**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HCM**

**KHOA ĐIỆN – ĐIỆN TỬ**

****

**BÁO CÁO CUỐI KÌ**

**ĐỒ ÁN ĐIỀU KHIỂN TỰ ĐỘNG**

**GVHD:** TS. ĐẶNG XUÂN BA

**SINH VIÊN THỰC HIỆN:**

LÊ NGUYỄN NGỌC NHƯ THÙY – 21151169

PHAN VĂN HOÀNG ANH – 21151070

**THÀNH PHỐ HỒ CHÍ MINH, THÁNG 12 – 2023**

|  |  |
| --- | --- |
| TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT  TP HỒ CHÍ MINH  KHOA ĐIỆN – ĐIỆN TỬ | CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM  Độc lập – Tự do – Hạnh phúc |
|  | *Tp. Hồ Chí Minh, Ngày Tháng 12 Năm 2023* |

**BÁO CÁO ĐỒ ÁN CUỐI KÌ**

**ĐIỀU KHIỂN TỰ ĐỘNG**

|  |  |  |
| --- | --- | --- |
| **Họ và tên sinh viên:** | Phan Văn Hoàng Anh  Lê Nguyễn Ngọc Như Thùy | MSSV: 21151070  MSSV: 21151169 |
| **Chuyên ngành:** | CNKT Điều khiển và Tự động hóa | **Mã ngành:** 151 |
| **Hệ đào tạo:** | Đại học chính quy | **Mã hệ:** |
| **Khóa:** | 2021 | **Lớp:** 21151CL3A |

I. ĐỀ TÀI: **MÔ HÌNH ROBOT HAI BÁNH TỰ CÂN BẰNG**

II. NHIỆM VỤ VÀ NỘI DUNG:

|  |
| --- |
| **GIẢNG VIÊN HƯỚNG DẪN** |
| TS. ĐẶNG XUÂN BA |

**LỜI CẢM ƠN**

Để có cơ hội vận dụng kiến thức đã học, tiếp xúc thực tế và hoàn thành *Đồ án Điều khiển tự động,* nhóm sinh viên vô cùng tri ân sự hướng dẫn tận tình trong suốt thời gian thực hiện mô hình và làm bài báo cáo đến giảng viên hướng dẫn – Thầy Đặng Xuân Ba. Nhóm thực hiện xin chân thành cảm ơn sự hỗ trợ tận tình hỗ trợ trong việc cung cấp kiến thức để hoàn thành tốt bài báo cáo. Sau khi tham gia thực hiện *Đồ án Điều khiển tự động* của thầy, chúng em đã tích lũy cho bản thân nhiều kiến thức bổ ích, tinh thần học tập nghiêm túc. Đây chắc chắn sẽ là hành trang quý báu để nhóm có thể vững bước sau này.

*Đồ án Điều khiển tự động* là môn học thú vị và có tính thực tế cao. Đảm bảo cung cấp đủ kiến thức, gắn liền với nhu cầu thực tiễn của sinh viên. Trong quá trình thực hiện bài báo cáo, nhận thấy mình đã cố gắng hết sức nhưng vì kiến thức vẫn còn hạn hẹp nên bài báo cáo vẫn còn nhiều điều thiết sót, chưa chính xác, kính mong Thầy xem xét và góp ý để bài báo cáo của chúng em được hoàn thiện hơn.

Sau cùng nhóm thực hiện xin chúc Thầy sức khoẻ, thành công và tiếp tục đào tạo những sinh viên giỏi đóng góp cho đất nước.

Nhóm thực hiện xin chân thành cảm ơn!

|  |
| --- |
| **Trân trọng**  **Nhóm thực hiện đồ án** |
| Phan Văn Hoàng Anh  Lê Nguyễn Ngọc Như Thùy |

**NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN**

|  |
| --- |
| *TP. Hồ Chí Minh, Ngày Tháng 12 Năm 2023* |
| **GIẢNG VIÊN HƯỚNG DẪN** |
| TS. ĐẶNG XUÂN BA |

**NỘI DUNG**

**DANH SÁCH HÌNH ẢNH**

**MỤC LỤC**

**CHƯƠNG 1**

**GIỚI THIỆU CHUNG**

* 1. **Đặt vấn đề**

Ngày nay, ngành tự động hóa - điều khiển tự động, mô hình con lắc ngược là một trong những đối tượng nghiên cứu điển hình và đặc thù bởi đặc tính động không ổn định của mô hình nên việc điều khiển được đối tượng này trên thực tế là chủ đề thu hút nhiều nghiên cứu.

Các mô hình con lắc ngược cơ bản áp dụng vào thực tế, ví dụ như mô hình xe-con lắc, con lắc ngược quay… ngoài nghiên cứu phát triển, việc khắc phục được những nhược điểm vốn có của các robot hai bánh kinh điển cũng rất được quan tâm. Cấu tạo robot hai bánh kinh điển bao gồm bánh dẫn động và bánh tự do (hay bất kì cái gì khác) để đỡ trọng lượng robot. Nếu trọng lượng được đặt nhiều vào bánh lái thì robot sẽ không ổn định và dễ bị ngã, còn nếu đặt vào nhiều bánh đuôi thì hai bánh chính sẽ mất khả năng bám. Nhiều thiết kế robot có thể di chuyển tốt trên địa hình phẳng nhưng không thể di chuyển lên xuống trên địa hình lồi lõm hoặc mặt phẳng nghiêng. Khi di chuyển lên đồi, trọng lượng robot dồn vào đuôi xe làm mất khả năng bám và trượt ngã.

* 1. **Mục tiêu nghiên cứu**

Mục tiêu của đề tài là xây dựng mô hình robot 2 bánh tự cân bằng dựa trên nền tảng lý thuyết mô hình con lắc ngược. Mục tiêu bài báo cáo đặt ra như sau:

− Tìm hiểu các mô hình xe, robot 2 bánh tự cân bằng và các nguyên lý cơ bản về cân bằng.

− Tính toán các thông số động lực học, xây dựng các hàm không gian - trạng thái (state-space) của mô hình.

− Tìm hiểu, lựa chọn các loại cảm biến và bộ điều khiển trung tâm. Trong đề tài này sẽ sử dụng cảm biến IMU 9 DOF và bộ điều khiển DSP F28335.

− Mô phỏng mô hình trên Matlab Simulink.

− Tìm hiểu và áp dụng Bộ lọc Kalman để lọc nhiễu cho cảm biến, xây dựng các thuật toán bù trừ để có giá trị góc chính xác.

− Xây dựng thuật toán điều khiển động cơ, giữ thăng bằng cho robot.

* 1. **Phương pháp nghiên cứu**

Xây dựng mô hình lý thuyết robot 2 bánh tự cân bằng và mô phỏng mô hình trên Matlab Simulink. Xây dựng mô hình thực:

− Thiết kế cơ khí: khung sườn của mô hình.

− Ứng dụng các mạch điện tử, mạch công suất, cảm biến, vi xử lý.

− Mạch cảm biến và bù trừ giá trị cảm biến.

* 1. **Bố cục bài báo cáo**
* Cơ sở lý thuyết
* Mô phỏng hệ thống
* Thiết kế phần cứng, thi công hệ thống
* Kết quả

**CHƯƠNG 2**

**CƠ SỞ LÝ THUYẾT**

* 1. **Đặc tính động lực học**
     1. **Mô hình hóa robot 2 bánh tự cân bằng trên địa hình phẳng**

Xây dựng hệ phương trình trạng thái mô tả hệ thống robot hai bánh tự cân bằng.

A diagram of a wheel and a wheel

Description automatically generated

*Hình Mô hình robot 2 bánh tự cân bằng trên mặt phẳng*

Trong bài báo cáo sử dụng các kí hiệu, đơn vị như sau:

|  |  |  |
| --- | --- | --- |
| **Kí hiệu** | **Đơn vị** | **Ý nghĩa** |
| m | kg | Khối lượng bánh xe |
| M | kg | Khối lượng của robot |
| R | m | Bán kính bánh xe |
| W | m | Chiều rộng của robot |
| D | m | Chiều ngang của robot |
| H | m | Chiều cao của robot |
| L | m | Khoảng cách từ trọng tâm robot đến trục bánh xe |
| fw |  | Hệ số ma sát giữa hai bánh xe và mặt phẳng di chuyển |
| fm |  | Hệ số ma sát giữa robot và động cơ DC |
| jm | kg.m2 | Moment quán tính của động cơ DC |
| Rm | ohm | Điện trở động cơ DC |
| Kb | V.sec/rad | Hệ số EMF của động cơ DC |
| Kt | Nm/A | Moment xoắn của động cơ DC |
| N |  | Tỉ số giảm tốc |
| G | m/s2 | Gia tốc trọng trường |
|  | rad | Góc trung bình của bánh trái và phải |
| l, r | rad | Góc của bánh trái và phải |
|  | rad | Góc nghiêng của phần thân robot |
|  | rad | Góc xoay của robot |
| xl, yl, zl | m | Tọa độ bánh trái |
| xr, yr, zr | m | Tọa độ bánh phải |
| xm, ym, zm | m | Tọa độ trung bình |
|  | Nm | Moment phát động theo các phương khác nhau |
| Fl,r | Nm | Moment phát động của động cơ bánh trái, phải |
| il, ir | A | Dòng điện động cơ bánh trái, phải |
| vl, vr | V | Điện áp động cơ bánh trái, phải |

*Bảng 1. Kí hiệu và ý nghĩa của các đại lượng*

Sử dụng phương pháp Euler-Lagrange để xây dựng mô hình động học. Giả sử tại thời điểm t = 0, robot di chuyển theo chiều dương trục x, ta có các phương trình sau:

Góc tịnh tiến trung bình của hai bánh xe và góc xoay của robot được xác định như sau:

= (1)

Trong đó tọa độ trung bình của Robot trong hệ quy chiếu:

= (2) và = (3)

Tốc độ bánh trái trong hệ quy chiếu:

= (4)

Tốc độ bánh phải trong hệ quy chiếu:

= (5)

Tọa độ tâm đối xững giữa hai động cơ trong hệ quy chiếu:

= (6)

Phương trình động năng của chuyển động tịnh tiến:

T1 = x’l2 + y’l2 + z’l2 ) + x’r2 + y’r2 + z’r2 ) + x’b2 + y’b2 + z’b2 ) (7)

Phương trình động năng của chuyển động quay:

T2 = Jwl2 + Jwr2 + Jψ’2 + J’2 + n2Jm (2 + n2Jm (2 (8)

Với n2Jm (2 ; n2Jm (2 (9) là động năng quay của phần cứng động cơ trái và phải

Phương trình thế năng:

U = mgzl +mgzr + mgzb (10)

Phương trình Lagrange:

L = T1 + T2 – U (11)

= F

= F (13)

= F (14)

Lấy đạo hàm L theo các biến ta được:

[(2m+M)R2 +2Jw + 2n2Jw] + (MLRcos – 2n2Jw) - MLR2 sin = F(15)

(MLRcos - 2n2Jm) + (ML2 + 2n2Jm) - MgLsin – ML22sincos = Fψ (16)

[mW2 + J + (Jw +n2Jm) + ML2sin2] 2 + 2ML2 sincos = F (17)

Momen động lực do động cơ DC sinh ra:

= (18)

Và

Fl = nKtil + fm(l ) – fm l (19)

Fr = nKtir + fm(r) – fm r(20)

= -nKtil – nKtir - fm(l ) - fm(r) (21)

Sử dụng phương pháp PWM để điều khiển động cơ nên chuyển từ dòng điện sang điện áp động cơ:

Lm (il,v )’’ = vl,v + Kb(l,r ) – Rmil,v (22)

Xem điện cảm phần ứng tương đối nhỏ (gần bằng 0), có thể bó qua, suy ra:

il,v = (23)

Từ đó, các momen lực sinh ra:

(24)

= (25)

Với và (26)

(27)

Thu được phương trình động lực học mô tả chuyển động của robot như sau:

[(2m + M)R2 +2Jw +2n2Jm] + (MLRcos - 2n2Jm) - MLR2sin = (28)

(MLRcos - 2n2Jm) + (ML2 + 2n2Jm) - MgLsin – ML22sincos = (29)

[mW2 + J + (Jw +n2Jm) + ML2sin2] 2 + 2ML2 sincos

= (30)

* + 1. **Mô hình hóa robot hai bánh tự cân bằng trên địa hình phẳng trong Matlab Simulink**

**A white square with black lines and a square with black text

Description automatically generated with medium confidence**

Hình 1Mô hình phi tuyến của robot hai bánh tự cân bằng trong Matlab Simulink

**A diagram of a computer

Description automatically generated**

Hình 2: Bên trong mô hình hệ robot 2 bánh tự cân bằng

Xây dựng mô hình robot 2 bánh tự cân bằng dựa trên mô hình toán học

function [teta\_dd,psi\_dd,phi\_dd] = fcn(vl,vr,teta,teta\_d,psi,psi\_d,phi,phi\_d)

%% Thông số hệ thống xe 2 bánh tự cân bằng dùng PID

m = 0.025; %Khoi luong banh xe

M = 0.8; %Khoi luong robot

R = 0.0325; %ban kinh ban xe

W = 0.15; %Chieu rong robot

D = 0.06; %Chieu sau robot

H = 0.13; %Chieu cao robot

L = 0.02; %khoang cach tu trong tam den truc banh xe

fw = 0.001; %He so ma sat giua banh xe voi mat phang

fm = 0.001; %he so ma sat giua dong co va robot

Jm = 0.5462; %moment quan tinh cua dong co

Jw = m\*R^2/2;

J\_psi = M\*L^2/3;

J\_phi = M\*(W^2+D^2)/12;

Rm = 20; %Dien tro dong co DC

Kb = 0.002; %he so emf cua dong co

Kt = 4.2; %Momen xoan cua dong co DC

n = 30; %Ty so giam toc

g = 9.81; %Gia toc trong truong

alpha = n\*Kt/Rm;

beta=n\*Kt\*Kb/Rm+fm;

a =alpha;

T=0.01;

x1\_init = 0.001; x2\_init = -0.0012; x4\_init = 0.002; x5\_init = -0.002; x7\_init = 0.002; x8\_init=-0.0014;

**teta\_dd** = (J\_psi\*a\*vl + J\_psi\*a\*vr - 2\*J\_psi\*beta\*teta\_d + 2\*J\_psi\*beta\*psi\_d - 2\*J\_psi\*fw\*teta\_d + L^2\*M\*a\*vl + L^2\*M\*a\*vr - 2\*L^2\*M\*beta\*teta\_d + 2\*L^2\*M\*beta\*psi\_d - 2\*L^2\*M\*fw\*teta\_d - 4\*Jm\*fw\*n^2\*teta\_d + L^3\*M^2\*R\*psi\_d^2\*sin(psi) - 2\*L\*M\*R\*beta\*teta\_d\*cos(psi) + 2\*L\*M\*R\*beta\*psi\_d\*cos(psi) - L^3\*M^2\*R\*phi\_d^2\*cos(psi)^2\*sin(psi) - L^2\*M^2\*R\*g\*cos(psi)\*sin(psi) + J\_psi\*L\*M\*R\*psi\_d^2\*sin(psi) + 2\*Jm\*L\*M\*g\*n^2\*sin(psi) + L\*M\*R\*a\*vl\*cos(psi) + L\*M\*R\*a\*vr\*cos(psi) + 2\*Jm\*L^2\*M\*n^2\*phi\_d^2\*cos(psi)\*sin(psi) + 2\*Jm\*L\*M\*R\*n^2\*psi\_d^2\*sin(psi))/(2\*J\_psi\*Jw + L^2\*M^2\*R^2 + 2\*Jw\*L^2\*M + J\_psi\*M\*R^2 + 2\*J\_psi\*Jm\*n^2 + 4\*Jm\*Jw\*n^2 + 2\*J\_psi\*R^2\*m + 2\*Jm\*L^2\*M\*n^2 + 2\*Jm\*M\*R^2\*n^2 + 2\*L^2\*M\*R^2\*m + 4\*Jm\*R^2\*m\*n^2 - L^2\*M^2\*R^2\*cos(psi)^2 + 4\*Jm\*L\*M\*R\*n^2\*cos(psi));

**psi\_dd** = -(2\*Jw\*a\*vl + 2\*Jw\*a\*vr - 4\*Jw\*beta\*teta\_d + 4\*Jw\*beta\*psi\_d + M\*R^2\*a\*vl + M\*R^2\*a\*vr - 2\*M\*R^2\*beta\*teta\_d + 2\*M\*R^2\*beta\*psi\_d + 2\*R^2\*a\*m\*vl + 2\*R^2\*a\*m\*vr + 4\*Jm\*fw\*n^2\*teta\_d - 4\*R^2\*beta\*m\*teta\_d + 4\*R^2\*beta\*m\*psi\_d - L\*M^2\*R^2\*g\*sin(psi) - 2\*Jw\*L\*M\*g\*sin(psi) - 2\*L\*M\*R\*beta\*teta\_d\*cos(psi) + 2\*L\*M\*R\*beta\*psi\_d\*cos(psi) - 2\*L\*M\*R\*fw\*teta\_d\*cos(psi) + L^2\*M^2\*R^2\*psi\_d^2\*cos(psi)\*sin(psi) - L^2\*M^2\*R^2\*phi\_d^2\*cos(psi)\*sin(psi) - 2\*Jw\*L^2\*M\*phi\_d^2\*cos(psi)\*sin(psi) - 2\*Jm\*L\*M\*g\*n^2\*sin(psi) - 2\*L\*M\*R^2\*g\*m\*sin(psi) + L\*M\*R\*a\*vl\*cos(psi) + L\*M\*R\*a\*vr\*cos(psi) - 2\*Jm\*L^2\*M\*n^2\*phi\_d^2\*cos(psi)\*sin(psi) - 2\*L^2\*M\*R^2\*m\*phi\_d^2\*cos(psi)\*sin(psi) - 2\*Jm\*L\*M\*R\*n^2\*psi\_d^2\*sin(psi))/(2\*J\_psi\*Jw + L^2\*M^2\*R^2 + 2\*Jw\*L^2\*M + J\_psi\*M\*R^2 + 2\*J\_psi\*Jm\*n^2 + 4\*Jm\*Jw\*n^2 + 2\*J\_psi\*R^2\*m + 2\*Jm\*L^2\*M\*n^2 + 2\*Jm\*M\*R^2\*n^2 + 2\*L^2\*M\*R^2\*m + 4\*Jm\*R^2\*m\*n^2 - L^2\*M^2\*R^2\*cos(psi)^2 + 4\*Jm\*L\*M\*R\*n^2\*cos(psi));

**phi\_dd** = -(W^2 \* beta \* phi\_d + W^2 \* fw \* phi\_d + R \* W \* a \* vl - R \* W \* a \* vr + 4 \* L^2 \* M \* R^2 \* psi\_d \* phi\_d \* cos(psi) \* sin(psi)) / (2 \* J\_phi \* R^2 + Jm \* W^2 + Jm \* W^2 \* n^2 + R^2 \* W^2 \* m + 2 \* L^2 \* M \* R^2 \* sin(psi)^2);

* 1. **Bộ lọc Kalman**
     1. **Giới thiệu bộ lọc Kalman**

Từ năm 1960 giáo sư Kalman để thu thập và kết hợp các thông tin từ cảm biến thánh phần. Khi phương trình định hướng và mẫu thống kê nhiễu trên cảm biến được xác định, bộ lọc Kalman sẽ cho ước lượng giá trị tối ưu như đang sử dụng một tín hiệu tinh khiết và độ phân bố không đổi. Trong hệ thống xe hai bánh tự cân bằng, tín hiệu ngõ ra của bộ lọc là tín hiệu từ cảm biến góc và cảm biến vận tốc góc đã được loại bỏ nhiễu nhờ nguồn tín hiệu hỗ trợ và xử lý trong bộ lọc thông qua quan hệ vận tốc góc bằng vi phân giá trị góc.

Bộ lọc Kalman ước tính quá trình bằng cách ử dụng một hình thức kiểm soát phản hồi. Các trạng thái của quá trình được ước tính bở bộ lọc tại một thời điểm nhất định và sau đó có được một phản hồi về các phép đo nhiễu. Do đó phương trình bộ lọc gồm có phương trình cập nhật thời gian và phương trình cập nhật đo lường.

Hoạt động của bộ lọc được mô tả:

A diagram of a diagram

Description automatically generated

*Hình*

Dưới đây là một số khái niệm chính liên quan đến bộ lọc Kalman:

* Trạng thái (State): Là các biến mà chúng ta muốn ước lượng trong hệ thống. Ví dụ, trong một hệ thống theo dõi vị trí của một đối tượng, trạng thái có thể bao gồm vị trí và vận tốc của đối tượng.
* Quá trình động (Dynamic Process): Là mô hình mô tả cách trạng thái của hệ thống thay đổi theo thời gian.
* Đo đạc (Measurement): Là các thông tin thu được từ các cảm biến hoặc bất kỳ nguồn nào khác mà chúng ta sử dụng để ước lượng trạng thái.
* Nhiễu quá trình (Process Noise): Là các biến ngẫu nhiên mô tả những biến động không dự đoán được trong hệ thống.
* Nhiễu đo đạc (Measurement Noise): Là các biến ngẫu nhiên mô tả sự không chắc chắn trong các đo đạc.
* Ma trận Covariance: Là một công cụ quan trọng để đo độ không chắc chắn của ước lượng. Nó thường được sử dụng để biểu diễn sự phân tán của các biến ngẫu nhiên. Bộ lọc Kalman hoạt động dựa trên hai bước chính: dự đoán và cập nhật. Trong bước dự đoán, nó dự đoán trạng thái tiếp theo của hệ thống dựa trên dự đoán trạng thái trước đó và mô hình động của hệ thống. Sau đó, trong bước cập nhật, nó kết hợp thông tin từ các quan sát thực tế để cập nhật ước lượng trạng thái.
  + 1. **Quá trình ước lượng**

Bộ lọc Kalman có thể ước lượng các trạng thái khi hệ thống đuộc mô hình đầy đủ dưới dạng phương trình vi phân tuyến tính. Khi một hệ thống được mô hình hóa trong không gian trạng thái và phương trình đại diện cho quá trình trạng thái của hệ thống trong ký hiệu nhiễu trắng.

(31)

F(t): mô hình chuyển đổi trạng thái.

B(t): mô hình tín hiệu điều khiển

W(t): nhiễu quá trình có phân bố Gausian với trung bình 0 và ma trận hiệp phương sai Q tại thời điểm k

Bộ lọc này không giả định các nhiễu lag Gausian nhưng bộ lọc tạo ra ước tính xác xuất có điều khiện chính xác trong trường hợp đặt biệt là tất cả các nhiễu đều phân bố Gausian.

* + 1. **Thuật toán Kalman được sử dụng trong chương trình**

Để giảm nhiễu từ cảm biến góc nghiên, nhóm sử dụng thư viện SimpleKalmanFilter của Arduino IDE

Chương trình Kalman được khởi tạo như sau:

#include <SimpleKalmanFilter.h>

SimpleKalmanFilter bo\_loc (2, 2, 0.001);

A graph with blue lines and red lines

Description automatically generated

Hình 3: Tín hiệu cảm biến trước và sau khi qua bộ lọc

* 1. **Giải thuật điều khiển** 
     1. **Cấu trúc bộ điều khiển PID cho robot hai bánh tự cân bằng**

Bộ điều khiển PID (PID - Proportinal Integral Derivative) là cơ chế phản hồi vòng điều khiển được sử dụng trong các hệ thống điều khiển có hồi tiếp. Bộ điều khiển giúp giảm tối đa sai số giữa tín hiệu đặt với tín hiệu ngõ ra thông qua các khâu tỷ lệ, vi phân, tích phân.

A diagram of mathematical equations

Description automatically generated

*Hình Bộ điều khiển PID*

P (Proportional): là phương pháp điều chỉnh tỉ lệ, giúp tạo ra tín hiệu điều chỉnh tỉ lệ với sai lệch đầu vào theo thời gian lấy mẫu.

I (Integral): là tích phân của sai lệch theo thời gian lấy mẫu. Điều khiển tích phân là phương pháp điều chỉnh để tạo ra các tín hiệu điều chỉnh sao cho độ sai lệch giảm về 0. Từ đó cho ta biết tổng sai số tức thời theo thời gian hay sai số tích lũy trong quá khứ. Khi thời gian càng nhỏ thể hiện tác động điều chỉnh tích phân càng mạnh, tương ứng với độ lệch càng nhỏ.

D (Derivative): là vi phân của sai lệch. Điều khiển vi phân tạo ra tín hiệu điều chỉnh sao cho tỉ lệ với tốc độ thay đổi sai lệch đầu vào. Thời gian càng lớn thì phạm vi điều chỉnh vi phân càng mạnh, tương ứng với bộ điều chỉnh đáp ứng với thay đổi đầu vào càng nhanh.

Ba bộ PID được sử dụng để điều khiển robot hai bánh tự cân bằng, bao gồm:

* Bộ PID điều khiển góc nghiêng (
* Bộ PID điều khiển vị trí ()
* Bộ PID điều khiển góc xoay ()

*Hình Cấu trúc bộ điều khiển PID cho hệ robot hai bánh tự cân bằng*

Hàm truyền đạt bộ điều khiển PID liên tục:

(31)

Rời rạc hóa đạo hàm theo thời gian:

(32)

Rời rạc hóa tích phân theo thời gian:

(33)

Phép biến đổi rời rạc (z – Transform)

X[x] = Z{x[k]} = (34)

z = Aejθ= A (cosθ+ jsinθ) (35)

Ta có: Z {x [k−n]} =z−n X [ z] và Z { x [k ]}=X [ z]  
   
Do đó:

(35)

ZY[z] =

Và

* U[z] + aE[z] +bE[z] +ce[z] (36)
* = u[k-1]+ ae[k] +be[k-1] + ce[k-2] (37)

Trong đó: a = ; b = ; c =

* + 1. **Bộ điều khiển PID rời rạc**

Bộ điều khiển PID với thông số KP, KI, KD cố định như trên, hệ thống chỉ làm việc tốt trong điều khiển hệ số KP, KI, KD đã được chỉnh định tối ưu và trong quá trình làm việc, các thông số trong mô hình không đổi.

Cấu trúc bộ điều khiển PID rời rạc với hệ số cố định sử dụng để điều khiển góc nghiêng, vị trí và góc xoay là như nhau và được hiện thực như sau trong Matlab Simulink.

A diagram of a computer program

Description automatically generated

*Hình Cấu trúc bên trong bộ điều khiển PID rời rạc với thông số cố định*

*Cấu trúc bộ điều khiển PID rời rạc được mô tả như sau:*

function u = fcn(u1, e, e1, e2, kp, ki, kd, T)

a = kp + ki\*(T/2) + kd/T;

b = -kp + ki\*T/2 + 2\*kd/T;

c = kd/T;

u = u1 + a\*e + b\*e1 + c\*e2;

* + - 1. **Trường hợp thông số mô hình được chọn cố định**

Với M= kg, fw= , thông số của các bộ điều khiển lần lượt là:

* Position PID controller [Kp= Ki= Kd=]
* Tilt PID controller [Kp= Ki= Kd=]
* Rotation PID controller [Kp= Ki= Kd=]

Bộ PID cố định sẽ điều khiển hệ thống hoạt động tốt ứng với các tín hiệu đầu vào là xung vuông hoặc sóng sine như nhau:

*Hình Đáp ứng ngõ ra vị trí với tín hiệu đặt vị trí là xung vuông*

*Hình Đáp ứng ngõ ra góc nghiêng với tín hiệu đặt góc nghiêng là xung vuông*

*Hình Đáp ứng ngõ ra góc xoay với tín hiệu đặt góc xoay là xung vuông*

* + - 1. **Kết luận**

**CHƯƠNG 3**

**THIẾT KẾ HỆ THỐNG**

**3.1 Yêu cầu thiết kế**

Để thiết kế phần cứng đảm bảo cho xe hoạt động cần lựa chọn các thiết bị phù hợp với các yêu cầu sau:

- Đảm bảo tổng khổi lượng của xe để động cơ có thể kéo cân bằng được.

- Nguồn cung cấp đủ cho hệ thống hoạt động.

- Khung xe chắc chắn.

- Cảm biến trả tín hiệu về tránh tối đa nhiễu và sai số.

- Động cơ moment đủ để kéo tải.

- Vi điều khiển và các cảm biến, đảm bảo xử lý tín hiệu nhanh.

**3.2 Sơ đồ khối hệ thống**

**3.2.1 Bản vẽ cơ khí**

A drawing of a vehicle with wheels

Description automatically generated

Hình : Bản vẽ 3D Robot 2 bánh tự cân bằng

A drawing of a rectangular object

Description automatically generated

Hình 4: Bản vẽ tầng trên robot

A drawing of a rectangular object

Description automatically generated

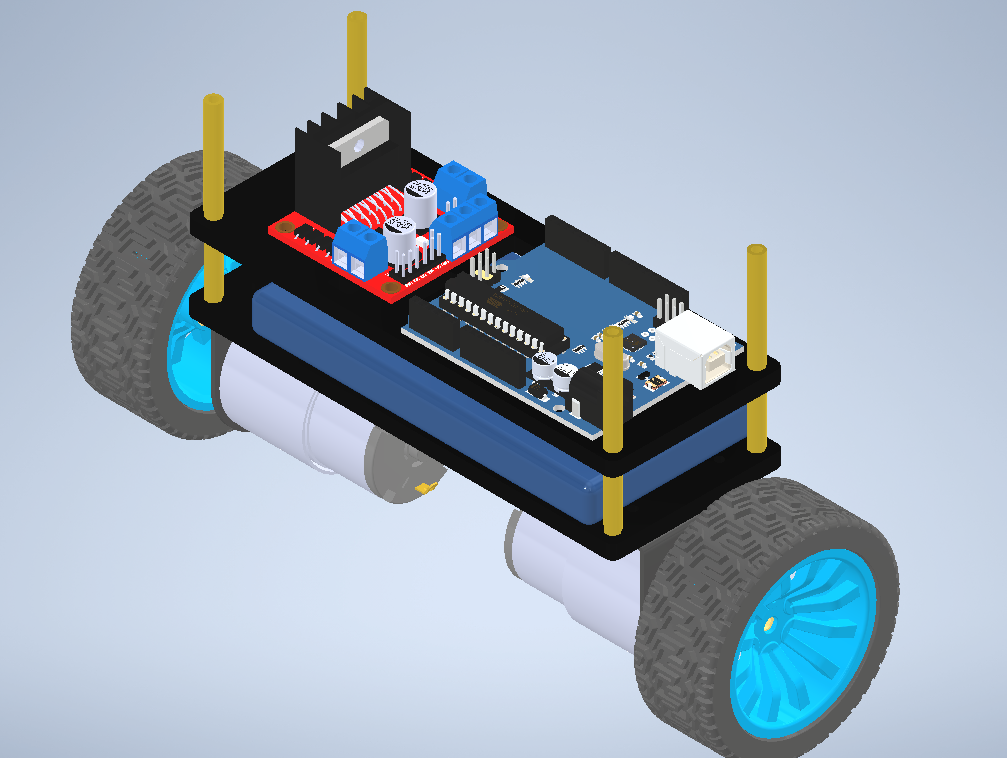
Hình 5: Bản vẽ tầng giữa robot

A drawing of a rectangular object

Description automatically generated

Hình 6: bản vẽ tầng dưới robot

- Tầng thứ hai: Khối điều khiển trung tâm



- Tầng đáy: Khối động cơ và cảm biến đọc xung (encoder) của động cơ, cảm biến đo góc nghiêng.

A blue rectangular object with wheels

Description automatically generatedA blue and black machine with wheels

Description automatically generated

***Chức năng của từng khối từng khối:***

- ***Bộ khung xe:*** cố định từng khối thành phần của xe với nhau, các thành phần cấu tạo khung được lựa chọn kích thước phù hợp đảm bảo không ảnh hướng nhiều trong quá trình xe cân bằng.

***- Khối cảm biến đo góc nghiêng:*** có nhiệm vụ đọc dữ liệu trả về và gửi tín hiệu đó về *Khối điều khiển trung tâm* xử lý ra giá trị góc nghiêng của xe.

***- Khối điều khiển trung tâm:*** là một vi điều khiển, có nhiệm vụ chính là xử lý tất cả dữ liệu đọc về từ *Khối đo góc nghiêng* và *Khối cảm biến đọc xung* của động cơ, xử lý các thuật toán điều khiển đưa ra tín hiệu để điều khiển *Khối điều khiển động cơ*.

***- Khối điều khiển động cơ:*** nhiệm vụ là nhận tín hiệu từ *Khối điều khiểu trung tâm* để cấp điện áp ra phù hợp để điều khiển Khối động cơ.

***- Khối động cơ và cảm biến đọc xung của động cơ:*** nhận điện áp cấp từ *Khối điều khiển động cơ* điều khiển xe di chuyển qua lại, nhanh, chậm, dừng để cho xe có thể đạt được trạng thái cân bằng. Ngoài ra còn có một bộ mã hoá quay encoder để trả về tín hiệu số xung đọc được, từ đó qua Khối điều khiển trung tâm xử lý có thể xác định chính xác được vị trí, tốc độ quay và chiều quay động cơ.

***- Khối nguồn:*** cấp nguồn ổn định cho các linh kiện và động cơ.

**3.3 Lựa chọn thiết bị phần cứng**

Bài báo cáo sử dụng các linh kiện: Arduino Uno R3, mạch cầu H L298N, cảm biến gia tốc MPU6050, động cơ DC Servo JGB37-520 DC, bộ thu phát tín hiệu Devo 7.

**3.3.1 Khối điều khiển trung tâm**

*Khối điều khiển trung tâm* là một vi điều khiển có nhiệm vụ xử lý dữ liệu đọc về từ cảm biến đo góc nghiêng và bộ mã hoá quay Encoder từ đó xử lý các thuật toán về điều khiển xuất ra tín hiệu đến *Khối điều khiển động cơ.*

Khối điều khiển trung tâm phải có tốc độ xử lý nhanh với hiệu suất làm việc cao. Để thực hiện đề tài này nhóm sử dụng Board Arduino Uno R3 với các ưu điểm sau:

- Có số lượng chân I/O kết nối nhiều thuận tiện trong việc đi dây hệ thống.

- Tốc độ xử lý phù hợp, có nhiều chân ngắt ngoại để đọc tín hiệu từ bộ mã hoá quay encoder trả về để điều khiển chính xác tốc độ, vị trí và chiều quay của động cơ.

- Thư viện hỗ trợ nhiều và cộng đồng người sử dụng lớn nên có thể dễ dàng lập trình và tìm kiếm thông tin nhanh chóng.

- Có nhiều chân hỗ trợ xuất xung PWM.

- Có nguồn cấp tương thích với các loại board mạch ngoại vi hỗ trợ khác nhau như tín hiệu mức cao (5V) hoặc tín hiệu mức thấp (3.3V).

- Hỗ trợ nhiều chuẩn giao tiếp (đặc biệt là UART có giao tiếp với máy tính thông qua USB hoặc thông qua bluetooth và I2C để giao tiếp với cảm biến góc nghiêng MPU).

- Dễ dàng cấp nguồn ngoài như pin, adapter AC-DC, …

- Có giao tiếp I2C để đọc tín hiệu trả về giá trị góc nghiêng từ cảm biến góc nghiêng MPU.

+ I2C sử dụng 2 đường truyền tín hiệu: Serial Clock Line và Serial Data Line ( tạo xung nhịp đồng hồ do Master phát đi và đường truyền nhận dữ liệu).

+ Thiết bị Master là 1 vi điều khiển, có nhiệm vụ điều khiển đường tín hiệu SCL gửi dữ liệu, lệnh thông qua đường SDA đến các thiết bị khác. Các thiết bị nhận dữ liệu, lệnh từ Master được gọi là Salve.

+ Khung truyền I2C:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Bắt đầu | 7 bit  địa chỉ | Bit Read/ Write | Bit ACK/NACK | 8 bit  dữ liệu | Bit ACK/NACK | Kết thúc |

**Thông số kỹ thuật**

- Vi điều khiển chính: Atmega328P.

- IC nạp và giao tiếp UART: ATmega16U2.

- Nguồn cấp: 5VDC từ cổng USB.

- Số chân đọc và ghi tín hiệu số (Digital I/O): 14 (trong đó 6 chân có khả năng xuất xung PWM).

- Số chân đọc tín hiệu tương tự (Analog Input): 6.

- Flash Memory: 32 KB trong đó 0.5 KB sử dụng cho bootloader.

- EEPROM: 1 KB.

- Số cổng UARTs giao tiếp nối tiếp phần cứng: 4.

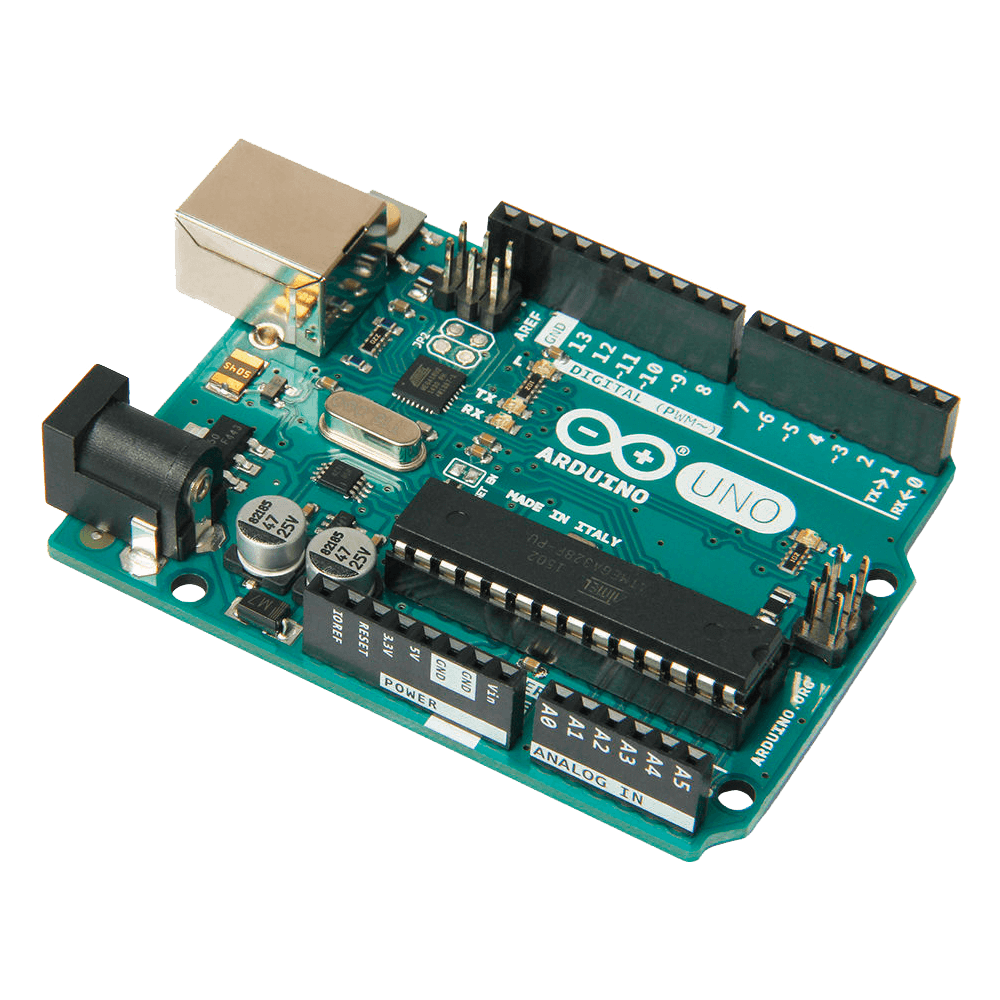
- Một thạch anh dao động với tần số 16 MHz.

- Một cổng kết nối USB, một đầu SPI, một đèn LED tích hợp ở chân 13.

- Giao tiếp: USB, SPI, I2C/TWI và UART.

- Kích thước: 68.6 x 53.4 mm.

- Khối lượng: 25 g.



*Hình 3.2 Arduino Uno R3*

**3.3.2 Khối cảm biến đo góc nghiêng**

Khối cảm biến đo góc nghiêng là một module gồm hai cảm biến đó là:

- Accelerometer: cảm biến đo gia tốc của module (bao gồm cả gia tốc trọng lực), có ba trục xyz tương ứng với ba chiều không gian.

- Gyroscope: cảm biến đo tốc độ quay của module quanh một trục, có ba trục xyz tương ứng với ba chiều không gian.

Khối điều khiển trung tâm sẽ nhận tín hiệu trả về từ hai cảm biến này và sẽ xử lý tính toán ra được giá trị góc nghiêng của xe.

Đối với yêu cầu hệ thống, chỉ để xác định giá trị góc nghiêng của xe thì nhóm lựa chọn ra được loại cảm biến GY-521 MPU6050 với các ưu điểm và nhược điểm sau:

- Về ưu điểm:

+ Có tích hợp sẵn phần cứng chuyên xử lý tín hiệu (Digital Motion Processor – DMP) do cảm biến thu thập và thực hiện các tính toán cần thiết giúp cải thiện tốc độ xử lý, phản hồi nhanh hơn và giảm bớt đáng kể phần xử lý tính toán của vi điều khiển.

+ Giao tiếp dễ dàng với vi điều khiển thông qua giao thức I2C phù hợp với Khối điều khiển trung tâm đã chọn.

+ Có nhiều nguồn thư viện để hỗ trợ trong việc xử lý tín hiệu và lập trình.

- Về nhược điểm:

+ Có giới hạn độ chính xác trong việc đo góc so với các cảm biến góc quay (Gyroscope) riêng lẻ hay cảm biến gia tốc (Accelerometer) độc lập.

+ Cả hai cảm biến đo gia tốc và tốc độ quay đều chịu ảnh hưởng của nhiễu.

**Thông số kỹ thuật**

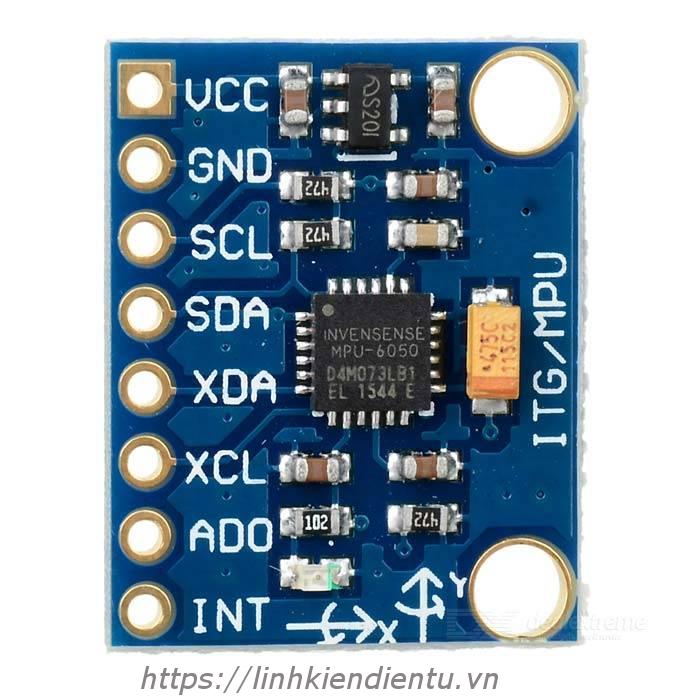
- Gồm 6 trục cảm biến (6DOF): cảm biến gia tốc (3 trục), cảm biến con quay hồi chuyển (3 trục).

- Bộ chuyển đổi tương tự sang số (ADC): 16 bit.

- Điện áp sử dụng: 3-5 VDC. - Điện áp giao tiếp: 3-5 VDC. - Chuẩn giao tiếp: I2C.

- Giá trị Gyroscope trong khoảng: ±250, ±500, ±1000, ±2000 o /sec.

- Giá trị Accelerometer trong khoảng: ±2, ±4, ±8, ±16 g.



*Hình Cảm biến góc nghiêng GY – 521 MPU6050*

**Bảng kết nổi chân của cảm biến MPU6050**

|  |  |
| --- | --- |
| **Chân** | **Kết nối và chức năng** |
| VCC | Chân cấp nguồn dương |
| GND | Chân cấp GND |
| SCL | Serial Clock Line |
| SDA | Serial Data Line |

**3.3.3 Khối điều khiển động cơ**

Khối điều khiển động cơ nhận tín hiệu PWM (điều chế độ rộng xung) từ Khối điều khiển trung tâm để dễ dàng điều khiển hai động cơ bánh trái và bánh phải quay thuận, quay nghịch, dừng hoặc theo tốc độ nhanh, chậm để xe dễ dàng đạt trạng thái cân bằng.

Giá trị trung bình của điện áp (và dòng điện) cung cấp cho tải được kiểm soát bằng cách thay đổi việc đóng cắt giữa nguồn và tải tắt với tốc độ rất nhanh. Thời gian đóng càng lâu so với thời gian cắt, thì tổng công suất cung cấp cho tải càng cao.

Ở đây, nhóm lựa chọn mạch điều khiển động cơ DC L298N (Dual Full Bridge Driver) với các ưu điểm sau:

- Dễ dàng sử dụng, dễ dàng lắp đặt, chi phí thấp, có hiệu năng cao.

- Một module có khả năng điều khiển hai động cơ.

- Tích hợp IC nguồn 7805 giúp cấp nguồn 5VDC cho các module khác.

Nhược điểm:

- Sử dụng trong một thời gian dài sẽ làm module nóng lên gây ra hao phí công suất.

- Hiệu suất không đạt được tối đa.

Thông số kỹ thuật:

- IC chính: L298 - Dual Full Bridge Driver.

- Điện áp đầu vào: 5 - 12VDC.

- Công suất tối đa mỗi cầu H: 25W.

- Dòng tối đa cho mỗi cầu H: 2A.

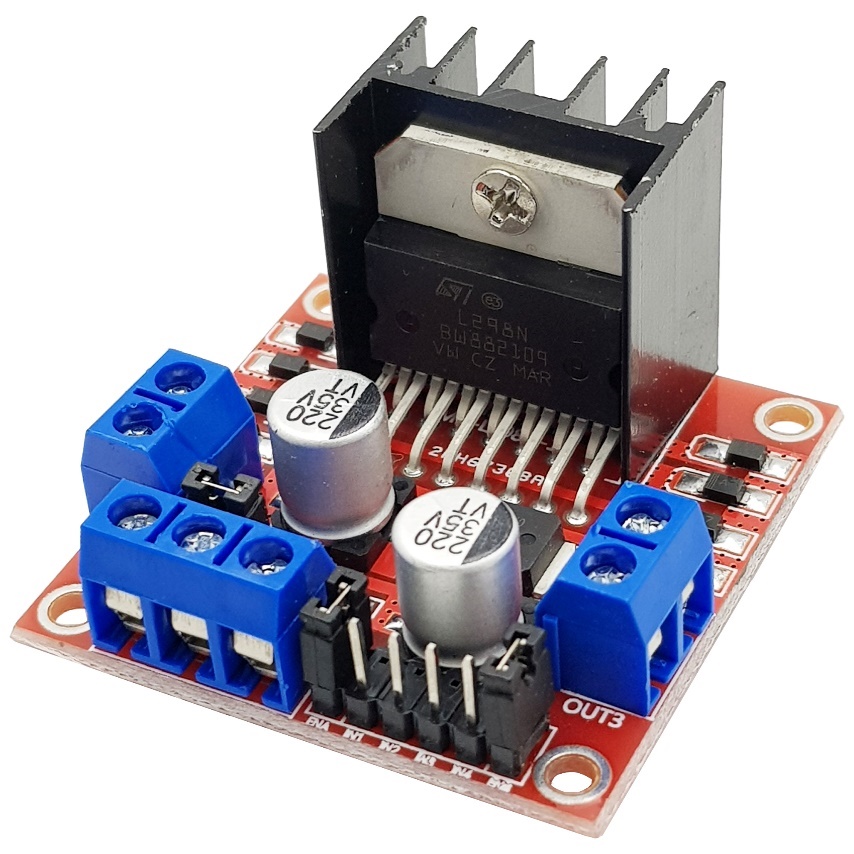
- Dòng tín hiệu điều khiển: 0 – 36mA.

- Điện áp của tín hiệu điều khiển: 5 – 7 VDC.

- Mức điện áp logic: Low -0.3V - 1.5V, High: 2.3V – Vss.

- Công suất hao phí: 20W khi nhiệt độ IC ở 75oC.

- Kích thước: 43x43x27mm.



*Hình Mạch cầu H điều khiển động cơ DC L298N (Dual Full Bridge Driver)*

|  |  |
| --- | --- |
| **Chân** | **Kết nối và chức năng** |
| 12V | Chân cấp nguồn dương (9 – 12V) |
| GND | Chân cấp GND (nối chung với GND Arduino) |
| 5V | Chân nguồn ra 5V |
| ENA | Chân điều khiển mạch cầu H động cơ 1 (cấp xung PWM cho động cơ 1) |
| IN1 | Chân nhận tín hiệu điều khiển động cơ 1 |
| IN2 | Chân nhận tín hiệu điều khiển động cơ 1 |
| ENB | Chân điều khiển mạch cầu H động cơ 1 (cấp xung PWM cho động cơ 2) |
| IN3 | Chân nhận tín hiệu điều khiển động cơ 2 |
| IN4 | Chân nhận tín hiệu điều khiển động cơ 2 |
| OUT1 | Chân nối với chân dương M1 động cơ 1 |
| OUT2 | Chân nối với chân âm M2 động cơ 1 |
| OUT3 | Chân nối với chân dương M1 động cơ 1 |
| OUT4 | Chân nối với chân âm M2 động cơ 1 |

*Bảng kết nối chân của mạch cầu H L298N*

Ngoài ra, nhóm còn lựa chọn bộ thu phát tín hiệu Devo 7 để điều khiển từ xa cho hệ xe 2 với ưu điểm:

* Được thiết kế để đáp ứng nhu cầu điều khiển từ xa cho các thiết bị không người lái.
* Với khả năng điều khiển 7 kênh, Devo 7 mang lại sự linh hoạt và chính xác, phù hợp cho việc kiểm soát máy bay điều khiển từ xa, trực thăng, hoặc xe điều khiển từ xa.
* Sử dụng công nghệ sóng 2.4GHz phổ biến, Devo 7 giúp đảm bảo ổn định và độ tin cậy trong mọi môi trường, với khả năng chống nhiễu tốt.
* Màn hình LCD thông minh hiển thị thông tin chi tiết, trong khi khả năng lập trình linh hoạt và kết nối ổn định tạo ra một trải nghiệm điều khiển mạnh mẽ và đáng tin cậy.

Rx 701 thường sử dụng dạng tín hiệu ngõ ra là xung PPM (Pulse Position Modulation). PPM là một dạng của điều chế xung trong hệ thống truyền thông, nơi tín hiệu được truyền bằng cách thay đổi vị trí của xung (pulse) trong một chuỗi xung liên tục.

**3.3.4 Khối động cơ và cảm biến đọc xung của động cơ**

Khối động cơ và cảm biến đọc xung của động cơ gồm:

- Hai động cơ bên phải và bên trái là cơ cấu chấp hành chính của xe để điều khiển xe cân bằng.

- Hai bộ mã hoá quay encoder gắn liền với mỗi động cơ giúp hồi tiếp xung về Khối điều khiển trung tâm từ đó xử lý tín hiệu giúp xác định chính xác vị trí, tốc độ quay và chiều quay của từng động cơ.

Vì yêu cầu đặt ra là động cơ gắn sẵn bộ mã hoá quay nên nhóm đã lựa chọn động cơ DC Servo JGB37-520 DC (loại 12VDC 333RPM) với ưu điểm như sau:

- Động cơ tích hợp thêm Encoder hai kênh A, B giúp đọc và điều khiển chính xác vị trí, chiều quay của động cơ.

- Cấu tạo bằng kim loại cho độ bền và ổn định cao.

- Động cơ được sử dụng nguyên liệu chất lượng cao với lõi dây đồng nguyên chất, lá thép 407, nam châm có độ từ tính mạnh… .

**Thông số kỹ thuật:**

- Động cơ:

+ Tỉ số truyền 30:1 (động cơ quay 30 vòng trục chính hộp giảm tốc quay 1 vòng).

+ Dòng không tải: 120mA

+ Dòng chịu đựng tối đa khi có tải:1A

+ Tốc độ không tải: 333RPM (333 vòng 1 phút)

+ Tốc độ chịu đựng tối đa khi có tải: 250RPM (250 vòng 1 phút)

+ Lực kéo Moment định mức: 3.5KG.CM

+ Lực léo Moment tối đa: 5KG.CM

+ Chiều dài hộp số L: 22mm

- Encoder:

+ Điện áp cấp cho Encoder hoạt động: 3.3 – 5 VDC.

+ Cảm biến từ trường Hall, có 2 kênh AB lệch nhau.

+ Đĩa Encoder trả 11 xung/ 1 kênh/ 1 vòng.

+ Số xung Encoder mỗi kênh trên 1 vòng quay trục chính: 11 x 30 = 330 xung.



Hình Động cơ DC Servo JGB37-520 12V 333RPM

Bảng kết nối chân của động cơ và Encoder

|  |  |
| --- | --- |
| **Chân** | **Kết nối và chức năng** |
| M1 (đỏ) | Dây cấp nguồn dương cho động cơ |
| GND (đen) | Dây cấp nguồn âm cho Encoder |
| C1/A (vàng) | Tín hiệu trả xung kênh A |
| C2/B (xanh lá) | Tín hiệu trả xung kênh B |
| VCC (xanh dương) | Dây cấp nguồn cho Encoder |
| M2 (trắng) | Dây cấp nguồn âm cho động cơ |

**3.3.5 Khối cấp nguồn**

Lựa chọn khối cấp nguồn gồm:

- Pin Lipo 12V.

- Công tắc bập bênh.

Ưu điểm của khối cấp nguồn đã được chọn:

- Sử dụng được linh động, gọn nhẹ hơn so với dùng nguồn tổ ong hay adapter.

- Lắp đặt, đi dây dễ dàng, không bị vướng dây nguồn khi xe chạy.

- Dễ dàng sử dụng, thay và sạc pin cho nguồn.

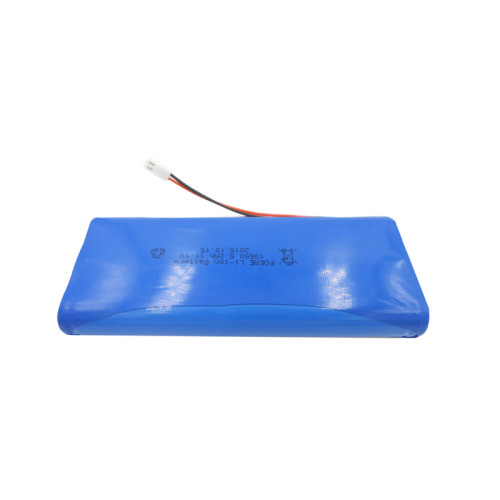
- Sử dụng công tắc giúp thuận tiện hơn trong việc đóng/tắt hệ thống.

Cấp nguồn cho hệ thống:

- Nguồn 12VDC được cấp vào chân Vcc và GND của L298N.

- Nguồn 5VDC từ L298N được cấp cho Arduino, mạch thu sóng, cảm biến gia tốc MPU6050 và hai encoder.

– Ngõ ra OUT1 OUT2 của mạch cầu H cấp nguồn cho động cơ 1, ngõ ra OUT3 OUT4 của mạch cầu H cấp nguồn cho động cơ 2.



Hình Pin Lipo 12V

**3.4 Bảng đi dây hệ thống**

**3.4.1 Phần nguồn**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Nguồn 9.3VDC** | **Arduino Mega** | **MPU6050** | **Bluetooth hc06** | **L298** | **Động cơ 1 và Encoder 1** | **Động cơ 2 và Encoder 2** |
| OUT+ | Power Jack |  |  | 12V |  |  |
| OUT- |  |  | GND |  |  |
|  | 5V | VCC |  |  | VCC | VCC |
|  | GND | GND | GND | GND | GND | GND |
|  |  |  | VCC | 5V |  |  |
|  |  |  |  | OUT1 |  |  |
|  |  |  |  | OUT2 |  |  |
|  |  |  |  | OUT3 |  | M1 |
|  |  |  |  | OUT4 |  | M2 |

*Bảng kết nối dây phần nguồn*

**3.4.2 Phần tín hiệu điều khiển**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Arduino Mega** | **MPU6050** | **Bluetooth hc06** | **L298** | **Động cơ 1 và Encoder 1** | **Động cơ 2 và Encoder 2** |
| 18 (TX1) |  | RX |  |  |  |
| 19 (TX2) |  | TX |  |  |  |
| 9 |  |  | ENA |  |  |
| 30 |  |  | IN1 |  |  |
| 32 |  |  | IN2 |  |  |
| 10 |  |  | ENB |  |  |
| 36 |  |  | INT3 |  |  |
| 34 |  |  | INT4 |  |  |
| 21 (SCL) | SCL |  |  |  |  |
| 20 (SDA) | SDA |  |  |  |  |
| 2 |  |  |  | C1/A |  |
| 22 |  |  |  | C2/B |  |
| 3 |  |  |  |  | C1/A |
| 26 |  |  |  |  | C2/B |

*Bảng kết nối dây phần tín hiệu điều khiển*

**3.4.3 Sơ đồ nối dây**

Khung sườn sử dụng chất liệu

|  |  |  |  |
| --- | --- | --- | --- |
| **Kí hiệu** | **Đơn vị** | **Ý nghĩa** | **Thông số** |
| m | kg | Khối lượng bánh xe |  |
| M | kg | Khối lượng của robot |  |
| R | m | Bán kính bánh xe |  |
| W | m | Chiều rộng của robot |  |
| D | m | Chiều ngang của robot |  |
| H | m | Chiều cao của robot |  |
| L | m | Khoảng cách từ trọng tâm robot đến trục bánh xe |  |
| fw |  | Hệ số ma sát giữa hai bánh xe và mặt phẳng di chuyển |  |
| fm |  | Hệ số ma sát giữa robot và động cơ DC |  |
| jm | kg.m2 | Moment quán tính của động cơ DC |  |
| Rm | ohm | Điện trở động cơ DC |  |
| Kb | V.sec/rad | Hệ số EMF của động cơ DC |  |
| Kt | Nm/A | Moment xoắn của động cơ DC |  |
| N |  | Tỉ số giảm tốc |  |
| G | m/s2 | Gia tốc trọng trường | 9.81 |

*Bảng giá trị thực tế mô hình cơ khí xe 2 bánh tự cân bằng*

*Hình ảnh thực tế của hệ xe 2 bánh tự cân bằng*

**3.5 Thi công hệ thống**

Sơ đồ kết nối phần cứng của mô hình

**CHƯƠNG 4**

**PHẦN MỀM**

**4.1 Yêu cầu thiết kế**

Về phần điều khiển hệ thống các yêu cầu thiết để xe đảm bảo cân bằng như sau:

* Đọc tín hiệu cảm biến MPU6050.
* Đọc giá trị xung encoder trả về.
* Xuất xung PWM điều khiển mạch cầu H điều khiển động cơ theo tốc độ chạy thuận, chạy nghịch, dừng động cơ.
* Giới hạn góc nghiêng để cân bằng xe từ -15 độ đến 15 độ.

**4.2 Xây dựng thuật toán điều khiển**

Để sử dụng và lập trình Arduino Uno thì phần mền dụng để lập trình là Arduino IDE ..., thuận tiện tỏng việc kết nối Arduino với máy tình bằng dây cáp USB.

Ngôn ngữ lập trình của Arduino là C/C++, lập trình đơn giản hơn nhiều so với các loại vi điều khác nhờ cộng đồng người dùng lớn và sẵn các thư viện hỗ trợ học tập lập trình đa dạng.

Chương trình điều khiển được đưa vào ở phần Phụ lục.

**4.2.1 Thuật toán chương trình chính**

*Hình Lưu đồ thuật toán chương trình chính*

Đọc và xử lý giá trị từ MPU6050

Ngắt nhận và xử lý tín hiệu từ Encoder

Khởi tạo Serial

Khởi tạo timer 2

Khởi tạo ngắt

Khỏi tạo I2C

Thời gian lấy mẫu 7ms

Điều khiển động cơ

Tính toán xung PWM mỗi bánh

Cập nhật các giá trị mới sau khoảng thời gian lấy mẫu

Bắt đầu

Khai báo thư viện

Khai báo port

Khai báo biến

**CHƯƠNG 5**

**KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

**5.1 Những kết quả đạt được**

**A machine with wires and wires

Description automatically generated**

Hình : Robot cân bằng trong thực tế

**A small black machine with wheels and wires

Description automatically generated**

Hình : Robot cân bằng trong thực tế

**A line of blue dots

Description automatically generated with medium confidence**

Hình : Đáp ứng hệ thống khi bật điện đến khi ổn định

**5.2 Hạn chế**

**5.3 Kết luận**

**5.4 Hướng phát triển**

**A screen shot of a graph

Description automatically generated******

Hình : Đáp ứng hệ thống khi có tác động ngoại lực

Hình : Đáp ứng vị trí



Hình : Đáp ứng góc quay

**5.2 Kết luận**

Sau quá trình nghiên cứu và thực hiện đề tài, nhóm chúng tôi xin tổng kết những thành tựu đáng chú ý như sau:

* Hoàn thiện thiết kế của mô hình xe hai bánh tự cân bằng, đảm bảo sự hoàn chỉnh và hiệu quả của nó.
* Thực hiện tính toán mô hình toán học và xây dựng hàm trạng thái cho hệ thống, giúp nâng cao hiểu biết về đặc tính và hoạt động của xe.
* Các thành phần điện và vi điều khiển đã hoạt động hiệu quả, bao gồm:
  + Cảm biến góc nghiêng trả về giá trị chính xác, cung cấp thông tin đáng tin cậy.
  + Điều chế độ rộng xung cho mạch cầu H ổn định, tránh tình trạng quá tải và quá nhiệt.
  + Vi điều khiển thực hiện các chức năng một cách ổn định, xử lý tín hiệu nhanh chóng.
* Mô hình có khả năng cân bằng tại chỗ và di chuyển theo tín hiệu điều khiển, thể hiện tính ứng dụng thực tế của nó.
* Tuy nhiên, do thời gian thực hiện đề tài có hạn, nên vẫn còn tồn tại một số hạn chế mà chúng tôi đang nỗ lực để cải thiện.

**5.3 Hạn chế**

Các hạn chế của đề tài trong quá trình nghiên cứu là:

- Xe được thiết kế chỉ cân bằng được trên bề mặt phẳng, không thể áp dụng cho địa hình gồ ghề, nghiêng, phức tạp

- Phần mô phỏng so với mô hình thực tế còn nhiều sai số do các thông số đưa ra chưa chính xác

- Xe chưa chịu được tác động lớn từ bên ngoài

**5.4 Hướng phát triển của đề tài**

- Thiết kế bộ điều khiển với bộ trọng số tối ưu nhất để hệ thống vận hành mượt mà, hạn chế được nhiễu tối đa.

- Khảo sát và thiết kế cho hệ xe hai bánh tự cân bằng trên các bề mặt môi trường như dốc nghiêng, gồ ghề, trơn trượt, ...

- Phát triển thêm nhiều chức năng cho hệ như mang vác vật, bám đối tượng, bám line,...

**PHỤ LỤC**

**TÀI LIỆU THAM KHẢO**