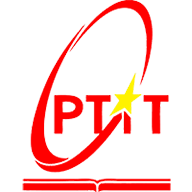


**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**KHOA CÔNG NGHỆ THÔNG TIN 1**



**ĐỀ TÀI TIỂU LUẬN:**

**ROBOT TRÁNH VẬT CẢN**

Môn học: Xây dựng các hệ thống nhúng

Giảng viên: Đỗ Tiến Dũng

Nhóm học phần: 02

Nhóm bài tập: 13

Họ và tên: Vũ Minh Hoàng

Mã sinh viên: B19DCCN285

Hà Nội, tháng 5 năm 2023

**Mục lục**

[I. Tổng quan đề tài. 2](#_Toc135589097)

[1. Giới thiệu đề tài. 2](#_Toc135589098)

[2. Mục đích chọn đề tài. 2](#_Toc135589099)

[3. Sơ lược các bước thực hiện. 2](#_Toc135589100)

[II. Giới thiệu về Arduino và các thành phần mạch 3](#_Toc135589101)

[1. Giới thiệu về Arduino 3](#_Toc135589102)

[1.1 Sơ lược về Arduino Uno R3 3](#_Toc135589103)

[1.2 I/O PINS 6](#_Toc135589104)

[2. Các thành phần của mạch 7](#_Toc135589105)

[2.1 Cảm biến siêu âm SFR 05 7](#_Toc135589106)

[2.2 Module điều khiển động cơ L298 8](#_Toc135589107)

[2.3 Động cơ giảm tốc 10](#_Toc135589108)

[2.4 Động cơ Servo SG90 11](#_Toc135589109)

[III. Nội dung 12](#_Toc135589110)

[1. Mô hình kết cấu Robot 12](#_Toc135589111)

[2. Sơ đồ khối và chức năng các khối 13](#_Toc135589112)

[2.1 Sơ đồ khối 13](#_Toc135589113)

[2.2 Chức năng các khối 13](#_Toc135589114)

[3. Thiết kế phần cứng điều khiển 14](#_Toc135589115)

[4. Chương trình điều khiển 17](#_Toc135589116)

[4.1 Code 17](#_Toc135589117)

[4.2 Giải thích 23](#_Toc135589118)

[IV. Kết quả đạt được 25](#_Toc135589119)

[1. Robot 25](#_Toc135589120)

[2. Kết quả chạy và đánh giá 25](#_Toc135589121)

[3. Mong muốn phát triển thêm 25](#_Toc135589122)

[4. Tài liệu tham khảo 26](#_Toc135589123)

# Tổng quan đề tài.

* 1. **Giới thiệu đề tài.**

Ngày nay, robot đã đạt được những thành tựu to lớn trong sản xuất công nghiệp cũng như trong đời sống. Sản xuất robot là ngành công nghiệp giá trị hàng tỉ USD và ngày càng phát triển mạnh, trong các họ robot chúng ta không thể không nhắc tới robot tránh vật cản với những đặc thù riêng mà các loại robot không có.

Mobile robot có thể di chuyển một cách linh hoạt, do đó tạo nên hoạt động lớn và cho đến nay nó đã dần khẳng định vai trò quan trọng không thể thiếu trong nhiều lĩnh vực, thu hút được nhiều sự đầu tư và nghiên cứu. Mobile robot cũng được chia làm nhiều loại: robot học đường đi, robot dò đường, robot tránh vật cản, robot tìm đường trong mê cung, … trong đó robot trán vật cản dễ dàng ứng dụng nhiều trong cuộc sống. Việc phát triển loại robot này sẽ phục vụ rất đắc lực cho con người.

* 1. **Mục đích chọn đề tài.**

Robot rò đường vừa có nhiều ứng dụng trong thực tế vừa dễ dàng để sinh viên vận dụng những kiến thức tiếp thu trên giảng đường vào nó. Với những kết cấu đơn giản nhưng lại có thể kết hợp với khá nhiều thành phần điện tử (sevor, cảm biến, arduino, module, …) nên những robot này rất phù hợp để sinh viên học tập và nghiên cứu thêm về ngành tự động hóa một cách cụ thể.

* 1. **Sơ lược các bước thực hiện.**

Trước tiên ta phải chế tạo được khung xe của robot. Khung xe phải đảm bảo bền chắc và đạt độ chính xác nhất định về việc bố trí các bánh xe và động cơ thông qua việc vẽ và lắp đặt động cơ.

Và cuối cùng là công đoạn lập trình dựa trên những kiến thức đã học.

# Giới thiệu về Arduino và các thành phần mạch

## Giới thiệu về Arduino

### 1.1 Sơ lược về Arduino Uno R3

Arduino Uno R3 là một bo mạch vi điều khiển dựa trên chip atmega328p. Uno R3 có 14 chân I/O digital (trong đó có 6 chân xuất xung PWM), 6 chân input analog, 1 thạch anh 16Mhz, 1 cổng USB, 1 jack nguồn DC, 1 nút reset.

A picture containing electronics, circuit, electronic engineering, electronic component

Description automatically generated

*Sơ đồ cấu trúc Arduino Uno R3*

**Thông số của Arduino**

|  |  |
| --- | --- |
| Vi điều khiển | Atmega328P |
| Điện áp hoạt động | 5V |
| Điện áp cấp(hoạt động tốt) | 7-12V |
| Điện áp cấp(giới hạn) | 6-12V |
| Chân I/O digital | 14 (có 6 chân xung PWM) |
| Chân Input Analog | 6(A0-A5) |
| Dòng điện mỗi chân I/0O | 20 mA |
| Dòng điện chân 3.3V | 50 mA |
| Bộ nhớ flash | 32 kB(Atmega328P)-trong đó  0.5kB dùng cho bootloader |
| SRAM | 2 kB(Atmega328P) |
| EEPROM | 1 kB(Atemega328P) |
| Tốc độ xung nhịp | 16 Mhz |
| Kích thước | 68.6 x 53.4 mm |
| Trọng lượng | 25g |

Sơ đồ chân vi điều khiển Atmega328P

A picture containing text, menu, font, screenshot

Description automatically generated

* Digital: Các chân I/O digital( chân số 2-13)được sử dụng làm chân nhập, xuất tín hiệu số thông qua các hàm chính: pinMode(), digitalWrite(), digitalRead(). Điện áp hoạt động lag 5V, dòng điện qua các chân này ở chế độ bình thường là 20mA, cấp dòng quá 40mA sẽ phá hỏng vi điều khiển.
* Analog: Uno có 6 chân Input analog (A0 – A5), độ phân giải mỗi chân là 10 bit (0 – 1023). Các chân này dùng để đọc tín hiệu điện áp 0 – 5V (mặc định) tương ứng với 1024 giá trị, sử dụng hàm analogRead().
* PWM: Các chân được đánh số 3,5,6,9,10,11; có chức năng cấp xung PWM (8bit) thông qua hàm analogWrite().
* UART: Atmega328P cho phép truyền dữ liệu thông qua 2 chân 0 (RX) và 1 chân (TX).

### 1.2 I/O PINS

Có hai cách cấp nguồn chính cho bo mạch Uno: cổng USB và jack DC.

Giới hạn điện áp cấp cho Uno là 6-20V. Tuy nhiên, dải điện áp khuyên dùng là 7-12V (tốt nhất là 9V). Lý do là nếu nguồn cấp dưới 7V thì điện áp ở chân 5V có thế thấp hơn 5V và mạch có thể hoạt động không ổn định; nếu nguồn cấp lớn hơn 12V có thể gấy nóng bo mạch hoặc phá hỏng.

Các chân nguồn trên Uno:

-Vin: chúng ta có thể cấp nguồn cho Uno thông qua chân này. Cách cấp nguồn này ít được sử dụng.

-5V: chân này có thể cho nguồn 5V từ bo mạch Uno. Việc cấp nguồn vào chân này hay chân 3.3V đều có thể phá hỏng bo mạch.

-3.3V: chân này cho nguồn 3.3V và dòng điện maximum là 50mA.

-GND: chân nối đất.

## Các thành phần của mạch

### 2.1 Cảm biến siêu âm SFR 05

A close-up of a circuit board

Description automatically generated with low confidence

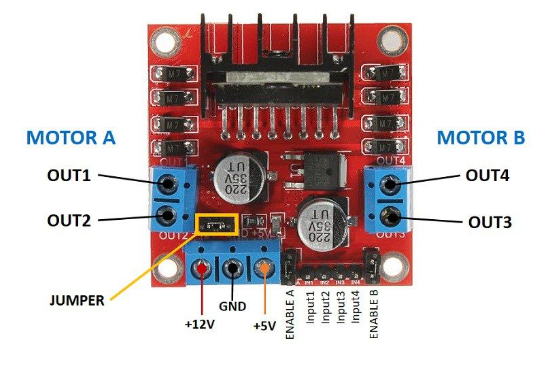
*Hình 2.2.1 Hình ảnh về SRF 05*

Thông số kỹ thuật:

* Điện áp hoạt động: 5VDC
* Khoảng cách phát hiện: 2cm-450cm
* Độ chính xác: ± 0.2cm
* Tín hiệu kích hoạt đầu vào: 10us xung TTL
* Kích thước: 43mm x 20mm x 17mm

Công dụng: phát ra sóng siêu âm và nhận sóng siêu âm phản hồi ngược lại khi gặp vật cản. Được ứng dụng trong việc đo khoảng cách, hay sử dụng để phát hiện, né tránh vật cản.

### 2.2 Module điều khiển động cơ L298

****

*Module điều khiển động cơ L298*

Thông số kỹ thuật:

* Module L298 có thể điều khiển 2 động cơ DC hoặc 1 động cơ bước, có 4 lỗ nằm ở 4 góc thuận tiện cho người sử dụng vị trí cố định cua module.
* Có gắn tản nhiệt chống nóng cho IC, giúp IC có thể điều khiển dòng đỉnh đạt 2A.
* IC L298N được gắn với các diode trên board giúp bảo vệ vi xử lý chống lại các dòng điện cảm ứng từ việc khởi động/ tắt động cơ.
* Driver: L298N tích hợp hai mạch cầu H
* Điện áp điều khiển: +5V ~ +12V
* Dòng tối đa cho mỗi mạch cầu H: 2A
* Điện áp của tín hiệu điều khiển: 0~ 36mA
* Công suất hao phí: 20W (khi nhiệt độ T=75°C)
* Nhiệt độ bảo quản: -25°C~ +130

Công dụng: IC L298 là một IC tích hợp nguyên khối gồm 2 mạch cầu H bên trong. Với điện áp làm tăng công suất nhỏ như động cơ DC loại vừa…

* Chân INPUT: IN1, IN2, IN3, IN4 được nối lần lượt với các chân 5,7,10,12 của L298. Đây là các chân tín hiệu nhận điều khiển
* Chân OUTPUT: OUT1, OUT2, OUT3, OUT4 (tương tự với các chân input) được nối với các chân 2,3,13,14 của L298. Các chân này sẽ được nối với động cơ.
* Hai chân ENA và ENB dùng để điều khiển mạch cầu H trong L298. Nếu ở mức logic “1” (nối với nguồn 5V) cho phép mạch cầu H hoạt động, nếu ở mức logic “0” thì mạch cầu H không hoạt động

Với bài toán của mình ở trên, các bạn chỉ cần lưu ý đến cách điều khiển chiều quay với L298:

* Khi ENA = 0: Động cơ không quay với mọi đầu vào.
* Khi ENA = 1:
* IN1 = 1; IN2 = 0: Động cơ quay thuận.
* IN1 = 0; IN2 = 1: Động cơ quay nghịch.
* IN1 = IN2: Động cơ dùng ngay tức thì.
* Với ENB cũng tương tự với IN3, IN4.

A picture containing text, diagram, line, plan

Description automatically generated

*Sơ đồ nguyên lý*

### 2.3 Động cơ giảm tốc

A picture containing cylinder, LEGO

Description automatically generated

*Động cơ giảm tốc*

Thông số kỹ thuật:

* Điện áp hoạt động: 3V~9V DC
* Momen xoắc cực đại: 800gf cm min 1:48(3V)
* Tốc độ không tải: 125 vòng/1 phút (3V)
* Dòng không tải: 70mA (250mA MAX)0

### 2.4 Động cơ Servo SG90

A picture containing text, diagram

Description automatically generated

*Động cơ servo SG90*

Thông số kỹ thuật:

* Khối lượng: 9g
* Kích thước: 22.2x11.8.32mm
* Momen xoắn: 1.8kg/cm
* Tốc độ hoạt động: 60° trong 0.1 giây
* Điện áp hoạt động: 4.8V (~5V)
* Nhiệt độ hoạt động: 0°C-55°C

# Nội dung

## Mô hình kết cấu Robot

A picture containing diagram, line, circle, technical drawing

Description automatically generated

*Mô hình kết cấu*

Phần khung sàn robot được làm bằng 1 miếng mika độ dày 3mm, chiều rộng 15cm, chiều dài 17cm, chiều cao của robot là 7cm tính từ mặt đất đến cảm biến. Robot gồm 3 bánh xe, 2 bánh phát động được điều khiển bằng hai động cơ riêng và một bánh cố định được gắn ở phía trước.

* Khi hai bánh xe trái và phải quay cùng chiều thì robot tiến hoặc lùi.
* Khi bánh trái quay tiến và bánh phải quay lùi thì robot di chuyển sang phải.
* Khi bánh trái quay lùi và bánh phải quay tiến thì robot di chuyển sang trái.

## Sơ đồ khối và chức năng các khối

### 2.1 Sơ đồ khối

A picture containing text, diagram, rectangle, screenshot

Description automatically generated

*Sơ đồ khối các chức năng của hệ thống*

### 2.2 Chức năng các khối

* Khối cảm biến chức năng vị trí: nhận biết đường đi thông qua cảm biến siêu âm.
* Khối điều khiển trung tâm: nhận được tín hiệu vào từ khối cảm biến vị trí, thực hiện thuật toán để điều khiển mạch công suất để đáp.
* Khối điều khiển công suất: nhận tín hiệu từ khối điều khiển trung tâm, điều khiển công suất khối thực thi.
* Khối hiển thị: nhận tín hiệu từ khối điều khiển trung tâm, thể hiện trạng thái vị trí của xe.
* Khối thực thi: 2 động cơ đọc lập được điều khiển tốc độ bằng khối điều khiển công suất.
* Khối nguồn: đưa điện áp vào khối hạ áp, cấp nguồn trực tiếp cho khối điều khiển công suất.
* Khối hạ áp: lấy điện áp từ khối nguồn, đưa khối điện áp ra 5V để nuôi khối cảm biến vị trí, khối điều khiển trung tâm, và khối hiển thị.

## Thiết kế phần cứng điều khiển

A picture containing text, electronic engineering, electronics, circuit component

Description automatically generated

*Sơ đồ mạch*

1. Khối nguồn:

* Sử dụng nguồn pin 12V. Pin được đặt trong cell holder để dễ dàng thay khi hết pin.

1. Khối thực thi:

* Xe tự hành này sử dụng hệ thống hai động cơ độc lập: bên trái và bên phải. Với hệ thống này, không cần thiết phải sử dụng bánh xe phía trước để điều hướng như ô tô hay xe đạp. Để điều hướng cho xe tự hành, 2 động cơ phải quay với tốc độ khác nhau. Ví dụ nếu động cơ bên phải quay nhanh hơn động cơ bên trái, xe sẽ rẽ trái. Việc điều khiển xe di chuyển và điều hướng trở thành điều khiển tốc độ 2 động cơ DC.

1. Khối điều khiển công suất:
   1. Bộ điều khiển trung tâm (MCU) không thể điều khiển trực tiếp 2 động cơ, vì vậy ta sử dụng một IC điều khiển động cơ (motor driver) cho dễ dàng điều khiển. Chọn IC điều khiển động cơ là L298, cho phép điều khiển hai động cơ DC. Ta sử dụng kỹ thuật PWM để thay đổi tốc độ động cơ bằng điều khiển số từ khối điều khiển trung tâm.
   2. IC L298 là tích hợp của hai mạch cầu H trong gói 16 chân. Tất cả các mạch kích, mạch cầu đều được tích hợp sẵn. L298 có điện áp danh nghĩa cao (lớn nhất 12V) và dòng điện danh nghĩa lớn nhất 2A nên thích hợp cho các ứng dụng công suất nhỏ như động cơ DC loại nhỏ và vừa. IC L298 phù hợp với những người thiếu kinh nghiệm làm mạch điện tử.
   3. Có 2 mạch cầu H nên mỗi chip L298 nên có thể điều khiển 2 đối tượng chỉ với 1 chip. Mỗi mạch cấu tạo bao gồm 1 đường nguồn Vs, một đường current senting (cảm biến dòng), phần cuối của mạch cầu H không được nối với GND mà bỏ trống cho người dùng nối một điện trở nhỏ gọi là sensing resistor.
   4. Hai động cơ được nối với các chân OUT, động cơ bên trái nối với OUT1, OUT2, động cơ bên phải nối với OUT3, OUT4. Các chân EnA, EnB cho phép hai mạch cầu hoạt động bằng cách kéo lên tín hiệu mức cao.
   5. A picture containing text, diagram, line, number

      Description automatically generatedSơ đồ nguyên lý cho bộ điều khiển công suất:

*Hình 3.3b Sơ đồ nguyên lý khố điều khiển công suất*

Chip L298 nhận tín hiệu từ khối điều khiển trung tâm, từ đó đưa ra tín hiệu thay đổi điện áp vào các động cơ DC để điều khiển tốc độ.

1. Khối điều khiển trung tâm:

* Vi xử lý để thực thi code điều khiển sẽ lấy input từ bộ cảm biến và đưa ra tín hiệu điều khiển chuyển động của xe. Ta sử dụng một vi điều khiển (MCU) là đủ đáp ứng yêu cầu. Họ vi điều khiển phổ biến là AVR từ sản xuất Atmel. Trong đó, ta chọn Atmega238 vì nó vừa đáp ứng đủ yêu cầu về tài nguyên và giá rẻ.
* Các tính năng của Atmega238:
* Vi điều khiển 8bit của AVR hiệu suất cao, tiêu thụ ít năng lượng.
* Với 23 chân có thể sử dụng cho các kết nối vào hoặc ra I/O, 32 thanh ghi, 3 bộ timer/counter có thể lấp trình, có các gắt nội và ngoại (2 lệnh trên một vector ngắt), giao thức truyền thông nối tiếp USART, SPI, I2C.
* Ngoài ra có thể sử dụng bộ biến đổi số tương tự 10-bit (ADC/DAC) mở rộng tới 8 kênh, khả năng lập trình được watchdog timer, hoạt động với 5 chế độ nguồn, có thể sử dụng tới 6 kênh điều chế độ rộng xung (PWM), hỗ trợ bootloader.

## Chương trình điều khiển

### 4.1 Code

#include <Servo.h>

Servo myservo; // Khai báo một đối tượng servo để điều khiển servo motor.

int pos = 0; // Biến để lưu trữ vị trí hiện tại của servo motor.

const int trig = 7;

const int echo = 4;

int enbA=3;

int in1 = 5;

int in2 = 6;

int in3 = 9;

int in4 = 10;

int enbB=11;

int dongcoservo = 8;

int gioihan = 20;//khoảng cách nhận biết vật

int i;

int x = 0;

unsigned long thoigian;

int khoangcach;

int khoangcachtrai, khoangcachphai;

int maxspeed=30;

void dokhoangcach();

void dithang(int duongdi);

void disangtrai();

void disangphai();

void dilui();

void resetdongco();

void quaycbsangphai();

void quaycbsangtrai();

void setup() {

myservo.attach(dongcoservo);

pinMode(trig, OUTPUT);

pinMode(echo, INPUT);

pinMode(in1, OUTPUT);

pinMode(in2, OUTPUT);

pinMode(in3, OUTPUT);

pinMode(in4, OUTPUT);

pinMode(enbA, OUTPUT);

pinMode(enbB, OUTPUT);

digitalWrite(in1, LOW);

digitalWrite(in2, LOW);

digitalWrite(in3, LOW);

digitalWrite(in4, LOW);

digitalWrite(enbA, LOW);

digitalWrite(enbB, LOW);

myservo.write(90);

delay(500);

Serial.begin(9600); }

void loop()

{

khoangcach = 0;

dokhoangcach();

Serial.println(khoangcach);

if (khoangcach > gioihan || khoangcach == 0)

{

dithang();

Serial.println("Di toi");

}

else

{

dunglai();delay(300);

quaycbsangtrai();

dokhoangcach();

Serial.println(khoangcach);

khoangcachtrai = khoangcach;

quaycbsangphai();

dokhoangcach();

Serial.println(khoangcach);

khoangcachphai = khoangcach;

if (khoangcachphai < 15 && khoangcachtrai < 15) {

dilui();delay(300);dunglai();delay(300);

Serial.println("Di lui");

}

else

{

if (khoangcachphai >= khoangcachtrai)

{ dilui();delay(300);dunglai();delay(300);

disangphai();

Serial.println("Di sang phai");

delay(400);dunglai();delay(300);

}

if (khoangcachphai < khoangcachtrai)

{ dilui();delay(300);dunglai();delay(300);

disangtrai();

Serial.println("Di sang trai");

delay(400);dunglai();delay(300);

}

}

}

}

void dithang()

{

analogWrite(enbA, 100);

analogWrite(enbB, 100);

digitalWrite(in1, 0);

digitalWrite(in2, 1);

digitalWrite(in3, 0);

digitalWrite(in4, 1);

}

void dunglai(){

analogWrite(enbA, 0);

analogWrite(enbB, 0);

digitalWrite(in1, 0);

digitalWrite(in2, 0);

digitalWrite(in3, 0);

digitalWrite(in4, 0);

}

void disangphai()

{

analogWrite(enbA, 100);

analogWrite(enbB, 0);

digitalWrite(in1, 0);

digitalWrite(in2, 1);

digitalWrite(in3, 0);

digitalWrite(in4, 0);

}

void disangtrai()

{

analogWrite(enbA, 0);

analogWrite(enbB, 100);

digitalWrite(in1, 0);

digitalWrite(in2, 0);

digitalWrite(in3, 0);

digitalWrite(in4, 1);

}

void dilui()

{

analogWrite(enbA, 100);

analogWrite(enbB, 100);

digitalWrite(in1, 1);

digitalWrite(in2, 0);

digitalWrite(in3, 1);

digitalWrite(in4, 0);

}

void dokhoangcach()

{

digitalWrite(trig, LOW);

delayMicroseconds(2);

digitalWrite(trig, HIGH);

delayMicroseconds(10);

digitalWrite(trig, LOW);

// Đo độ rộng xung HIGH ở chân echo.

thoigian = pulseIn(echo, HIGH);

khoangcach = thoigian / 2 / 29.412;

}

void quaycbsangtrai()

{

myservo.write(150);

delay(1000);

dokhoangcach();

myservo.write(90);

}

void quaycbsangphai()

{

myservo.write(30);

delay(1000);

dokhoangcach();

myservo.write(90);

}

void resetservo()

{

myservo.write(90);

}

### Giải thích

* Các biến trig và echo được sử dụng để kết nối với cảm biến khoảng cách ultrasonic. Các biến enbA, in1, in2, in3, in4, enbB sử dụng để kết nối và điều khiển động cơ. Biến dongcoservo là chân điều khiển servo motor. gioihan là giá trị ngưỡng để xác định khi nào robot phải dừng lại gần vật cản. maxspeed là giá trị tốc độ tối đa của động cơ.
* Các hàm sau đó được định nghĩa trong mã:
* dokhoangcach(): được sử dụng để đo khoảng cách từ cảm biến khoảng cách ultrasonic. Nó gửi một xung đo lường và tính thời gian phản hồi để tính toán khoảng cách.
* dithang(int duongdi): Hàm này dùng để di chuyển robot đi thẳng với tốc độ và hướng duyệt vào tham số duongdi.
* disangtrai(), disangphai(): được sử dụng để quay servo motor sang trái và sang phải để quét môi trường xung quanh.dilui(): Hàm này dùng để di chuyển robot lùi lại.
* quaycbsangtrai(), quaycbsangphai(): Hàm này dùng để quay servo motor sang trái hoặc sang phải.
* setup() được gọi một lần khi chương trình khởi động. Nó cấu hình các chân của Arduino và servo motor, khởi tạo giá trị ban đầu và khởi động kết nối Serial để gửi dữ liệu. Hàm attach() được sử dụng để kết nối đối tượng servo với chân điều khiển servo motor.
* Trong hàm loop(), chương trình lặp lại việc đo khoảng cách từ cảm biến. Nếu khoảng cách lớn hơn giới hạn gioihan hoặc bằng 0, robot di chuyển thẳng và in ra "Di toi" trên Serial. Nếu khoảng cách nhỏ hơn giới hạn gioihan, robot dừng lại và quay servo motor sang trái và phải để đo khoảng cách từ hai hướng. Khoảng chạnh phải và chạnh trái được lưu trong các biến khoangcachphai và khoangcachtrai. Nếu cả hai khoảng cách này đều nhỏ hơn 15, tức là có vật cản cả phía phải và phía trái, robot sẽ di chuyển lùi và in ra "Di lui" trên Serial.
* Nếu khoảng cách phía phải lớn hơn hoặc bằng khoảng cách phía trái, robot sẽ di chuyển sang phải bằng cách gọi hàm disangphai() và in ra "Di sang phai" trên Serial. Nếu khoảng cách phía phải nhỏ hơn khoảng cách phía trái, robot sẽ di chuyển sang trái bằng cách gọi hàm disangtrai() và in ra "Di sang trai" trên Serial.
* Trong các trường hợp di chuyển sang phải hoặc sang trái, robot cũng sẽ thực hiện lệnh dilui() để di chuyển lùi một khoảng rồi dừng lại trước khi thực hiện di chuyển theo hướng mới.
* Hàm dunglai() được sử dụng để dừng lại robot bằng cách tắt động cơ.
* Hàm resetservo() được sử dụng để đưa servo motor về vị trí ban đầu, tức là góc 90 độ.
* Các lệnh Serial.println() được sử dụng để gửi các thông điệp và giá trị từ robot đến giao diện Serial để theo dõi và gỡ lỗi trong quá trình chạy chương trình.
* Toàn bộ quá trình di chuyển và xử lý dữ liệu khoảng cách được lặp lại trong hàm loop() để robot có thể tiếp tục quét và di chuyển trong môi trường một cách tự động.

# Kết quả đạt được

1. **Robot**

* Robot hoàn thành kịp tiến độ. Quá trình làm việc nhóm hiệu quả, phần gắn mạch và đấu nối mạch, lập trình được phân chia đảm bảo sự liên kết giữa các thành viên trong nhóm
* Cảm biến siêu âm được bố trí khoa học, giảm tình trạng nhiễu và gây cản trở servo hoạt động

1. **Kết quả chạy và đánh giá**

* Robot chạy đúng quy trình và chính xác, đúng thuật toán thiết kế, hoàn thành nhiệm vụ và thể hiện được khả năng hoạt động mà không cần can thiệp của con người
* Hạn chế:
  + Do thiếu kinh nghiệm nên robot không đạt chỉ tiêu thẩm mĩ.
  + Robot chạy chưa ổn định
* Quá trình thực hiện đề tài đã giúp chúng em thu được kết quả tốt bổ sung nhiều kiến thức chưa hiểu rõ trong qua trình học tập và hiểu rõ quá trình thiết kế đến thực hiện 1 ý tưởng kĩ thuật đơn giản. Môn học đã cho sinh viên chúng môi trường làm việc thân thiện, sáng tạo, kích thích sự tìm tòi, sáng tạo.

1. **Mong muốn phát triển thêm**

Trong quá trình chạy, một số trường hợp robot bị vướng vào chướng ngại vật trên đường đi. Nhóm chúng em muốn trang bị và kết hợp cho robot thêm camera có kết nối với điện thoại thông minh, thêm chức năng điều khiển robot trên điện thoại để tiến hành tháo gỡ vướng mắc cho robot.

## Tài liệu tham khảo

1. <http://tbe.vn/chia-se-kien-thuc/7056-phuong-phap-dieu-xung-pwm-la-gi.html>
2. <http://extremeelectronics.co.in/avr-tutorials/line-following-robot-using-avr-atmega8/>
3. <http://robocon.vn/detail/ic55-ic-l293d-motor-driver.html>
4. <http://html.alldatasheet.com/html-pdf/>80247/ATMEL/ATMEGA8/126/1/ ATMEGA8.html