

**BỘ GIÁO DỤC & ĐÀO TẠO**  
**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP. HỒ CHÍ MINH**  
**KHOA ĐIỆN – ĐIỆN TỬ**  
**BỘ MÔN TỰ ĐỘNG ĐIỀU KHIỂN**



**BÁO CÁO ĐIỀU KHIỂN THÔNG MINH**

**Đề Tài:**

**MÔ HÌNH ROBOT 2 BÁNH TỰ CÂN BẰNG**  
**SỬ DỤNG BỘ ĐIỀU KHIỂN LQR**

**GVHD:** PSG.TS Nguyễn Minh Tâm

**SVTH:**

Nguyễn Văn Pháp  
Vũ Tiến Phát

**MSSV:**

21151303  
21151309

**Tp. Hồ Chí Minh tháng 6 năm 2024**

## **LỜI CAM ĐOAN**

Đề tài này là do nhóm của chúng em tự thực hiện dựa vào một số tài liệu tham khảo và chúng em xin cam đoan đề tài này không sao chép bất kỳ công trình đã có trước đó. Nếu có sao chép nhóm chúng em hoàn toàn chịu trách nhiệm.

## **LỜI CẢM ƠN**

Trong suốt quá trình thực hiện đề tài, mặc dù gặp nhiều khó khăn nhưng được sự giúp đỡ từ thầy, cô, các anh chị đi trước cũng như các bạn nên Đồ án cũng đã hoàn thành đúng tiến độ. Nhóm thực hiện xin cảm ơn thầy Nguyễn Văn Đông Hải qua những video hướng dẫn cực kỳ hay và bổ ích.

Đồng thời, nhóm cũng xin chân thành cảm ơn các thầy cô trong Khoa Điện Điện Tử đã tạo điều kiện, cung cấp những kiến thức cơ bản cần thiết để nhóm có đủ kiến thức để thực hiện quá trình nghiên cứu.

Ngoài ra, nhóm còn nhận được sự hướng dẫn nhiệt tình từ các anh (chị) đi trước. Anh (chị) cũng giới thiệu thêm các tài liệu tham khảo trong việc thực hiện nghiên cứu.

# MỤC LỤC

LỜI CAM ĐOAN .....	ii
LỜI CẢM ƠN .....	iii
DANH MỤC HÌNH ẢNH .....	v
DANH MỤC BẢNG .....	vi
CHƯƠNG 1: TỔNG QUAN .....	1
1.1. Lý do chọn đề tài .....	1
1.2. Mục tiêu đề tài .....	2
1.3. Phương pháp thực hiện .....	3
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT .....	4
2.1. Mô hình toán học .....	4
2.2. Bộ lọc Kalman .....	12
2.2.1. Giới thiệu về bộ lọc Kalman .....	12
2.2.2. Quá trình ước lượng .....	12
2.2.3. Thuật toán Kalman rời rạc: .....	13
CHƯƠNG 3. MÔ PHỎNG MATLAB .....	16
3.1. Tiến hành mô phỏng .....	16
3.2. Kết quả mô phỏng .....	19
CHƯƠNG 4. MÔ HÌNH THỰC TẾ .....	20
CHƯƠNG 5. KẾT LUẬN .....	23
TÀI LIỆU THAM KHẢO .....	24

## DANH MỤC HÌNH ẢNH

Hình 1. Mô hình xe 2 bánh tự cân bằng trong mặt phẳng.....	5
Hình 2. Quy trình thực hiện bộ lọc Kalman.....	14
Hình 3. Tổng quan chu trình thực hiện bộ lọc Kalman hoàn chỉnh.....	15
Hình 4. Thông số hệ thống xe 2 bánh tự cân bằng.....	16
Hình 5. Ma trận A.....	16
Hình 6. Ma trận B.....	17
Hình 7. Thông số ma trận Q và ma trận R.....	17
Hình 8. Kết quả tính ma trận K của LQR.....	17
Hình 9. Mô phỏng Simulink.....	18
Hình 10. Khối Xe_2_Banh_Tu_Can_Bang.....	18
Hình 11. Thông số mô hình trong khối Matlab function.....	19
Hình 12. Kết quả mô phỏng.....	19

## **DANH MỤC BẢNG**

Bảng 1: Ký hiệu và ý nghĩa các đại lượng .....	6
--	---

## CHƯƠNG 1. TỔNG QUAN

### 1.1. Lý do chọn đề tài

Ngày nay, vấn đề về ùn tắc giao thông và ô nhiễm môi trường tại các thành phố lớn do các phương tiện giao thông cá nhân gây ra đang được quan tâm hàng đầu. Vì vậy, việc tìm hiểu và chế tạo ra các loại phương tiện di chuyển cá nhân ít chiếm diện tích không gian và không tạo ra khí thải gây ô nhiễm môi trường đang được quan tâm nghiên cứu bởi nhiều người. Một sản phẩm đã được đưa ra sau sự nghiên cứu, được gọi là xe hai bánh cân bằng, giúp con người di chuyển linh hoạt hơn, ít chiếm không gian và thân thiện với môi trường. Sự thiết thực và hấp dẫn của đề tài này đã được nhận thấy cũng như có liên quan đến những kiến thức được học tại trường, do đó việc thực hiện đề tài này đã được quyết định bởi nhóm chúng em, với mong muốn tìm hiểu về loại hình di chuyển thông minh này.

### 1.2. Mục tiêu đề tài

Mục tiêu của đề tài là xây dựng mô hình robot cân bằng hai bánh dựa trên lý thuyết về con lắc ngược. Các mục tiêu bao gồm:

- Đưa ra mô hình robot và nguyên lý hoạt động tự cân bằng.
- Tính toán các thông số động học, phương trình trạng thái không gian của mô hình.
- Tính toán, chọn cảm biến, bộ điều khiển động cơ và bộ điều khiển trung tâm. Trong đề tài, cảm biến sử dụng là MPU 6050, bộ điều khiển trung tâm là STM32 (STM32F407VET6).
- Mô phỏng mô hình trên Matlab Simulink.
- Nghiên cứu và ứng dụng bộ lọc Kalman để lọc nhiễu từ cảm biến.
- Đưa ra thuật toán điều khiển động cơ và giữ cân bằng cho robot.
- Robot có thể hoạt động một cách tự động.



### 1.3. Phương pháp thực hiện

- Thiết kế và thi công phần cứng cho xe hai bánh tự cân bằng.
- Xác định góc nghiêng của xe sử dụng cảm biến MPU 6050 kết hợp với bộ lọc Kalman để lọc nhiễu.
- Sử dụng thuật toán LQR để điều khiển xe cân bằng.
- Sử dụng bộ điều khiển PI để tăng độ đáp ứng của thuật toán LQR.

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

### 2.1. Mô hình toán học

- Lý thuyết về LQR (*Linear – quadratic regulator*)

Đối tượng tuyến tính mô tả bởi phương trình trạng thái:

$$\dot{x}(t) = Ax(t) + Bu(t)$$

Ta phải thiết kế bộ điều khiển phản hồi trạng thái K sao cho hàm mục tiêu J là nhỏ nhất.

Xác định hàm chỉ tiêu chất lượng:

$$J = \int_0^{\infty} (x^T Q x + u^T R u + 2x^T N u) dt$$

Trong đó Q và R là các ma trận trọng số.

Quy tắc điều khiển phản hồi để giảm thiểu giá trị của hàm J là:

$$u(t) = -Kx(t)$$

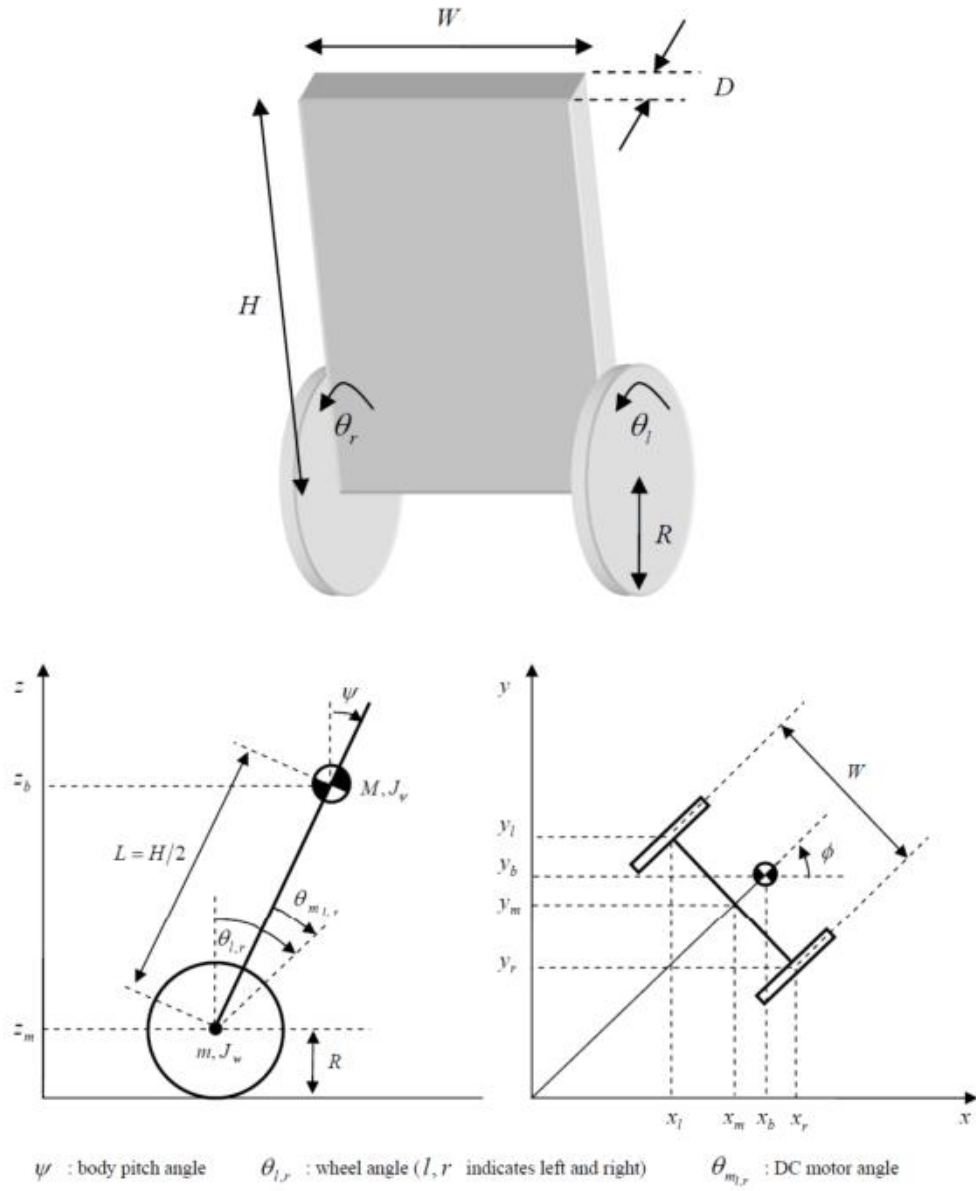
K được xác định là :

$$K = R^{-1}B^T S$$

Khi S là hằng số thì  $\dot{S} = 0$  ta có phương trình Riccati như sau:

$$A^T S + SA - SBR^{-1}B^T S + Q = 0$$

- Sơ đồ và hệ quy chiếu xe 2 bánh tự cân bằng



Hình 1. Mô hình xe 2 bánh tự cân bằng trong mặt phẳng

## CHƯƠNG 2. CƠ SỞ LÝ THUYẾT

*Bảng 1: Ký hiệu và ý nghĩa các đại lượng*

Ký hiệu	Đơn vị	Ý nghĩa
m	Kg	Khối lượng của bánh xe
M	Kg	Khối lượng của Robot
R	m	Bán kính bánh xe
W	m	Chiều ngang của Robot
D	m	Chiều rộng của Robot
H	m	Chiều cao của Robot
L	m	Khoảng cách từ trọng tâm Robot đến trục bánh xe
$f_w$		Hệ số ma sát giữa bánh xe và mặt phẳng di chuyển
$f_m$		Hệ số ma sát giữa Robot và động cơ DC
$J_m$	$\text{kgm}^2$	Moment quán tính động cơ DC
$R_m$	$\Omega$	Điện trở động cơ DC
$K_b$	Vs/rad	Hệ số EMF của động cơ DC
$K_t$	Nm/A	Moment xoắn của động cơ DC
N		Tỉ số giảm tốc
G	$\text{m/s}^2$	Gia tốc trọng trường
$\theta$	rad	Góc trung bình của bánh trái và bánh phải
$\theta_{l,r}$	rad	Góc của bánh trái và bánh phải
$\psi$	rad	Góc nghiêng của phần thân Robot
$\phi$	rad	Góc xoay của Robot
$x_l, y_l, z_l$	m	Tọa độ bánh trái
$x_r, y_r, z_r$	m	Tọa độ bánh phải
$x_m, y_m, z_m$	m	Tọa độ trung bình
$F_\theta, F_\psi, F_\phi$	Nm	Moment phát động theo các phương khác nhau
$F_{l,r}$	Nm	Moment phát động của động cơ bánh trái, phải
$i_l, i_r$	A	Dòng điện động cơ bánh trái, phải
$v_l, v_r$	V	Điện áp động cơ bánh trái, phải

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:

$$\begin{bmatrix} \theta \\ \phi \end{bmatrix} = \begin{bmatrix} \frac{1}{2}(\theta_l + \theta_r) \\ \frac{R}{W}(\theta_l - \theta_r) \end{bmatrix} \#(2.1)$$

Tọa độ trung bình của Robot trong hệ quy chiếu:

$$\begin{bmatrix} x_m \\ y_m \\ z_m \end{bmatrix} = \begin{bmatrix} \int \dot{x}_m dt \\ \int \dot{y}_m dt \\ R \end{bmatrix} \#(2.2)$$

$$\text{Và } \begin{bmatrix} \dot{x}_m \\ \dot{y}_m \end{bmatrix} = \begin{bmatrix} R\dot{\theta} \cos \phi \\ R\dot{\theta} \sin \phi \end{bmatrix} \#(2.3)$$

Tọa độ bánh trái trong hệ quy chiếu:

$$\begin{bmatrix} x_l \\ y_l \\ z_l \end{bmatrix} = \begin{bmatrix} x_m - \frac{W}{2} \sin \phi \\ y_m + \frac{W}{2} \cos \phi \\ z_m \end{bmatrix} \#(2.4)$$

Tọa độ bánh phải trong hệ quy chiếu:

$$\begin{bmatrix} x_r \\ y_r \\ z_r \end{bmatrix} = \begin{bmatrix} x_m + \frac{W}{2} \sin \phi \\ y_m - \frac{W}{2} \cos \phi \\ z_m \end{bmatrix} \#(2.5)$$

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

$$\begin{bmatrix} x_b \\ y_b \\ z_b \end{bmatrix} = \begin{bmatrix} x_m + L \sin \psi \cos \phi \\ y_m + L \sin \psi \sin \phi \\ z_m + L \cos \psi \end{bmatrix} \#(2.6)$$

Phương trình động năng của chuyển động tịnh tiến:

$$T_1 = \frac{1}{2}m(\dot{x}_l^2 + \dot{y}_l^2 + \dot{z}_l^2) + \frac{1}{2}m(\dot{x}_r^2 + \dot{y}_r^2 + \dot{z}_r^2) + \frac{1}{2}M(\dot{x}_b^2 + \dot{y}_b^2 + \dot{z}_b^2) \#(2.7)$$

Phương trình động năng của chuyển động quay:

$$T_2 = \frac{1}{2}J_w\dot{\theta}_l^2 + \frac{1}{2}J_w\dot{\theta}_r^2 + \frac{1}{2}J_\psi\dot{\psi}^2 + \frac{1}{2}J_\phi\dot{\phi}^2 + \frac{1}{2}n^2J_m(\dot{\theta}_l - \dot{\psi})^2 + \frac{1}{2}n^2J_m(\dot{\theta}_r - \dot{\psi})^2 \quad (2.8)$$

Với  $\frac{1}{2}n^2J_m(\dot{\theta}_l - \dot{\psi})^2$  là động năng quay của phần ứng động cơ trái

$\frac{1}{2}n^2J_m(\dot{\theta}_r - \dot{\psi})^2$  là động năng quay của phần ứng động cơ phải

Phương trình thế năng:

$$U = mgz_l + mgz_r + Mgz_b \quad (2.9)$$

Phương trình Lagrange:

$$L = T_1 + T_2 - U \quad (2.10)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) - \frac{\partial L}{\partial \theta} = F_\theta \quad (2.11)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\psi}}\right) - \frac{\partial L}{\partial \psi} = F_\psi \quad (2.12)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\phi}}\right) - \frac{\partial L}{\partial \phi} = F_\phi \quad (2.13)$$

Lấy đạo hàm L theo các biến  $\theta, \psi, \phi$  ta được:

$$[(2m + M)R^2 + 2J_w + 2n^2J_m]\ddot{\theta} + (MLR \cos \psi - 2n^2J_m)\ddot{\psi} - MLR\dot{\psi}^2 \sin \psi = F_\theta \quad (2.14)$$

$$(MLR \cos \psi - 2n^2J_m)\ddot{\theta} + (ML^2 + J_\psi + 2n^2J_m)\ddot{\psi} - MgL \sin \psi - ML^2\dot{\phi}^2 \sin \psi \cos \psi = F_\psi \quad (2.15)$$

$$\left[\frac{1}{2}mW^2 + J_\phi + \frac{W^2}{2R^2}(J_w + n^2J_m) + ML^2 \sin^2 \psi\right]\ddot{\phi} + 2ML^2\dot{\psi}\dot{\phi} \sin \psi \cos \psi = F_\phi \quad (2.16)$$

Momen động lực do động lực DC sinh ra:

$$\begin{bmatrix} F_\theta \\ F_\psi \\ F_\phi \end{bmatrix} = \begin{bmatrix} F_l + F_r \\ F_\psi \\ \frac{W}{2R}(F_l - F_r) \end{bmatrix} \quad (2.17)$$

Và:

$$F_l = nK_t i_l + f_m(\dot{\psi} - \dot{\theta}_l) - f_w \dot{\theta}_l \quad (2.18)$$

$$F_r = nK_t i_r + f_m(\dot{\psi} - \dot{\theta}_r) - f_w \dot{\theta}_r \quad (2.19)$$

$$F_\psi = -nK_t i_l - nK_t i_r - f_m(\dot{\psi} - \dot{\theta}_l) - f_m(\dot{\psi} - \dot{\theta}_r) \quad (2.20)$$

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

$$L_m \dot{i}_{l,r} = v_{l,r} + K_b(\dot{\psi} - \dot{\theta}_{l,r}) - R_m i_{l,r} \# (2.21)$$

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

$$i_{l,r} = \frac{v_{l,r} + K_b(\dot{\psi} - \dot{\theta}_{l,r})}{R_m} \# (2.22)$$

Từ đó, các moment lực sinh ra:

$$F_\theta = \alpha(v_l + v_r) - 2(\beta + f_w)\dot{\theta} + 2\beta\dot{\psi} \# (2.23)$$

$$F_\psi = -\alpha(v_l + v_r) + 2\beta\dot{\theta} - 2\beta\dot{\psi} \# (2.24)$$

$$F_\phi = \frac{W}{2R}\alpha(v_l - v_r) - \frac{W}{2R}(\beta + f_w)\dot{\phi} \# (2.25)$$

Với  $\alpha = \frac{nK_t}{R_m}$  và  $\beta = \frac{nK_t K_b}{R_m} + f_m$

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

$$\begin{aligned} [(2m + M)R^2 + 2f_w + 2n^2 J_m]\ddot{\theta} + (MLR \cos \psi - 2n^2 J_m)\ddot{\psi} - MLR\dot{\psi}^2 \sin \psi \\ = \alpha(v_l + v_r) - 2(\beta + f_w)\dot{\theta} + 2\beta\dot{\psi} \# (2.26) \end{aligned}$$

$$\begin{aligned} (MLR \cos \psi - 2n^2 J_m)\ddot{\theta} + (ML^2 + J_\psi + 2n^2 J_m)\ddot{\psi} - MgL \sin \psi - ML^2\dot{\phi}^2 \sin \psi \cos \psi \\ = -\alpha(v_l + v_r) + 2\beta\dot{\theta} - 2\beta\dot{\psi} \# (2.27) \end{aligned}$$

$$\begin{aligned} \left[ \frac{1}{2}mW^2 + J_\phi + \frac{W^2}{2R^2}(J_w + n^2 J_m) + ML^2 \sin^2 \psi \right] \ddot{\phi} + 2ML^2 \dot{\psi} \dot{\phi} \sin \psi \cos \psi \\ = \frac{W}{2R}\alpha(v_l - v_r) - \frac{W^2}{2R^2}(\beta + f_w)\dot{\phi} \# (2.28) \end{aligned}$$

- *Tuyến tính hóa hệ thống*

Giả sử ta đặt các biến trạng thái như sau:

$x_1 = \theta$ ,  $x_2 = \dot{\theta}$ , là góc quay và vận tốc góc quay bánh xe

$x_3 = \psi$ ,  $x_4 = \dot{\psi}$ , là góc nghiêng và vận tốc nghiêng của thân Robot

$x_5 = \phi$ ,  $x_6 = \dot{\phi}$ , là góc xoay và vận tốc xoay của Robot

$$\Rightarrow \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f_1(x_1, x_2, x_3, x_4, x_5, x_6, v_l, v_r) \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = f_2(x_1, x_2, x_3, x_4, x_5, x_6, v_l, v_r) \\ \dot{x}_5 = x_6 \\ \dot{x}_6 = f_3(x_1, x_2, x_3, x_4, x_5, x_6, v_l, v_r) \end{cases} \#(2.29)$$

Với  $x = [x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6]^T = [\theta \ \dot{\theta} \ \psi \ \dot{\psi} \ \phi \ \dot{\phi}]^T$

Nếu chọn điểm làm việc là:

$$x_0 = [0 \ 0 \ 0 \ 0 \ 0 \ 0]^T, u_0 = [0 \ 0]^T$$

Ta có thể tuyến tính hóa hệ thống về dạng:  $\dot{x} = Ax + Bu$

Với:  $u = \begin{bmatrix} v_l \\ v_r \end{bmatrix}$

Tìm ma trận A như sau:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ \left. \frac{\partial f_1}{\partial x_1} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_1}{\partial x_2} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_1}{\partial x_3} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_1}{\partial x_4} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_1}{\partial x_5} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_1}{\partial x_6} \right|_{\substack{u=u_0 \\ x=x_0}} \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \left. \frac{\partial f_2}{\partial x_1} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_2}{\partial x_2} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_2}{\partial x_3} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_2}{\partial x_4} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_2}{\partial x_5} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_2}{\partial x_6} \right|_{\substack{u=u_0 \\ x=x_0}} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \left. \frac{\partial f_3}{\partial x_1} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_3}{\partial x_2} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_3}{\partial x_3} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_3}{\partial x_4} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_3}{\partial x_5} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_3}{\partial x_6} \right|_{\substack{u=u_0 \\ x=x_0}} \end{bmatrix}$$

Tìm ma trận B như sau:

$$B = \begin{bmatrix} 0 & 0 \\ \left. \frac{\partial f_1}{\partial v_l} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_1}{\partial v_r} \right|_{\substack{u=u_0 \\ x=x_0}} \\ 0 & 0 \\ \left. \frac{\partial f_2}{\partial v_l} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_2}{\partial v_r} \right|_{\substack{u=u_0 \\ x=x_0}} \\ 0 & 0 \\ \left. \frac{\partial f_3}{\partial v_l} \right|_{\substack{u=u_0 \\ x=x_0}} & \left. \frac{\partial f_3}{\partial v_r} \right|_{\substack{u=u_0 \\ x=x_0}} \end{bmatrix}$$



Lúc này ta có ma trận trọng số như sau:

$$Q = \begin{bmatrix} Q_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & Q_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & Q_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & Q_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & Q_5 & 0 \\ 0 & 0 & 0 & 0 & 0 & Q_6 \end{bmatrix} \text{ và } R = \begin{bmatrix} R_1 & 0 \\ 0 & R_2 \end{bmatrix}$$

Với các thông số  $Q_1, Q_2, Q_3, Q_4, Q_5, Q_6, R_1, R_2$  để tinh chỉnh cho bộ điều khiển LQR. Trong đó tham số  $Q_1, Q_2, Q_3, Q_4, Q_5, Q_6$  được coi là trọng số tối ưu tương ứng cho 6 biến trạng thái  $\theta, \dot{\theta}, \psi, \dot{\psi}, \phi, \dot{\phi}$ . Với mô hình hệ xe hai bánh ta có ma trận  $Q$  là ma trận 6x6 (tương ứng với 6 biến trạng thái) và  $R$  là 2x2 (tương ứng với 2 biến ngõ vào). Sau khi chọn được các tham số điều khiển tương ứng, chúng ta có thể xây dựng được tham số phản hồi  $K$  với tín hiệu điều khiển  $u = -Kx$ .

Thông số  $K$  được tính toán dựa vào phương trình Riccati. Tuy nhiên Matlab đã hỗ trợ việc tính toán thủ công bằng hàm  $LQR(A,B,Q,R)$  khi các ma trận  $A, B, Q, R$  đã được tìm ra.

### 2.2. Bộ lọc Kalman

#### 2.2.1. Giới thiệu về bộ lọc Kalman

Năm 1960, Rudolf E. Kálmán (1930 – 2016) xuất bản bài báo nổi tiếng của mình mô tả một giải pháp đệ quy (recursive) cho vấn đề lọc dữ liệu tuyến tính rời rạc.

Kalman Filter là một trong những thuật toán ước lượng quan trọng và phổ biến nhất. Bộ lọc Kalman tạo ra các ước tính về các biến ẩn dựa trên các phép đo không chính xác và không chắc chắn.

Ngày nay, Kalman Filter được sử dụng trong theo dõi mục tiêu (Radar), hệ thống định vị, hệ thống điều khiển, đồ họa máy tính và nhiều hơn nữa.

#### 2.2.2. Quá trình ước lượng

Mô hình của đối tượng tuyến tính rời rạc được biểu diễn thông qua hệ phương trình trạng thái sau:

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1} \quad \#(2.30)$$

$$z_k = Hx_k + v_k \quad \#(2.31)$$

Trong đó:

- $x$ : biến trạng thái.
- $u$ : biến đầu vào.
- $z$ : trạng thái đầu ra.
- $w, v$ : nhiễu quá trình và nhiễu đo lường.

Giả sử các nhiễu quá trình và nhiễu đo lường có các phương sai tương ứng là  $Q$  và  $R$ .

$$p(w) \sim N(0, Q) \quad \#(2.32)$$

$$p(v) \sim N(0, R) \quad \#(2.33)$$

Những tính toán căn bản của bộ lọc:

Định nghĩa:

- $\hat{x}_k^-$  là giá trị ước lượng của  $x_k$  trước khi ta xử lý giá trị đo tại thời điểm  $k$ .
- $\hat{x}_k^+$  là giá trị ước lượng của  $x_k$  sau khi ta xử lý giá trị đo tại thời điểm  $k$ .
- $\hat{x}_k$  là giá trị ước lượng trạng thái sau tại bước  $k$  có được sau khi so sánh với giá trị đo  $z_k$

Chúng ta có sai số ước lượng trạng thái trước và sau:

$$\begin{cases} e_k^- \equiv x_k - \hat{x}_k^- \\ e_k \equiv x_k - \hat{x}_k \end{cases} \#(2.34)$$

Tương quan sai số ước lượng trước “priori”:

$$P_k^- = E\{e_k^- e_k^{-T}\} \#(2.35)$$

Tương quan sai số ước lượng sau “posteriori”:

$$P_k = E\{e_k e_k^T\} \#(2.36)$$

Khi lấy đạo hàm phương trình bộ lọc Kalman, với mục đích tìm một phương trình để tính toán trạng thái ước lượng posteriori  $\hat{x}_k$  thể hiện sự tương quan giữa giá trị ước lượng priori  $\hat{x}_k^-$  và độ sai lệch giữa giá trị đo thực  $z_k$  và giá trị đo ước lượng  $H\hat{x}_k^-$ :

$$\hat{x}_k = \hat{x}_k^- + K(z_k - H\hat{x}_k^-) \#(2.37)$$

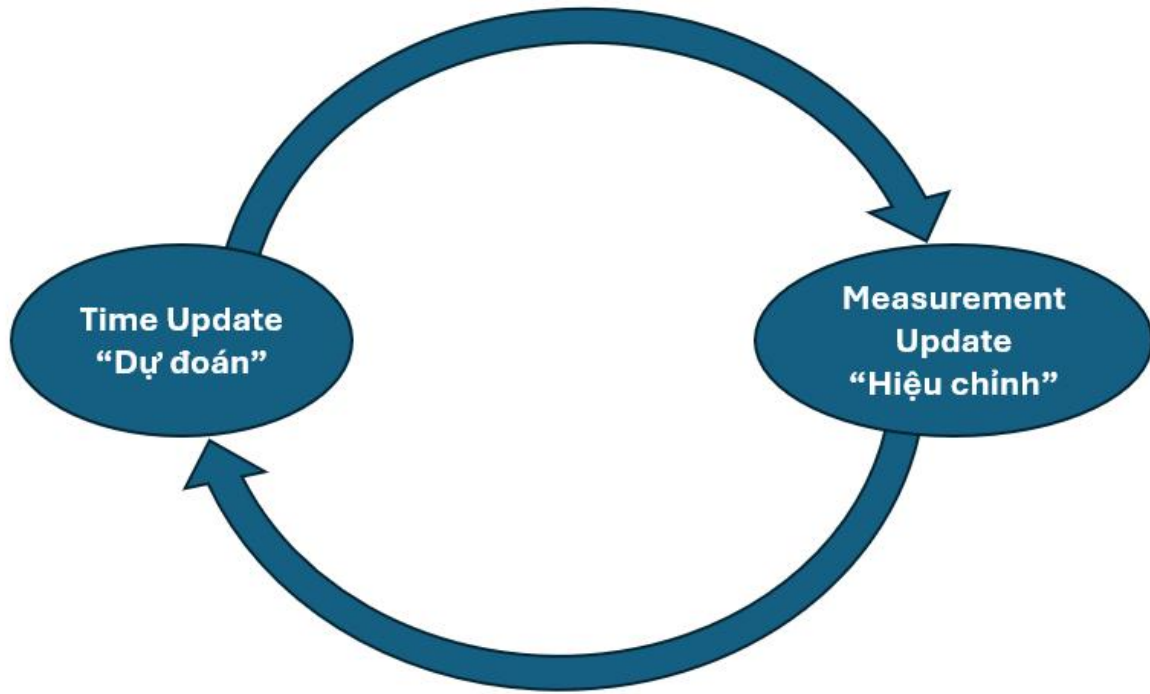
Ma trận K trong phương trình trên là ma trận độ lợi hay hệ số trộn để tối thiểu hóa phương trình tương quan sai số posteriori. Biểu thức tính K như sau:

$$K_k = P_k^- H^T (H P_k^- H^T + R)^{-1} \#(2.38)$$

### 2.2.3. Thuật toán Kalman rời rạc:

Bộ lọc Kalman ước lượng tiến trình bằng cách sử dụng dạng điều khiển hồi tiếp: bộ lọc ước lượng các trạng thái của quá trình tại một vài thời điểm và sau đó chứa tín hiệu hồi tiếp trong các dạng của giá trị đo lường. Do đó, phương trình bộ lọc Kalman chia làm hai nhóm: *phương trình cập nhật thời gian* và *phương trình cập nhật giá trị đo lường*. *Phương trình cập nhật thời gian* chịu trách nhiệm cho việc dự báo trước (về mặt thời gian) của trạng thái hiện tại và ước lượng sai số tương quan để chứa vào bộ ước lượng trước priori cho bước thời gian tiếp theo. *Phương trình cập nhật giá trị đo lường* chịu trách nhiệm cập nhật cho tín hiệu hồi tiếp, nghĩa là cập nhật giá trị mới vào giá trị ước lượng trước priori để tạo tín hiệu ước lượng sau posteriori tốt hơn.

*Phương trình cập nhật thời gian* cũng có thể được coi là phương trình dự đoán. Trong khi đó *phương trình cập nhật giá trị đo lường* thì được xem như là phương trình hiệu chỉnh. Vì vậy, thuật toán ước lượng cuối cùng đều giống nhau ở thuật toán dự đoán và hiệu chỉnh để giải quyết vấn đề số học như hình vẽ dưới đây:



Hình 2. Quy trình thực hiện bộ lọc Kalman

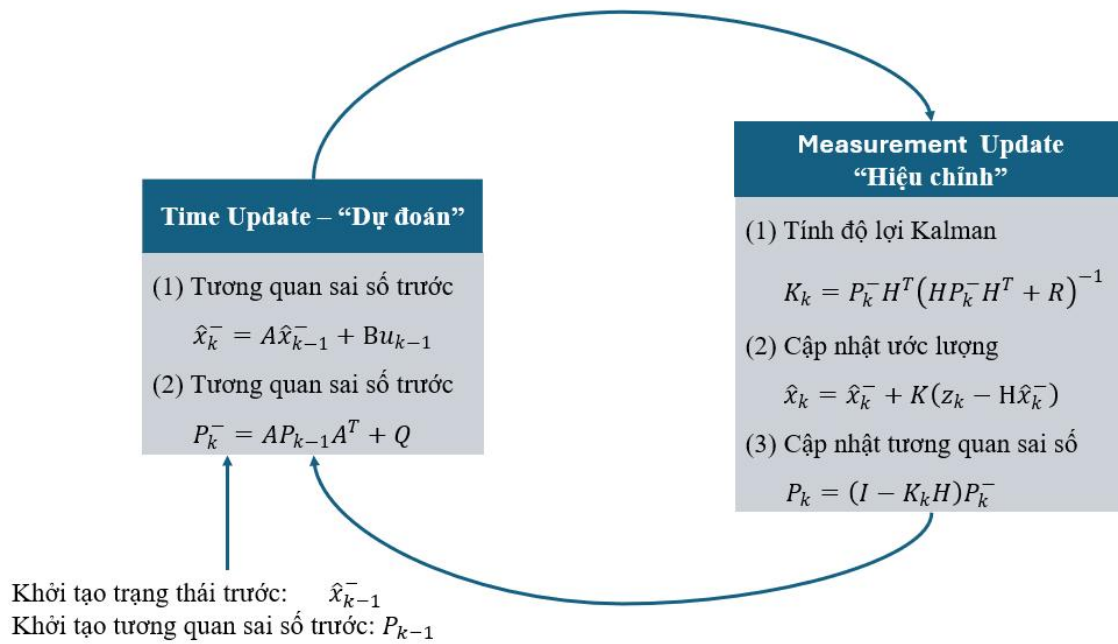
Phương trình cập nhật thời gian cho bộ lọc Kalman rời rạc:

$$\begin{aligned}\hat{x}_k^- &= A\hat{x}_{k-1}^- + Bu_{k-1} \\ P_k^- &= AP_{k-1}A^T + Q\end{aligned}\quad \#(2.39)$$

Phương trình cập nhật giá trị đo lường cho bộ lọc Kalman rời rạc:

$$\begin{aligned}K_k &= P_k^- H^T (HP_k^- H^T + R)^{-1} \\ \hat{x}_k &= \hat{x}_k^- + K(z_k - H\hat{x}_k^-) \\ P_k &= (I - K_k H)P_k^-\end{aligned}\quad \#(2.40)$$

Tổng quan chu trình thực hiện bộ lọc Kalman hoàn chỉnh như hình sau đây:



Hình 3. Tổng quan chu trình thực hiện bộ lọc Kalman hoàn chỉnh

## CHƯƠNG 3. MÔ PHỎNG MATLAB

### 3.1. Tiến hành mô phỏng

```
%% Thông số hệ thống xe 2 bánh tự cân bằng dùng LQR
m = 0.0345; %Khối lượng bánh xe
M = 0.875; %Khối lượng robot
R = 0.0325; %bán kính bánh xe
W = 0.225; %Chiều rộng robot
D = 0.084; %Chiều sâu robot
H = 0.132; %Chiều cao robot
L = 0.091; %khoảng cách từ trọng tâm đến trục bánh xe
fw = 0.18; %Hệ số ma sát giữa bánh xe với mặt phẳng
fm = 0.002; %hệ số ma sát giữa động cơ và robot
Jm = 0.000082; %moment quán tính của động cơ
Jw = m*R^2/2;
J_psi = M*L^2/3;
J_phi = M*(W^2+D^2)/12;
Rm = 13; %Điện trở động cơ DC
Kb = 1.91; %hệ số emf của động cơ
Kt = 0.216; %Momen xoắn của động cơ DC
n = 33.64; %Tỷ số giảm tốc
g = 9.81; %Gia tốc trọng trường
alpha = n*Kt/Rm; beta=n*Kt*Kb/Rm+fm; a =alpha;
T=0.01;
```

Hình 4. Thông số hệ thống xe 2 bánh tự cân bằng.

- Sau khi dùng Matlab để tính toán thì ta được ma trận A, B như sau:

A =

0	1.0000	0	0	0	0
0	-32.7243	48.4785	8.8862	0	0
0	0	0	1.0000	0	0
0	-19.7154	49.4379	-2.6272	0	0
0	0	0	0	0	1.0000
0	0	0	0	0	-13.4327

Hình 5. Ma trận A

```
B =
      0      0
    2.3219    2.3219
      0      0
   -0.6865   -0.6865
      0      0
   -0.8679    0.8679
```

Hình 6. Ma trận B

- Chọn ma trận Q và R:

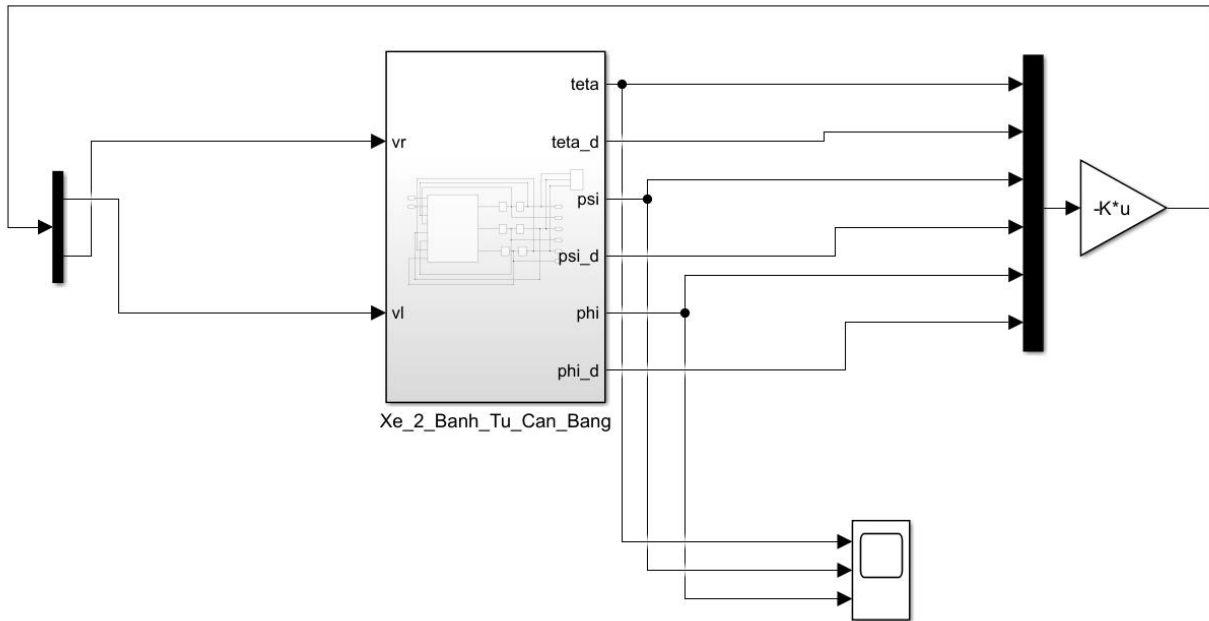
```
%% Tính thông số K của LQR
R__ = [100 0; 0 100]; |
Q = [ 1000 0 0 0 0 0;
      0 10 0 0 0 0;
      0 0 10000 0 0 0;
      0 0 0 1 0 0;
      0 0 0 0 100000 0;
      0 0 0 0 0 10]
K = lqr(A,B,Q,R__)

% Chọn thông số ban đầu
x1_init = 0.001; x2_init = -0.0012; x4_init = 0.002; x5_init = -0.002; x7_init = 0.002; x8_init=-0.0014;
```

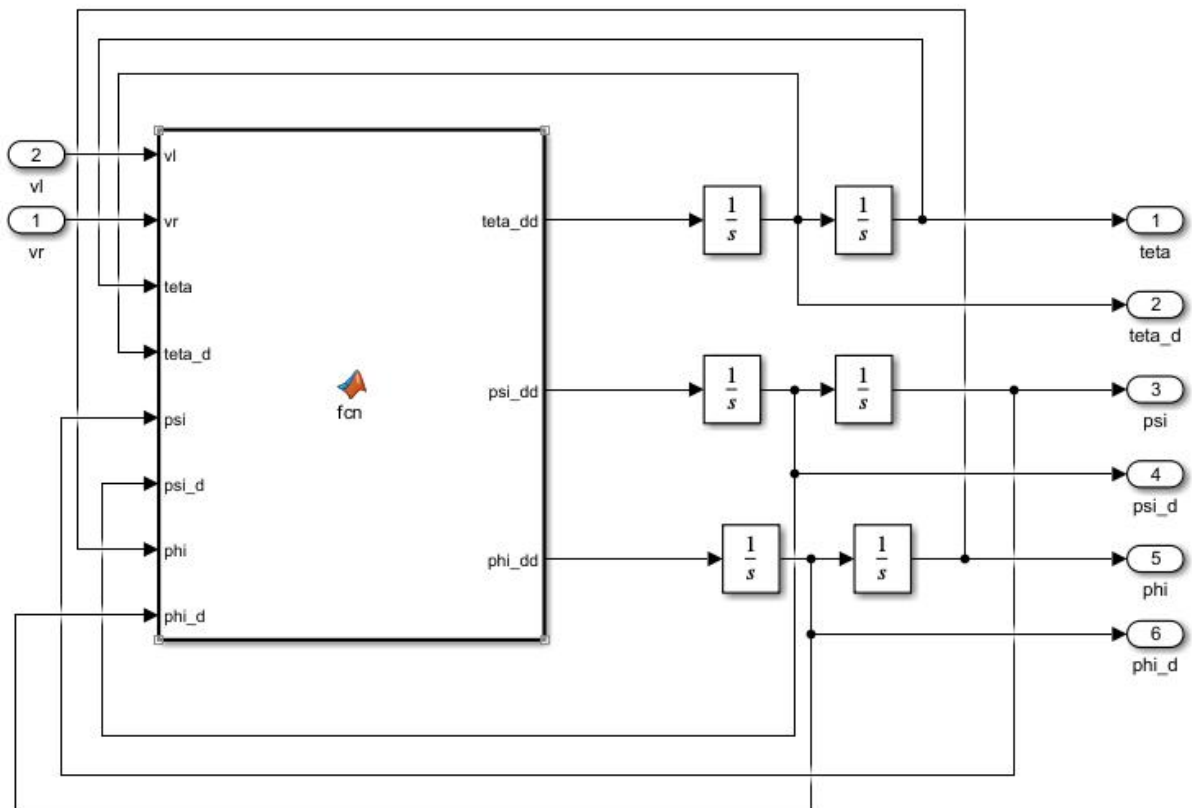
Hình 7. Thông số ma trận Q và ma trận R.

```
K =
   -2.2361    0.8694   -17.0800   -1.6988   -22.3607   -1.5188
   -2.2361    0.8694   -17.0800   -1.6988    22.3607    1.5188
```

Hình 8. Kết quả tính ma trận K của LQR.



Hình 9. Mô phỏng Simulink



Hình 10. Khối Xe\_2\_Banh\_Tu\_Can\_Bang



## CHƯƠNG 3. MÔ PHỎNG MATLAB

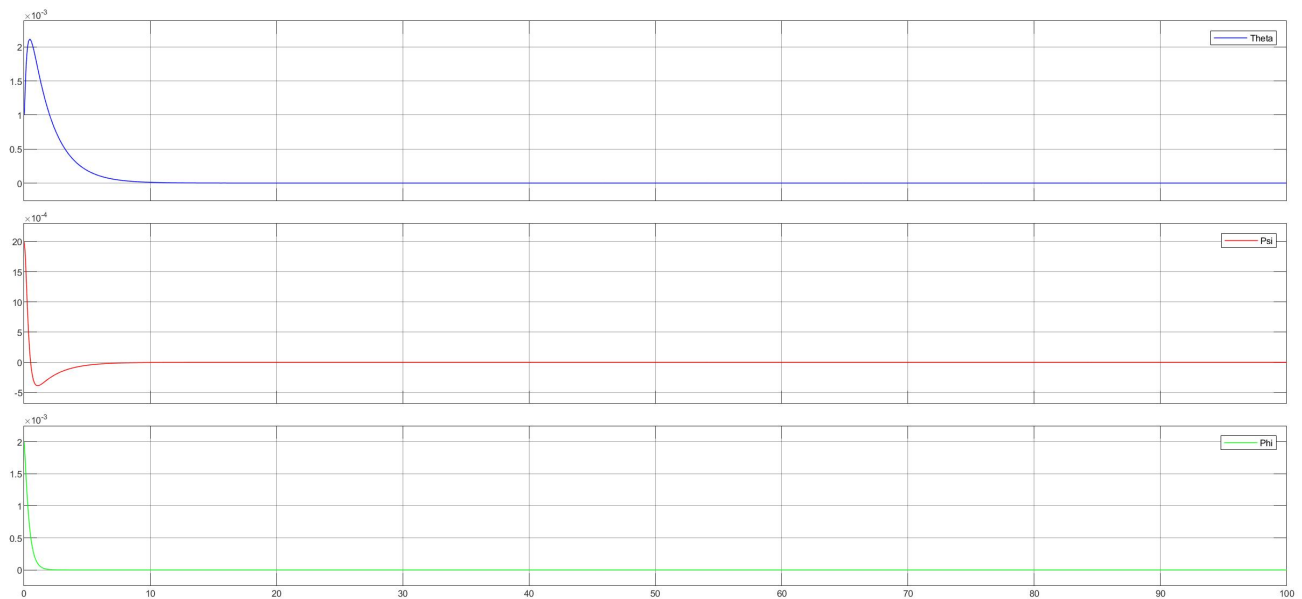
```
function [teta_dd,psi_dd,phi_dd] = fcn(vl,vr,teta,teta_d,psi,psi_d,phi,phi_d)

%% Thông số hệ thống xe 2 bánh tự cân bằng dùng LQR
m = 0.0345; %Khối lượng bánh xe
M = 0.875; %Khối lượng robot
R = 0.0325; %Bán kính bánh xe
W = 0.225; %Chiều ngang robot
D = 0.084; %Chiều rộng robot
H = 0.132; %Chiều cao robot
L = 0.091; %Khoảng cách từ trọng tâm đến trục bánh xe
fw = 0.18; %Hệ số ma sát giữa bánh xe với mặt phẳng
fm = 0.002; %Hệ số ma sát giữa động cơ và robot
Jm = 0.000082; %Moment quán tính của động cơ
Jw = m*R^2/2;
Jpsi = M*L^2/3;
Jphi = M*(W^2+D^2)/12;
Rm = 13; %Điện trở động cơ DC
Kb = 1.91; %Hệ số emf của động cơ
Kt = 0.216; %Moment xoắn của động cơ DC
n = 33.64; %Tỷ số giảm tốc
g = 9.81; %Gia tốc trọng trường
alpha = n*Kt/Rm; beta=n*Kt*Kb/Rm+fm; a =alpha;
T=0.01;

teta_dd = (Jpsi*a*vl + Jpsi*a*vr - 2*Jpsi*beta*teta_d + 2*Jpsi*beta*psi_d - 2*Jpsi*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*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 - phi_dd = -(W^2*beta*phi_d + W^2*fw*phi_d + R*M*a*vl - R*M*a*vr + 4*L^2*M*R^2*psi_d*phi_d*cos(psi)*sin(psi))/(2*Jphi*R^2 + Jm*W^2 + Jm*W^2*n^2 + R^2*W^2*m + 2*L^2*M*R^2*sin(psi)^2);
```

Hình 11. Thông số mô hình trong khối Matlab function

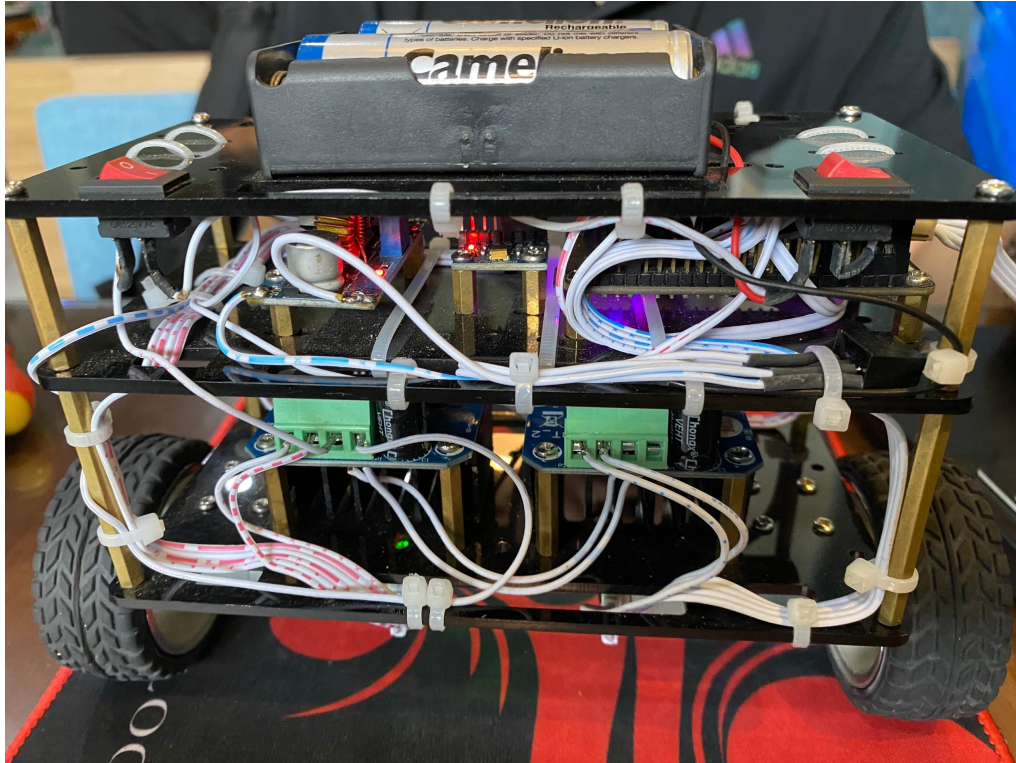
### 3.2. Kết quả mô phỏng



Hình 12. Kết quả mô phỏng.

**Nhận xét:** Tín hiệu đáp ứng của hệ thống tương đối tốt, các biến trạng thái của xe cân bằng tại 0, không có sai số xác lập, thời gian đạt xác lập nhanh, có xuất hiện vọt lố nhưng không đáng kể.

## CHƯƠNG 4. MÔ HÌNH THỰC TẾ



Hình 13. Mô hình Robot trên thực tế

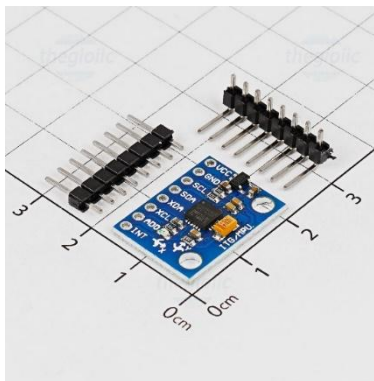
### 4.1. Cấu trúc phần cứng

Nguồn cung cấp sử dụng 3 viên pin 18650 3.7V/2200mA mắc nối tiếp với nhau.

Sử dụng mạch giảm áp 12V  $\rightarrow$  5V để cấp cho MPU6050 và STM32.

Robot sử dụng vi điều khiển STM32F407VET6 làm trung tâm thu thập dữ liệu và điều khiển khi hoạt động của Robot.

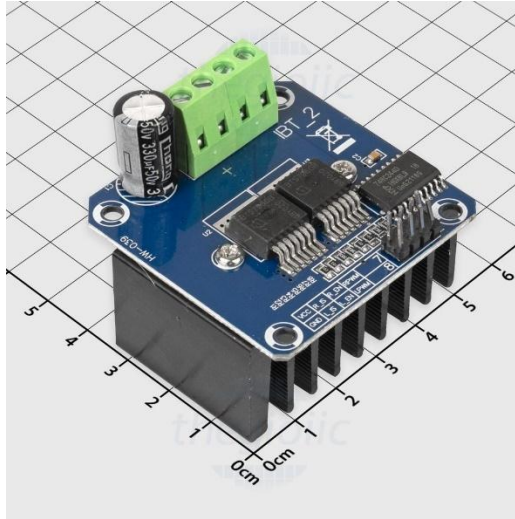
- *Cảm biến con quay gia tốc MPU6050*



Hình 14. MPU 6050

MPU6050 là cảm biến được sử dụng để đo 6 thông số: 3 trục Góc quay (Gyro), 3 trục gia tốc hướng (Accelerometer), là loại cảm biến gia tốc phổ biến nhất trên thị trường hiện nay.

- **Driver *BTS7960***



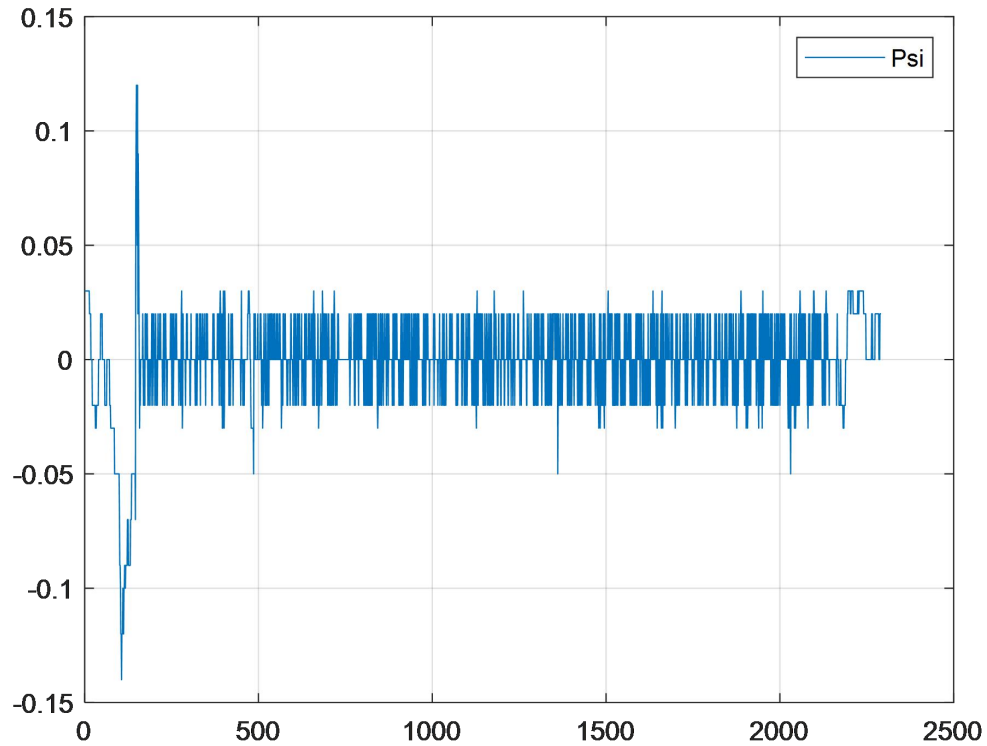
*Hình 15. BTS7960*

BTS7960 là một mô-dun điều khiển động cơ DC, cầu H dòng cao tích hợp đầy đủ cho các ứng dụng truyền động động cơ cần sử dụng dòng cao.

Mạch này chứa một MOSFET bên cao kênh p và một MOSFET bên thấp kênh n, với một vi mạch trình điều khiển tích hợp trong một gói. Việc giao tiếp với vi điều khiển được thực hiện dễ dàng nhờ vi mạch điều khiển tích hợp có các đầu vào mức logic, bảo vệ chống quá nhiệt, quá áp, quá dòng và ngắn mạch...

Mạch chịu được điện áp cung cấp từ 6V đến 27V, đặc biệt có khả năng tự ngắt khi áp dưới 5.5V và mở lại sau khi áp lớn hơn 5.5V. Đồng thời chịu được dòng tải cao lên đến 45A.

#### 4.2. Kết quả thực nghiệm



*Hình 16. Ngõ ra góc nghiêng khi Robot giữ thăng bằng*

**Nhận xét:** Robot hoạt động ổn định, sai số rất nhỏ, chỉ rơi vào khoảng  $\pm 0.025$  rad. Tuy nhiên đôi lúc vẫn bị mất ổn định do một số tác động từ bên ngoài.

## CHƯƠNG 5. KẾT LUẬN

### 5.1. Kết quả đạt được

Qua quá trình nghiên cứu và thực hiện đề tài, nhóm xem xin được tổng hợp những thành quả đã đạt được:

- Thiết kế hoàn chỉnh và hoàn thiện được mô hình xe hai bánh tự cân bằng.
- Tính toán mô hình toán học, hàm trạng thái cho hệ thống.
- Thực hiện thành công mô phỏng hệ thống trên Matlab Simulink.
- Hoàn thiện lập trình cho xe hai bánh tự cân bằng trên vi điều khiển STM32.

### 5.2. Hạn chế

Các mặt hạn chế còn gặp phải trong quá trình nghiên cứu đề tài là:

- Trong quá trình chạy, xe giữ cân bằng khá tốt, tuy nhiên vẫn còn gặp phải tình trạng vọt lố của góc nghiêng.
- Khả năng đáp ứng khi tác động nhiễu còn yếu.
- Khó giữ cân bằng trên những địa hình phức tạp như: gồ ghề, lượn sóng, nghiêng,...
- Xe chưa có khả năng tự di chuyển, tự đứng dậy (swing-up).

### 5.3. Hướng phát triển

Một vài hướng phát triển trong tương lai như sau:

- Xây dựng khả năng tự hành và tự đứng dậy (swing-up).
- Thay đổi bánh tròn thành bánh đa hướng (mecanum) giúp xe di chuyển linh hoạt hơn.
- Xây dựng bộ điều khiển giúp xe đáp ứng cao hơn khi gặp tác động nhiễu và di chuyển được trên các mặt phẳng phức tạp.

**TÀI LIỆU THAM KHẢO**

1. Alex Becker, *Hướng dẫn về bộ lọc Kalman (Kalman Filter)*, Kalman Filter Tutorial, [https://www.kalmanfilter.net/VI/default\\_vi.aspx](https://www.kalmanfilter.net/VI/default_vi.aspx), 3/6/2024.
2. Nguyen Van Dong Hai, *điều khiển LQR xe hai bánh tự cân bằng*, <https://www.youtube.com/watch?v=81GAAs486a8>, 3/6/2024.
3. Thegioiic, *GY-521 MPU-6050 Cảm Biến Con Quay Gia Tốc*, <https://www.thegioiic.com/gy-521-mpu-6050-cam-bien-con-quay-gia-toc>, 3/6/2024.
4. Thegioiic, *BTS7960 Module Điều Khiển Động Cơ DC 43A*, <https://www.thegioiic.com/bts7960-module-dieu-khien-dong-co-dc-43a>, 3/6/2024.