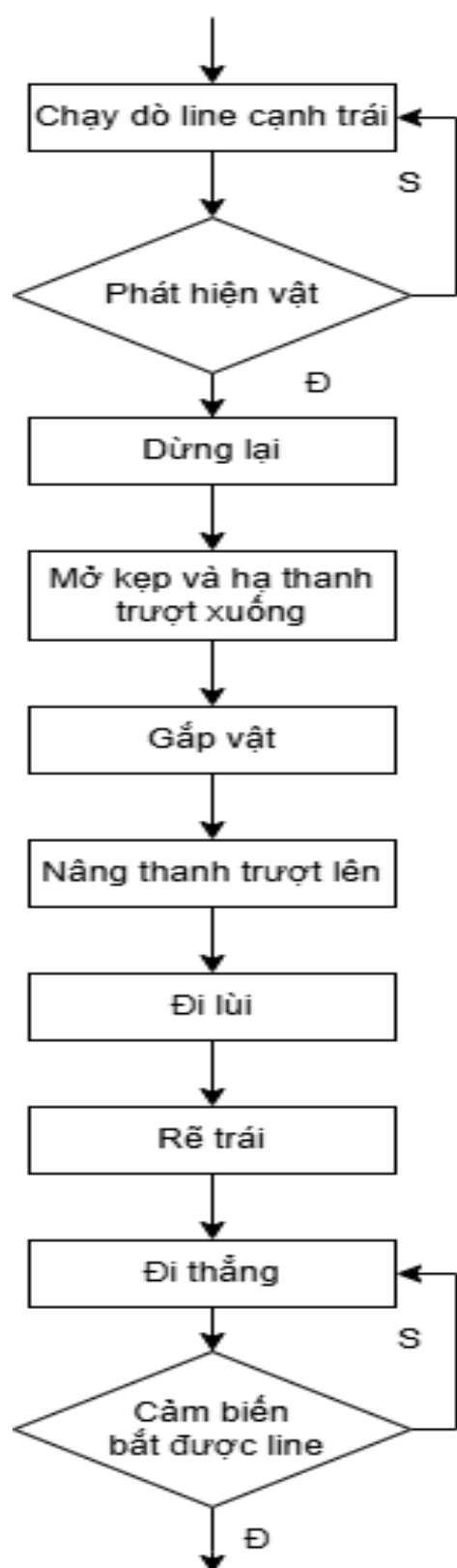
3.2.4. Khối ngõ ra (Output)

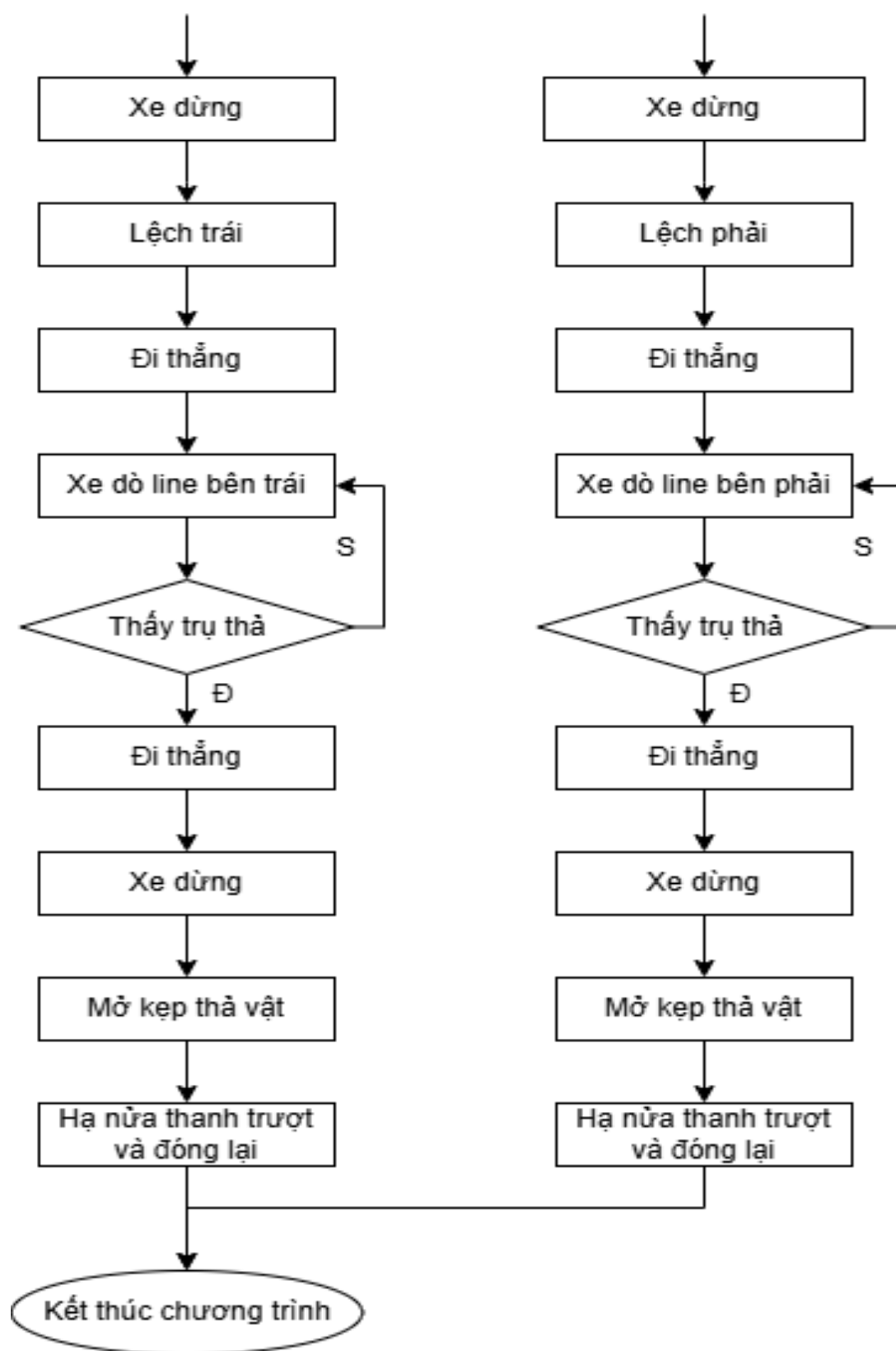
Động cơ giảm tốc vàng nhận mức tín hiệu điện áp sau khi xử lý của Arduino Mega 2560 thông qua mạch hạ áp DC LM 2596 và mạch điều khiển động cơ L298N sẽ giúp xe di chuyển và bắt theo line.

Động cơ giảm tốc vàng ở phần cơ cấu gấp vật nhận mức điện áp sau khi xử lý của Arduino Mega 2560 thông qua mạch điều khiển động cơ L298N và qua mạch hạ áp DC LM 2596 sẽ giúp cơ cấu trượt lên xuống và gấp vật linh hoạt và hợp lý.

3.3. Lưu đồ giải thuật







Hình 3.3. Lưu đồ giải thuật

3.3.1. Giải thích lưu đồ giải thuật

Trường hợp 1: Đặt xe ở phần sân A (xe đặt ở phần sân bên trái)

Gạt công tắc mở nguồn cho robot dò line và cơ cấu gấp. Nhấn nút ở phần sân bên trái khi đó xe chạy thẳng tới line cuối. Khi đó 3 cảm biến dò line đều dò thấy line trắng ($CB1 == 0 \ \&\& \ CB2 == 0 \ \&\& \ CB3 == 0$) xe dừng 300ms sau đó đi lùi lại 130s để bắt đúng line. Xe rẽ trái trong 350ms và bắt đầu dò line bên trái khi nào mà cảm biến vật cản phát hiện (có vật) thì xe dừng lại 300ms.

Sau đó xe sẽ kiểm tra trạng thái sao cho thanh trượt được nâng lên, mở kẹp và khi kiểm tra xong sẽ hạ thanh trượt xuống và đóng kẹp gấp vật. Sau khi gấp xong sẽ trượt thanh trượt lên, mang vật lên cao và xe đi lùi 450ms để thoát khỏi line ô chứa vật, xe rẽ trái 500ms và đi thẳng để tiến đến line giữa.

Xe đi thẳng đến khi 1 trong 3 cảm biến bắt được line thì xe sẽ lùi lại 400ms, lệch trái 700ms và đi thẳng 100ms để bắt được line và chạy dò line trái. Khi cảm biến vật cản phát hiện trụ thả thì xe sẽ đi thẳng thêm 100ms để xe chắc chắn sát trụ và dừng lại. Cơ cấu bắt đầu mở kẹp ra và thả vật xuống, sau đó sẽ hạ thanh trượt xuống 200ms và đóng kẹp lại để đối thủ không thể thả vật vào và lấy thêm điểm.

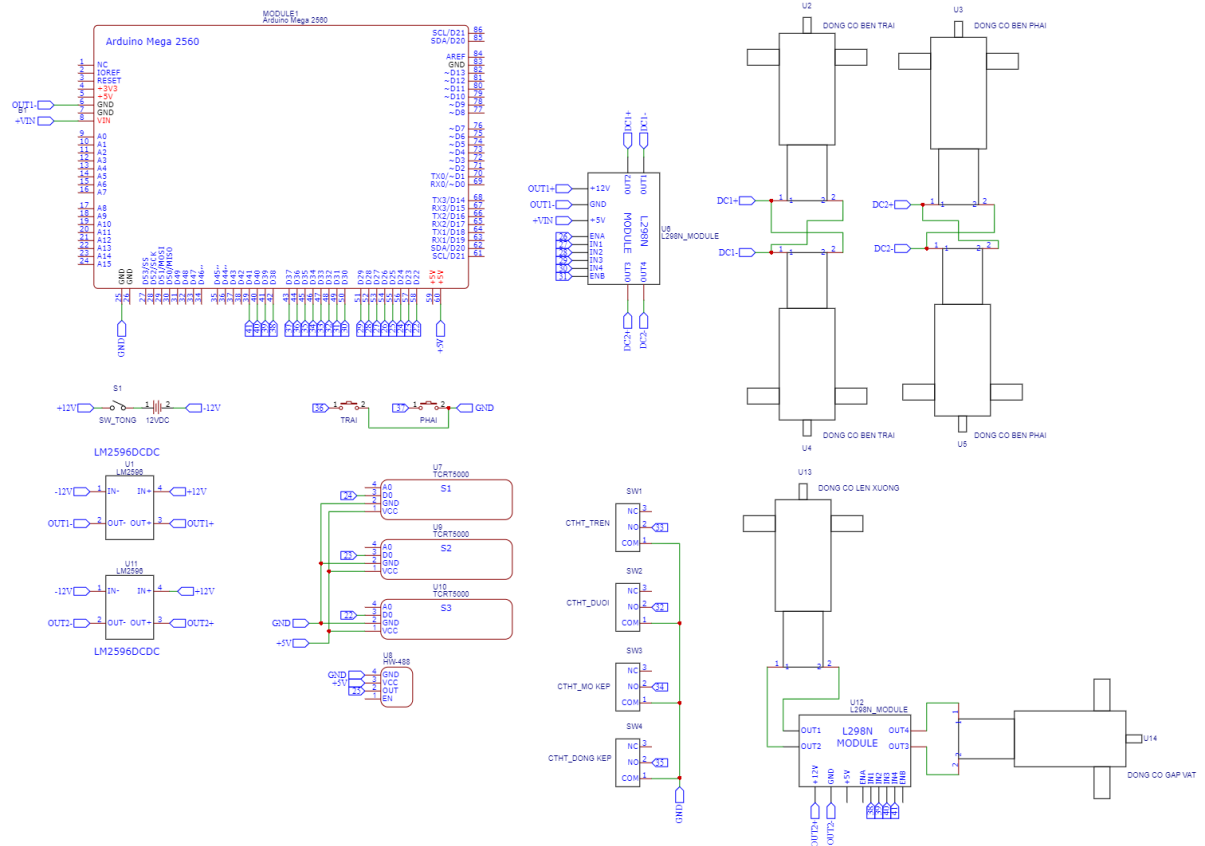
Trường hợp 2: Đặt xe ở phần sân B (xe đặt ở phần sân bên phải)

Gạt công tắc mở nguồn cho robot dò line và cơ cấu gấp. Nhấn nút ở phần sân bên phải khi đó xe chạy thẳng tới line cuối. Khi đó 3 cảm biến dò line đều dò thấy line trắng ($CB1 == 0 \ \&\& \ CB2 == 0 \ \&\& \ CB3 == 0$) xe dừng 300ms sau đó đi lùi lại 130s để bắt đúng line. Xe rẽ phải trong 350ms và bắt đầu dò line bên phải khi nào mà cảm biến vật cản phát hiện (có vật) thì xe dừng lại 300ms.

Sau đó xe sẽ kiểm tra trạng thái sao cho thanh trượt được nâng lên, mở kẹp và khi kiểm tra xong sẽ hạ thanh trượt xuống và đóng kẹp gấp vật. Sau khi gấp xong sẽ trượt thanh trượt lên, mang vật lên cao và xe đi lùi 350ms để thoát khỏi line ô chứa vật, xe rẽ phải 450ms và đi thẳng để tiến đến line giữa.

Xe đi thẳng đến khi 1 trong 3 cảm biến bắt được line thì xe sẽ lùi lại 320ms, lệch phải 700ms và đi thẳng 100ms để bắt được line và chạy dò line phải. Khi cảm biến vật cản phát hiện trụ thả thì xe sẽ đi thẳng thêm 100ms để xe chắc chắn sát trụ và dừng lại. Cơ cấu bắt đầu mở kẹp ra và thả vật xuống, sau đó sẽ hạ thanh trượt xuống 200ms và đóng kẹp lại để đối thủ không thể thả vật vào và lấy thêm điểm.

3.4. Sơ đồ đấu dây



Hình 3.4. Sơ đồ đấu dây

Chi tiết về sơ đồ nối dây:

Khối xử lý:

Thứ tự các dây tín hiệu kết nối với Board Arduino Mega 2560

Bảng 3.1. Chân nối với Arduino

Số chân	Chức năng
22	Dò line 3
23	Dò line 2
24	Dò line 1
25	Cảm biến vật cản
26	EnA1
27	In1
28	In2
29	In3
30	In4
31	EnB1
32	CTHT dưới
33	CTHT trên
34	CTHT mở
35	CTHT đóng
36	Nút nhấn trái

37	Nút nhấn phải
38	in1
39	in2
40	in3
41	in4

Arduino được cấp nguồn bởi nguồn 5v từ Module L298N thông qua jack cắm DC, ngoài các chân tín hiệu điều khiển trên thì Arduino còn cung cấp nguồn và GND cho các cảm biến, công tắc hành trình và nút nhấn.

Khởi nguồn:

3 pin 18650 được mắc nối tiếp thành 12.6v là nguồn chính cung cấp cho toàn bộ robot. Các pin được mắc qua mạch bảo vệ và sạc pin 3s với công suất 20A. Từ đó nguồn sẽ qua công tắc tổng và tiến hành cấp nguồn song song đồng thời cho 2 module giảm áp DC LM2596 để xuống còn khoảng 7v để phù hợp hơn với điều kiện xe chạy thực tế.

Khởi cảm biến:

Cảm biến dò line: 3 cảm biến được đánh dấu lần lượt là S1, S2, S3 theo thứ tự từ phải qua trái. Mỗi cảm biến đảm nhận chức năng dò line riêng biệt (S3 dò line phải, S2 dò line giữa, S1 dò line trái) để xe có thể chạy và bắt đúng theo đường line cố định. Ngõ ra cảm biến được lấy từ chân Digital (D0) để cấp tín hiệu về cho Arduino.

Cảm biến hồng ngoại phát hiện vật cản: Được cấp nguồn từ Arduino và nhóm đã điều chỉnh 2 biến trở trên cảm biến sao cho phù hợp và bắt được vật chính xác nhất để cơ cấu gấp vật. Tín hiệu được lấy từ chân OUT của cảm biến để đưa về Arduino.

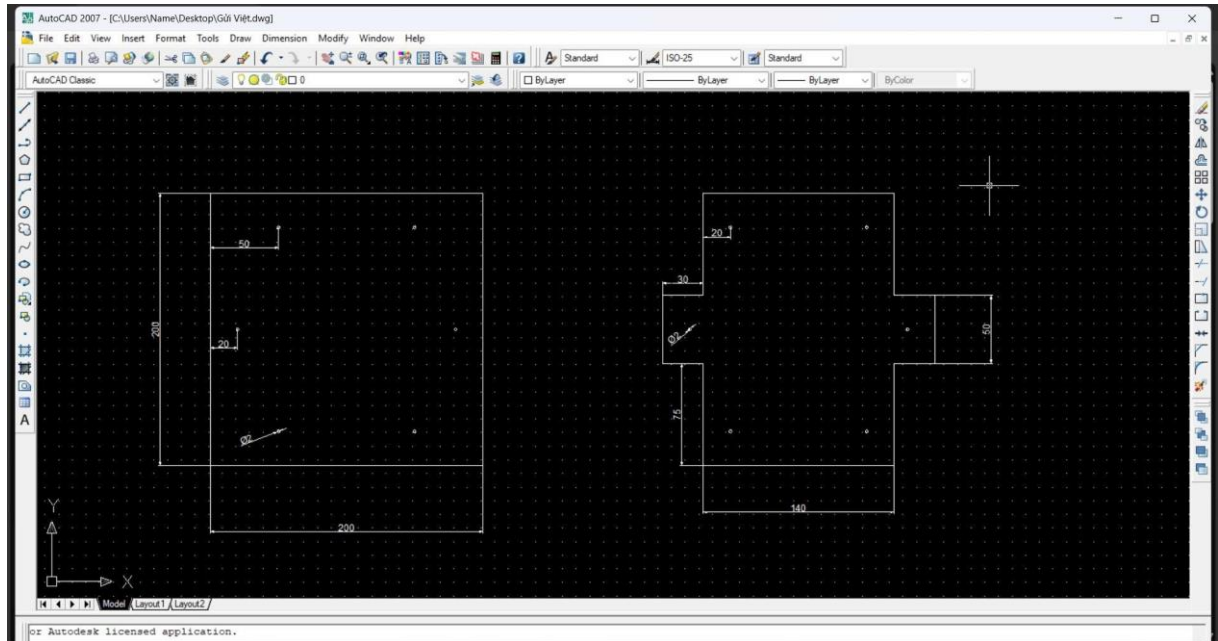
Khởi động cơ:

Động cơ di chuyển: Bố trí trên xe với lần lượt 2 động cơ mắc song song nhau bên trái và bên phải để xe có thể di chuyển lên xuống và quay trái quay phải linh hoạt. 2 bên động cơ được nối lần lượt vào 2 ngõ ra của module L298N. Ở phía điều khiển, các chân enA, enB được nối với Arduino để điều khiển tốc độ động cơ. Các chân in1, in2, in3, in4 cũng được nối lần lượt là chân 27,28,29,30 để điều khiển chiều quay của động cơ. Cấp nguồn cho động cơ là đầu ra của module giảm áp DC LM2596 nối với chân +12V và GND của L298N.

Động cơ cơ cấu gấp vật: Cơ cấu gấp với hai động cơ phụ trách nhiệm vụ khác nhau. Động cơ dọc: Trượt lên xuống thông qua ray trượt và hệ thống bánh răng, ở mỗi cuối hành trình sẽ có một công tắc hành trình có nhiệm vụ gửi tín hiệu về vị trí và trạng thái của cơ cấu gấp. Động cơ ngang: Động cơ quay giúp cơ cấu đóng và mở ra giúp việc gấp và thả vật trở nên linh hoạt và chính xác hơn. 2 động cơ được nối lần lượt vào 2 ngõ ra của module L298N và các chân in1, in2, in3, in4 cũng được nối lần lượt vào các chân 38,39,40,41 của Arduino để điều khiển chiều quay của động cơ. Cấp nguồn cho mạch là đầu ra của module giảm áp DC LM2596 còn lại.

3.5. Mô hình phần cứng

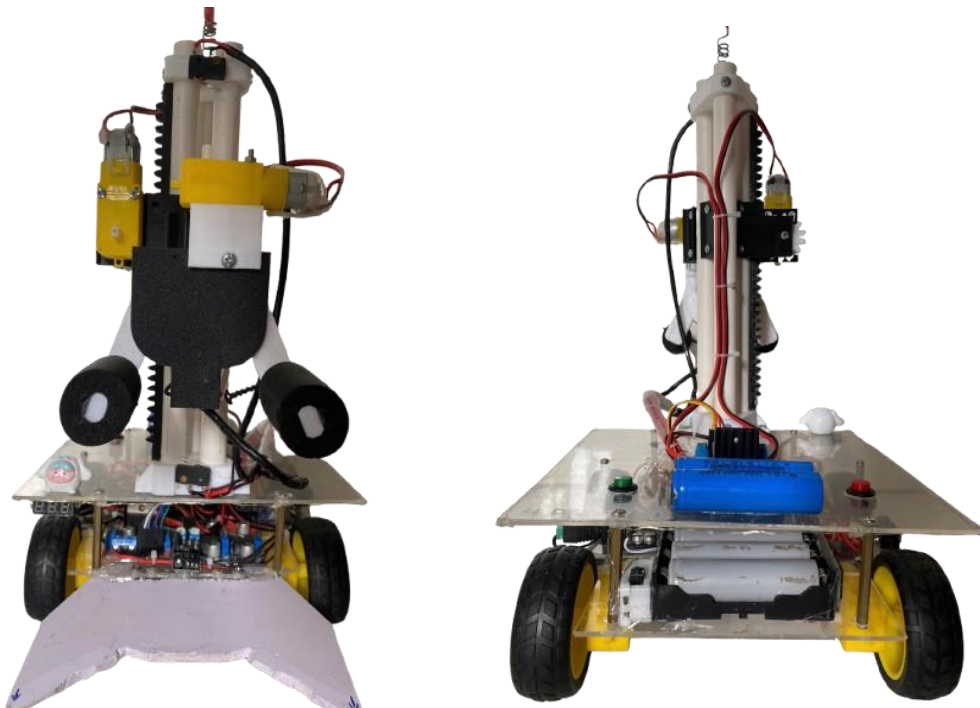
Thiết kế trên phần mềm AutoCAD:



Hình 3.5. Thiết kế phần cứng trên Autocad

Khung xe được thiết kế hình vuông với kích thước tổng thể là 20cm x 20cm ở mặt trên và ở mặt dưới được thiết kế để sao cho có thể đặt vừa động cơ và các linh kiện sao cho hài hòa và cân đối nhất.

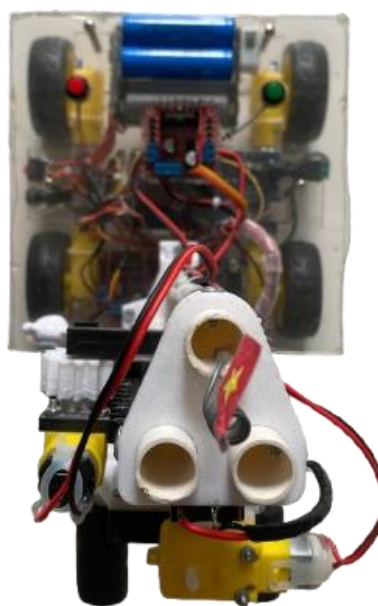
Thiết kế robot hoàn chỉnh:



Hình 3.6. Mặt trước và sau xe robot



Hình 3.7. Mặt trái và phải của xe robot



Hình 3.8. Mặt trên của xe robot

CHƯƠNG 4. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

4.1. Kết quả đạt được

Robot dò line gấp vật chạy đúng đường line theo quy định, có kích thước và khối lượng phù hợp và đúng theo quy định. Cảm biến và chương trình hoạt động tốt giúp nhận biết theo dõi đường line và vật, đảm bảo robot di chuyển theo đúng con đường định sẵn.

Khung xe với các bộ phận và các cấu trúc chịu tải được thiết kế có độ cứng và ổn định cao, giúp robot di chuyển một cách chính xác trên đường line.

Điều này giúp robot hoạt động ổn định và chính xác trong quá trình dò line và gấp vật, đáp ứng được các yêu cầu và mục đích của đề tài.

4.1.1. Ưu điểm

Mô hình được thiết kế và thi công cứng cáp và chắc chắn.

Robot hoạt động khá ổn định.

Di chuyển được chính xác theo line, gấp được vật và đưa vật tới đích.

Có thể linh hoạt thay đổi chương trình, thay đổi sân đấu dễ dàng.

Thao tác khởi động nhanh và tiện lợi.

Theo dõi linh hoạt mức điện áp để có thể đưa ra phương án sạc phù hợp.

Có tính năng cản trở đối thủ thả vật giúp có điểm số cao hơn trong cuộc thi.

4.1.2. Nhược điểm

Khả năng gấp vật còn nhiều hạn chế, còn chưa có độ chính xác cao

Cảm biến hồng ngoại đôi khi còn gặp nhiễu

Pin dung lượng thấp nên robot nhanh hết pin dẫn đến không thực hiện đúng theo mong muốn.

Khi gấp vật do sức nặng dồn về phía trước nên khi di chuyển xe dễ bị bẩy lên phần sau đuôi.

4.2. Hướng phát triển đề tài

Cải tiến hệ thống điều khiển sử dụng thuật toán nâng cao áp dụng các thuật toán như PID (Proportional-Integral-Derivative) để cải thiện độ chính xác khi dò line. Nâng cấp phần cứng cảm biến chính xác hơn sử dụng cảm biến cao cấp hơn để cải thiện độ chính xác và độ tin cậy khi dò line và phát hiện vật thể. Động cơ mạnh hơn chọn các loại động cơ có công suất lớn hơn để nâng cao khả năng di chuyển và gấp vật của robot.

Tích hợp các tính năng theo dõi qua wifi hay bluetooth lên các thiết bị di động hoặc máy tính để biết thông số theo thời gian thực của robot.

4.3. Kết luận

Trong đồ án này, nhóm đã tích góp được nhiều kết quả quan trọng. Đầu tiên đã thành công trong việc chế tạo và lập trình thành công “Mô hình xe robot tự hành dò line và gấp vật”. Robot được thiết kế với cấu trúc cơ khí cơ bản đảm bảo độ cứng vững và tính ổn định trong quá trình hoạt động.

Thử nghiệm và kiểm tra khả năng dò line của robot và kết quả cho thấy nó có khả năng chạy đúng đường line định sẵn một cách chính xác. Điều này chứng tỏ tính chính xác và đáng tin cậy của robot trong việc điều hướng và di chuyển trên đường line. Ngoài ra, robot cũng đã thể hiện khả năng gấp vật một cách ổn định và chính xác. Tính linh hoạt và đa nhiệm của robot cho phép nó thực hiện nhiều nhiệm vụ khác nhau liên quan đến gấp vật một cách hiệu quả.

Tổng kết, đồ án đã đạt được kết quả quan trọng trong việc thiết kế và lập trình “xe robot dò line gấp vật”. Các kết quả này đánh dấu bước tiến đáng kể trong nghiên cứu và ứng dụng robot trong lĩnh vực dò line và gấp vật. Đồng thời, nó cũng mở ra triển vọng và tiềm năng cho việc phát triển và ứng dụng của robot dò line gấp vật trong tương lai.

TÀI LIỆU THAM KHẢO

- [1] Vina FE - Giới thiệu Arduino Mega 2560. <https://dientutuonglai.com/gioi-thieu-arduino-mega-2560>
- [2] Nshop – Cảm biến vật cản hồng ngoại HW- 488. <https://nshopvn.com/product/cam-bien-vat-can-hong-ngoai-hw-488>
- [3] EasyEDA – About EasyEDA. <https://easyeda.com/page/about>
- [4] VinaFE – Phần mềm lập trình Arduino IDE là gì? <https://dientutuonglai.com/arduino-ide-la-gi>
- [5] RTC Technology – Nguyên lý hoạt động của robot tự hành tránh vật cản. <https://rtc.edu.vn/nguyen-ly-hoat-dong-cua-robot-tu-hanh-tranh-vat-can>
- [6] Advancecad – Xe dò line với Arduino. <https://advancecad.edu.vn/xe-do-line-voi-arduino>
- [7] Trần Hoàn, Giáo trình Cad trong kỹ thuật điều khiển và tự động hóa, Trường đại học Công Thương Tp.HCM

PHỤ LỤC

```
//-----Khai báo L298N động cơ-----
int in11 = 27;
int in21 = 28;
int in31 = 29;
int in41 = 30;
int ena1 = 26;
int enb1 = 31;
//-----Khai báo L298N cơ cấu gấp-----
int in12 = 38;
int in22 = 39;
int in32 = 40;
int in42 = 41;
//-----Khai báo chân cảm biến từ phải qua-----
int s1 = 24;
int s2 = 23;
int s3 = 22;
//-----Khai báo chân cảm biến vật cản-----
int svat = 25;
//-----Set nút điều khiển-----
int le = 36;
int ri = 37;
//-----Kết nối công tắc hành trình-----
int ctht_len = 33; // thanh trượt lên xuống
int ctht_xuong = 32;
int ctht_tha = 34; // tay kẹp sản phẩm
int ctht_gap = 35;
//-----Biến trạng thái công tắc hành trình-----
int trangthai_ctht_len = 1;
int trangthai_ctht_xuong = 1;
int trangthai_ctht_tha = 1;
```

```

int trangthai_ctht_gap = 1;
//-----Đặt tên cho các giá trị-----
int cb_vat;
int nut_r;
int nut_l;
int CB1;
int CB2;
int CB3;

//-----Khởi tạo-----
void setup() {
    Serial.begin(9600);
    //-----Chức năng chân drive động cơ-----
    pinMode(in11, OUTPUT);
    pinMode(in21, OUTPUT);
    pinMode(in31, OUTPUT);
    pinMode(in41, OUTPUT);
    pinMode(ena1, OUTPUT);
    pinMode(enb1, OUTPUT);
    //-----Chức năng chân drive cơ cấu-----
    pinMode(in12, OUTPUT);
    pinMode(in22, OUTPUT);
    pinMode(in32, OUTPUT);
    pinMode(in42, OUTPUT);
    //-----Chức năng cảm biến line và vật-----
    pinMode(s1, INPUT);
    pinMode(s2, INPUT);
    pinMode(s3, INPUT);
    pinMode(svat, INPUT);
    //-----Chức năng nút điều khiển-----
    pinMode(le, INPUT_PULLUP);
    pinMode(ri, INPUT_PULLUP);

```

```

//-----Chức năng công tác hành trình-----
pinMode(ctht_len, INPUT_PULLUP);
pinMode(ctht_xuong, INPUT_PULLUP);
pinMode(ctht_tha, INPUT_PULLUP);
pinMode(ctht_gap, INPUT_PULLUP);

//-----Tốc độ động cơ ban đầu-----
analogWrite(ena1, 250); // tốc độ động cơ a ban đầu 120 (0-255)
analogWrite(enb1, 250); // tốc độ động cơ b ban đầu 120 (0-255)
delay(1);
}

//-----Chương trình chính-----
void loop() {
  cb_vat = digitalRead(svai);
  nut_l = digitalRead(le);
  nut_r = digitalRead(ri);
  read();

  //-----Dò line trái (khi nhấn nút trái)-----
  if (nut_l == 0) {
    // Đi thẳng ban đầu
    dithang();
    delay(3550);

    // Dừng line trên
    while (true) {
      read();
      if (CB1 == 0 && CB2 == 0 && CB3 == 0) {
        dungdc();
        delay(300);
        dilui();
      }
    }
  }
}

```

```

    delay(130);
    dungdc();
    delay(250);
    dungdc();
    delay(300);
    retrai();
    delay(350);
    dungdc();
    delay(500);
    break;
}
dolinetrai();
}
// Thấy vật
while (true) {
    dolinetrai();
    if (digitalRead(svat) == 0) {
        break;
    }
}

// Gấp vật
if (digitalRead(svat) == 0) {
    dungdc();
    delay(300);
    gapvat();
    delay(200);
    dilui();
    delay(450);
    dungdc();
    delay(300);
    retrai();
}

```

```

    delay(500);
    dithang();
    delay(400);
}
while (true) {
    dithangcham();
    if (digitalRead(s1)==0||digitalRead(s2)==0||digitalRead(s3)==0) {
        break;
    }
}
dungdc();
delay(300);
dilui();
delay(400);
dungdc();
delay(300);
lechtrai();
delay(700);
dungdc();
delay(300);
dithang();
delay(100);

while (true) {
    dolinetrai();
    if (digitalRead(svat) == 0) {
        break;
    }
}

// Thả vật
if (digitalRead(svat) == 0) {

```

```

dithang();
delay(100);
dungdc();
delay(300);
thavat();
delay(100);
can_doi_thu();
delay(200);
dungdc();
delay(30000);
}
}
//-----Dò line phải (khi nhấn nút phải)-----
if (nut_r == 0) {
// Đi thẳng ban đầu
dithang();
delay(3500);

// Dừng line trên
while (true) {
    read();
    if (CB1 == 0 && CB2 == 0 && CB3 == 0) {
        dungdc();
        delay(300);
        dilui();
        delay(130);
        dungdc();
        delay(300);
        rephai();
        delay(350);
        dungdc();
        delay(500);
    }
}
}

```

```

        break;
    }
    dolinephai();
}
// Thấy vật
while (true) {
    dolinephai();
    if (digitalRead(svat) == 0) {
        break;
    }
}

// Gấp vật
if (digitalRead(svat) == 0) {
    dungdc();
    delay(300);
    gapvat();
    delay(200);
    dilui();
    delay(350);
    dungdc();
    delay(300);
    rephai();
    delay(450);
    dithang();
    delay(400);
}
while (true) {
    dithang();
    if (digitalRead(s1)==0||digitalRead(s2)==0||digitalRead(s3)==0) {
        break;
    }
}

```

```

}
dungdc();
delay(300);
dilui();
delay(320);
dungdc();
delay(300);
lechphai();
delay(700);
dungdc();
delay(300);
dithang();
delay(100);

while (true) {
    dolinephai();
    if (digitalRead(svat) == 0) {
        break;
    }
}

// Thả vật
if (digitalRead(svat) == 0) {
    dithang();
    delay(100);
    dungdc();
    delay(300);
    thavat();
    delay(100);
    can_doi_thu();
    delay(200);
    dungdc();

```



```

        delay(30000);
    }
}
//-----Đọc giá trị cảm biến-----
void read() {
    CB1 = digitalRead(s1);
    CB2 = digitalRead(s2);
    CB3 = digitalRead(s3);
}

//=====Dò line trái=====
void dolinetrai() {
    int IR1 = digitalRead(s1); // Cảm biến phải
    int IR2 = digitalRead(s2); // Cảm biến giữa
    int IR3 = digitalRead(s3); // Cảm biến trái

    if (IR2 == 0 && IR1 == 1 && IR3 == 1) {
        dithang();
    } else if (IR1 == 0 && IR2 == 0 && IR3 == 1) {
        lechphai();
    } else if (IR1 == 0 && IR2 == 1 && IR3 == 1) {
        rephai();
    } else if (IR3 == 0 && IR2 == 0 && IR1 == 1) {
        lechtrai();
    } else if (IR3 == 0 && IR2 == 1 && IR1 == 1) {
        retrai();
    } else if (IR1 == 1 && IR2 == 1 && IR3 == 1) {
        dungdc();
    } else if (IR1 == 0 && IR2 == 0 && IR3 == 0) {
        dungdc();
    } else {
        dungdc();
    }
}

```

```

    }
}

//=====Dò line phải=====

void dolinephai() {
    int IR1 = digitalRead(s1); // Cảm biến phải
    int IR2 = digitalRead(s2); // Cảm biến giữa
    int IR3 = digitalRead(s3); // Cảm biến trái

    if (IR2 == 0 && IR1 == 1 && IR3 == 1) {
        dithang();
    } else if (IR1 == 0 && IR2 == 0 && IR3 == 1) {
        lechphai();
    } else if (IR1 == 0 && IR2 == 1 && IR3 == 1) {
        rephai();
    } else if (IR3 == 0 && IR2 == 0 && IR1 == 1) {
        lechtrai();
    } else if (IR3 == 0 && IR2 == 1 && IR1 == 1) {
        retrai();
    } else if (IR1 == 1 && IR2 == 1 && IR3 == 1) {
        dungdc();
    } else if (IR1 == 0 && IR2 == 0 && IR3 == 0) {
        dungdc();
    } else {
        dungdc();
    }
}

//=====Hàm điều khiển động cơ=====

void dithang() {
    analogWrite(ena1, 255);
    analogWrite(enb1, 255);
}

```

```
digitalWrite(in11, LOW);  
digitalWrite(in21, HIGH);  
digitalWrite(in31, LOW);  
digitalWrite(in41, HIGH);  
}
```

```
void dithangcham() {  
    analogWrite(ena1, 150);  
    analogWrite(enb1, 150);  
    digitalWrite(in11, LOW);  
    digitalWrite(in21, HIGH);  
    digitalWrite(in31, LOW);  
    digitalWrite(in41, HIGH);  
}
```

```
void dilui() {  
    analogWrite(ena1, 170);  
    analogWrite(enb1, 170);  
    digitalWrite(in11, HIGH);  
    digitalWrite(in21, LOW);  
    digitalWrite(in31, HIGH);  
    digitalWrite(in41, LOW);  
}
```

```
void dungdc() {  
    analogWrite(ena1, 0);  
    analogWrite(enb1, 0);  
    digitalWrite(in11, LOW);  
    digitalWrite(in21, LOW);  
    digitalWrite(in31, LOW);  
    digitalWrite(in41, LOW);  
}
```

```
void rephai() {  
    analogWrite(ena1, 255);  
    analogWrite(enb1, 255);  
    digitalWrite(in11, HIGH);  
    digitalWrite(in21, LOW);  
    digitalWrite(in31, LOW);  
    digitalWrite(in41, HIGH);  
}
```

```
void retrai() {  
    analogWrite(ena1, 255);  
    analogWrite(enb1, 255);  
    digitalWrite(in11, LOW);  
    digitalWrite(in21, HIGH);  
    digitalWrite(in31, HIGH);  
    digitalWrite(in41, LOW);  
}
```

```
void lechtrai() {  
    analogWrite(ena1, 255);  
    analogWrite(enb1, 100);  
    digitalWrite(in11, LOW);  
    digitalWrite(in21, HIGH);  
    digitalWrite(in31, LOW);  
    digitalWrite(in41, HIGH);  
}
```

```
void lechphai() {  
    analogWrite(ena1, 100);  
    analogWrite(enb1, 255);  
    digitalWrite(in11, LOW);  
    digitalWrite(in21, HIGH);  
}
```

```
digitalWrite(in31, LOW);
digitalWrite(in41, HIGH);
}
```

```
//=====Hàm điều khiển cơ cấu gấp=====
```

```
void gapvat() {
    // Thả vật
    digitalWrite(in32, HIGH);
    digitalWrite(in42, LOW);
    while (digitalRead(ctht_tha) == HIGH) {
        delay(10);
    }
    // Nâng thanh trượt lên
    digitalWrite(in12, LOW);
    digitalWrite(in22, HIGH);
    while (digitalRead(ctht_len) == HIGH) {
        delay(10);
    }
    // Hạ thanh trượt xuống
    digitalWrite(in12, HIGH);
    digitalWrite(in22, LOW);
    while (digitalRead(ctht_xuong) == HIGH) {
        delay(10);
    }
    digitalWrite(in12, LOW);
    digitalWrite(in22, LOW);

    // Gấp vật
    digitalWrite(in32, LOW);
    digitalWrite(in42, HIGH);
    while (digitalRead(ctht_gap) == HIGH) {
```

```

    delay(10);
}
digitalWrite(in32, LOW);
digitalWrite(in42, LOW);

// Nâng thanh trượt lên
digitalWrite(in12, LOW);
digitalWrite(in22, HIGH);
while (digitalRead(ctht_len) == HIGH) {
    delay(10);
}
// digitalWrite(in12, LOW);
// digitalWrite(in22, LOW);
}
//-----Thả vật-----
void thavat() {
    digitalWrite(in12, LOW);
    digitalWrite(in22, HIGH);
    while (digitalRead(ctht_len) == HIGH) {
        delay(200);
    }

    // Thả vật
    digitalWrite(in32, HIGH);
    digitalWrite(in42, LOW);
    while (digitalRead(ctht_tha) == HIGH) {
        delay(10);
    }
    digitalWrite(in32, LOW);
    digitalWrite(in42, LOW);
    digitalWrite(in12, LOW);
    digitalWrite(in22, LOW);
}

```

```

}
//===== Cản phá đối thủ=====
void can_doi_thu() {
    // Hạ thanh trượt xuống
    digitalWrite(in12, HIGH);
    digitalWrite(in22, LOW);
    delay(200); /
    digitalWrite(in12, LOW);
    digitalWrite(in22, LOW);
    // Gấp vật
    digitalWrite(in32, LOW);
    digitalWrite(in42, HIGH);
    while (digitalRead(ctht_gap) == HIGH) {
        delay(10);
    }
    digitalWrite(in32, LOW);
    digitalWrite(in42, LOW);
}

```