

MỤC LỤC

DANH MỤC HÌNH ẢNH, ĐỒ THỊ	3
DANH MỤC BẢNG	5
DANH SÁCH CÁC TỪ VIẾT TẮT	6
LỜI CAM ĐOAN	7
LỜI CẢM ƠN.....	8
CHƯƠNG 1: TỔNG QUAN	9
1.1. Mô tả đề tài.	9
1.2. Mục tiêu nghiên cứu.....	9
1.3. Đối tượng nghiên cứu	9
1.4. Phạm vi nghiên cứu.	9
1.5. Phương pháp nghiên cứu.	9
1.6. Ý nghĩa khoa học và thực tiễn của đề tài.....	9
1.7 Tổng quan về robot.....	10
1.7.1 Giới thiệu chung về robot di động.....	10
1.7.2 Các bộ phận điều khiển điện tử và cơ cấu chấp hành của robot di động	13
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ THIẾT KẾ HỆ THỐNG.....	18
2.1. Bài toán tìm đường đi ngắn nhất	18
2.1.1. Thuật toán A*	18
2.1.2. Thuật toán Dijkstra	19
2.1.3 Sơ đồ Voronoi.....	19
2.1.4. Lựa chọn thuật toán nghiên cứu	19
2.2. Xử lý bản đồ	20
2.2.1. Xử lý hình ảnh	20
2.2.2 Cải thiện ảnh.....	24
2.3. Xây dựng thuật toán	28
2.3.1. Xử lý bản đồ thực tế.	28
2.3.2. Thuật toán A*	30
2.4. Thiết kế robot.....	32
2.4.1. Linh kiện.....	32

2.4.2. Thiết kế Robot	40
CHƯƠNG 3: THỬ NGHIỆM VÀ THẢO LUẬN	42
3.1. Thử nghiệm.....	42
3.2. Đánh giá.....	50
3.3. Hướng phát triển trong tương lai.....	50
LỜI KẾT	51

DANH MỤC HÌNH ẢNH, ĐỒ THỊ

CHƯƠNG 1: TỔNG QUAN

Hình 1. 1 Các khối cơ bản trong robot di động.	11
Hình 1. 2 Robot hút bụi	12
Hình 1. 3 Robot Curiosity thăm dò sao hỏa của NASA.....	13
Hình 1. 4 Sơ đồ nguyên lý và cấu tạo của một động cơ một chiều.	14
Hình 1. 5 Sơ đồ nguyên lý của mạch cầu H, K1, K2, K3, K4 là các khóa điện tử.	15

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ THIẾT KẾ HỆ THỐNG

Hình 2. 1 Các giai đoạn chính trong xử lý ảnh.....	20
Hình 2. 2 Các thành phần chính của hệ thống xử lý ảnh.	21
Hình 2. 3 (a) Ảnh gốc. (b) Lược đồ mức xám của ảnh.	23
Hình 2. 4 Lược đồ mức xám của ảnh	23
Hình 2. 5 Giãn ảnh.....	25
Hình 2. 6 Ảnh gốc.....	26
Hình 2. 7 Ảnh sau khi được giãn ảnh	26
Hình 2. 8 Ảnh thu được sau khi co ảnh	27
Hình 2. 9 Bản đồ thực tế sảnh khu D tầng 4.....	28
Hình 2. 10 Bản đồ 10x10 mô phỏng bản đồ thực tế.....	29
Hình 2. 11 Dữ liệu số được xuất ra	29
Hình 2. 12 Khu vực hoạt động của robot	30
Hình 2. 13 Bản đồ được duyệt và lưu vào đồ thị D.....	30
Hình 2. 14 Bản đồ được duyệt khi chọn vị trí bắt đầu	31
Hình 2. 15 Đường đi tìm được	31
Hình 2. 16 Mạch điều khiển động cơ DTU Drive.....	32
Hình 2. 17 Động Cơ DC Servo Giảm Tốc Hành Tinh Planetary GP36.....	33
Hình 2. 18 Vi điều khiển Arduino Mega 2560.....	34
Hình 2. 19 Sơ đồ chân vi điều khiển Arduino Mega 2560.....	34
Hình 2. 20 Sơ đồ chân Arduino Mega.....	36
Hình 2. 21 Bản vẽ khung robot.	40

CHƯƠNG 3: THỬ NGHIỆM VÀ THẢO LUẬN

Hình 3. 1 Thực nghiệm lần 1	42
Hình 3. 2 Thực nghiệm lần 2	43
Hình 3. 3 Thực nghiệm lần 3.....	43
Hình 3. 4 Thực nghiệm lần 4.....	44
Hình 3. 5 Thực nghiệm lần 5.....	44
Hình 3. 6 Hành lang khu D tầng 4.....	45
Hình 3. 7 Robot thực nghiệm	46
Hình 3. 8 Robot xuất phát.....	47

Hình 3. 9 Robot gặp vật cản	48
Hình 3. 10 Robot xoay tránh vật cản	48
Hình 3. 11 Robot tránh được vật cản.....	49
Hình 3. 12 Robot xoay tiến về đích	49
Hình 3. 13 Robot về đích.....	50

DANH MỤC BẢNG

Bảng 1. 1 Trạng thái điều khiển động cơ	17
Bảng 3. 1 Bảng kết quả mô phỏng	45
Bảng 3. 2 Bảng kết quả thực nghiệm.....	46

DANH SÁCH CÁC TỪ VIẾT TẮT

OpenCV	Open Computer Vision
CDIO	Conceive Design Implement Operate
GPS	Global Positioning System

LỜI CAM ĐOAN

Tôi thực hiện đề tài “ Thiết kế giải thuật tìm đường đi tối ưu cho robot dò đường” cam đoan những nội dung nghiên cứu và thực hiện đề tài trong cuốn báo cáo này là do tôi tự nghiên cứu và thực hiện không sao chép bất kì khóa luận tốt nghiệp hay đồ án tốt nghiệp nào. Nếu có phát hiện nội dung sao chép hay gian lận nào tôi xin chịu hoàn toàn trách nhiệm trước hội đồng kỉ luật của nhà trường.

Đà Nẵng, ngày ... tháng ... năm 2020

Sinh viên thực hiện

(Ký và ghi rõ họ tên)

LỜI CẢM ƠN

Để có thể hoàn thành đồ án, ngoài sự nỗ lực của bản thân, không thể không kể đến những sự hỗ trợ, giúp đỡ vô cùng to lớn.

Đầu tiên em xin gửi lời cảm ơn tới các thầy cô, ban giám hiệu Trường Đại học Duy Tân nói chung cũng như trung tâm Điện – Điện tử nói riêng đã tận tình truyền giảng dạy, tạo điều kiện, giúp đỡ em trong suốt 5 năm học qua. Những kiến thức mà thầy cô truyền thụ chính là nền tảng quan trọng để em có thể thực hiện đồ án này cũng như trong công việc sau này.

Cảm ơn sự đóng góp thầm lặng của các tác giả đã chia sẻ kiến thức và tài nguyên miễn phí trên Internet.

Xin cảm ơn gia đình và những người bạn đã luôn động viên cổ vũ, là chỗ dựa tinh thần tin cậy nhất của em trong suốt quá trình thực hiện đồ án.

Em xin dành lời cảm ơn đặc biệt tới thầy TS. Hà Đắc Bình – giảng viên trực tiếp hướng dẫn em thực hiện đồ án này. Ngoài những chỉ dẫn vô cùng tận tâm, em còn cảm nhận được sự tin tưởng, niềm kỳ vọng vào sinh viên và phong cách làm việc rất chuyên nghiệp ở thầy.

Em sẽ không thể đạt được kết quả như ngày hôm nay nếu như không có sự giúp đỡ của tất cả mọi người. Một lần nữa, em xin chân thành cảm ơn!

Đà Nẵng, tháng 12 năm 2020

Sinh viên thực hiện

Trần Minh Tâm

CHƯƠNG 1: TỔNG QUAN

1.1. Mô tả đề tài.

Đề tài “Thiết kế giải thuật tìm đường đi tối ưu cho robot dò đường” là xử lý bản đồ thực tế thành ma trận dữ liệu số, từ đó nạp ma trận dữ liệu số vào máy tính. Máy tính sẽ xử lý và đưa ra output là lộ trình đường đi tối ưu (tối ưu về thời gian, quãng đường hoặc cả hai).

Sau khi nhận được tín hiệu từ máy tính. Robot sẽ đi theo lộ trình này để đến đích.

1.2. Mục tiêu nghiên cứu.

Tiến hành xử lý ảnh(bản đồ). Thiết kế giải thuật tìm đường đi tối ưu (ngắn nhất hoặc nhanh nhất) cho robot.

1.3. Đối tượng nghiên cứu

Giải thuật tìm đường đi tối ưu A^*

Robot tự hành

1.4. Phạm vi nghiên cứu.

Nghiên cứu, thiết kế và chế tạo robot hoạt động độc lập trong điều kiện phòng thí nghiệm, vùng hoạt động của robot có diện tích 36m², có vật cản.

1.5. Phương pháp nghiên cứu.

Thực hiện nghiên cứu dựa trên mô hình CDIO

Khảo sát, tìm hiểu, thu thập thông tin từ Internet, sách báo, ...

Tổng hợp và phân tích thông tin

Thiết kế và thi công Thử nghiệm, đánh giá và hiệu chỉnh sản phẩm

1.6. Ý nghĩa khoa học và thực tiễn của đề tài.

Giải pháp tối ưu đường đi giúp robot tiết kiệm thời gian cũng như quãng đường di chuyển, từ đó tiết kiệm chi phí cũng như nâng cao hiệu quả công việc. Ngoài ra, giải pháp còn có thể được ứng dụng trong các công cụ định hướng cho phương tiện giao thông. Những dữ liệu thu thập được của robot sẽ giúp hệ thống cập nhật lại điều kiện thực tế, có thể tích hợp công nghệ học máy (machine learning).

1.7 Tổng quan về robot

Cuộc cách mạng công nghiệp là những bước phát triển vượt bậc của con người, cuộc cách mạng cho phép ứng dụng rộng rãi các robot trong nhiều lĩnh vực của cuộc sống. Yếu tố này phù hợp với nhận thức về vấn đề an toàn trong công việc, robot thay thế cho con người trong sản xuất, trong lao động, trong các nhà máy... điều đó đã góp phần vào sự xuất hiện nhu cầu sử dụng các robot. Ngày nay trong đời sống có rất nhiều công việc do tính chất công việc nặng nhọc, môi trường làm việc khó khăn độc hại rất nguy hiểm và tác hại trực tiếp đối với con người. Một số nơi địa hình quá khó khăn để di chuyển (vùng rừng núi, sa mạc, vùng có lỗ núi, động đất, vùng bị cháy rừng, vùng nhiễm phóng xạ, khu vực bị khủng bố...). Ở những nơi này con người rất cần tới sự hỗ trợ của robot. Với mô hình robot đa năng có sự tuỳ biến cao, dễ lắp đặt và vận chuyển, được thiết kế và lập trình từ máy tính cá nhân, robot có thể làm việc độc lập theo chu trình được cài đặt sẵn hoặc theo sự điều khiển từ xa qua vô tuyến từ người điều khiển, tính chất và công việc cụ thể được thay đổi dễ dàng, khả năng kết nối với các thiết bị chuyên dụng linh hoạt. Robot có kích thước tương đối và làm việc được trong nhiều môi trường khắc nghiệt về thời tiết, khí hậu độc hại và nguy hiểm đối với con người. Ngoài ra robot có thể được thiết kế cho phù hợp với các công việc mang tính chất tự động hoá cao, có thể ứng dụng vào các dây chuyền sản xuất tự động ở các nhà máy, xí nghiệp, các khu công nghiệp, khu chế xuất... Hiện nay hầu hết các thiết bị robot và dây chuyền tự động hoá được sử dụng trong các ngành công nghiệp sản xuất và chế biến đều được nhập từ nước ngoài rất đắt tiền. Trong khi nhu cầu ở nước ta đang rất cao và trong nước có khả năng chế tạo sản xuất để phù hợp với điều kiện làm việc ở Việt Nam. Lĩnh vực thiết kế, chế tạo robot và các thiết bị điều khiển tự động rất mới mẻ và có tiềm năng rộng lớn không những ở trong nước mà còn cả trên thế giới. Việc thâm nhập, nghiên cứu và chế tạo một số mô hình điều khiển tự động như robot thông minh, robot thăm dò, robot sản xuất là một hướng cần thiết nhằm rút ngắn khoảng cách giữa khoa học công nghệ trong nước và trên thế giới trong lĩnh vực này.

1.7.1 Giới thiệu chung về robot di động

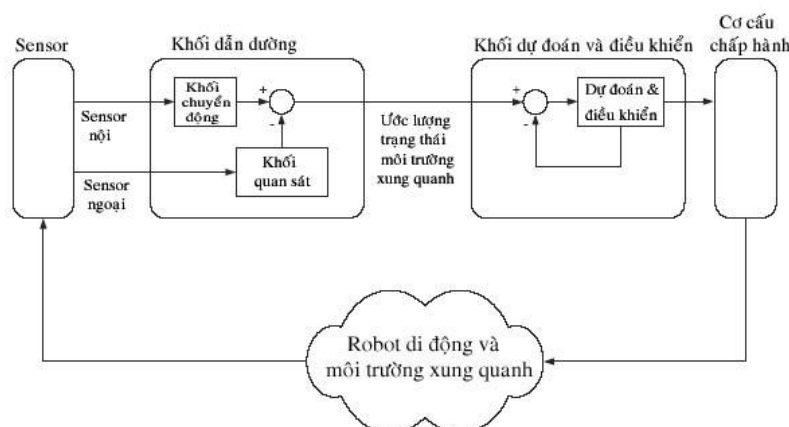
Có một số sự khác biệt quan trọng giữa các yêu cầu của việc lắp đặt robot cố định truyền thống với các yêu cầu của các hệ thống robot di động. Một trong những mối quan tâm hàng đầu là sự không biết trước môi trường vận hành của robot di động. Đối với các hệ thống robot cố định, người ta thường xây dựng (thiết kế) một không gian làm việc nhỏ

để thực hiện công việc và robot cố định thường thực hiện các công việc lặp đi lặp lại trong môi trường xác định trước. Đối với các hệ thống robot di động, việc nhận biết được môi trường làm việc là một yếu tố quyết định tới các “hành động” của robot, chỉ khi nhận biết được đầy đủ các thông tin về môi trường xung quanh thì robot di động mới có thể thích ứng được trong các môi trường làm việc khác nhau.

Theo khái niệm, robot di động phải có một số bộ phận chuyển động. Chuyển động có thể dưới dạng bánh xe, chân, cánh hoặc một số cơ cấu khác. Việc lựa chọn cơ cấu chuyển động là dựa vào chức năng của robot và các công việc của robot cần phải thực hiện. Trong nhiều môi trường làm việc công nghiệp, bánh xe là dạng chuyển động thích hợp nhất. Đối với các hệ thống nghiên cứu ứng dụng khác, chân hoặc cánh có thể giúp cho robot di động chuyển động được trên địa hình mà robot không có khả năng đi qua.

Một nhân tố quan trọng trong thiết kế của các hệ thống robot di động là khả năng mang theo các thiết bị. Thiết bị bao gồm nguồn điện cũng như tất cả các phần cứng là các bộ cảm biến và các bộ xử lý mà robot đòi hỏi. Các thiết bị lắp đặt của robot cố định có thể được kết nối trực tiếp với các nguồn điện và bộ xử lý, không gian làm việc thường được xác định trước và có thể được kiểm soát. Hầu hết các robot di động đều có một trọng tải (payload) nhất định, điều này có thể dẫn tới những sự giới hạn trong thiết kế của hệ thống, đặc biệt trong các ứng dụng hay bị giới hạn về kích cỡ và cân nặng.

Các thành phần cơ bản, thông dụng trong các hệ thống robot di động sử dụng trong thực tế được chỉ ra như trên hình 1.



Hình 1. 1 Các khối cơ bản trong robot di động.

Thông tin về trạng thái của hệ thống được cung cấp bởi các cảm biến nội trong và cảm biến ngoại. Thông tin này được một khối dẫn đường (có thể là các môđun phần cứng hoặc phần mềm, nhưng thường là một chương trình phần mềm có khả năng tính toán) sử dụng để ước lượng trạng thái của hệ thống. Tín hiệu này sau đó được môđun lập kế hoạch và điều khiển sử dụng để phát ra các lệnh gửi tới các bộ phận thao tác của robot.

Các khối cơ bản cho các hệ thống robot di động bao gồm các bộ phận cảm biến, bộ phận dẫn đường, các môđun lập kế hoạch, điều khiển và các bộ chấp hành dùng để di chuyển robot và tác động tới môi trường của nó. Trong khi một số các phần cụ thể của robot di động sẽ phụ thuộc vào mục đích ứng dụng, một hệ thống robot di động điển hình đòi hỏi phải có tất cả các khối trên thì mới vận hành được. Thông tin về trạng thái môi trường xung quanh của robot được cung cấp với các cảm biến nội và cảm biến ngoại. Thông tin này được bộ xử lý trung tâm (có cài đặt sẵn các chương trình, thuật toán dẫn đường...) sử dụng để ước lượng trạng thái của hệ thống, không chỉ bao gồm trạng thái của bản thân robot mà cả trạng thái của môi trường xung quanh nó. Ước lượng này được sử dụng để lập kế hoạch hoạt động của robot và phát ra các lệnh cho các cơ cấu chấp hành của robot.

Thiết kế của các hệ thống robot di động phần nhiều sẽ bị chi phối bởi mục đích ứng dụng. Không có giới hạn về số lượng các kết hợp giữa các thiết bị thao tác, các bộ phận cảm biến và các thuật toán điều khiển cho các hệ thống robot di động. Hơn nữa, luồng thông tin trong phạm vi hệ thống sẽ thay đổi tùy thuộc vào quá trình thực thi, sẽ có một số thiết kế robot khác đáng kể so với những robot trình bày trong hình 1. Trong một số trường hợp, dữ liệu từ các bộ phận cảm biến được trực tiếp cung cấp cho bộ lên kế hoạch và điều khiển, cho phép hệ thống phản ứng nhanh hơn với những thay đổi mạnh mẽ của môi trường.

Một số robot di động phổ biến:



Hình 1. 2 Robot hút bụi



Hình 1. 3 Robot Curiosity thăm dò sao hỏa của NASA

1.7.2 Các bộ phận điều khiển điện tử và cơ cấu chấp hành của robot di động

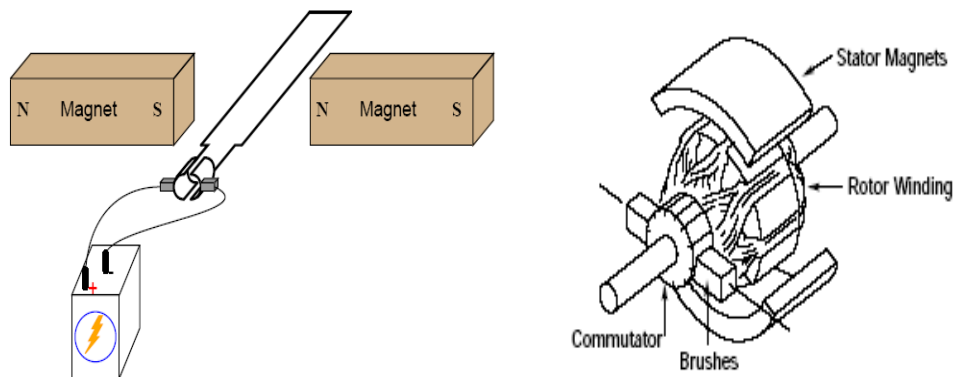
1.7.2.1. Bộ phận chấp hành của robot

Bộ phận chấp hành của robot di động trong trường hợp này thực chất là các motor điện, loại máy chuyển điện năng thành cơ năng thường dưới dạng chuyển động tròn. Một motor được cấu tạo gồm hai phần chính: Rotor và Stator (hình 2) Rotor là phần động nằm ở trong lõi gồm các cuộn dây cuốn quanh một lõi thép, lõi này được cuốn vào trục motor. Stator là phần tĩnh được cấu tạo từ nam châm vĩnh cửu tạo ra một từ trường trong động cơ. Khi có dòng điện chạy qua một dòng điện được sinh ra trong stator và từ trường bên trong động cơ sẽ tác động lên cuộn dây làm cho động cơ quay. Có hai loại động cơ thường được dùng trong điều khiển đó là:

- ✓ Động cơ một chiều (DC motor).
- ✓ Động cơ bước (Step motor).

1.7.2.2. Động cơ một chiều

Là loại động cơ được cấu tạo từ cuộn dây và hệ thống nam châm vĩnh cửu. Loại motor này hoạt động dựa vào dòng điện đưa vào cuộn dây ở stator, do điện trở của cuộn dây trong stator là không đổi nên tốc độ của động cơ phụ thuộc vào điện áp của nguồn đưa vào điều khiển động cơ hoặc sẽ được điều khiển bởi độ rộng của xung điện áp được đưa vào, thường công suất của động cơ một chiều là khá lớn. Nhược điểm lớn nhất của động cơ một chiều là điều khiển sẽ khó chính xác điều này sẽ được khắc phục khi sử dụng động cơ bước. Hình 1.4 chỉ ra sơ đồ nguyên lý và cấu tạo của động cơ một chiều.



Hình 1. 4 Sơ đồ nguyên lý và cấu tạo của một động cơ một chiều.

1.7.2.3. Động cơ bước

Khác với động cơ một chiều, động cơ bước được cấu tạo từ nhiều cuộn dây đối xứng, mỗi một cặp cuộn dây tương ứng với một pha trong chuyển động việc điều khiển động cơ bước thông qua các xung có độ lệch pha với nhau lần lượt qua các cặp cuộn dây. Chính nhờ việc cho các tín hiệu lần lượt chạy qua các cặp cuộn dây đã tạo ra các bước trong chuyển động. Tùy thuộc vào độ chính xác yêu cầu mà mỗi loại động cơ bước có góc bước khác nhau, góc bước càng nhỏ thì độ chính xác càng lớn. Đây chính là ưu điểm lớn nhất so với động cơ một chiều, việc sử dụng động cơ bước sẽ cho phép điều khiển chính xác tới vị trí yêu cầu. Tuy nhiên do đặc điểm cấu tạo mà động cơ bước bị giới hạn nhiều về mặt công suất nên không thể chế tạo các động cơ bước có công suất cao. Chính vì vậy trong các thiết bị đòi hỏi độ chính xác cao, công suất nhỏ như bộ phận quay đầu từ trong đĩa cứng người ta thường sử dụng động cơ bước.

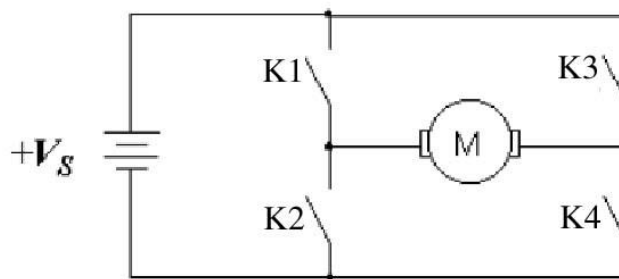
Những đặc tính quan trọng của động cơ điện đó là:

- ✓ Điện áp sử dụng, dòng tiêu thụ: đây là hai đặc tính rất quan trọng vì nó xác định giới hạn làm việc của động cơ và giới hạn làm việc của mạch khuếch đại công suất. Chính hai thông số này cho ta biết được công suất tối đa của mạch công suất để có thể điều khiển động cơ một cách an toàn.
- ✓ Tốc độ tối đa có thể đạt được: chính là tốc độ khi không có tải.
- ✓ Lực xoắn của động cơ: chính là mômen lực lớn nhất mà động cơ có thể tạo ra, đây là thông số quan trọng quyết định tới khả năng tải của động cơ.

1.7.2.4. Bộ phận điều khiển điện tử

Do đặc điểm của động cơ bước, mặc dù điều khiển chính xác nhưng không có công suất lớn hơn nữa do được điều khiển theo từng bước (góc quay) nên sẽ không tạo ra được các chuyển động liên tục một cách trơn tru. Trong khóa luận này sẽ sử dụng các động cơ một chiều như là bộ phận thao tác của robot. Động cơ một chiều có ưu điểm là dễ điều khiển (chỉ cần cung cấp một dòng điện vào cuộn dây là động cơ một chiều sẽ hoạt động) và để tăng độ chính xác cho động cơ một chiều thì một bộ lập mã (encoder) được sử dụng như là bộ phận phản hồi từ động cơ về khối điều khiển.

Mạch công suất được sử dụng là mạch cầu H (H-Bridge). Sở dĩ phải sử dụng mạch cầu H là vì với mạch cầu H sẽ cho phép điều khiển động cơ một chiều có thể quay được theo hai chiều khác nhau. Sơ đồ nguyên lý của mạch cầu H được chỉ ra như trên hình 1.5.



Hình 1. 5 Sơ đồ nguyên lý của mạch cầu H, K1, K2, K3, K4 là các khóa điện tử.

Nguyên tắc hoạt động của mạch cầu H như sau:

- Khi K₁ và K₄ đóng, K₂ và K₃ mở động cơ quay theo một chiều.
- Khi K₂ và K₃ đóng, K₁ và K₄ mở động cơ quay theo chiều ngược lại.
- K₁, K₂ hoặc, K₃, K₄ không được đồng thời mở.

Dựa vào nguyên tắc trên và yêu cầu cần phải thiết kế một mạch công suất dựa trên các linh kiện sẵn có để có thể sử dụng cho hệ robot di động, khóa luận đã thiết kế thành công một mạch cầu H tương đối tối ưu có một số thông số kỹ thuật nổi bật sau:

- Đầu vào điều khiển là 2 bits tương ứng với 4 trạng thái điều khiển (tiến, lùi, phanh và không phanh) của động cơ.
- Có thể điều khiển được động cơ công suất tương đối lớn (600W).

- Tần số hoạt động $\leq 200\text{KHz}$.
- Dải điện thế đưa vào động cơ có thể từ 8V đến 35V.

Với những động cơ có công suất không quá lớn thì không cần dùng các bộ cách ly để cách ly phần công suất với phần điều khiển.

Trong sơ đồ trên 4 khóa điện tử K_1, K_2, K_3, K_4 chính là 4 transistor trường ở đây dùng hai loại JFET kênh N và JFET kênh P. Một bit điều khiển đưa vào sẽ làm cho chỉ một trong 2 (hoặc JFET kênh N hoặc JFET kênh P) trên một nửa cầu được mở. Mặc dù JFET được mở bằng thế giữa cực G và cực S, tuy nhiên trên thực tế có một điện dung rất nhỏ giữa hai cực G và S, chính điện dung này làm cho quá trình đóng và mở JFET bị trễ làm cho quá trình thay đổi trạng thái đóng, mở của JFET là không tức thời, vì vậy sẽ có thời điểm hai JFET trên cùng một kênh sẽ cùng mở, xảy ra hiện tượng ngắn mạch trên một kênh. Để khắc phục nhược điểm trên trong khóa luận sử dụng thêm hai transistor lưỡng cực trước khi tín hiệu được đưa vào cực G để quá trình phóng và nạp điện của tụ điện trên cực GS là tức thời.

Mỗi lỗi vào được đưa qua một tích phân rồi sau đó đưa các bộ so sánh với ngưỡng so sánh lần lượt được đặt ở +2V và +3V cũng để giải quyết vấn đề hai transistor trên cùng một nửa kênh cùng mở. Thực chất mạch tích phân dùng để làm một mạch trễ. Tín hiệu sau khi được trễ sẽ được so sánh với hai mức so sánh khác nhau (+2V và +3V) nên luôn đảm bảo một transistor trên một kênh đóng trước khi transistor còn lại mở, tránh được trường hợp cả hai transistor trên một kênh cùng mở. Chính điều này làm tăng tần số hoạt động của mạch cầu, với thiết kết trên thì mạch cầu có thể hoạt động tốt với tần số tín hiệu đưa vào là 200KHz mà các linh kiện công suất không bị nóng.

Các mạch NOR dùng để mã hóa các tín hiệu lỗi vào cho phép với 2 bits điều khiển nhưng có thể tạo ra bốn trạng thái điều khiển khác nhau với động cơ. Cụ thể với hai bits điều khiển là I_0 và I_1 ta có các trạng thái điều khiển sau:

Bảng 1. 1 Trạng thái điều khiển động cơ

I_0	I_1	Trạng thái điều khiển động cơ
0	0	Không phanh, động cơ ở trạng thái thả
0	1	Động cơ quay tiến.
1	0	Động cơ quay lùi
1	1	Động cơ ở trạng thái phanh, hai đầu dây bị chập lại.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ THIẾT KẾ HỆ THỐNG

2.1. Bài toán tìm đường đi ngắn nhất

Định nghĩa:

-Bài toán tìm đường đi ngắn nhất là bài toán tìm 1 đường đi giữa 2 đỉnh sao cho tổng các trọng số của các cạnh tạo ra đường đi đó nhỏ nhất.

-Cho trước một đồ thị có trọng số (nghĩa là một tập đỉnh V , một tập cạnh E , và một hàm trọng số có giá trị thực $f : E \rightarrow \mathbb{R}$), cho trước một đỉnh v thuộc V , tìm một đường đi P từ v tới mỗi đỉnh v' thuộc V sao cho:

$$\sum_{p \in P} f(p)$$

là nhỏ nhất trong tất cả các đường nối từ v tới v' . Bài toán đường đi ngắn nhất giữa mọi cặp đỉnh là một bài toán tương tự, trong đó ta phải tìm các đường đi ngắn nhất cho mọi cặp đỉnh v và v' .

-Các thuật toán quan trọng nhất giải quyết bài toán này là:

- Thuật toán Dijkstra.
- Giải thuật tìm kiếm A^* .
- Sơ đồ Voronoi.

2.1.1. Thuật toán A^*

Trong khoa học máy tính, A^* (đọc là A sao) là thuật toán tìm kiếm trong đồ thị. Thuật toán này tìm một đường đi từ một nút khởi đầu tới một nút đích cho trước (hoặc tới một nút thỏa mãn một điều kiện đích). Thuật toán này sử dụng một "đánh giá heuristic" để xếp loại từng nút theo ước lượng về tuyến đường tốt nhất đi qua nút đó. Thuật toán này duyệt các nút theo thứ tự của đánh giá heuristic này. Do đó, thuật toán A^* là một ví dụ của tìm kiếm theo lựa chọn tốt nhất (best-first search).

Thuật toán A^* được mô tả lần đầu vào năm 1968 bởi Peter Hart, Nils Nilsson, và Bertram Raphael. Trong bài báo của họ, thuật toán được gọi là thuật toán A; khi sử dụng thuật toán này với một đánh giá heuristic thích hợp sẽ thu được hoạt động tối ưu, do đó mà có tên A^* .

Năm 1964, Nils Nilsson phát minh ra một phương pháp tiếp cận dựa trên khám phá để tăng tốc độ của thuật toán Dijkstra. Thuật toán này được gọi là A1. Năm 1967 Bertram Raphael đã cải thiện đáng kể thuật toán này, nhưng không thể hiện thị tối ưu.

Ông gọi thuật toán này là A2. Sau đó, trong năm 1968 Peter E. Hart đã giới thiệu một đối số chứng minh A2 là tối ưu khi sử dụng thuật toán này với một đánh giá heuristic thích hợp sẽ thu được hoạt động tối ưu. Chứng minh của ông về thuật toán cũng bao gồm một phần cho thấy rằng các thuật toán A2 mới là thuật toán tốt nhất có thể được đưa ra các điều kiện. Do đó ông đặt tên cho thuật toán mới là A*(A sao, A-star).

2.1.2. Thuật toán Dijkstra

Thuật toán Dijkstra, mang tên của nhà khoa học máy tính người Hà Lan Edsger Dijkstra vào năm 1956 và ấn bản năm 1959, là một thuật toán giải quyết bài toán đường đi ngắn nhất nguồn đơn trong một đồ thị có hướng không có cạnh mang trọng số âm. Thuật toán thường được sử dụng trong định tuyến với một chương trình con trong các thuật toán đồ thị hay trong công nghệ Hệ thống định vị toàn cầu (GPS).

2.1.3 Sơ đồ Voronoi

Trong toán học, sơ đồ Voronoi, đặt tên theo nhà toán học người Nga Georgy Voronoi, là một cách phân tách một không gian metric theo khoảng cách tới một tập hợp rời rạc các vật thể cho trước trong không gian. Tập hợp các vật thể có thể là tập hợp rời rạc các điểm. Trong ngành thủy văn, sơ đồ này còn được gọi là đa giác Thiessen theo tên nhà khí tượng học người Mỹ Alfred H. Thiessen.

Trong trường hợp đơn giản nhất, khi các vật thể là các điểm, ta có một tập hợp S gồm các điểm trên mặt phẳng, được gọi là các điểm Voronoi. Mỗi điểm s ứng với một ô Voronoi, hay còn gọi là ô Dirichlet, ký hiệu là $V(s)$, bao gồm tất cả các điểm gần s hơn tất cả các điểm Voronoi khác. Các cạnh của sơ đồ Voronoi là tập các điểm có khoảng cách tới hai điểm Voronoi gần nhất là như nhau. Các đỉnh của sơ đồ Voronoi là các điểm có khoảng cách tới ít nhất ba điểm Voronoi gần nhất là như nhau.

2.1.4. Lựa chọn thuật toán nghiên cứu

Em chọn giải thuật tìm kiếm A* để thiết kế giải thuật tìm đường đi tối ưu cho robot bởi vì:

- Giải thuật tìm kiếm A* tìm kiếm nhanh, tối ưu, dễ hiểu và dễ sử dụng.
- Yêu cầu đối với phần cứng thấp, tiết kiệm chi phí.
- Hoạt động tốt trong môi trường cố định ít thay đổi.

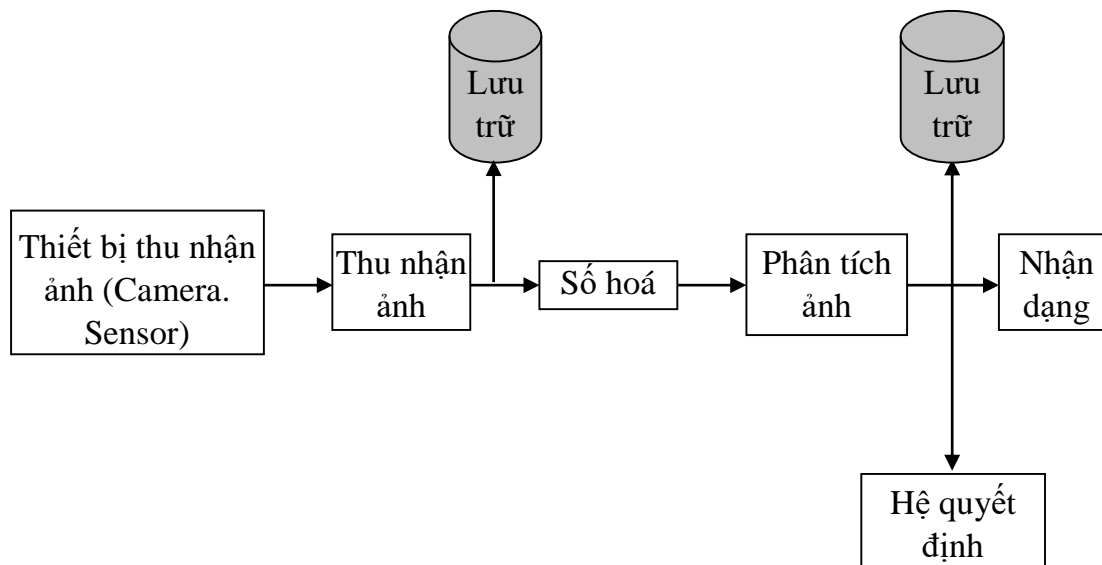
2.2. Xử lý bản đồ

2.2.1. Xử lý hình ảnh

2.2.1.1. Tổng quan về một hệ thống xử lý ảnh

Xử lý ảnh là một khoa học còn tương đối mới mẻ so với nhiều ngành khoa học khác, nhất là trên qui mô công nghiệp, song trong xử lý ảnh đã bắt đầu xuất hiện những máy tính chuyên dụng. Để có thể hình dung cấu hình một hệ thống xử lý ảnh chuyên dụng hay một hệ thống xử lý ảnh dùng trong nghiên cứu, đào tạo, trước tiên chúng ta xem xét các bước cần thiết trong xử lý ảnh.

Trước hết là quá trình thu nhận ảnh. Ảnh có thể thu nhận qua camera. Thường ảnh thu nhận qua camera là tín hiệu tương tự (loại camera ống kiểu CCIR) nhưng cũng có thể là tín hiệu số hoá (loại CCD-Charge Coupled Device).



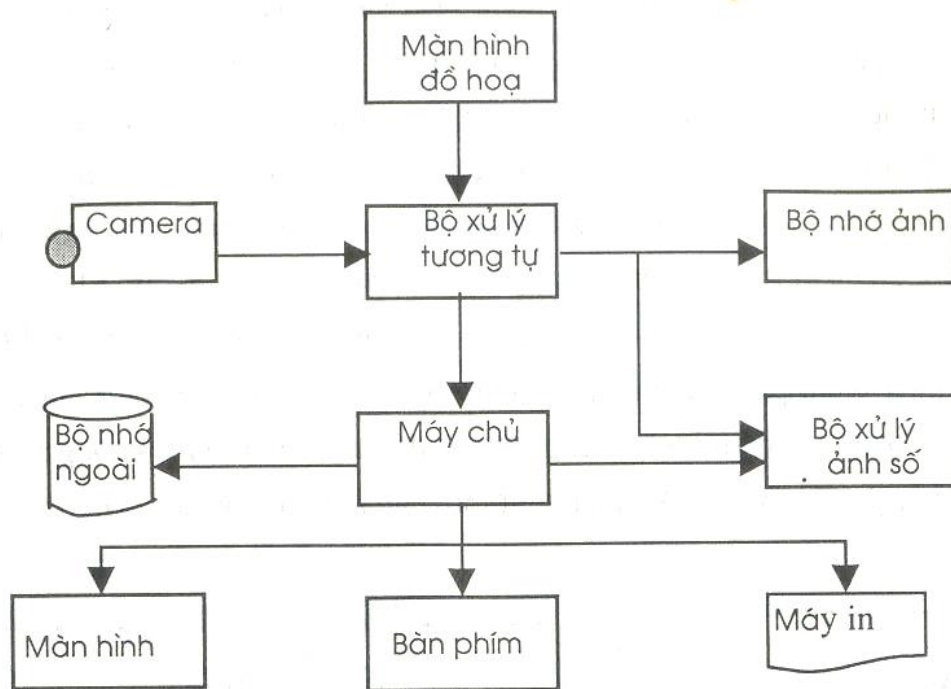
Hình 2. 1 Các giai đoạn chính trong xử lý ảnh

Ảnh cũng có thể thu nhận từ vệ tinh qua các bộ cảm biến (sensor) hay ảnh, tranh được quét trên máy scanner. Tiếp theo là quá trình số hoá để biến đổi tín hiệu tương tự sang tín hiệu số trước khi chuyển sang giai đoạn xử lý, phân tích hay lưu trữ lại.

Quá trình phân tích ảnh thực chất bao gồm nhiều công đoạn nhỏ. Trước hết là công việc tăng cường ảnh để nâng cao chất lượng ảnh. Do những nguyên nhân khác nhau: có thể do chất lượng thiết bị thu nhận ảnh, do nguồn sáng hay do nhiễu, ảnh có thể bị suy

biến. Do vậy cần phải tăng cường và khôi phục lại ảnh để làm nổi bật một số đặc tính chính của ảnh hay làm cho ảnh gần giống nhất với trạng thái gốc-trạng thái trước khi ảnh bị biến dạng. Giai đoạn tiếp theo là phát hiện các đặc tính như biên, phân vùng ảnh, trích chọn các đặc tính...

Cuối cùng, tùy theo mục đích của ứng dụng, sẽ là giai đoạn nhận dạng, phân lớp hay các quyết định khác. Các giai đoạn chính của quá trình xử lý ảnh có thể mô tả ở hình 2.1. Với các giai đoạn trên, một hệ thống xử lý ảnh gồm các thành phần tối thiểu như hình 2.2.



Hình 2. 2 Các thành phần chính của hệ thống xử lý ảnh.

- Đối với một hệ thống xử lý ảnh thu nhận qua camera. Có 2 loại camera: camera ống loại CCIR và camera CCD. Loại camera ứng với chuẩn CCIR quét ảnh với tần số 1/25 và mỗi ảnh gồm 625 dòng. Loại CCD gồm các photodiode và làm tương ứng một cường độ sáng tại một điểm ảnh với một phần tử ảnh. Như vậy ảnh là tập hợp các điểm ảnh. Số pixel tạo nên một ảnh gọi là độ phân giải.
- Bộ xử lý tương tự: Bộ phận này thực hiện các chức năng sau:

- Chọn camera thích hợp nếu hệ thống có nhiều camera.
- Chọn màn hình hiển thị tín hiệu.
- Thu nhận tín hiệu video, thực hiện lấy mẫu và mã hoá.
- Tiền xử lý ảnh khi thu nhận: dùng kỹ thuật băng tra.
- Bộ xử lý ảnh số: Gồm nhiều bộ xử lý chuyên dụng: xử lý lọc, trích chọn đường bao, nhị phân hoá ảnh. Các bộ xử lý này làm việc với tốc độ 1/25 giây.
- Máy chủ: Đóng vai trò điều khiển các thành phần miêu tả ở trên.
- Bộ nhớ ngoài: Dữ liệu ảnh cũng như các kiểu dữ liệu khác, để có thể chuyển giao cho các quá trình khác cần được lưu trữ.

2.2.1.2 Các vấn đề trong xử lý ảnh

2.2.1.2.1 Một số khái niệm

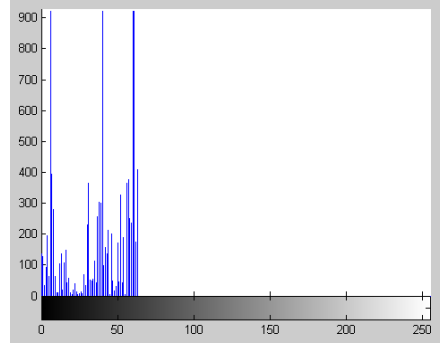
- Pixel: Phần tử ảnh. Ảnh trong thực tế là một ảnh liên tục về không gian và về giá trị độ sáng. Để có thể xử lý ảnh bằng máy tính cần thiết phải tiến hành số hoá ảnh. Trong quá trình số hoá, người ta biến đổi tín hiệu liên tục sang tín hiệu rời rạc thông qua quá trình lấy mẫu và lượng tử hoá mà về nguyên tắc bằng mắt thường không phân biệt được hai mức kề nhau. Mỗi pixel gồm một cặp tọa độ x, y và màu. Một ảnh là một tập hợp các điểm ảnh. Khi được số hoá nó thường được biểu diễn bởi bảng hai chiều $I(n, p)$: n dòng và p cột. Ta nói ảnh gồm $n \times p$ pixels. Người ta thường ký hiệu $I(x, y)$ để chỉ một pixel.
- Gray level: Mức xám. Mức xám là kết quả sự mã hoá tương ứng một cường độ sáng của mỗi điểm ảnh với một giá trị số - kết quả của quá trình lượng tử hoá. Các mã hoá kinh điển thường dùng là 16, 32 hay 64 mức, mã hoá 256 mức là phổ dụng nhất do mỗi pixel sẽ được mã hoá bởi 8 bit.

2.2.1.2.2 Lược đồ mức xám (histogram)

Lược đồ mức xám của một ảnh là một hàm cung cấp tần suất xuất hiện của mỗi mức xám. Lược đồ xám được biểu diễn trong một hệ tọa độ vuông góc x, y . Trong hệ tọa độ này trục hoành biểu diễn số mức xám từ 0 đến N , N là số mức xám. Trục tung biểu diễn số điểm ảnh cho một mức xám (số điểm ảnh có cùng mức xám) hoặc cách biểu diễn khác là trục tung là tỷ lệ số điểm ảnh có cùng mức xám trên tổng số điểm ảnh.



(a)

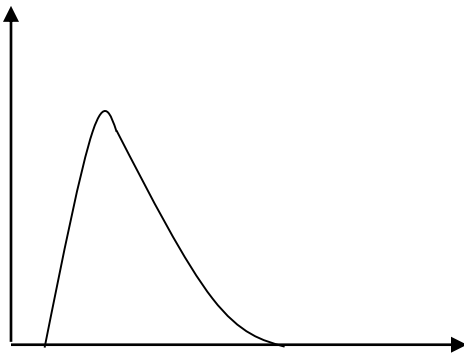


(b)

Hình 2. 3 (a) Ảnh gốc. (b) Lược đồ mức xám của ảnh.

Lược đồ xám cung cấp rất nhiều thông tin về phân bố mức xám của ảnh. Theo thuật ngữ của xử lý ảnh gọi là tính động của ảnh. Tính động của ảnh cho phép phân tích trong khoảng nào đó phân bố phần lớn các mức xám của ảnh: ảnh rất sáng hay ảnh rất đậm.

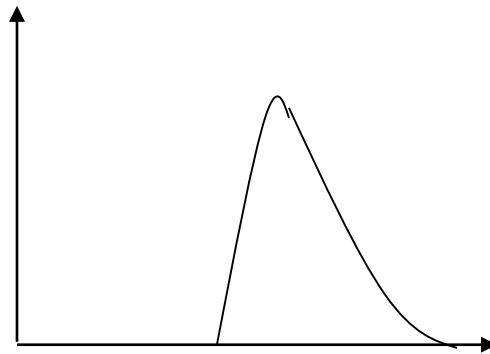
Số điểm ảnh



Mức xám

(a) Ảnh đậm.

Số điểm ảnh



Mức xám

(b) Ảnh nhạt

Hình 2. 4 Lược đồ mức xám của ảnh

Nếu ảnh sáng, lược đồ xám nằm bên phải (mức xám cao) còn ảnh đậm lược đồ xám nằm bên trái (mức xám thấp). Theo định nghĩa của lược đồ xám, thuật toán xây dựng lược đồ xám có thể mô tả như sau:

Bắt đầu

H là bảng chứa lược đồ xám (là vectơ có N phần tử)

a. Khởi tạo mảng

for (i:=1;i<=N;i++) H[i]=0;

b. Tạo bảng

Với mỗi điểm ảnh I(x,y) tính $H[I(x,y)] = H[I(x,y)] + 1$;

c. Tính giá trị Max của bảng H. Sau đó hiện bảng trong khoảng từ 0 đến Max

Kết thúc:

Lọc đồ xám là một công cụ hữu hiệu dùng trong nhiều công đoạn của xử lý ảnh như tăng cường ảnh.

2.2.2 Cải thiện ảnh

Nâng cao chất lượng ảnh là một bước quan trọng tạo tiền đề cho xử lý ảnh. Mục đích chính là nhằm làm nổi bật một số đặc tính của ảnh như thay đổi độ tương phản, lọc nhiễu, nổi biên, làm trơn biên, khuếch đại ảnh... Tập hợp các kỹ thuật này tạo nên giai đoạn tiền xử lý ảnh. Dưới đây sẽ trình bày tổng quan về các kỹ thuật tiền xử lý ảnh.

2.2.2.1. Cải thiện ảnh dùng toán tử điểm

Toán tử điểm là toán tử không bộ nhớ, ở đó một mức xám $u \in [0, N]$ sang mức xám $v \in [0, N]$, $v=f(u)$. Ánh xạ tùy theo các ứng dụng khác nhau mà có dạng khác nhau.

2.2.2.3. Cải thiện ảnh dùng toán tử không gian

Thường ảnh thu nhận có nhiều nhiễu cần phải loại bỏ nhiễu hay ảnh không được sắc nét bị mờ cần làm rõ các chi tiết biên. Các toán tử không gian trong kỹ thuật tăng cường ảnh được phân theo nhóm công dụng: làm trơn nhiễu, nổi biên. Để làm trơn nhiễu hay tách nhiễu người ta sử dụng các bộ lọc tuyến tính (lọc trung bình, lọc thông thấp) hay lọc phi tuyến (trung vị, giả trung vị, đồng hình v.v...).

2.2.2.4. Giãn ảnh

Giãn ảnh là kỹ thuật nhằm loại bỏ các điểm đen bị vây bởi các điểm trắng. Trong kỹ thuật này một cửa sổ hay còn gọi là mặt nạ (mask) được rê đi khắp ảnh và thực hiện đối sánh một pixel với các điểm lân cận (không tính điểm ở tâm). Thuật toán giãn ảnh có thể được mô tả như sau:

For all pixels $I(x,y)$ {I là ảnh lối vào}

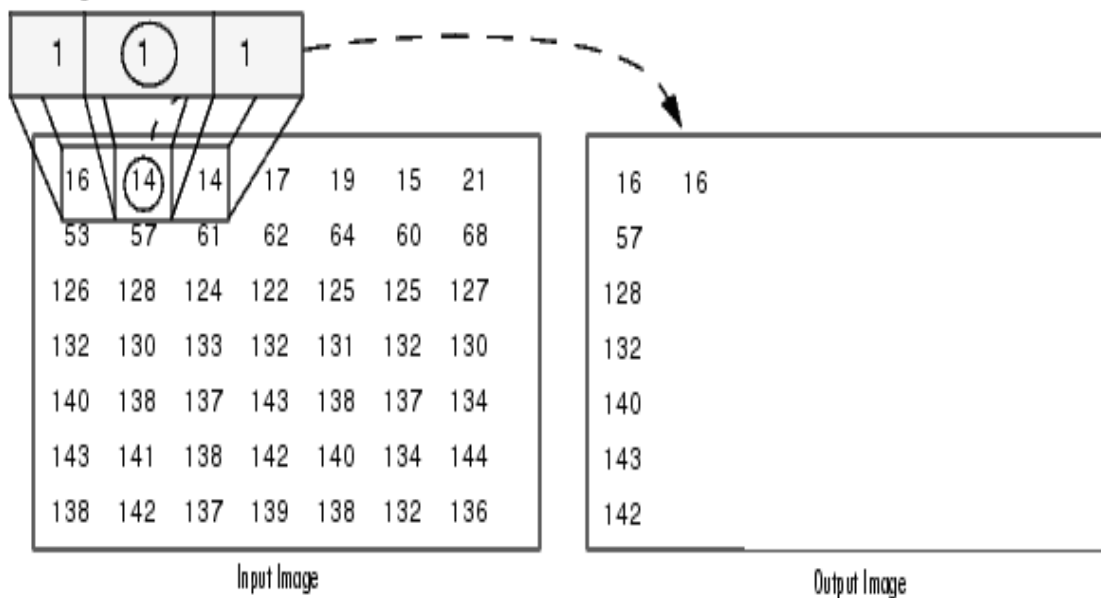
{

- Tính $F(x,y) = \text{Max}\{ I(x,y)_{in\ m*n} \}$;

- $O(x,y) = F(x,y)$; {O là ảnh lối ra}

}

Ví dụ sử dụng một cửa sổ 1 x 3 sánh với phần tử 1 x 2 $I(1,2)$ trong ma trận điểm ảnh đa cấp xám thì phần tử $I(1,2)$ đang có giá trị là 14 sẽ có giá trị là giá trị lớn nhất trong khung cửa sổ 1 x 3 đó là giá trị 16.



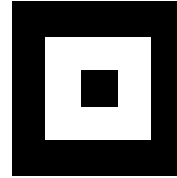
Hình 2. 5 Giãn ảnh

Ví dụ với một ảnh nhị phân 10x10 như sau:

```

0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 0 0
0 0 1 1 1 1 1 1 0 0
0 0 1 1 0 0 1 1 0 0
0 0 1 1 0 0 1 1 0 0
0 0 1 1 1 1 1 1 0 0
0 0 1 1 1 1 1 1 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0

```



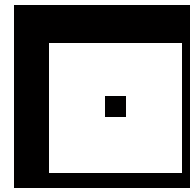
Hình 2. 6 Ảnh gốc

Sử dụng một cửa sổ là ma trận đơn vị 2×2 I(2,2) di trên toàn bộ ảnh và thực hiện thuật toán giãn ảnh ta thu được ảnh như sau:

```

0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 1 0
0 0 1 1 1 1 1 1 1 0
0 0 1 1 1 1 1 1 1 0
0 0 1 1 1 0 1 1 1 0
0 0 1 1 1 1 1 1 1 0
0 0 1 1 1 1 1 1 1 0
0 0 1 1 1 1 1 1 1 0
0 0 0 0 0 0 0 0 0 0

```



Hình 2. 7 Ảnh sau khi được giãn ảnh

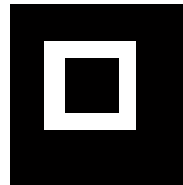
2.2.2.5. Co ảnh

Co ảnh là thao tác đối ngẫu của giãn ảnh nhằm loại bỏ điểm trắng bị vây quanh bởi các điểm đen. Trong kĩ thuật này một cửa sổ $m \times n$ được rê đi khắp ảnh và thực hiện sánh một pixel của ảnh với các điểm lân cận. Thuật toán co ảnh có thể được mô tả như sau:

For all pixels $I(x,y)$ { I là ảnh lối vào}
{
- Tính $F(x,y) = \text{Min}\{ I(x,y)_{in\ m \times n} \}$;
- $O(x,y) = F(x,y)$; { O là ảnh lối ra}
}

Thực hiện co ảnh với ảnh gốc ở phần trên và vẫn sử dụng ma trận đơn vị $I(2,2)$ làm cửa sổ ta thu được ảnh như sau:

0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0	0
0	0	1	0	0	0	1	0	0	0
0	0	1	0	0	0	1	0	0	0
0	0	1	0	0	0	1	0	0	0
0	0	1	1	1	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0



Hình 2. 8 Ảnh thu được sau khi co ảnh

2.3. Xây dựng thuật toán

2.3.1. Xử lý bản đồ thực tế.

Để máy tính có thể xử lý dữ liệu và thực hiện thuật toán thì, bản đồ thực tế phải được chuyển sang dạng số. Vì thế ta phải thực hiện xử lý hình ảnh và chuyển bản đồ thực tế sang dữ liệu số.

-Bản đồ thực tế:



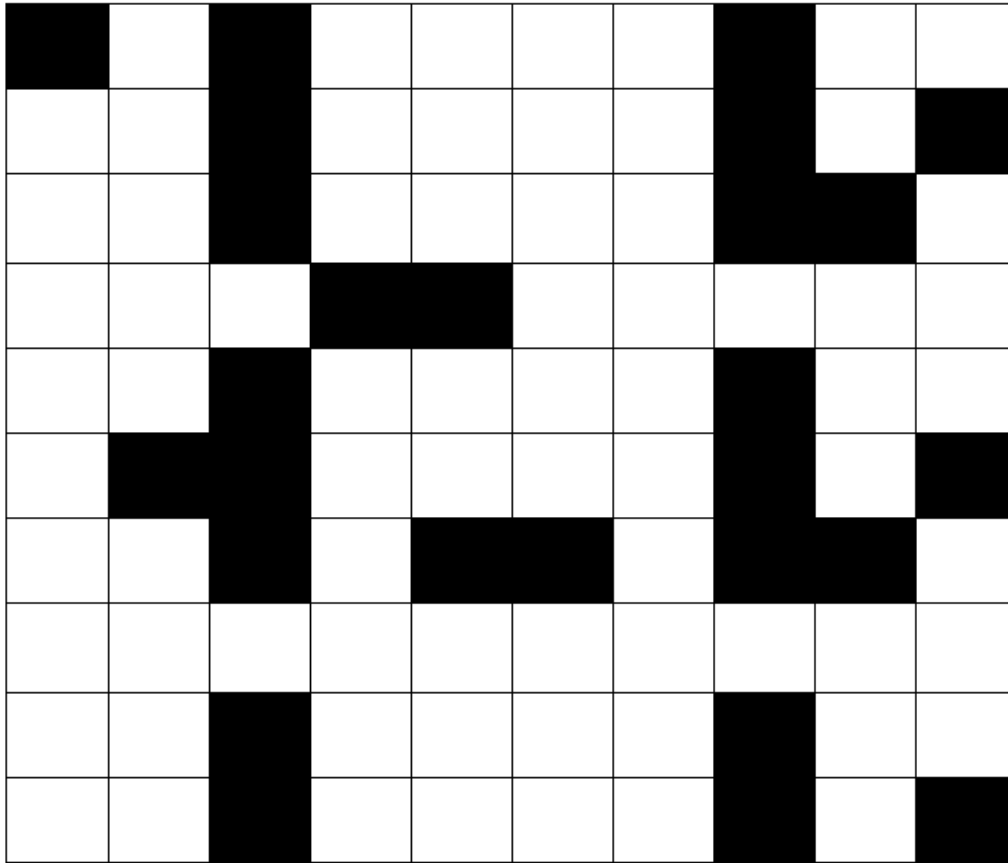
Hình 2. 9 Bản đồ thực tế sảnh khu D tầng 4

+Diện tích tổng: 600x600 cm

+Diện tích 1 ô: 60x60 cm

+Diện tích vật cản chiếm giữ: 10x10 cm

-Từ bản đồ thực tế ta xây dựng được bản đồ 10x10 như sau:



Hình 2. 10 Bản đồ 10x10 mô phỏng bản đồ thực tế.

+Trong đó ô màu đen đại diện cho ô có vật cản chiếm giữ và được gán giá trị logic “1”, các ô còn lại thuộc về không gian tự do, có giá trị “0”.

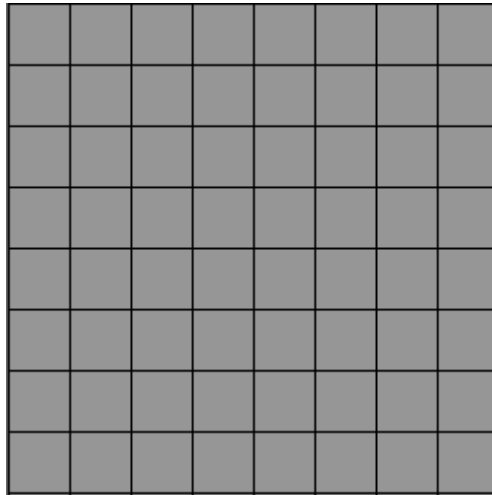
-Tiếp theo, ta sử dụng thư viện opencv và C++ để chuyển bảng đồ 10x10 sang dữ liệu số và ta thu được bản đồ dữ liệu số:

1	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	1	0	1
0	0	1	0	0	0	0	1	1	0
0	0	0	1	1	0	0	0	0	0
0	0	1	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	1
0	0	1	0	1	1	0	1	1	0
0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	1	0	1

Hình 2. 11 Dữ liệu số được xuất ra

2.3.2. Thuật toán A*

-Máy tính sẽ đảm nhận việc xử lý giải thuật và đưa ra chỉ thị cho robot. Robot sẽ di chuyển trên một bản đồ dạng lưới caro.

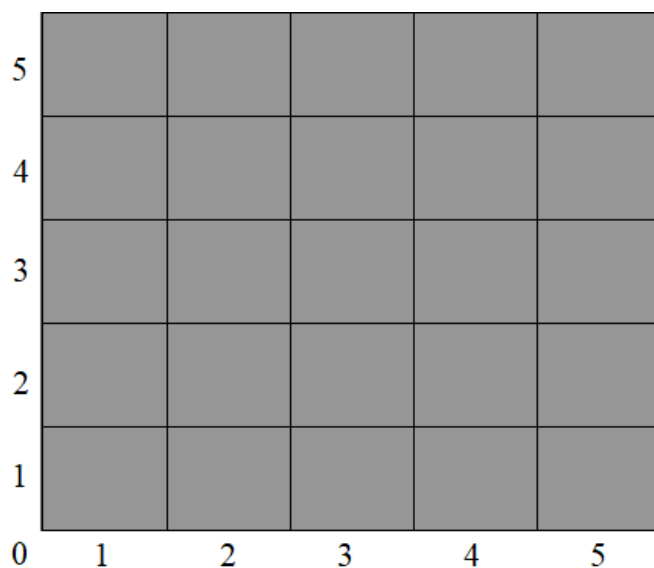


Hình 2. 12 Khu vực hoạt động của robot

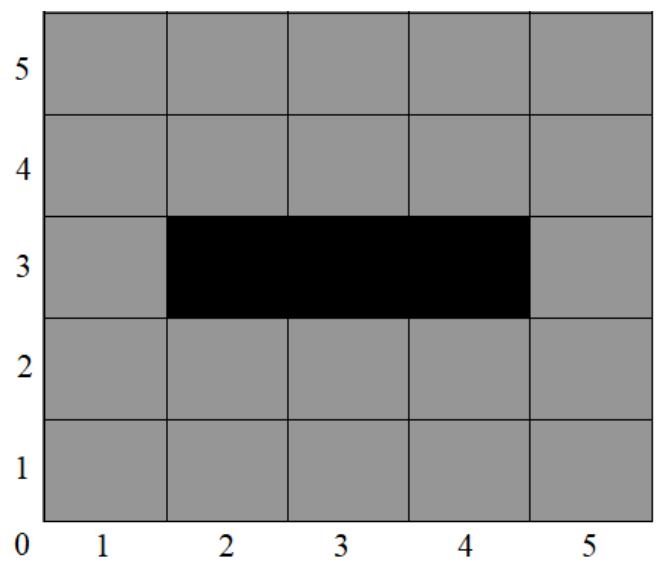
-Người dùng sẽ nhập vào máy tính các input bao gồm: kích thước bản đồ, tọa độ hiện tại của robot, tọa độ của đích đến, tọa độ của các vật cản. Máy tính sẽ xử lý và đưa ra output là lộ trình đường đi tối ưu (tối ưu về thời gian, quãng đường hoặc cả hai).

-Mô tả thuật toán:

+Dữ liệu bản đồ nhập vào sẽ được duyệt và lưu vào đồ thị D.



a, Không có vật cản

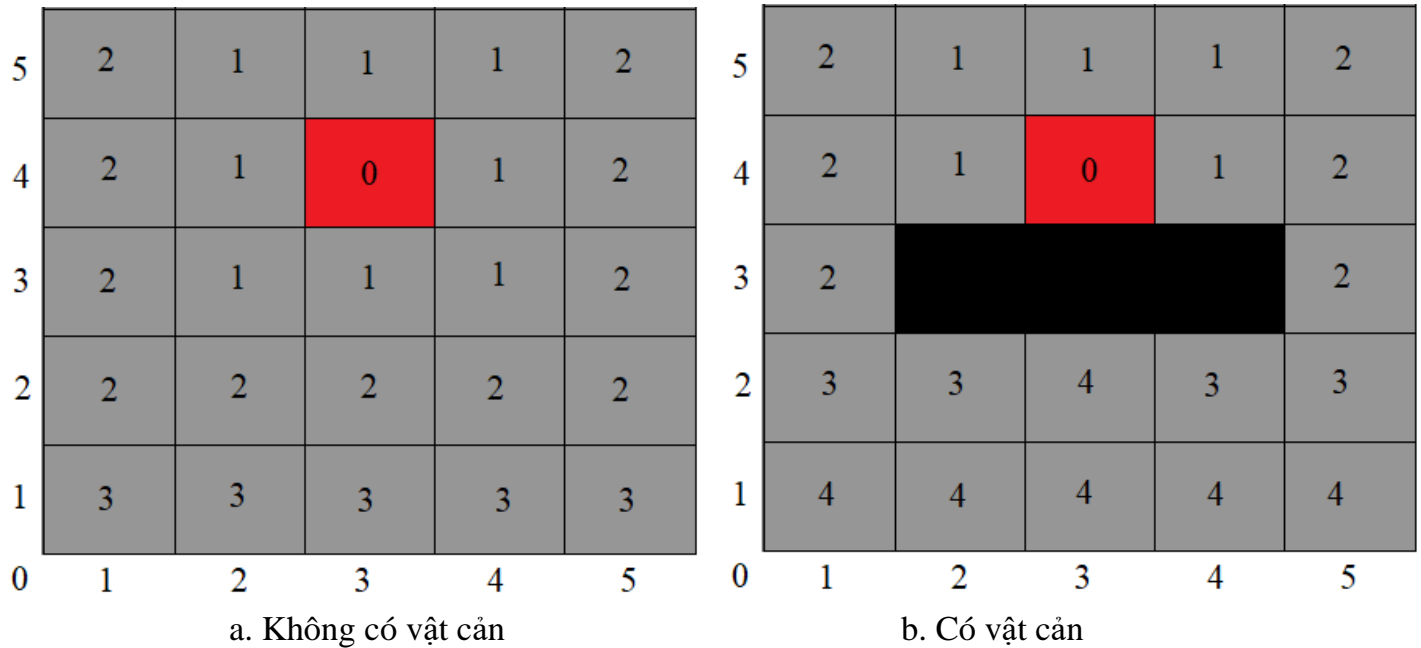


b, Có vật cản

Hình 2. 13 Bản đồ được duyệt và lưu vào đồ thị D

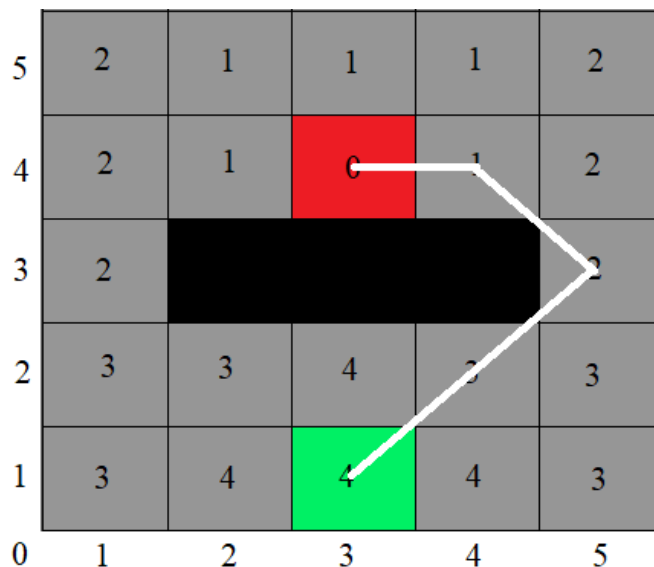
- Trong đó vùng màu xám là vùng tự do không có vật cản, vùng màu đen là vùng có vật cản chiếm giữ.

+Chọn vị trí bắt đầu (đỏ) của robot, thuật toán sẽ duyệt đồ thị D và gán độ dài ngắn nhất robot đi từ điểm bắt đầu đến tất cả các ô trong đồ thị, sau đó lưu vào một mảng B.



Hình 2. 14 Bản đồ được duyệt khi chọn vị trí bắt đầu

+Từ đó khi chọn vị trí đích (xanh) của robot, robot dễ dàng xác định được đường đi thông qua độ dài khoảng cách đã duyệt và lưu trong mảng B.

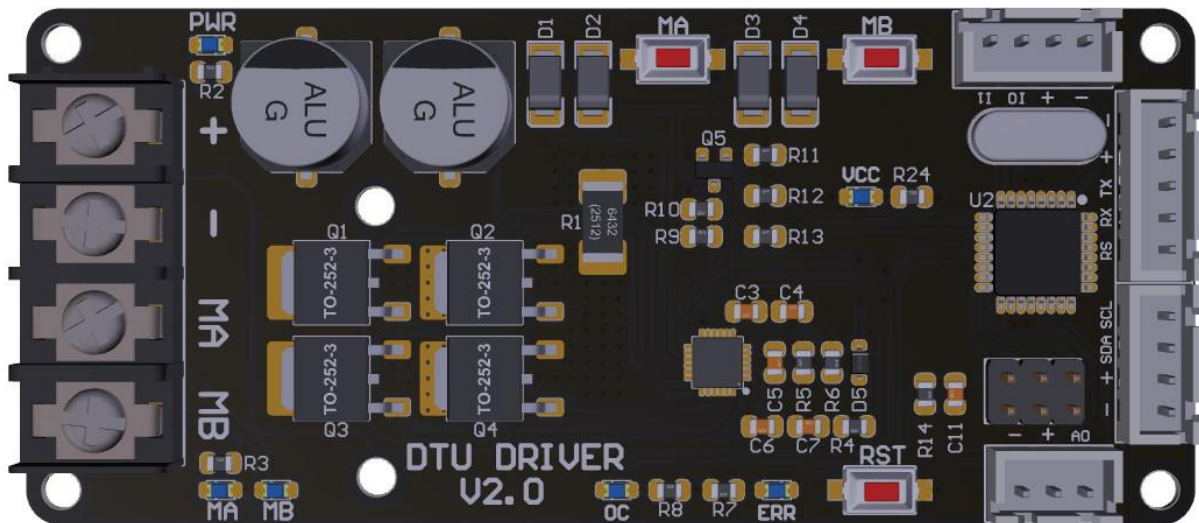


Hình 2. 15 Đường đi tìm được

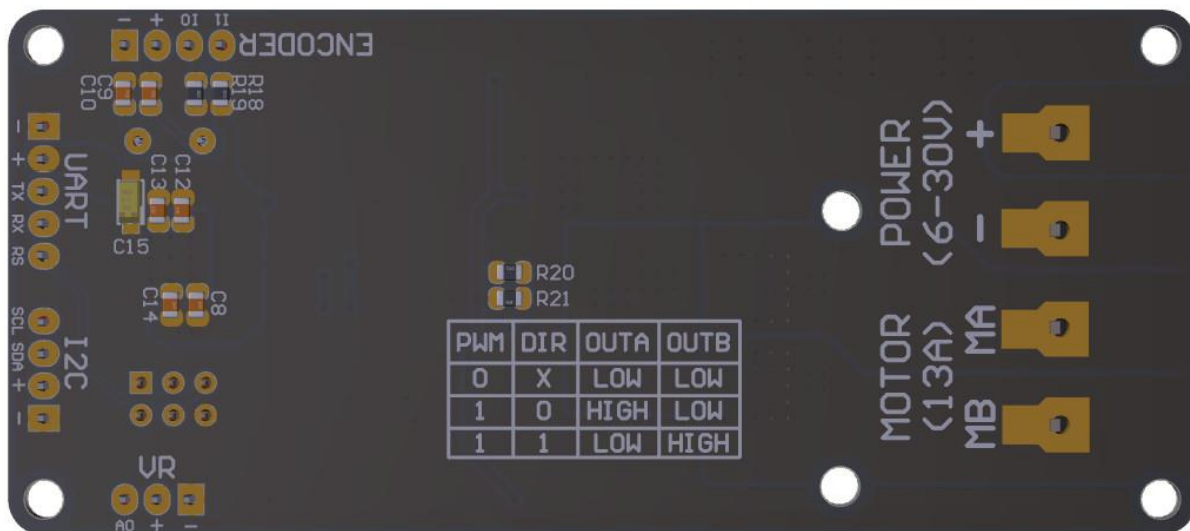
2.4. Thiết kế robot

2.4.1. Linh kiện

2.4.1.1 Mạch điều khiển động cơ DTU Drive



a. Mặt trước



b. Mặt sau

Hình 2. 16 Mạch điều khiển động cơ DTU Drive.

2.4.1.2. Động Cơ DC Servo Giảm Tốc Hành Tinh Planetary GP36

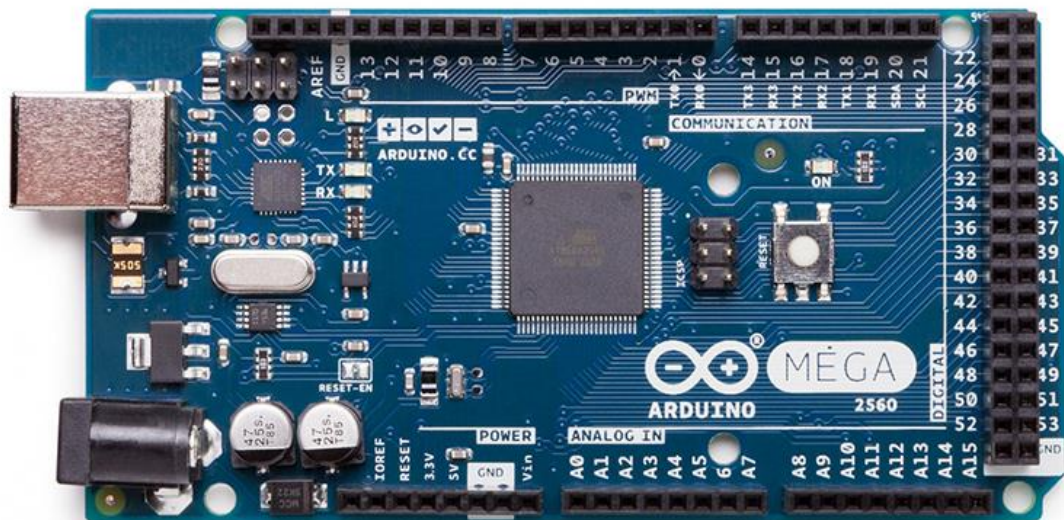


Hình 2. 17 Động Cơ DC Servo Giảm Tốc Hành Tinh Planetary GP36

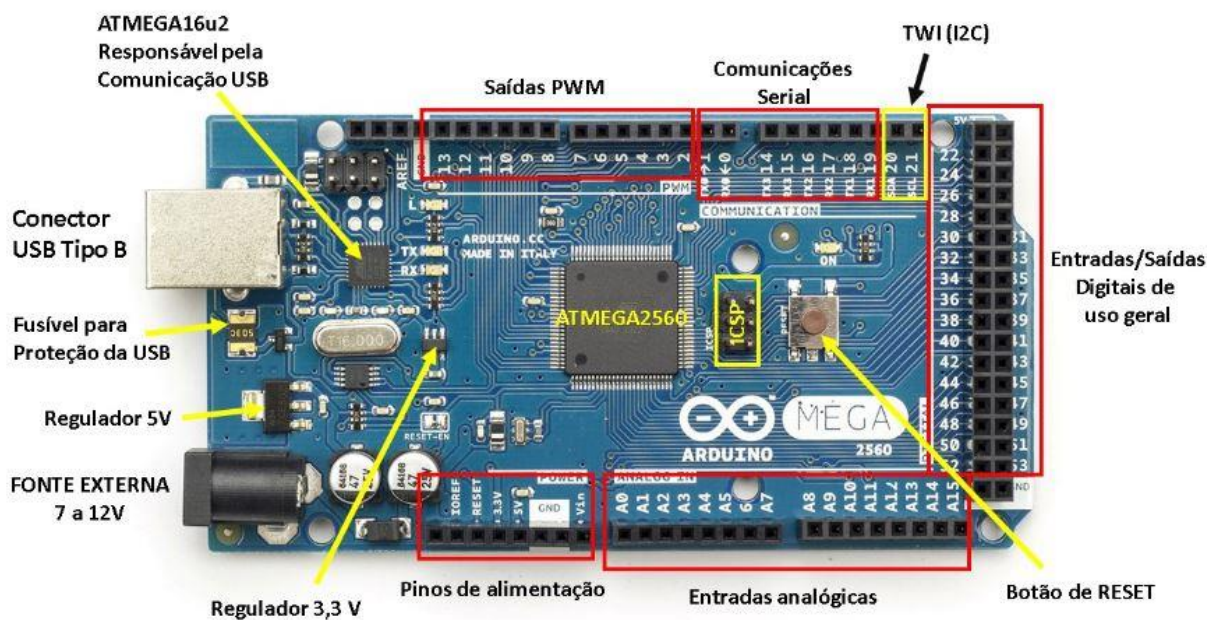
Thông số kỹ thuật:

- Điện áp sử dụng: 12VDC.
- Dòng tiêu thụ: 3A
- Stalling current: 7A (dòng khi động cơ bị kẹt hoặc quá tải đứng im).
- Encoder quang 500CPR (Count Per Round), 2 kênh A-B.
- Điện áp cấp cho Encoder: 5VDC, tín hiệu xuất ra 2 kênh TTL 0/5VDC
- Hộp số kim loại giảm tốc hành tinh Planetary.
- Tỷ số giảm tốc 1:14 hoặc 1: 27 (tùy chọn)
- Moment lực kéo:
 - 6.2Kg.cm (1:14 gear ratio) - 280rpm
 - 12.2Kg.cm (1:27 gear ratio) - 145rpm

2.4.1.3. Vi điều khiển Arduino Mega 2560



Hình 2. 18 Vi điều khiển Arduino Mega 2560



Hình 2. 19 Sơ đồ chân vi điều khiển Arduino Mega 2560

2.4.1.4. Giới thiệu vi điều khiển Arduino Mega 2560

Trước hết, tại sao tại sao chúng ta nên lựa chọn Arduino Mega 2560?

Vì đây là một bo mạch được tích hợp nhiều tính năng nổi bật. Tính năng đầu tiên là thiết kế hệ thống I / O lớn với 16 bộ chuyển đổi tương tự và 54 bộ chuyển đổi digital

hỗ trợ UART và các chế độ giao tiếp khác. Thứ hai, Arduino Mega 2560 có sẵn RTC và các tính năng khác như bộ so sánh, timer, ngắt để điều khiển hoạt động, tiết kiệm điện năng và tốc độ nhanh hơn với xung thạch anh 16 Mhz.

Các tính năng khác bao gồm hỗ trợ JTAG để lập trình, gỡ lỗi và xử lý sự cố. Với bộ nhớ FLASH lớn và SRAM, bo này có thể xử lý chương trình hệ thống lớn một cách dễ dàng. Nó cũng tương thích với các loại bo mạch khác nhau như tín hiệu mức cao (5V) hoặc tín hiệu mức thấp (3.3V) với chân nạp I / O.

Brownout và watchdog giúp hệ thống đáng tin cậy và mạnh mẽ hơn. Nó hỗ trợ ICSP cũng như lập trình vi điều khiển USB với PC.

Arduino Mega 2560 là một sự thay thế của Arduino Mega cũ. Nó thường được sử dụng cho các dự án rất phức tạp.

-Thông số kỹ Thuật:

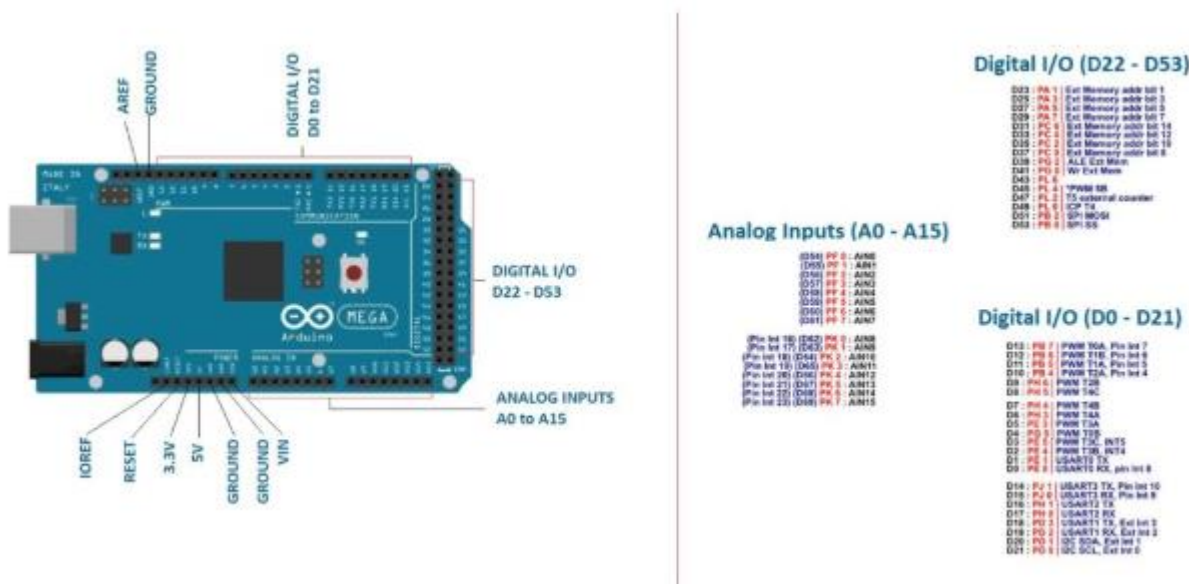
Arduino Mega	Tính năng, đặc điểm
Vi điều khiển	AVR ATmega 2560 (8bit)
Nguồn cung cấp	7-12V (Bộ điều chỉnh sẵn có cho bộ điều khiển)
Số chân I/O số	54
Số chân I/O tương tự	16
Xung clock	16 MHz (nhà sản xuất cài đặt là 1MHz)
Bộ nhớ flash	128 KB
SRAM	8 KB
Giao tiếp	USB (Lập trình với ATmega 8), ICSP (lập trình), SPI, I2C và USART
Bộ Timer	2 (8bit) + 4 (16bit) = 6 Timer
PWM	12 (2-16 bit)
ADC	16 (10 bit)
USART	4
Ngắt thay đổi chân	24

Arduino Mega 2560 cũng được tích hợp các tính năng bổ sung như bộ so sánh, ngắt, chế độ tiết kiệm điện, cảm biến nhiệt độ, RTC và nhiều tính năng khác.

-Chân nguồn:

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

-Sơ đồ chân Arduino Mega:



Hình 2. 20 Sơ đồ chân Arduino Mega

-Chân điều khiển:

RESET: Arduino Mega Mega 2560 có sẵn mạch reset với nút ấn để thiết lập lại hệ thống và chân này có thể được sử dụng khi kết nối các thiết bị khác để thiết lập lại bộ điều khiển.

XTAL1, XTAL2: Thạch anh(16Mhz) được kết nối với xung clock cung cấp cho bộ điều khiển.

AREF: Chân này được dùng khi sử dụng ADC để chuyển đổi tín hiệu với điện áp tham chiếu bên ngoài mà không muốn sử dụng điện áp tham chiếu nội bộ 1.1V hoặc 5V.

-Các chân Digital (70):

Chân số: Từ 0-53 (số) và 0-15 (tương tự) có thể được sử dụng làm đầu vào hoặc đầu ra cho thiết bị được thiết lập bằng các hàm Mode (), digitalWrite (), digitalRead ().

Ứng dụng:

Thiết bị đầu ra: Relay, LED, buzzer, LCD và các thiết bị khác.

Thiết bị đầu vào: Nút ấn, cảm biến siêu âm, cần điều khiển và các thiết bị khác

Thí dụ:

Đầu ra tín hiệu thấp trên Arduino mega

```
pinMode (0, OUTPUT);
```

```
digitalWrite (0, LOW);
```

Đầu vào đọc tín hiệu trên Arduino mega

```
pinMode (0, INPUT);
```

```
digitalRead (0);
```

-Chân tương tự (16):

Từ 0-15 (analog) có thể được sử dụng như chân đầu vào tương tự cho bộ ADC, nếu không sử dụng nó hoạt động như chân digital bình thường. Nó được thiết lập bởi các hàm `pinMode ()` khai báo chân, `analogRead ()` để đọc trạng thái chân và nhận giá trị kỹ thuật số cho tín hiệu analog. Lưu ý phải cẩn thận để lựa chọn điện áp tham chiếu bên trong hoặc bên ngoài và chân Aref.

Ứng dụng :

Thiết bị đầu vào: Cảm biến nhiệt độ, cảm biến (như ldr, irled và độ ẩm) và các thiết bị khác

Thí dụ :

```
pinMode (0, INPUT);
```

```
analogRead (0);
```

-Chân có Chức năng thay thế:

+Chân SPI: Chân 22-SS, 23-SCK, 24-MOSI, 25-MISO

Các chân này được sử dụng cho giao tiếp nối tiếp với giao thức SPI để liên lạc giữa 2 thiết bị trở lên. SPI cho phép bit phải được thiết lập để bắt đầu giao tiếp với các thiết bị khác.

Ứng dụng: Lập trình điều khiển AVR, giao tiếp với những người khác ngoại vi như LCD và thẻ SD.

+Chân I2C:

Chân 20 cho SDA và 21 cho SCK (Tốc độ 400khz) để cho phép liên lạc hai dây với các thiết bị khác. Hàm được sử dụng là `wire.begin ()` để bắt đầu chuyển đổi I2C, với `wire.Read ()` để đọc dữ liệu i2c và `wire.Write ()` để ghi dữ liệu i2c.

Ứng dụng:

Thiết bị đầu ra: LCD và liên lạc giữa nhiều thiết bị với hai dây.

Thiết bị đầu vào: RTC và các thiết bị khác.

Thí dụ:

I2c chỉ đọc dữ liệu

```
Wire.begin ();
```

```
Wire.requestFrom (2, 1); // dữ liệu 1byte
```

```
Wire.Read ();
```

+PWM chân :

Chân 2-13 có thể được sử dụng như đầu ra PWM với hàm analogWrite () để ghi giá trị pwm từ 0-255.

Ứng dụng:

Thiết bị đầu ra: Điều khiển tốc độ của động cơ, ánh sáng mờ, pid cho hệ thống điều khiển hiệu quả.

+Chân USART:

Chân 0 - RXD0, chân 1 - TXD0

Chân 19 - RXD1, chân 18 - TXD1

Chân 17 - RXD2, chân 16 - TXD2

Chân 15 - RXD3, chân 14 - TXD3

Chân này được sử dụng cho giao tiếp nối tiếp giữa bo mạch với máy tính hoặc hệ thống khác để chia sẻ và ghi dữ liệu. Nó được sử dụng với hàm serialBegin () để cài đặt tốc độ truyền và bắt đầu truyền thông với serial.Println () để in mảng ký tự (mảng char) ra thiết bị khác.

Ứng dụng:

Bộ điều khiển truyền thông và máy tính

Ví dụ :

```
Serial.begin (9600);
```

```
Serial.Println ("hello");
```

+Chân ngắt :

Chân digital: 0,22,23,24,25,10,11,12,13,15,14

Chân analog: 6,7,8,9,10,11,12,13,14,15

Chân này được sử dụng để ngắt. Để kích hoạt chân ngắt phải cài đặt bật ngắt toàn cục.

Ứng dụng :

Bộ mã hóa vòng quay, nút bấm dựa trên ngắt và các nút khác.

Thí dụ :

```
pinMode (0, OUTPUT);
```

```
pinMode (1, INPUT_PULLUP);
```

```
attachInterrupt (digitalpinToInterrupt (1), LOW, CHANGE);
```

+Chân ngắt phần cứng:

Chân 18 - 21,2,3 ngắt phần cứng được sử dụng cho các ứng dụng ngắt. Ngắt phần cứng phải được bật với tính năng ngắt toàn cục để ngắt quãng từ các thiết bị khác.

Ứng dụng:

Nhấn nút cho chương trình ISR, đánh thức bộ điều khiển bằng thiết bị bên ngoài như cảm biến siêu âm và các thiết bị khác.

Thí dụ:

```
pinMode (0, OUTPUT);
```

```
pinMode (1, INPUT_PULLUP);  
attachInterrupt (digitalpinToInterrupt (1), LOW, LOW);
```

-Các phần của Arduino Mega:

+Giắc cắm **nguồn DC**:

Cấp nguồn cho Arduino Mega từ 7-12V qua cổng này. Arduino Mega R3 có bộ điều chỉnh điện áp nguồn cấp 5V và 3.3V cho bộ điều khiển Arduino và bộ cảm biến.

+**AVR 2560**:

Đây là vi điều khiển chính được sử dụng để lập trình và chạy tác vụ cho hệ thống. Đây là bộ não của hệ thống để điều khiển tất cả các thiết bị khác trên mạch.

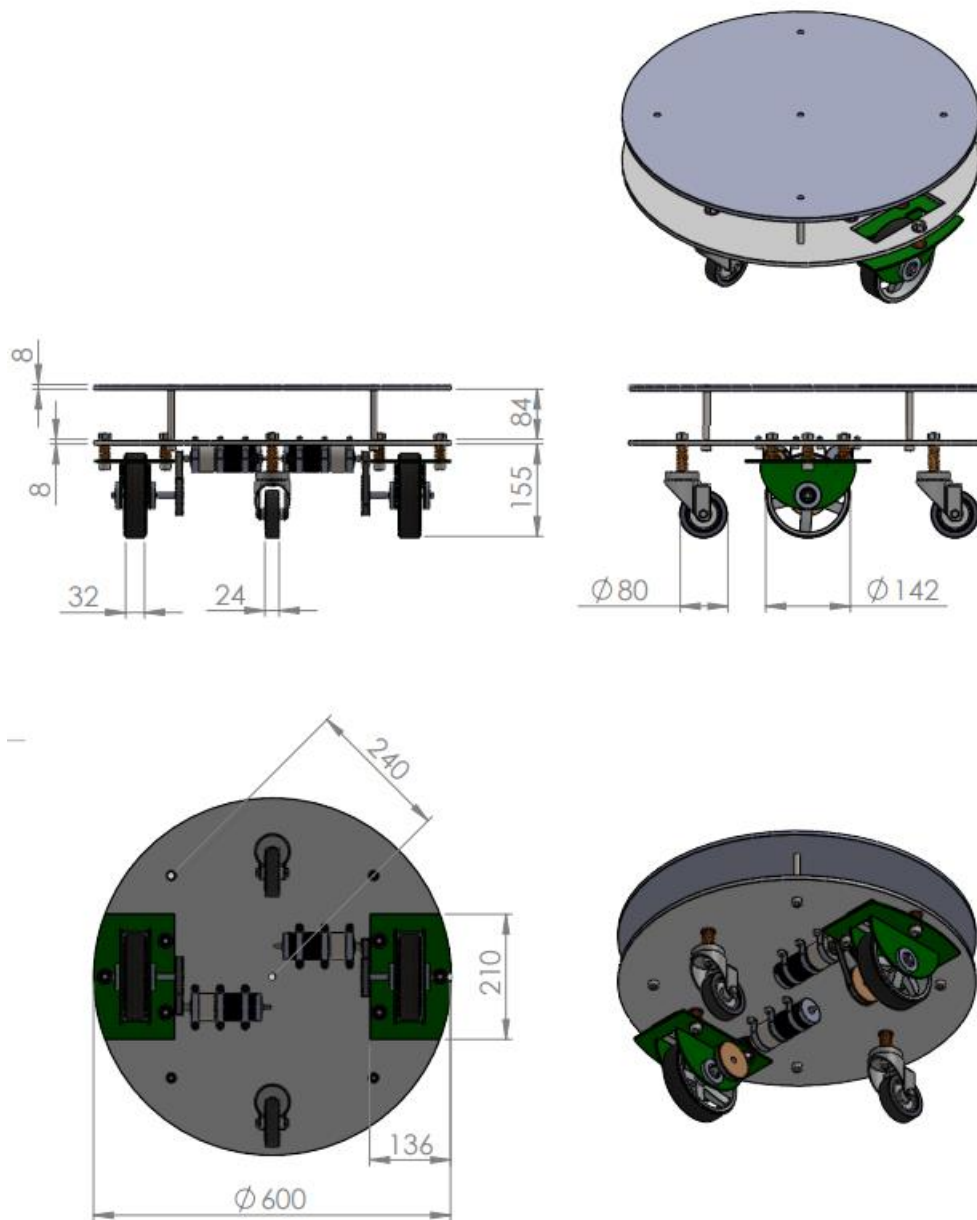
+**ATmega8**:

Vi điều khiển này được sử dụng để liên lạc giữa bộ điều khiển chính và các thiết bị khác. Bộ điều khiển này được lập trình cho giao tiếp USB và các tính năng lập trình nối tiếp.

+**ICSP 1 (ATmega8) và 2 (AVR 2560)**:

Nó có các tính năng của lập trình sử dụng bus nối tiếp với lập trình AVR sử dụng giao tiếp SPI. AVR 2560 được lập trình để chạy hệ thống và ATmega 8 được lập trình để liên lạc và lập trình nối tiếp.

2.4.2. Thiết kế Robot



Hình 2. 21 Bản vẽ khung robot.

-Thông số robot:

+Chiều cao: 250 mm

+Đường kính: 600 mm

+Đường kính bánh xe lớn: 142 mm

+Đường kính bánh xe nhỏ: 80 mm

-Robot được cấu tạo gồm 2 phần:

+Phần động cơ: Gồm 2 động Cơ DC Servo Giảm Tốc Hành Tinh Planetary GP36 gắn vào 2 bánh xe lớn đường kính 100 mm, cùng với 2 bánh xe nhỏ đường kính 60mm có thể quay tự do.

+Phần điều khiển:

- Động cơ được gắn vào bộ điều khiển động cơ DTU Drive, mỗi động cơ là 1 bộ điều khiển.
- Vi điều khiển Arduino Mega 2560 sẽ tiếp nhận tín hiệu từ máy tính và điều khiển động cơ thông qua module DTU Drive.

CHƯƠNG 3: THỬ NGHIỆM VÀ THẢO LUẬN

3.1. Thử nghiệm

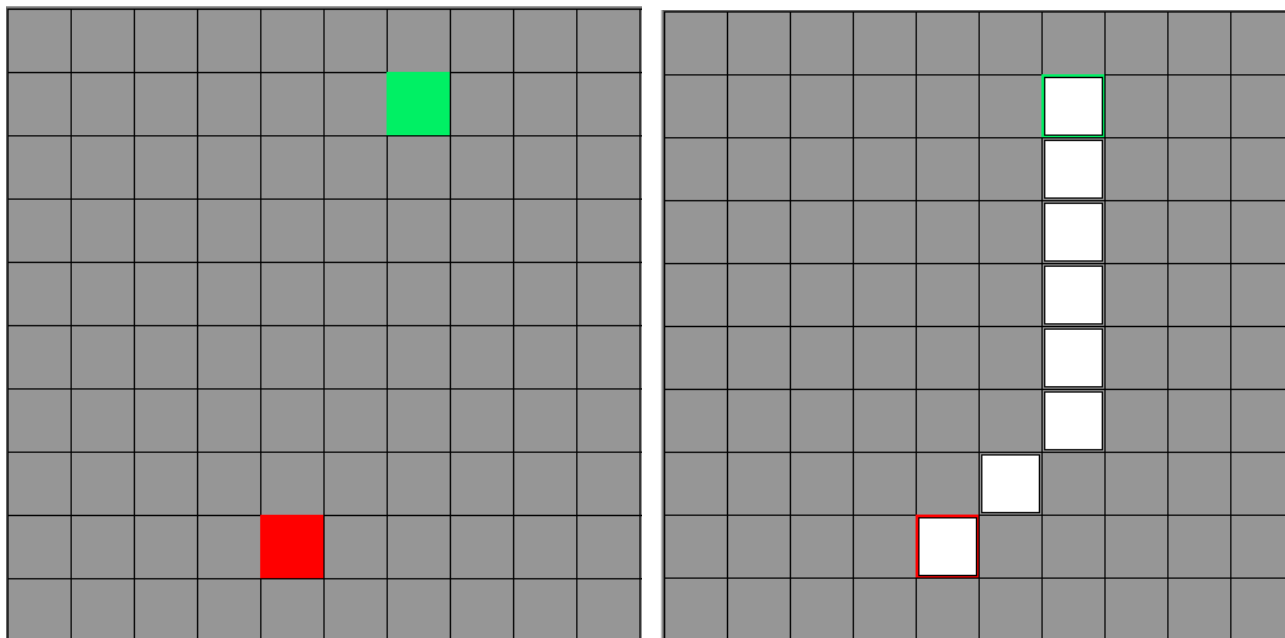
a, Mô phỏng

-Các lần thử nghiệm:

Đỏ: vị trí bắt đầu.

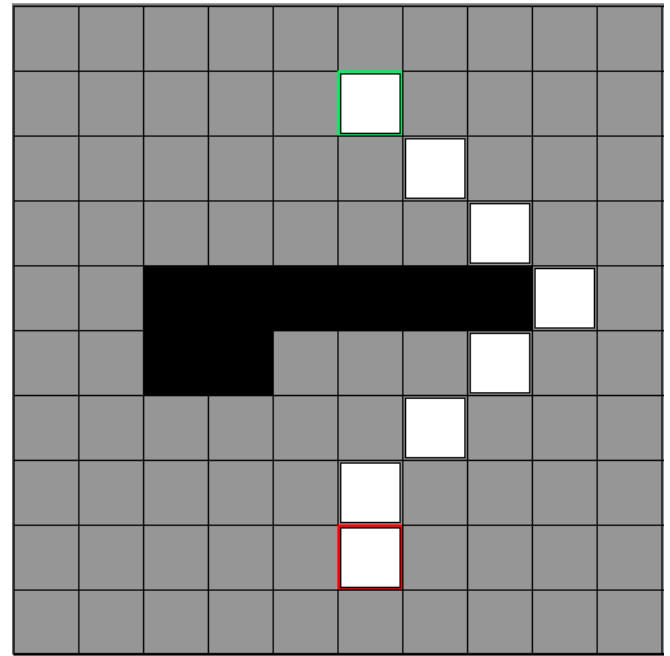
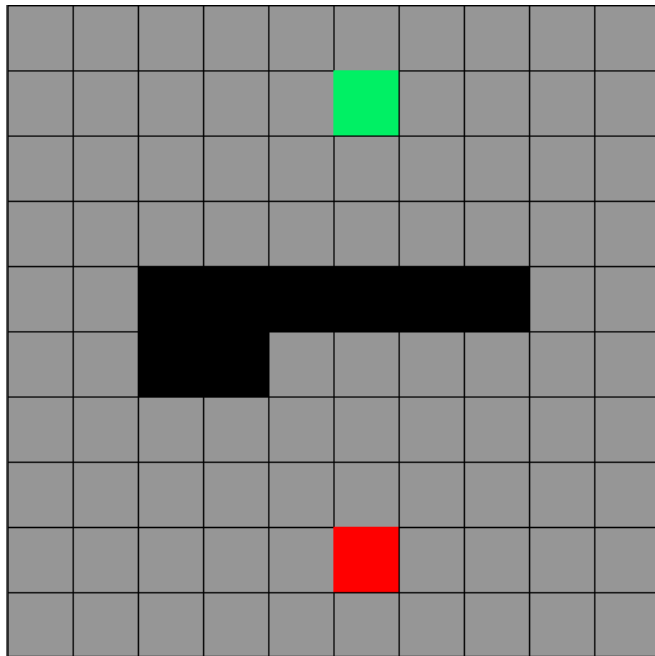
Xanh: vị trí kết thúc.

+Lần 1:



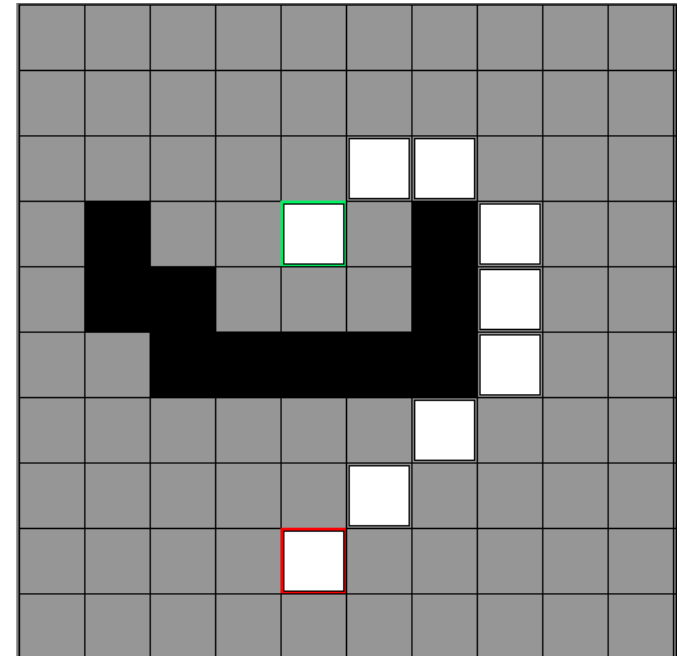
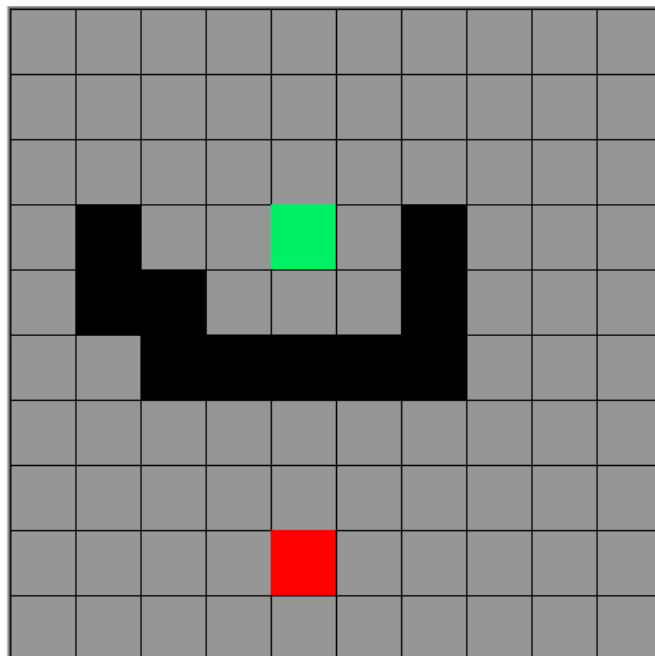
Hình 3. 1 Thử nghiệm lần 1

+Lần 2:



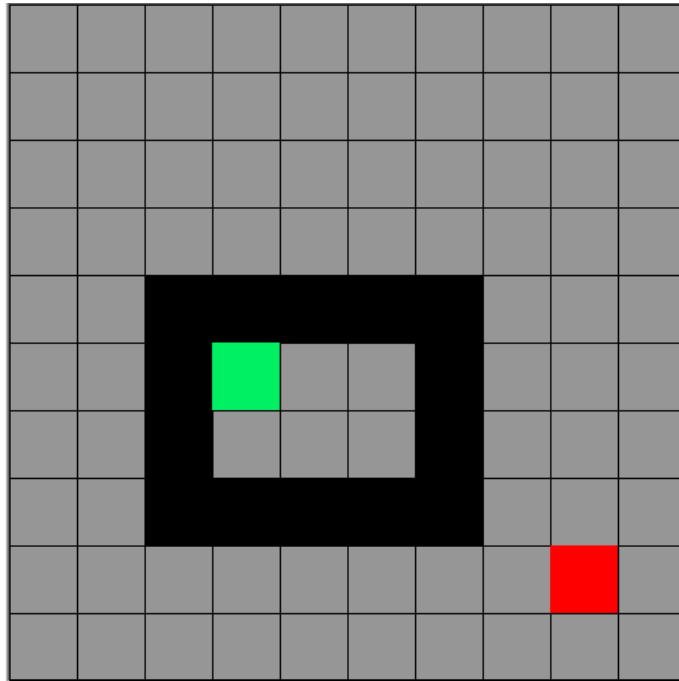
Hình 3. 2 Thực nghiệm lần 2

+Lần 3:



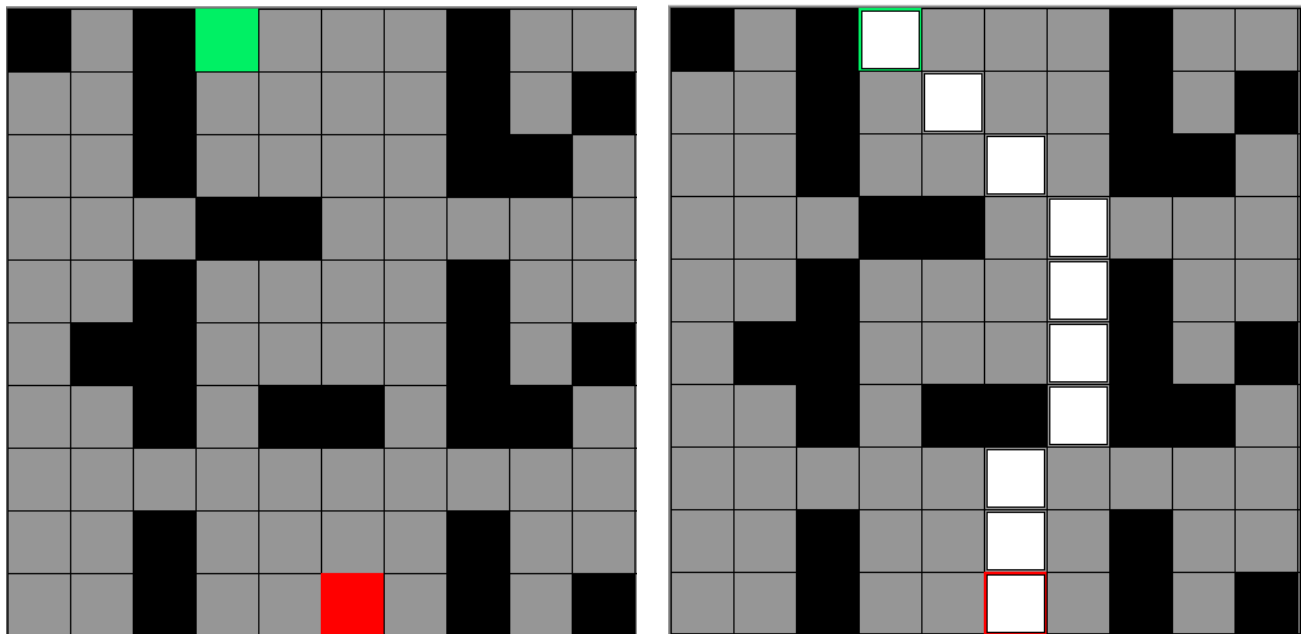
Hình 3. 3 Thực nghiệm lần 3

+Lần 4:



Hình 3. 4 Thực nghiệm lần 4

+Lần 5:



Hình 3. 5 Thực nghiệm lần 5

Bảng 3. 1 Bảng kết quả mô phỏng

Lần thử nghiệm	Thành công	Thất bại
1	x	
2	x	
3	x	
4		x
5	x	

-Đánh giá kết quả:

+ Qua 5 lần thử nghiệm thuật toán đã tìm được đường đi thành công 4 lần và thất bại 1 lần.

+ Lý do tìm kiếm thất bại là vì vị trí kết thúc nằm trong vùng có vật cản dày, không có vùng tự do để robot đi qua được.

+ Thuật toán chạy tốt tìm kiếm nhanh và tối ưu.

b, Thực tế

-Môi trường thực nghiệm:



Hình 3. 6 Hành lang khu D tần 4

+Diện tích tổng: 600x600 cm

+Diện tích 1 ô: 60x60 cm

+Vật cản nhân tạo là hình hộp vuông 10x10x10 cm, diện tích chiếm giữ trong 1 ô là 10x10 cm, ô đặt vật cản được tính là ô vật cản.

Môi trường thử nghiệm là hành lang khu D tầng 4 với các vật cản nhân tạo (hình 3.6).

-Robot thực nghiệm:



Hình 3. 7 Robot thực nghiệm

+Diện tích robot: 2826 cm²

+Chiều cao robot: 250 mm

-Kết quả thực nghiệm

Bảng 3. 2 Bảng kết quả thực nghiệm

Lần thử nghiệm	Thành công	Thất bại
1	x	
2	x	
3	x	
4		x
5	x	

Qua các lần thử nghiệm robot đều vận hành thành công, tuy nhiên vào lần thử nghiệm thứ 4 do robot to đã va chạm với vật cản.

-Robot vận hành:

+Dữ liệu đường đi lấy từ thực nghiệm 5



Hình 3. 8 Robot xuất phát



Hình 3. 9 Robot gặp vật cản



Hình 3. 10 Robot xoay tránh vật cản



Hình 3. 11 Robot tránh được vật cản



Hình 3. 12 Robot xoay tiến về đích



Hình 3. 13 Robot về đích

3.2. Đánh giá

- Thuật toán chạy đúng đã vạch ra được đường đi ngắn nhất, tối ưu nhất.
- Robot chạy đúng theo đường đi của thuật toán và hoàn thành đường đi.
- Việc thiết kế robot to làm cho khả năng robot bị va chạm với vật cản, cần nghiên cứu thiết kế robot phù hợp với môi trường làm việc.

3.3. Hướng phát triển trong tương lai.

Sử dụng bộ điều khiển PID, đọc tín hiệu encoder hồi tiếp từ động cơ để có thể điều khiển động cơ một cách chính xác.

Thiết kế giao diện để người dùng có thể dễ dàng sử dụng.

Giao tiếp giữa robot và máy chủ thông qua wifi thay cho Bluetooth để tăng cự ly giao tiếp cũng như tương thích với những nền tảng truyền thông mới hiện nay.

Tích hợp camera để chụp ảnh liên tục địa hình và xử lý hình ảnh để xử lý ngay lập tức trước các thay đổi địa hình.

LỜI KẾT

Sau bốn tháng nghiên cứu em cũng đã hoàn thành đồ án. Thực sự, kết quả đồ án đã không được như kỳ vọng ban đầu. Một phần nguyên nhân là do em đã chưa phân tích kỹ đề tài, do đó chưa có sự chuẩn bị tốt nhất. Em đã không thể lường được việc có quá nhiều vấn đề phát sinh trong quá trình thực hiện đồ án. Những ngày đầu khi thực hiện đồ án, em đã vấp phải rất nhiều khó khăn, nhiều vấn đề phát sinh mà kiến thức em chưa đủ để giải quyết, cần phải tìm tòi học hỏi và vận dụng rất nhiều mới có thể hoàn thành đồ án. Từ đó em hiểu rằng không những học kiến thức ở trường là xong, mà phải biết vận dụng, học hỏi từ trong thực tế, những khó khăn khi thực hiện đồ án này chỉ là những viên sỏi nhỏ, còn những khó khăn to lớn hơn đang chờ trên con đường trở thành một kỹ sư giỏi.