

TRƯỜNG ĐẠI HỌC TRÀ VINH
KHOA KỸ THUẬT VÀ CÔNG NGHỆ



BÁO CÁO TỔNG KẾT
ĐỀ TÀI KHOA HỌC VÀ CÔNG NGHỆ CẤP TRƯỜNG

TÊN ĐỀ TÀI
THIẾT KẾ, THI CÔNG VÀ ĐIỀU KHIỂN MỜ HỆ
CON NÊM NGƯỢC

CHỦ NHIỆM ĐỀ TÀI: Ths. ĐẶNG HỮU PHÚC
ĐƠN VỊ: BỘ MÔN ĐIỆN TỬ - VIỄN THÔNG

Trà Vinh, ngày tháng năm 2012

TRƯỜNG ĐẠI HỌC TRÀ VINH
KHOA KỸ THUẬT VÀ CÔNG NGHỆ



BÁO CÁO TỔNG KẾT
ĐỀ TÀI KHOA HỌC VÀ CÔNG NGHỆ CẤP TRƯỜNG

TÊN ĐỀ TÀI
THIẾT KẾ, THI CÔNG VÀ ĐIỀU KHIỂN MỜ HỆ
CON NÊM NGƯỢC

Xác nhận của cơ quan chủ trì
(ký tên và đóng dấu)

Chủ nhiệm đề tài
(ký tên, họ tên)

ĐẶNG HỮU PHÚC

Trà Vinh, ngày 19 tháng 02 năm 2012



LỜI CẢM ƠN

Trước hết, tôi xin chân thành cảm ơn Ban Giám Hiệu, Phòng Khoa học Công nghệ và Đào tạo sau đại học, Phòng Kế hoạch Tài vụ, Khoa Kỹ thuật và Công nghệ của Trường Đại học Trà Vinh đã tạo điều kiện tốt nhất cho tôi thực hiện đề tài nghiên cứu khoa học này.

Tôi xin gửi đến thầy TS. Hoàng Minh Trí, giảng viên Trường Đại học Bách Khoa TP. Hồ Chí Minh lời cảm ơn sâu sắc. Thầy đã giúp đỡ và hướng dẫn tôi hoàn thành đề tài nghiên cứu khoa học này.

Cuối cùng cho tôi xin được gửi lời cảm ơn tới những người thân, gia đình, bạn bè và đồng nghiệp đã động viên, khuyến khích, giúp đỡ tôi trong suốt quá trình học tập và nghiên cứu.

Tôi xin chân thành cảm ơn!

Tác giả

Đặng Hữu Phúc



DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT

STT	Ký hiệu	Diễn giải
1	θ	Góc quay con nệm
2	$\dot{\theta}$	Vận tốc góc con nệm
3	x	Là khoảng cách từ vật nặng (“con chạy”) tới vị trí cân bằng
4	\dot{x}	Vận tốc di chuyển của vật nặng (“con chạy”)
5	d	Khoảng cách giữa điểm gốc quay (“trục quay”) và mặt trượt
6	c	Khoảng cách giữa điểm gốc quay và trọng tâm của khung con nệm
7	m	Khối lượng vật nặng (“con chạy”)
8	M	Khối lượng khung con nệm
9	g	Gia tốc trọng trường
10	J_B	Mô-ment quán tính của khung con nệm ($J_B \approx Mc^2$)
11	b_1	Hệ số ma sát của con nệm với tâm quay
12	b_2	Hệ số ma sát của vật nặng với mặt trượt
13	F	Mô-ment (điều khiển) của động cơ DC tác động vào con chạy
14	K_m	Hệ số cảm ứng của động cơ DC
15	U	Điện áp điều khiển đưa vào động cơ DC
16	K	Tổng động năng
17	P	Tổng thế năng
18	T	mô men
19	q	biến trạng thái
20	L	lực suy rộng công thức Euler – Lagrange
21	M_o	ma trận điều khiển
22	M_c	ma trận quan sát
23	T_s	là thời gian xác lập của hệ thống
24	P_o (%)	độ vọt lố
25	k, α, η	là các hằng số dương tự chọn
26	S	mặt trượt
27	c_1, c_2	các hằng số dương
28	s_1, s_2	mặt trượt con
29	u	luật điều khiển
30	u_1	Luật điều khiển cho lớp thứ nhất
31	u_{sm1}, u_{sm2}	Luật chuyển điều khiển của lớp trượt thứ nhất và thứ hai
32	u_{eq1}, u_{eq2}	luật điều khiển cân bằng lớp thứ nhất và thứ 2
33	ZE	Zero

34	NB	Âm lớn
35	NM	Âm vừa
36	NS	Âm nhỏ
37	PS	Dương nhỏ
38	PM	Dương vừa
39	PB	Dương lớn
40	NE	Âm
41	PO	dương

DANH MỤC CÁC BẢNG

STT	Ký hiệu	Diễn giải	Trang
1	Bảng 1.1	Quy trình thực hiện đề tài	9
1	Bảng 2.1	Các thông số của hệ con nôm ngược	18
2	Bảng 2.2	Các luật sử dụng trong điều khiển mờ	33
3	Bảng 3.1	So sánh các bộ điều khiển	39
3	Bảng 3.2	Thông số thiết kế mô hình con nôm ngược thực nghiệm	40
4	Bảng 4.1	Báo cáo kinh phí đã sử dụng trong đề tài	50

DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ

STT	Ký hiệu	Diễn giải	Trang
1	Hình 1.1	Mô hình con nêm ngược sử dụng động cơ DC với dây xích và bánh răng để kéo vật nặng	3
2	Hình 1.2	Mô hình con nêm ngược sử dụng động cơ DC kết hợp với vật nặng tạo thành xe trượt trên mặt phẳng ngang của con nêm ngược	4
3	Hình 1.3	Mô hình con nêm ngược sử dụng động cơ DC với dây đai răng kéo vật nặng	4
4	Hình 1.4 (a)	Mô hình con nêm ngược sử dụng động cơ DC kết hợp với vật nặng tạo thành xe trượt trên mặt phẳng ngang của con nêm ngược	5
5	Hình 1.4 (b)	Mô hình con nêm ngược sử dụng động cơ DC với dây đai răng kéo vật nặng	5
6	Hình 1.5	Mô hình con nêm ngược sử dụng động cơ DC và dây đai kéo 2 vật nặng ở hai mặt nghiêng của con nêm	6
7	Hình 2.1	Nguyên lý hoạt động của con nêm ngược tự cân bằng	10
8	Hình 2.2	Mô hình hệ con nêm ngược	11
9	Hình 2.3	Bộ điều khiển đặt cực cho mô hình con nêm ngược tự cân bằng	20
10	Hình 2.4	Sự ảnh hưởng của cực lên hệ thống	20
11	Hình 2.5	Sơ đồ mô phỏng bộ điều khiển trượt	24
12	Hình 2.6	sơ đồ bên trong khối “Sliding Mode ODH	24

		Controller”	
13	Hình 2.7	Hiện tượng chattering của bộ điều khiển trượt	25
14	Hình 2.8	Sơ đồ mô phỏng bộ điều khiển Trượt-Mờ-PID	27
15	Hình 2.9	Sơ đồ bên trong khối “Sliding Mode-Fuzzy-PID Controller”	28
16	Hình 2.10	Khối mô tả bộ điều khiển mờ sẽ thiết kế	29
17	Hình 2.11	Sơ đồ mô phỏng điều khiển mờ hệ nê m ngược	36
18	Hình 3.1	So sánh 3 bộ điều khiển với $\theta \leq 18^\circ$	37
19	Hình 3.2	So sánh 3 bộ điều khiển với $\theta > 18^\circ$	38
20	Hình 3.3	Sơ đồ thiết kế mô hình hệ con nê m ngược	40
21	Hình 3.4 (a)- (b)- (c) (d) -(e) -(f) (g)- (h)	Các Mô hình con nê m ngược thực nghiệm đã thi công	41-42- 43-44- 45
22	Hình 3.5	Sơ đồ khối bộ điều khiển	45
23	Hình 3.6	Board mạch cảm biến góc nghiêng	46
24	Hình 3.7	Board mạch điều khiển	46
25	Hình 3.8	kết quả thực hiện bộ điều khiển Trượt-Mờ-PID	48

PH N I

T NG QUAN TÀI

1.1. S c n thi t c a tài

Hi n nay, t n c ta ang trong th i k công nghi p hoá, hi n i hoá. Cùng v i s phát tri n c a khoa h c k thu t, i u khi n t ng c ng phát tri n m nh m v i hàng lo t thi t b i n t t ng ã và ang c t o ra nh m ph c v cho cu c s ng và s n xu t c a con ng i.

T o ra các ng d ng trong th c t t v i c nghi n c u các mô hình thu nh luôn là ph ng pháp mà các nhà khoa h c quan tâm hàng u, v i th khi thi t k các b i u khi n, ta th ng s d ng các mô hình t ng t nh mô hình th c t , các mô hình thu nh này s c xây d ng trên c s mô ph ng ph ng th c ho t ng gi ng nh mô hình s c dùng trong th c t ch ng h n nh i u ki n làm vi c, ch làm vi c c a i t ng...

H con nê m ng c là m t mô hình thu nh và c ng d ng trong h th ng t ng cân b ng c a tàu bi n, robot... ng th i c ng là v n ang c quan tâm nghi n c u trên th gi i. Tuy nhiên m c thành công còn nhi u h n ch khâu thi công th c nghi m v i h th ng có tính phi tuy n ph c t p, bên c nh ó khâu thi t k òi h i ph i có chính xác tuy t i và áp ng i u khi n ph i nhanh.

C n c vào ch ng trình ào t o ngành i n T T ng Hóa c a tr ng i H c Trà Vinh và tình hình trang thi t b ph c v cho vi c h c t p c a sinh viên, c ng nh nghi n c u c a giáo viên v chuyên ngành t ng hóa còn r t thi u th n, do ó tài nghi n c u này s góp ph n giúp cho sinh viên và giáo viên có th ti p c n v i công ngh i u khi n hi n i, góp ph n nâng cao v th v ào t o ngành T ng Hóa c a tr ng i h c Trà Vinh so v i các tr ng i h c khác trong n c. Do ó khi tài nghi n c u này c th c hi n thành công s gi i quy t c các v n sau:

- Ứng dụng các lý thuyết về robot khi nhìn vào robot trong thực tế.
- Cung cấp trang thiết bị phục vụ cho việc nghiên cứu ứng dụng của giáo viên và học tập của sinh viên trường Đại học Trà Vinh mà không phải mua hay thuê ngoài. Các thiết bị là:
 - + Phục vụ nghiên cứu chuyên ngành về robot.
 - + Phục vụ công tác giảng dạy các môn học: lý thuyết về robot, vi robot, kỹ thuật robot, lập trình vi robot...

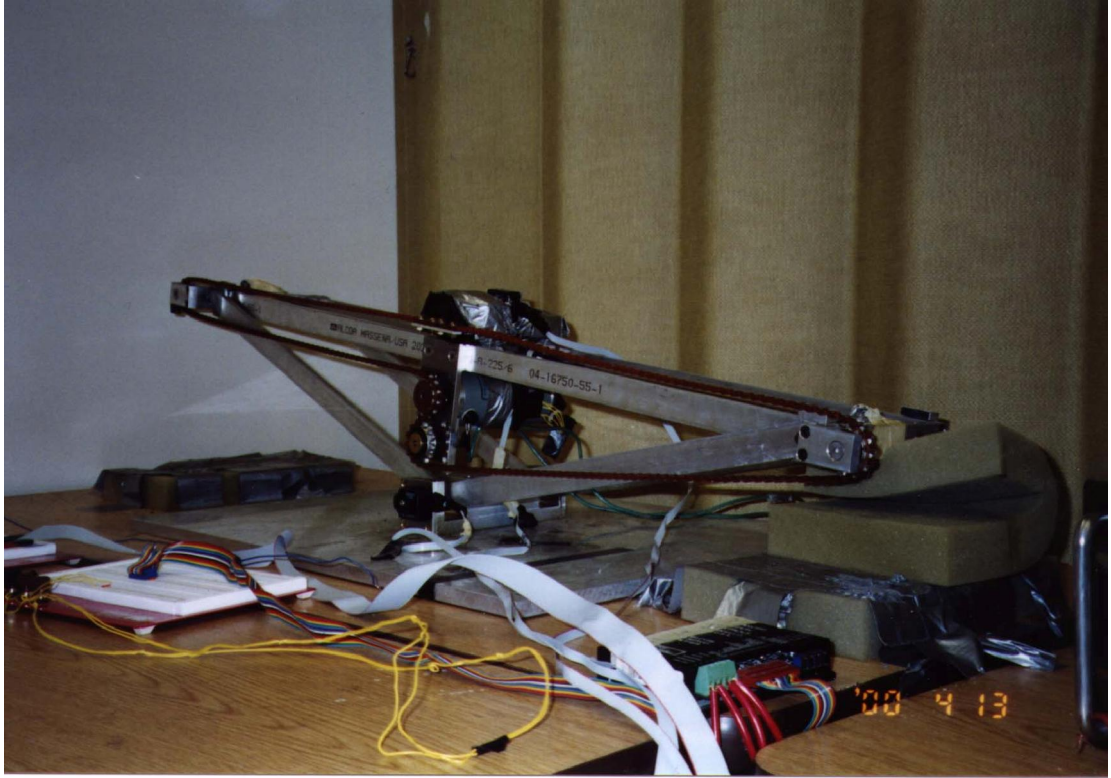
Vấn đề kinh tế: có thể phát triển sản phẩm nghiên cứu này để tạo ra hình thức cân bằng tàu bi n, xe 2 bánh cân bằng, hình thức cân bằng trên tay máy, robot...

1.2. Các công trình nghiên cứu liên quan

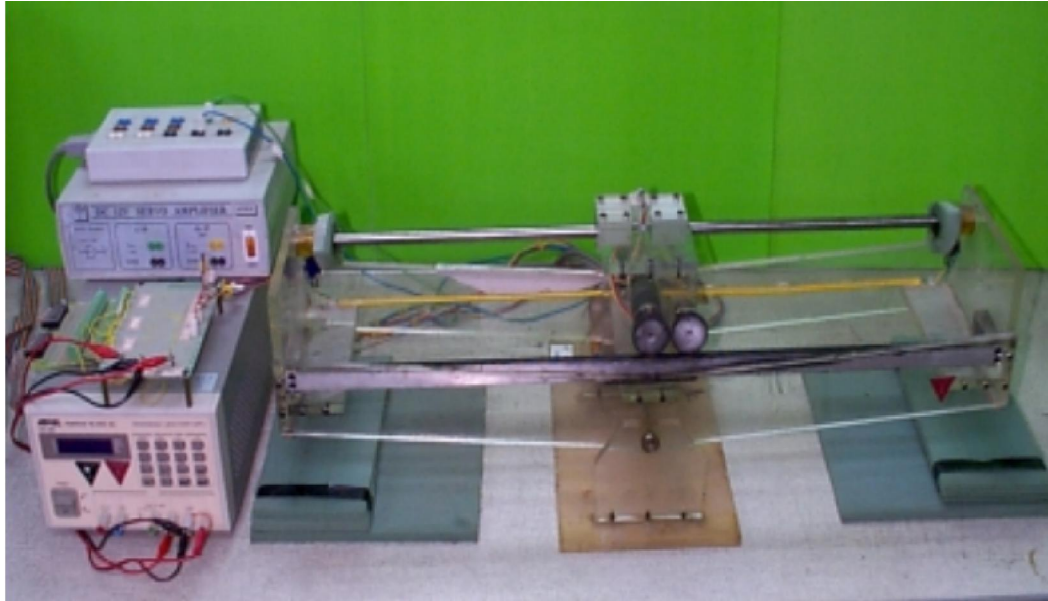
1.2.1. Một số mô hình cân bằng trên tay máy

Hệ thống cân bằng trên tay máy có 3 dạng chính như sau:

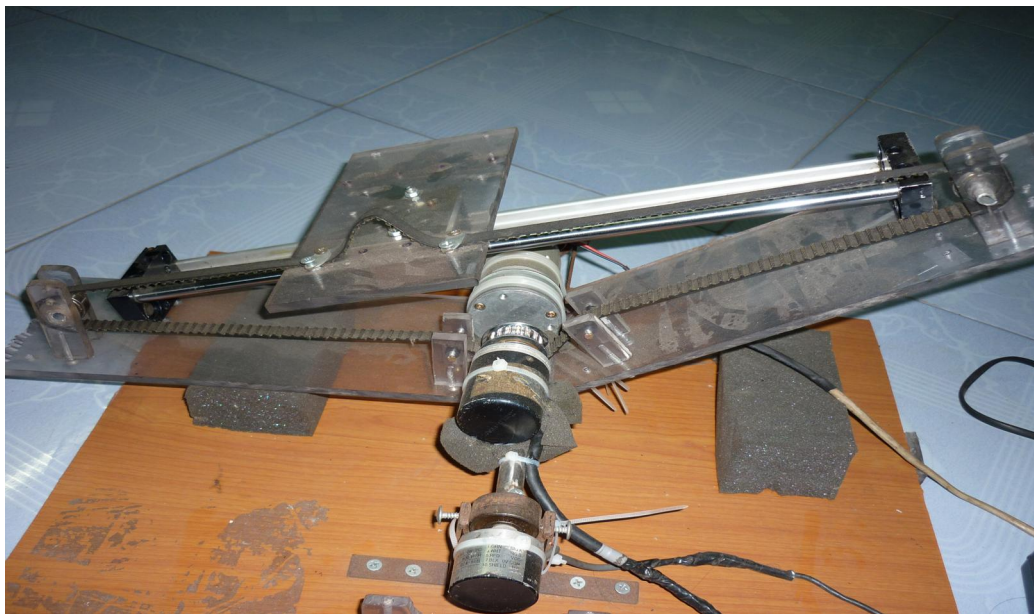
- Hệ thống cân bằng trên tay máy dùng motor DC-Servo kéo một vít truyền thông qua sợi dây và ròng rọc, giúp vít truyền di chuyển trên mặt phẳng ngang của con lắc nh m cân bằng con lắc.
- Hệ thống cân bằng trên tay máy dùng motor DC- Servo kéo hai vít truyền thông qua sợi dây và ròng rọc, giúp vít truyền di chuyển trên hai mặt phẳng nghiêng của con lắc nh m cân bằng con lắc.
- Hệ thống cân bằng trên tay máy dùng motor DC- Servo kết hợp vít truyền tạo thành chiếc xe di chuyển trên mặt phẳng ngang của con lắc nh m giúp con lắc cân bằng.



Hình 1.1: Mô hình con nôm ng c s d ng ng c DC v i dây xích và bánh r ng kéo v t n ng (tài li u tham kh o [16])



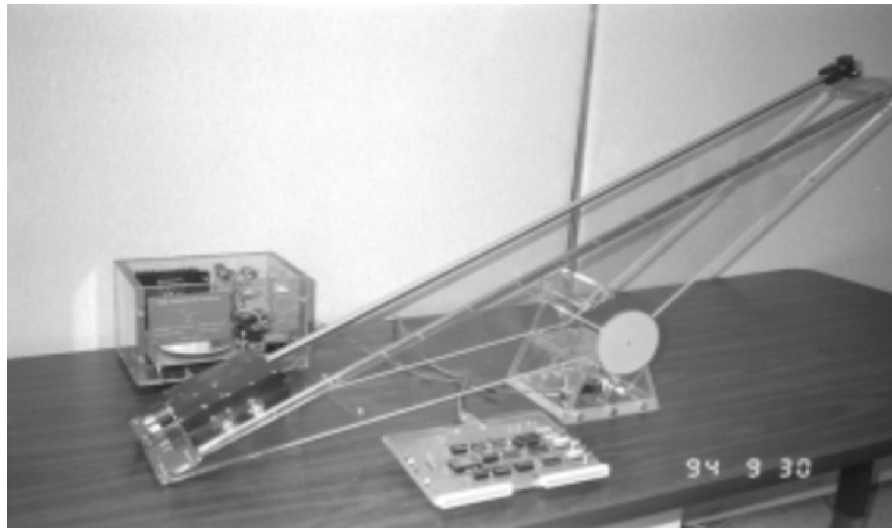
Hình 1.2: Mô hình con nêm ngang sử dụng động cơ DC kết hợp với truyền động thành xe trượt trên mặt phẳng ngang của con nêm ngang (tài liệu tham khảo [10])



Hình 1.3: Mô hình con nêm ngang sử dụng động cơ DC với dây đai truyền động (tài liệu tham khảo [15])

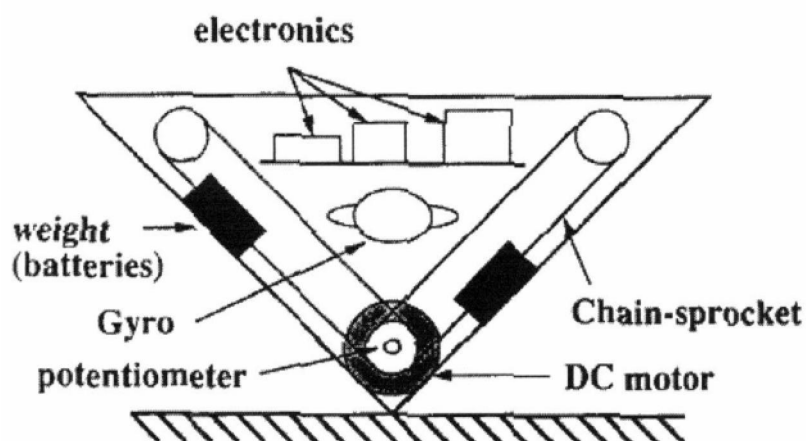


Hình 1.4 (a)



Hình 1.4 (b)

Hình 1.4: (a) Mô hình con nôm ng c s d ng ng c DC k th p v i v t n ng t o thành xe tr t trên m t ph ng ngang c a con nôm ng c (b) : Mô hình con nôm ng c s d ng ng c DC v i dây ai r ng kéo v t n ng (tài li u tham kh o [12],[21], [22])



Hình 1.5: Mô hình con nôm ng c s d ng ng c DC và dây ai kéo 2 v t n ng hai m t nghiêng c a con nôm (tài li u tham kh o [17])

✓ **Nh n xét:**

Mô hình con nôm ng c t cân b ng th c nghi m c thi t k r t a d ng. Trong s các mô hình con nôm ng c c tham kh o trên thì mô hình con nôm ng c hình 1.4 (a) và hình 1.4 (b) ã c ng t i video trên trang web “<http://www.youtube.com/watch?v=hgrJaTUCXPs>” và “<http://www.youtube.com/watch?v=0xU2HOzjhh4&feature=related>”.

1.2.2. M t s báo cáo nghiên c u khoa h c có liên quan

H con nôm ng c t cân b ng là v n ang c quan tâm và nghiên c u trên th gi i, nên có nhi u bài báo, báo cáo khoa h c vi t v v n này. Các bài báo và báo cáo khoa h c liên quan n tài này th ng t p trung vào nh ng n i dung sau:

- Mô hình hoá h th ng ng h c c a h nôm ng c. Các tài li u tham kh o [15], [16], [17], [10], [17], [14].
- S d ng ph ng pháp i u khi n LQR - Tài li u tham kh o [16].
- S d ng ph ng pháp i u khi n m - Các tài li u tham kh o [15], [16].
- S d ng ph ng pháp i u khi n m -tr t - Tài li u tham kh o [17].
- S d ng ph ng pháp i u khi n m -thích nghi - Tài li u tham kh o [16].

- S d ng ph ng pháp i u khi n m -thích nghi-n ron. - Tài li u tham kh o [15].
- S d ng ph ng pháp i u khi n n ron-tr t - Tài li u tham kh o [10].
- S d ng ph ng pháp GA - Tài li u tham kh o [16].

✓ **Nh n xét:**

Các báo cáo nghiên c u khoa h c v h con nê m ng c nêu trên h u h t là c a các tác gi ngoài n c, và ã s d ng nhi u ph ng pháp khác nhau i u khi n gi cân b ng con nê m ng c.

- ***V u i m:***
 - Mô hình hoá c h th ng con nê m ng c.
 - S d ng ph ng pháp i u khi n hi n i.
 - Cho k t qu i u khi n mô ph ng t t.
- ***V nh c i m:***
 - áp ng i u khi n còn có dao ng l n.
 - Th i gian xác l p ch m (h n 5 giây).
 - Ch a có s so sánh gi a các ph ng pháp i u khi n khác nhau (gi a ph ng pháp i u khi n hi n i và ph ng pháp i u khi n kinh i n).
- ***So sánh v i tài ang th c hi n:***
 - tài “ Thi t k , thi công và i u khi n m h con nê m ng c” c th c hi n trên c s k th a mô hình hoá h th ng c a các tài nghiên c u tr c ó, s d ng nhi u ph ng pháp i u khi n khác nhau và so sánh k t qu gi a các ph ng pháp, th i gian xác l p nhanh h n (t 1.5s – 2.5s). ng th i, trong khuôn kh tài nghiên c u này, tác gi còn xu t m t ph ng pháp i u khi n m i là “ i u khi n k t h p Tr t – M - PID”.

1.3. Nội dung nghiên cứu

- Tìm hiểu lý thuyết điều khiển.
- Áp dụng lý thuyết điều khiển vào việc mô phỏng điều khiển hệ con nê-m-ơng cân bằng trên phần mềm Matlab.
- Áp dụng chế tạo hệ con nê-m-ơng cân bằng mô hình thực.

1.4. Mục tiêu của tài

Các mục tiêu chính của tài liệu là:

- Mô hình hoá hệ thống con nê-m-ơng cân bằng.
- Tìm hiểu và nghiên cứu 3 phương pháp điều khiển là: điều khiển cực (Pole-Placement control), điều khiển trượt (Sliding Mode control) và điều khiển mờ (fuzzy control).
- Sử dụng các phương pháp điều khiển nêu trên và công cụ mô phỏng Matlab/Simulink, tiến hành thiết kế bộ điều khiển góc nghiêng của con nê-m-ơng cân bằng điều khiển con nê-m-ơng cân bằng.
- Chế tạo mô hình hệ con nê-m-ơng cân bằng trên cơ sở ứng dụng điều khiển mờ điều khiển hệ thống cân bằng.

1.5. Quy trình thực hiện

Quy trình thực hiện tài c thực hiện theo thứ tự các công việc sau:

Bảng 1.1: Quy trình thực hiện tài

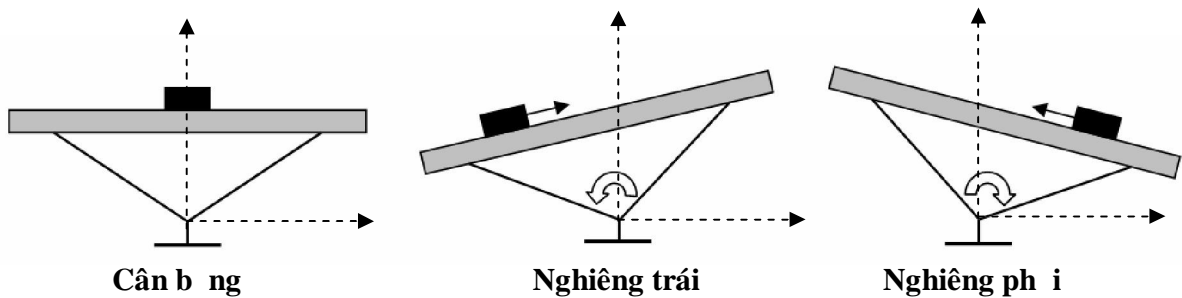
STT	Nội dung công việc	Sản phẩm chính	Thời gian
1	Tìm kiếm tài liệu (phần việc tính toán lý thuyết và thi công mô hình)	Thông tin và tính toán thiết kế và thi công mô hình	02/2011 đến 03/2011
2	Khảo sát mô hình toán học con nôm ngữ	Mô hình toán học	03/2011 đến 04/2011
3	Thiết kế mô phỏng trên Matlab (phiên bản sinh viên)	Bản thiết kế mô phỏng	04/2011 đến 06/2011
4	Mua linh kiện, vật tư, thiết kế và thi công mô hình con nôm ngữ	Mô hình cơ khí	06/2011 đến 10/2011
5	Viết chương trình điều khiển	Board mạch và chương trình	10/2011 đến 01/2012
6	Chẩn đoán và viết báo cáo	Báo cáo	01/2012 đến 02/2012

PHẦN II

PHƯƠNG PHÁP VÀ KỸ THUẬT THỰC HIỆN

2.1. Nguyên lý hoạt động của con nêm nâng cột cân bằng

Con nêm nâng cột hoạt động dựa trên cơ sở cân bằng trọng tâm của con nêm, bộ cảm biến DC-Servo sẽ giúp vận động di chuyển qua lại trên mặt phẳng ngang của con nêm dựa trên trọng tâm và giúp con nêm cân bằng.



Hình 2.1: Nguyên lý hoạt động của con nêm nâng cột cân bằng

Khi con nêm nâng cột ở vị trí cân bằng thì góc nghiêng θ của con nêm (so với trục ngang trọng tâm của con nêm) bằng 0.

Khi con nêm nâng cột ở vị trí nghiêng trái thì góc nghiêng $\theta < 0$, lúc này ta phải điều khiển vận động di chuyển qua lại của cân bằng trọng tâm của con nêm nâng cột. Tương tự cho trường hợp con nêm nâng cột nghiêng phải ($\theta > 0$).

K_m : H s c m ng c a ng c DC.

U: i n áp i u khi n a vào ng c DC.

Ch n chi u d ng là chi u ng c v i chi u kim ng h .

H con nê n ng c c miêu t nh Hình 2.2. H th ng g m m t v t n ng tr t d c theo thanh tr t ngang n m trên cùng c a con nê m, v trí v t n ng c i u khi n b i m t motor k t h p v i v t n ng t o thành m t chi c xe có th di chuy n qua l i. M c tiêu i u khi n là làm cho con nê m gi cân b ng theo tr c ng. Ta có th i u khi n gi con nê m v cân b ng theo tr c ng thông qua i u khi n v trí xe tr t di chuy n trên b m t tr t làm tr ng tâm h th ng thay i v trí.

ng d ng c a h con nê m ng c dùng gi cân b ng cho h th ng tàu th y.

2.2.3. Các ph ng trình bi n tr ng thái c a h th ng

Áp d ng ph ng pháp Euler – Lagrange ta có:

$$L = K - P$$

$$\frac{d}{dt} \left(\frac{\partial K}{\partial \dot{q}_i} \right) - \frac{\partial K}{\partial q_i} + \frac{\partial P}{\partial q_i} = T_i \quad (2.1)$$

Trong ó:

K: t ng ng n ng

P: t ng th n ng

T: mô men

q: bi n tr ng thái

i v i h th ng con nê m ng c ta có $q_1 = \theta$; $q_2 = x$

✓ Tính toán các n ng l ng:

- ng n ng K: $K = K_1 + K_2$ (2.2)

$$K_1 = \frac{1}{2} J_B (\dot{\theta})^2 = \frac{1}{2} M c^2 (\dot{\theta})^2 \quad (2.3)$$

Xét h con nê m nh Hình 2.2 ta có:

$$\begin{aligned}
 AB &= OB \tan(\theta) = x \tan(\theta) \Rightarrow AC = AB + BC = x \tan(\theta) + d \\
 \Rightarrow y_m &= CH = AC \cos(\theta) = (x \tan(\theta) + d) \cos(\theta) = x \sin(\theta) + d \cos(\theta) \\
 x_m &= OH = OA - AH = (OB / \cos(\theta)) - AC \sin(\theta) \\
 &= (x / \cos(\theta)) - (x \tan(\theta) + d) \sin(\theta) \\
 \Rightarrow \begin{cases} y_m = x \sin(\theta) + d \cos(\theta) \approx x\theta + d \\ x_m = (x / \cos(\theta)) - (x \tan(\theta) + d) \sin(\theta) \approx x - x\theta^2 - d\theta \approx x - d\theta \end{cases} \\
 (\forall \theta \text{ i } \sin(\theta) \approx \theta \text{ [rad]; } \cos(\theta) \approx 1)
 \end{aligned}$$

$$\begin{aligned}
 \Rightarrow \begin{cases} \dot{x}_m \approx \frac{d}{dt}(x\theta + d) = (\dot{x}\theta + x\dot{\theta}) \\ \dot{x}_m \approx \frac{d}{dt}(x - d\theta) = (\dot{x} - d\dot{\theta}) \end{cases} \\
 \Rightarrow K_2 = \frac{1}{2} m \left((\dot{x}_m)^2 + (\dot{x}_m)^2 \right) = \frac{1}{2} m \left((\dot{x}\theta + x\dot{\theta})^2 + (\dot{x} - d\dot{\theta})^2 \right) \\
 K_2 = \frac{1}{2} m \left((\dot{x}\theta)^2 + 2x\dot{\theta}\dot{x} + (x\dot{\theta})^2 + (\dot{x})^2 - 2d\dot{\theta}\dot{x} + (d\dot{\theta})^2 \right) \\
 \Leftrightarrow K_2 \approx \frac{1}{2} m \left((\dot{x})^2 - 2d\dot{\theta}\dot{x} + (d\dot{\theta})^2 + (x\dot{\theta})^2 \right) \quad (2.4)
 \end{aligned}$$

$$\forall \theta \text{ i } (2x\dot{\theta}\dot{x} \approx 0 ; (x\dot{\theta})^2 = (\dot{x})^2 (\theta)^2 \approx 0)$$

• Th n ng P:

$$\begin{aligned}
 P &= P_1 + P_2 \quad ; P_1 = (Mgc) \cos(\theta); P_2 = (mg)(x \sin(\theta) + d \cos(\theta)) \\
 \Leftrightarrow P &= ((Mgc) \cos(\theta)) + (mg)(x \sin(\theta) + d \cos(\theta)) \\
 &= (Mgc) \cos(\theta) + (mg)x \sin(\theta) + (mgd) \cos(\theta) \quad (2.5)
 \end{aligned}$$

K t h p (2.1), (2.2) và (2.5) ta có:

$$\begin{aligned}
 \text{"Euler - Lagrange":} \quad \begin{cases} \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = T_1 \\ \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{x}} \right) - \frac{\partial L}{\partial x} = T_2 \end{cases} \quad (2.6)
 \end{aligned}$$

$$\Leftrightarrow \begin{cases} \left(\frac{d}{dt} \left(\frac{\partial K}{\partial \dot{\theta}} \right) - \frac{\partial K}{\partial \theta} \right) - \left(\frac{d}{dt} \left(\frac{\partial P}{\partial \dot{\theta}} \right) - \frac{\partial P}{\partial \theta} \right) = Fd - b_1 \dot{\theta} \\ \left(\frac{d}{dt} \left(\frac{\partial K}{\partial \dot{x}} \right) - \frac{\partial K}{\partial x} \right) - \left(\frac{d}{dt} \left(\frac{\partial P}{\partial \dot{x}} \right) - \frac{\partial P}{\partial x} \right) = F - b_2 \dot{x} \end{cases} \quad (2.7)$$

Trong đó ($L = K - P$; $F = K_m U$: moment c a xe)

T (2.3), (2.4) và (2.5) ta có:

$$\begin{cases} K = K_1 + K_2 = \frac{1}{2} Mc^2 (\dot{\theta})^2 + \frac{1}{2} m ((\dot{x})^2 - 2d\dot{\theta}\dot{x} + d^2 (\dot{\theta})^2 + x^2 (\dot{\theta})^2) \\ P = P_1 + P_2 = (Mgc) \cos(\theta) + (mg)x \sin(\theta) + (mgd) \cos(\theta) \end{cases}$$

$$\Rightarrow \left(\frac{\partial P}{\partial \dot{\theta}} \right) = \left(\frac{\partial P}{\partial \dot{x}} \right) = 0$$

T (2.7) suy ra:

$$\left\{ \begin{array}{l} \frac{d}{dt} \left(\frac{\partial \left(\frac{1}{2} Mc^2 (\dot{\theta})^2 + \frac{1}{2} m ((\dot{x})^2 - 2d\dot{\theta}\dot{x} + d^2 (\dot{\theta})^2 + x^2 (\dot{\theta})^2) \right)}{\partial \dot{\theta}} \right) - \frac{\partial \left(\frac{1}{2} Mc^2 (\dot{\theta})^2 + \frac{1}{2} m ((\dot{x})^2 - 2d\dot{\theta}\dot{x} + d^2 (\dot{\theta})^2 + x^2 (\dot{\theta})^2) \right)}{\partial \theta} \\ + \frac{\partial ((Mgc) \cos(\theta) + (mg)x \sin(\theta) + (mgd) \cos(\theta))}{\partial \theta} = Fd - b_1 \dot{\theta} \\ \\ \frac{d}{dt} \left(\frac{\partial \left(\frac{1}{2} Mc^2 (\dot{\theta})^2 + \frac{1}{2} m ((\dot{x})^2 - 2d\dot{\theta}\dot{x} + d^2 (\dot{\theta})^2 + x^2 (\dot{\theta})^2) \right)}{\partial \dot{x}} \right) - \frac{\partial \left(\frac{1}{2} Mc^2 (\dot{\theta})^2 + \frac{1}{2} m ((\dot{x})^2 - 2d\dot{\theta}\dot{x} + d^2 (\dot{\theta})^2 + x^2 (\dot{\theta})^2) \right)}{\partial x} \\ + \frac{\partial ((Mgc) \cos(\theta) + (mg)x \sin(\theta) + (mgd) \cos(\theta))}{\partial x} = F - b_2 \dot{x} \end{array} \right.$$

$$\Leftrightarrow \begin{cases} \frac{d}{dt} (Mc^2 (\dot{\theta}) - m d \dot{x} + m d^2 (\dot{\theta}) + m x^2 (\dot{\theta})) - 0 - (Mgc) \sin(\theta) + \\ + (mg)x \cos(\theta) - (mgd) \sin(\theta) = Fd - b_1 \dot{\theta} \\ \frac{d}{dt} (m(\dot{x} - d\dot{\theta})) - m x (\dot{\theta})^2 + mg \sin(\theta) = F - b_2 \dot{x} \end{cases}$$

$$\Leftrightarrow \begin{cases} (Mc^2\ddot{\theta} + md^2\ddot{\phi} + mx^2\ddot{\theta}) - md\ddot{\phi} + 2mx\dot{\theta}\dot{\phi} + (mg)x\cos(\theta) - \\ (Mgc)\sin(\theta) - (mgd)\sin(\theta) = Fd - b_1\dot{\phi} \\ (m\ddot{x} - md\ddot{\phi}) - mx(\dot{\theta})^2 + mg\sin(\theta) = F - b_2\dot{\phi} \end{cases}$$

$$\Leftrightarrow \begin{cases} (md^2\ddot{\phi} + Mc^2\ddot{\theta} + mx^2\ddot{\theta}) - md\ddot{\phi} + 2mx\dot{\theta}\dot{\phi} + b_1\dot{\phi} + mgx\cos(\theta) - \\ Mgc\sin(\theta) - mgd\sin(\theta) = Fd \\ m\ddot{x} + b_2\dot{\phi} - md\ddot{\phi} - mx(\dot{\theta})^2 + mg\sin(\theta) = F \end{cases} \quad (2.8)$$

T (2.8) suy ra:

$$\left\{ \begin{aligned} \ddot{\theta} &= \frac{(-b_1\dot{\phi} - 2mx\dot{\theta}\dot{\phi} - b_2d\dot{\phi} + mdx(\dot{\theta})^2 - mgx\cos(\theta) + Mgc\sin(\theta))}{(Mc^2 + mx^2)} \\ &+ \left(\frac{2d}{(Mc^2 + mx^2)} \right) F \\ \ddot{\phi} &= \frac{-(b_1d\dot{\theta} + 2mdx\dot{\theta}\dot{\phi} + b_2d^2\dot{\phi} - md^2x(\dot{\theta})^2 + mgdx\cos(\theta) - Mgdc\sin(\theta))}{(Mc^2 + mx^2)} \\ &- \frac{b_2\dot{\phi}}{m} + x(\dot{\theta})^2 - g\sin(\theta) + \left(\frac{2d^2}{(Mc^2 + mx^2)} + \frac{1}{m} \right) F \end{aligned} \right. \quad (2.9)$$

t bi n tr ng thái: $X = [\theta \quad \dot{\phi} \quad x \quad \ddot{\phi}]^T = [x_1 \quad x_2 \quad x_3 \quad x_4]^T$.

Và: $F = K_m U$

H (2.9) c vi t l i nh sau:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f_1(X) + g_1(X)U \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = f_2(X) + g_2(X)U \end{cases} \quad (2.10)$$

V i U là i n áp i u khi n a vào ng c .

Suy ra (2.10) là phương trình phi tuyến của hệ thống con nêm ngang cân bằng.

Trong đó:

$$f_1(X) = \frac{(-b_1x_2 - 2mx_3x_4x_2 - b_2dx_4 + mdx_3x_2^2 - mgx_3 \cos(x_1) + Mgc \sin(x_1))}{(Mc^2 + mx_3^2)}$$

$$g_1(X) = K_m \left(\frac{2d}{(Mc^2 + mx_3^2)} \right)$$

$$f_2(X) = \frac{d(-b_1x_2 - 2mx_3x_4x_2 - b_2dx_4 + mdx_3x_2^2 - mgx_3 \cos(x_1) + Mgc \sin(x_1))}{(Mc^2 + mx_3^2)} - \frac{b_2x_4}{m} + x_3x_2^2 - g \sin(x_1)$$

$$g_2(X) = K_m \left(\frac{2d^2}{(Mc^2 + mx_3^2)} + \frac{1}{m} \right)$$

2.2.4. Tuyến tính hoá hệ thống tỉ mỉ cân bằng

tỉ mỉ cân bằng của hệ thống là: $\theta_0 \approx 0$; $x_0 \approx 0$

Thích hiện tuyến tính hoá ta có:

$$\cos(\theta) \cong 1; \sin(\theta) \cong \theta; x \cong (\theta)^2 \cong x^2 \cong 0$$

Khi đó (2.9) có vi phân sau:

$$\Leftrightarrow \begin{cases} \dot{x} = \frac{(-b_1\dot{\theta} - b_2d\dot{x} - mgx + Mgc\theta)}{(Mc^2)} + \left(\frac{2d}{(Mc^2)} \right) F \\ \dot{x} = \frac{-(b_1d\dot{\theta} + b_2d^2\dot{x} + mgdx - Mgcd\theta)}{(Mc^2)} - \frac{b_2\dot{x}}{m} - g\theta + \left(\frac{2d^2}{(Mc^2)} + \frac{1}{m} \right) F \end{cases}$$

$$\Leftrightarrow \left\{ \begin{array}{l} \mathfrak{R} = \left[\left(\frac{Mgc}{Mc^2} \right) \theta + \left(\frac{-b_1}{Mc^2} \right) \mathfrak{R} + \left(\frac{-mg}{Mc^2} \right) x + \left(\frac{-b_2 d}{Mc^2} \right) \mathfrak{R} \right] + \left(\frac{2d}{Mc^2} \right) F \\ \mathfrak{R} = \left[\left(\frac{Mgdc}{Mc^2} - g \right) \theta + \left(\frac{-b_1 d}{Mc^2} \right) \mathfrak{R} + \left(\frac{-mgd}{Mc^2} \right) x + \left(\frac{-b_2 d^2}{Mc^2} - \frac{b_2}{m} \right) \mathfrak{R} \right] + \left(\frac{2d^2}{Mc^2} + \frac{1}{m} \right) F \end{array} \right.$$

$$\Leftrightarrow \left\{ \begin{array}{l} \mathfrak{R} = \left[\left(\frac{g}{c} \right) \theta + \left(\frac{-b_1}{Mc^2} \right) \mathfrak{R} + \left(\frac{-mg}{Mc^2} \right) x + \left(\frac{-b_2 d}{Mc^2} \right) \mathfrak{R} \right] + \left(\frac{2d}{Mc^2} \right) F \\ \mathfrak{R} = \left[\left(\frac{gd}{c} - g \right) \theta + \left(\frac{-b_1 d}{Mc^2} \right) \mathfrak{R} + \left(\frac{-mgd}{Mc^2} \right) x + \left(- \left(\frac{b_2 d^2}{Mc^2} + \frac{b_2}{m} \right) \right) \mathfrak{R} \right] + \left(\frac{2d^2}{Mc^2} + \frac{1}{m} \right) F \end{array} \right.$$

t bi n tr ng thái:

$$X = \left[\theta \quad \mathfrak{R} \quad x \quad \mathfrak{R} \right]^T = \left[x_1 \quad x_2 \quad x_3 \quad x_4 \right]^T$$

Suy ra ph ng trình tuyen tính hoá h th ng:

$$\Rightarrow \begin{bmatrix} \mathfrak{R} \\ \mathfrak{R} \\ \mathfrak{R} \\ \mathfrak{R} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \left(\frac{g}{c} \right) & \left(\frac{-b_1}{Mc^2} \right) & \left(\frac{-mg}{Mc^2} \right) & \left(\frac{-b_2 d}{Mc^2} \right) \\ 0 & 0 & 0 & 1 \\ \left(\frac{gd}{c} - g \right) & \left(\frac{-b_1 d}{Mc^2} \right) & \left(\frac{-mgd}{Mc^2} \right) & \left(- \left(\frac{b_2 d^2}{Mc^2} + \frac{b_2}{m} \right) \right) \end{bmatrix} \begin{bmatrix} \theta \\ \mathfrak{R} \\ x \\ \mathfrak{R} \end{bmatrix} + \begin{bmatrix} 0 \\ \left(\frac{2d}{Mc^2} \right) \\ 0 \\ \left(\frac{2d^2}{Mc^2} + \frac{1}{m} \right) \end{bmatrix} F$$

$$\Rightarrow \begin{bmatrix} \theta \\ \dot{\theta} \\ x \\ \dot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \left(\frac{g}{c}\right) & \left(\frac{-b_1}{Mc^2}\right) & \left(\frac{-mg}{Mc^2}\right) & \left(\frac{-b_2 d}{Mc^2}\right) \\ 0 & 0 & 0 & 1 \\ \left(\frac{gd}{c} - g\right) & \left(\frac{-b_1 d}{Mc^2}\right) & \left(\frac{-mgd}{Mc^2}\right) & \left(-\left(\frac{b_2 d^2}{Mc^2} + \frac{b_2}{m}\right)\right) \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \\ x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ K_m \left(\frac{2d}{Mc^2}\right) \\ 0 \\ K_m \left(\frac{2d^2}{Mc^2} + \frac{1}{m}\right) \end{bmatrix} U \quad (2.11)$$

2.2.4. Bảng thông số của hệ con nhem ng c t cân b ng

Bảng 2.1: Các thông số của hệ con nhem ng c [10], [15], [23]

Ký hiệu	Giá trị	Thông số hệ thống
M	3 kg	Khối lượng con nhem
m	0.6 kg	Khối lượng vật nhem
c	0.05 m	Khoảng cách giữa trục quay và trục tâm con nhem
d	0.12 m	Khoảng cách giữa trục quay và vật nhem
b ₁	0.05 N/m/s	Hệ số ma sát trục quay
b ₂	45 N/m/s	Hệ số ma sát vật nhem và trục quay
g	9.81 m/s ²	Giá trị trọng trường
K _m	5 Nm/A	Hệ số mô men của cuộn dây DC

2.3. B i u k h i n t c c

th c h i n v i c t h i t k b i u k h i n t c c, t a s t u y n t i n h h o á m ô h ì n h p h i t u y n c a h c o n n ê m n g c t h à n h m ô h ì n h k h ô n g g i a n t r n g t h á i.

T p h ã n g t r ì n h (2.10) v à (2.11) t a c ó c á c m a t r n t r n g t h á i:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \left(\frac{g}{c}\right) & \left(\frac{-b_1}{Mc^2}\right) & \left(\frac{-mg}{Mc^2}\right) & \left(\frac{-b_2d}{Mc^2}\right) \\ 0 & 0 & 0 & 1 \\ \left(\frac{gd}{c} - g\right) & \left(\frac{-b_1d}{Mc^2}\right) & \left(\frac{-mgd}{Mc^2}\right) & \left(-\left(\frac{b_2d^2}{Mc^2} + \frac{b_2}{m}\right)\right) \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 196.2 & -6.67 & -784.8 & -720 \\ 0 & 0 & 0 & 1 \\ 13.734 & -0.8 & -94.176 & -161.4 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ K_m \left(\frac{2d}{Mc^2}\right) \\ 0 \\ K_m \left(\frac{2d^2}{Mc^2} + \frac{1}{m}\right) \end{bmatrix} = \begin{bmatrix} 0 \\ 160 \\ 0 \\ 27.53 \end{bmatrix}; C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ v à } D = [0]$$

✓ K i m t r a t i n h i u k h i n c v à q u a n s á t c c a h t h ã n g, t a t i n h:

- M a t r n: $M_c = [B, AB, A^2B, A^3B]$

- M a t r n: $M_o = \begin{bmatrix} C \\ CA \\ CA^2 \\ CA^3 \end{bmatrix}$

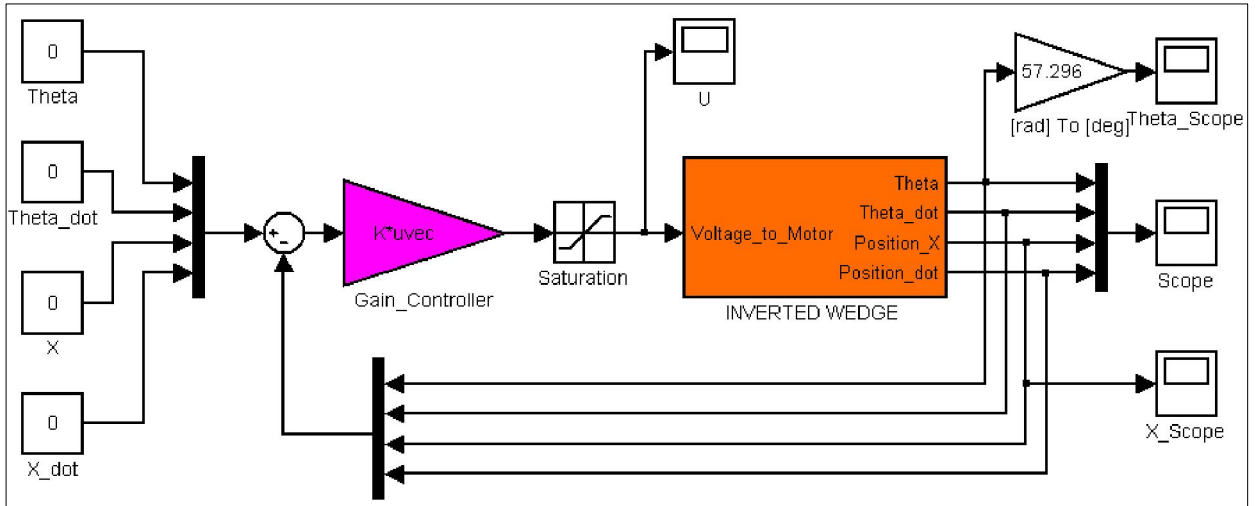
S d ã n g M a t l a b, k i m t r a c: $rank(M_c) = rank(M_o) = 4$. S u y r a h

t h ã n g (2.10) l à c ó t h i u k h i n v à q u a n s á t c. N h v y v i c t h i t k b

i u k h i n t c c l à c ó t h t h c h i n c.

S b i u khi n t c c cho h th ng công th c (2.10) c trình bày

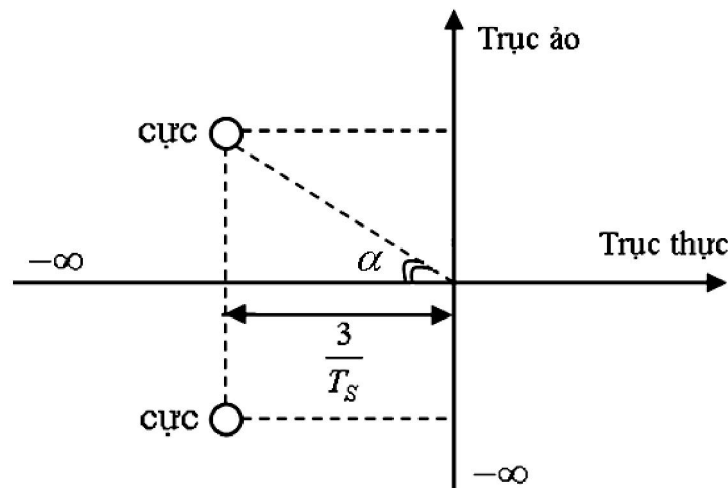
hình 2.3 v i các vector tr ng thái c i u khi n là: $[\theta, \dot{\theta}, x, \dot{x}]^T$



Hình 2.3: B i u khi n t c c cho mô hình con nêm ng c t cân b ng

Trong ó, giá tr c a khâu b o hoà là ± 24 , ây là giá tr i n áp c c i c p cho ng c DC s d ng trong mô hình con nêm ng c trong th c nghi m.

thi tk b i u khi n t c c, ta c n ph i l a ch n b c c n nh và phù h p v i h th ng. Hình 2.4 th hi n s nh h ng c a các i m c c lên áp ng c a h th ng.



Hình 2.4: S nh h ng c a c c lên h th ng

V i:

T_s : là th i gian xác l p c a h th ng.

α : nh h ng n v t l P_o (%) c a h th ng theo công th c: $P_o = e^{\frac{-\pi}{tg(\alpha)}}$

i v i mô hình con nên ng c t cân b ng thì vi c gi cân b ng là quan tr ng nh t. Cho nên áp ng c a góc nghiêng con nê m ph i có th i gian xác l p nhanh và v t l ph i nh . ng th i con nê m không thay i v trí c a xe (v t n ng) t c th i quá nhanh nên áp ng c a v trí xe (v t n ng) s không có yêu c u quá kh t khe v th i gian xác l p c ng nh v t l . Do ó ta có th ch n nh sau:

+ ***i v i áp ng c a góc nghiêng con nê m:***

- Ch n th i gian xác l p c a góc nghiêng c a con nê m là 0.5 [giây], suy ra

$$\text{giá tr l n ph n th c c a i m c c là: } \frac{3}{T_s} = \frac{3}{0.5} = 6$$

- v t l P (%) c a áp ng góc nghiêng c a con nê m là 2%:

$$P_o = e^{\frac{-\pi}{tg(\alpha)}} = 2\% \Rightarrow tg(\alpha) = 0.8026, \text{ suy ra giá tr l n ph n o c a i m c c}$$

này là:

$$6 \times tg(\alpha) = 6 \times 0.8026 = 4.8$$

Suy ra giá tr c p c c c ch n là: **-6 + 4.8*i ; -6 - 4.8*i**

+ ***i v i v trí xe (v t n ng):***

- Ch n th i gian xác l p c a v trí bánh xe robot là 1 [giây], suy ra giá tr

$$\text{l n ph n th c c a i m c c là: } \frac{3}{T_s} = \frac{3}{1} = 3$$

- v t l P (%) c a áp ng v trí v t n ng:

$$P_o = e^{\frac{-\pi}{tg(\alpha)}} = 4\% \Rightarrow tg(\alpha) = 0.975$$

Suy ra l n giá tr ph n o c a i m c c này là: $3 \times tg(\alpha) = 3 \times 0.975 = 2.925 \cong 3$

Suy ra giá trị cực tiểu của hàm là: $-3 + 3i$; $-3 - 3i$.

cho hình thức nghiệm thì tất cả các cực tiểu nằm ở phần bên trái mặt phẳng phức. Tóm lại, giá trị cực tiểu của hàm như sau:

$$V = [-3 + 3i; -3 - 3i; -6 + 4.8i; -6 - 4.8i]$$

Sử dụng công thức Ackerman tính luật điều khiển: $K = \text{acker}(A, B, V)$

Sử dụng Matlab thực hiện phép toán, ta có luật điều khiển K là:

$$K = [K_1; K_2; K_3; K_4] = [0.84; 0.0074; -4.4484; -5.4933]$$

Suy ra tín hiệu điều khiển ngược kéo véc tơ ngược tính theo công thức:

$$U = [(K_1(x_{ref} - x) + K_2(\dot{x}_{ref} - \dot{x})) + [K_3(\theta_{ref} - \theta) + K_4(\dot{\theta}_{ref} - \dot{\theta})] \quad (2.12)$$

2.4. Bộ điều khiển trượt

Vì bộ điều khiển trượt không thể điều khiển cân bằng con nêm góc khi góc nghiêng lớn hơn 18° , cho nên ta phải thiết kế bộ điều khiển mờ có khả năng điều khiển tự động. Đây tác giả chọn phương pháp điều khiển trượt.

$$\text{Tập phương trình (2.10) là } \begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f_1(X) + g_1(X)U \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = f_2(X) + g_2(X)U \end{cases}$$

Ta có:

$$\text{Chọn mặt trượt } S = x_2 - \varphi(x_1) \quad (2.13)$$

$$\text{Nếu } S = 0 \Rightarrow x_2 = \varphi(x_1) \Rightarrow \dot{x}_1 = \varphi(x_1) \quad (2.14)$$

Chọn $\varphi(x_1)$ sao cho phương trình (2.14) có nghiệm $x_e = [0]$ là điểm cân bằng ổn định tiệm cận.

$$\Leftrightarrow \text{chọn } \varphi(x_1) = -k x_1 \text{ (k là hằng số dương)}. \quad (2.15)$$

$$\Rightarrow \dot{x}_1 = -k x_1 = h(x) \quad (2.16)$$

Điểm $x_e = [0]$ là điểm cân bằng ổn định tiệm cận của phương trình (2.16) khi và chỉ khi:

$$\text{Xét ma trận } A = \frac{\partial h(x)}{\partial(x)} = -k < 0 \Rightarrow \text{điểm } x_e = [0] \text{ là điểm cân bằng ổn định}$$

tiệm cận.

\Rightarrow phương trình (2.15) thỏa mãn điều kiện ổn định.

$$\text{Từ (2.13) và (2.15) Chọn mặt trượt: } S = x_2 + kx_1 \quad (2.17)$$

$$\Rightarrow \dot{S} = \dot{x}_2 + k\dot{x}_1 = f_1(X) + g_1(X)U + kx_2 \quad (2.18)$$

$\Rightarrow S$ có bậc trượt bằng 1, thỏa mãn điều kiện ổn định.

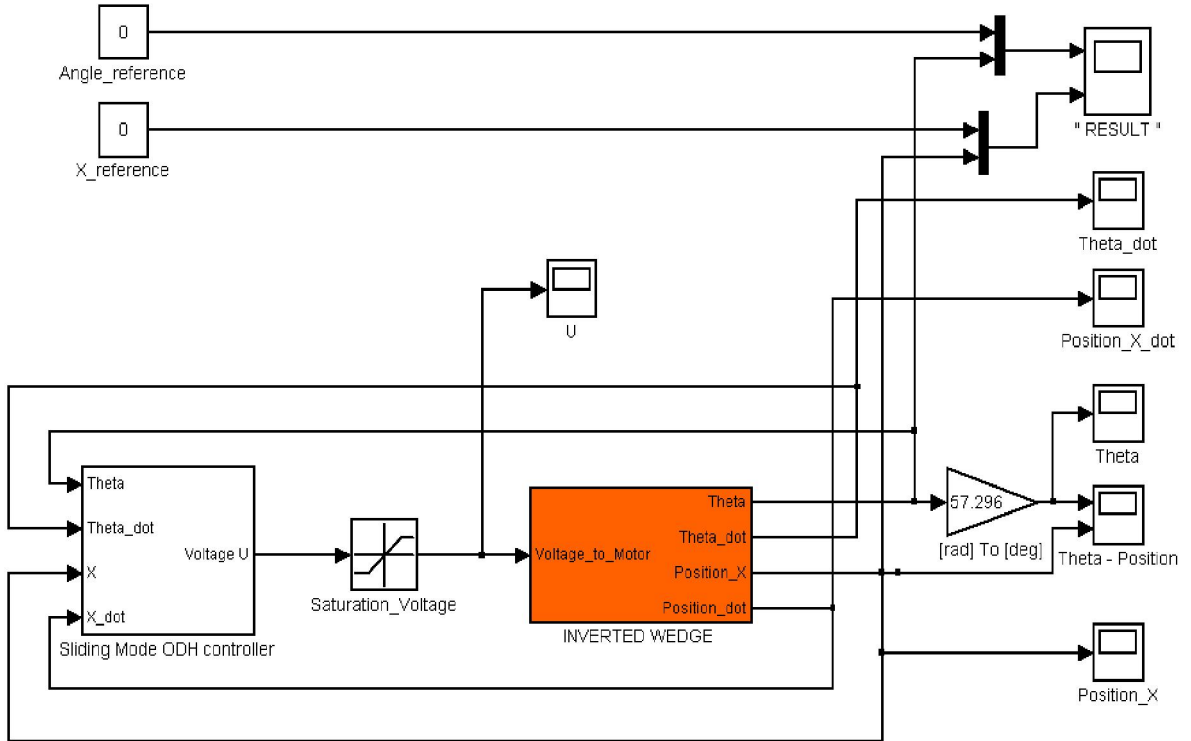
Chọn $\dot{S} = -\alpha \text{Sign}(S)$, với α là hằng số dương tùy chọn.

Từ (2.18) suy ra luật điều khiển trượt:

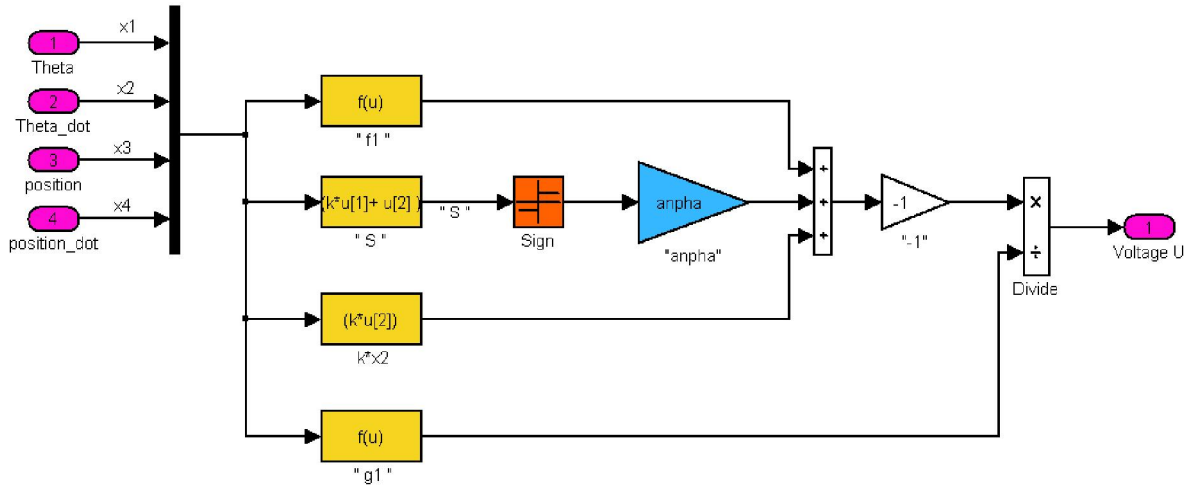
$$U = \frac{1}{g_1(X)} [-\alpha \text{Sign}(S) - f_1(X) - kx_2] \quad (2.19)$$

✓ S mô phỏng:

Các h s ch n mô phỏng: $\alpha = 92, k = 3$



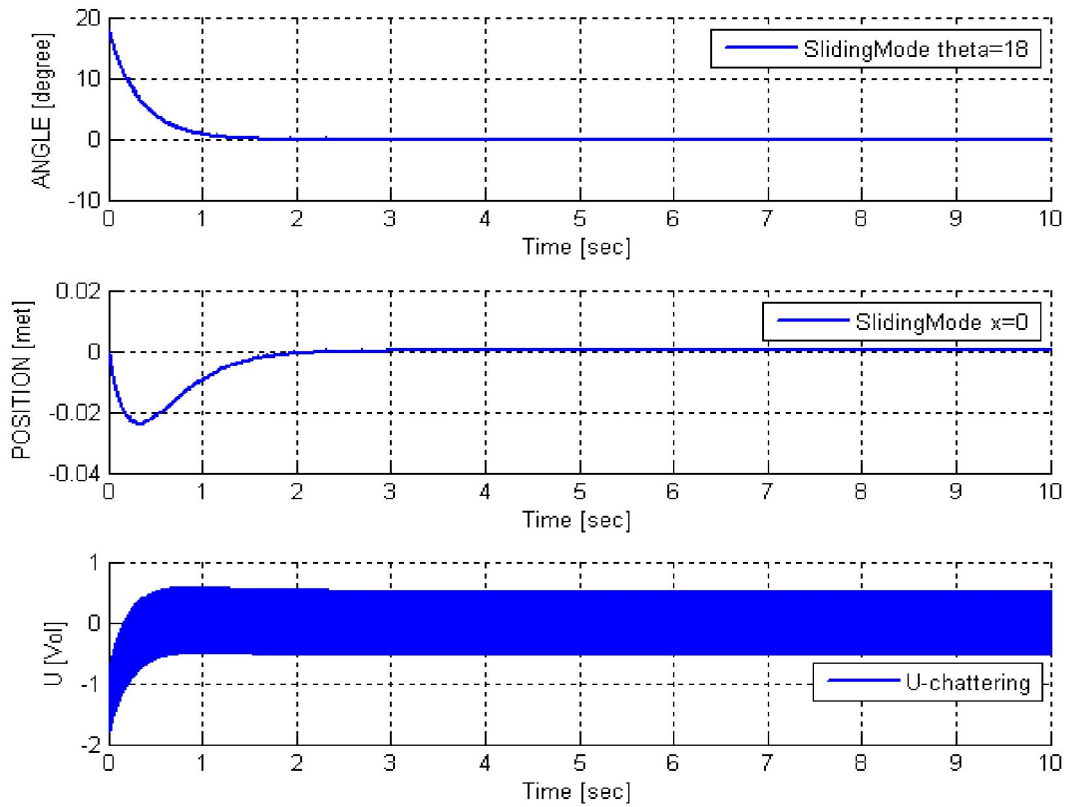
Hình 2.5: S mô phỏng b i u khi n tr t



Hình 2.6: s bên trong kh i “Sliding Mode ODH Controller”

2.5. B i u khi n k t h p Tr t - M - PID

B i u khi n tr t có th i u khi n cân b ng con nê m ng c t t, tuy nhiên có nh c i m là tín hi u i u khi n có hi n t ng chattering (hình 2.7). kh c ph c hi n t ng chattering, trong khuôn kh tài này, tác gi xu t s d ng b i u khi n k t h p Tr t - M - PID.



Hình 2.7: Hi n t ng chattering c a b i u khi n tr t

▼ **B i u k h i n k t h p Tr t - M - PID**

B i u k h i n k t h p Tr t - M - PID c thi t k d a tr ên c s c a b i u k h i n tr t m c 2.4 v i c u tr úc nh sau:

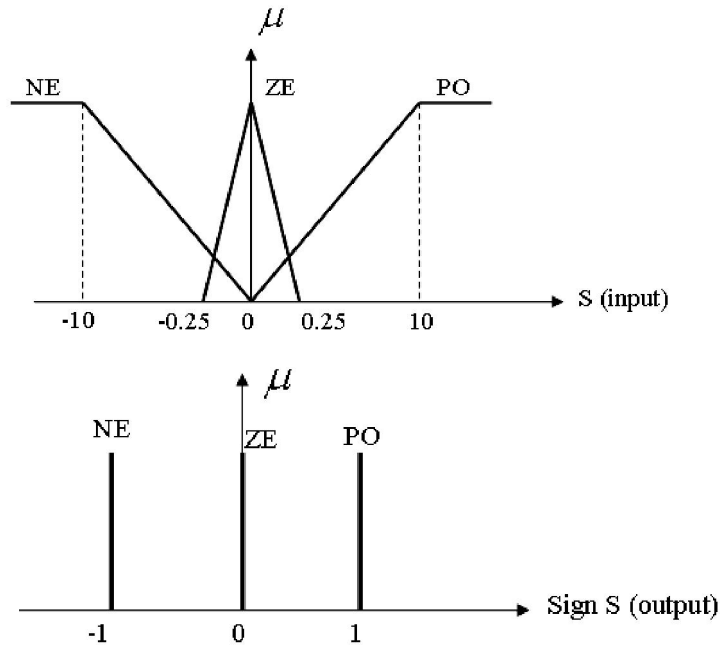
- Hàm $\text{Sign}(S)$ c thay b ñng b i u k h i n m theo quy t c m Sugeno v i:

§ Ngõ vào là S : $S = (\text{âm}, \text{zero}, \text{d ñng}) = (\text{NE}, \text{ZE}, \text{PO})$.

§ Ngõ ra là $\text{Sign}(S)$: $\text{output} = (\text{âm}, \text{zero}, \text{d ñng}) = (\text{NE}, \text{ZE}, \text{PO})$.

§ Ph ñng pháp suy đi n m MAX – MIN.

§ Ph ñng pháp gi i m trung bình tr ñng s .



+ If S is ZE then output is ZE.

+ If S is NE then output is NE.

+ If S is PO then output is PO.

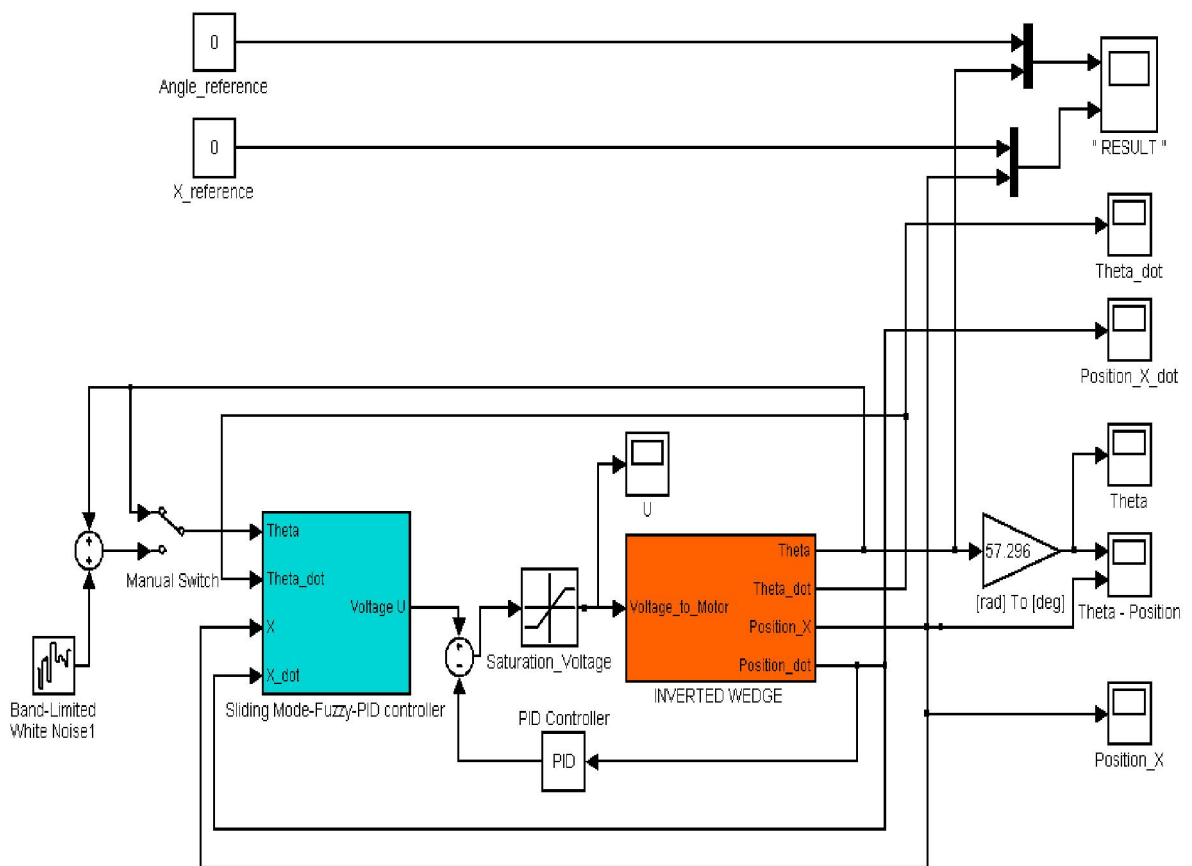
- B i u k h i n PID th c h i n n h i m v i u k h i n n ñnh v n t c ñi chuy ñc a v t ñng (xe tr t tr ên b m t con ñm).

- Lu t i u k h i n $U = G \cdot U_{\text{fuzzy_Sliding}} - U_{\text{PID}}$

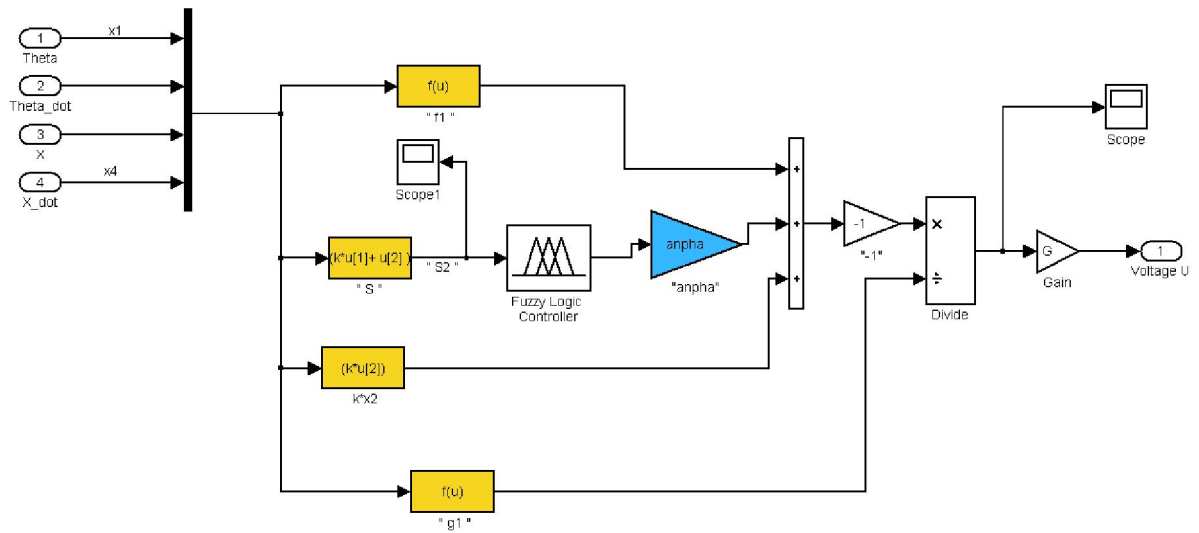
V i các thông s ch n là: $\alpha = 0.1$; $k = 10$; $K_p = 10$; $K_i = 0.05$; $K_D = 0.001$;

$$G = 3$$

Các thông s trên c ch n t c s c l ng c a b i u khi n tr t m c 2.4 và t i n hành ch nh nh c ó c t n hi u i u khi n nh mong mu n.



Hình 2.8: S mô ph ng b i u khi n Tr t-M -PID



Hình 2.9: Sơ đồ bên trong khối “Sliding Mode-Fuzzy-PID Controller”

2.6. Bài tập khi n m

Bài tập khi n m tr t có th i u khi n cân b ng con n m ng c t t v i góc nghiêng r ng, tuy nhiên các áp ng tín hi u i u khi n l i có hi n t ng dao ng cho nên ta c n ph i thi t k m t b i u khi n t t h n, ây tác gi ch n ph ng pháp i u khi n m .

2.6.1. Phân tích thi t k b i u khi n m

H con n m ng c t cân b ng ph thu c vào v trí xe và góc l ch con n m vì v y khi thi t k b i u khi n m chúng ta ch n gi i pháp là b i u khi n MISO có 4 ngõ vào ph n h i tín hi u v và 1 ngõ ra áp i u khi n.

- Bài tập khi n m 4 tín hi u ngõ vào:
 1. V trí xe x
 2. V n t c xe $\dot{x}(t)$
 3. Góc l ch $\theta(t)$
 4. V n t c góc $\dot{\theta}(t)$
- Bài tập khi n m có 1 bi n ngõ ra là tín hi u áp i u khi n u c a motor

- Luật mô-đun xây dựng dựa trên kinh nghiệm và hoạt động củaalcon nêmluật mô-đun. Bộ điều khiển có 4 ngõ vào nên ta chọn số bit ngõ vào là 3 có số luật mô-đun là 81.
- Ta xây dựng bộ điều khiển theo quy tắc Sugeno. Số bit ngõ ra ta chọn là 7 bit ngõ ra với 7 mức khác nhau của motor.



Hình 2.10 : Kh i mô t b i u khi n m s thi t k

2.6.2. Thi t k b i u khi n m

B i u khi n có 4 bi n vào: sai s v trí góc, vị phân c a sai s vị trí góc , sai s v trí v t n ng , vị phân sai s v trí v t n ng .

- V trí góc l ch : $[-\pi/2, \pi/2]$ (rad).
- V n t c góc : $[-1, 1]$ (rad/s).
- V trí v t n ng : $[-0.3, 0.3]$ (m).
- V n t c v t n ng : $[-1, 1]$ (m/s).

B i u khi n có 1 bi n ra: i n áp c p cho ñng c .

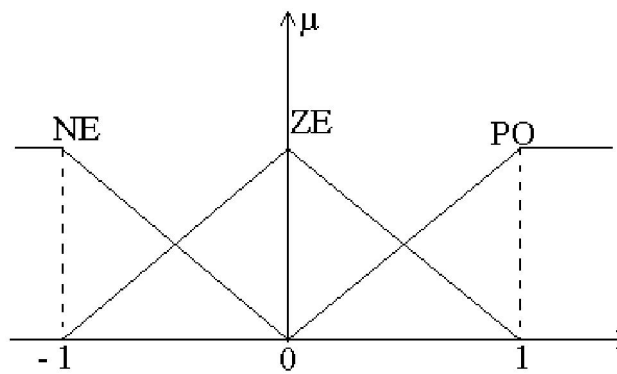
- i n áp c p cho ñng c DC: $[-24, 24]$ (VDC).

Chu n hoá t p c s c a các bi n vào ra: ta chu n hoá bi n ngõ vào/ra m i n $[-1, 1]$ nên ta ch n h s khu ch i kh i t i n và h u x lý:

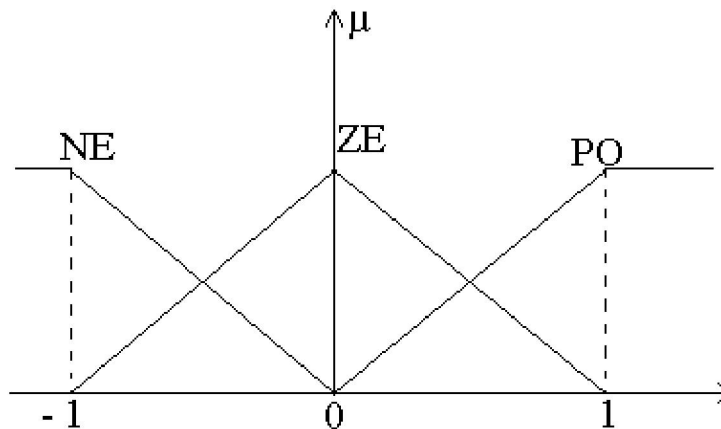
- H s t i n x lý: $K_1 = 0.6; K_2 = 1; K_3 = 2; K_4 = 1$
- H s h u x lý: $K_5 = 24$.

Xây dựng biên ngôn ngữ cho 4 biến vào là : sai số vị trí góc, vị phân cực sai số vị trí góc, sai số vị trí vận tốc, vị phân sai số vị trí vận tốc và biên ngôn ngữ áp dụng khi cần.

- Biên ngôn ngữ ngõ vào sai số vị trí góc (θ) có 3 giá trị :
 - 1.PO (dương)
 - 2.ZE (zero)
 - 3.NE (âm)



- Biên ngôn ngữ ngõ vào vị phân cực sai số vị trí góc ($\dot{\theta}$) có 3 giá trị :
 - 1.PO (dương)
 - 2.ZE (zero)
 - 3.NE (âm)

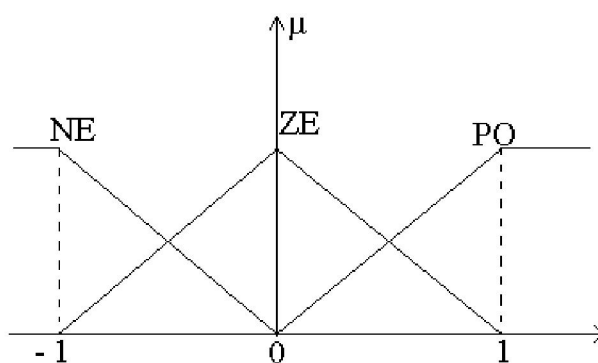


- Biến ngôn ngữ ngõ vào sai số vị trí xe (vận tốc) x có 3 giá trị :

1.PO (dương)

2.ZE (zero)

3.NE (âm)

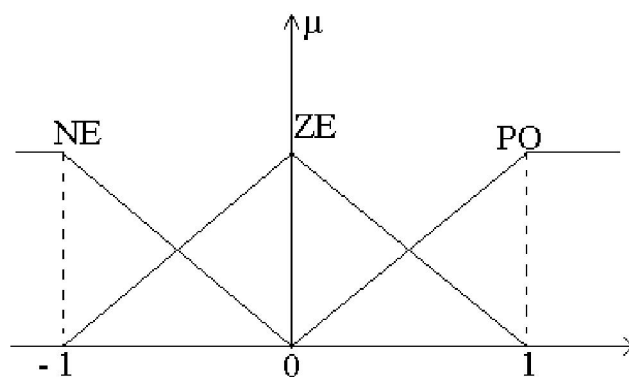


- Biến ngôn ngữ ngõ vào vị phân sai số vị trí xe δ có 3 giá trị :

1.PO (dương)

2.ZE (zero)

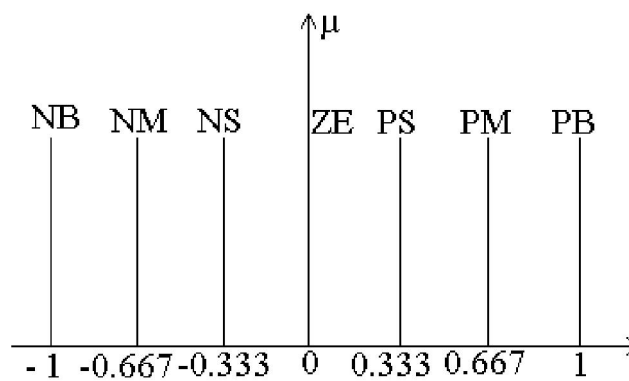
3.NE (âm)



- Biến ngôn ngữ ngõ ra biến ngõ ra áp lực khi cần có 7 giá trị :

1.NB (âm lớn)

- 2.NM (âm v a)
- 3.NS (âm nh)
- 4.ZE (zero)
- 5.PS (d ng nh)
- 6.PM (d ng v a)
- 7.PB (d ng l n)



Ta có miền giá trị của các biến vào ra:

Góc quay θ : $[-\pi/2, \pi/2]$ (rad)

Vận tốc góc $\dot{\theta}$: $[-1, 1]$ (rad/s)

Vị trí xe x : $[-0.3, 0.3]$ (m)

Vận tốc xe \dot{x} : $[-1, 1]$ (m/s)

Ta xây dựng các luật inference như sau:

✓ Bảng lu t m : xây d ng trên c s th c nghi m và kinh nghi m chuyên gia.

B ng 2.2: Các lu t s d ng trong i u khi n m

STT	Ngõ vào				Ngõ ra
	Góc l ch	V n t c góc	V trí v t	V n t c v t	Áp i u khi n
1	NE	NE	NE	NE	NB
2	NE	NE	NE	ZE	NM
3	NE	NE	NE	PO	NS
4	NE	NE	ZE	NE	ZE
5	NE	NE	ZE	ZE	PS
6	NE	NE	ZE	PO	PM
7	NE	NE	PO	NE	PS
8	NE	NE	PO	ZE	PM
9	NE	NE	PO	PO	PB
10	NE	ZE	NE	NE	NM
11	NE	ZE	NE	ZE	NM
12	NE	ZE	NE	PO	NS
13	NE	ZE	ZE	NE	ZE
14	NE	ZE	ZE	ZE	PS
15	NE	ZE	ZE	PO	PM
16	NE	ZE	PO	NE	PS
17	NE	ZE	PO	ZE	PM
18	NE	ZE	PO	PO	PB
19	NE	PO	NE	NE	NM
20	NE	PO	NE	ZE	NS
21	NE	PO	NE	PO	NS
22	NE	PO	ZE	NE	ZE
23	NE	PO	ZE	ZE	ZE

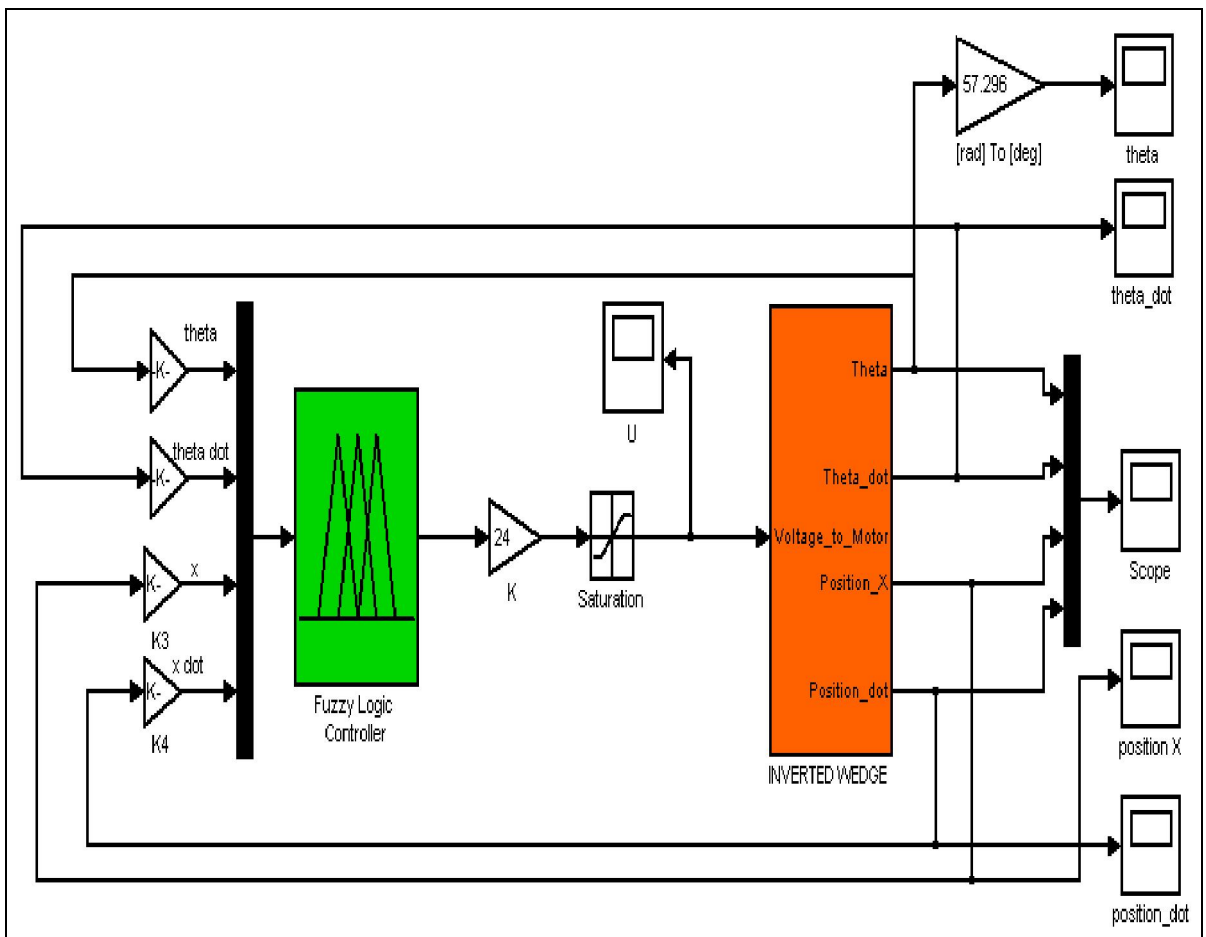
24	NE	PO	ZE	PO	PS
25	NE	PO	PO	NE	ZE
26	NE	PO	PO	ZE	PS
27	NE	PO	PO	PO	PM
28	ZE	NE	NE	NE	NS
29	ZE	NE	NE	ZE	ZE
30	ZE	NE	NE	PO	PS
31	ZE	NE	ZE	NE	ZE
32	ZE	NE	ZE	ZE	PS
33	ZE	NE	ZE	PO	PM
34	ZE	NE	PO	NE	ZE
35	ZE	NE	PO	ZE	PS
36	ZE	NE	PO	PO	PM
37	ZE	ZE	NE	NE	NB
38	ZE	ZE	NE	ZE	NB
39	ZE	ZE	NE	PO	NM
40	ZE	ZE	ZE	NE	NM
41	ZE	ZE	ZE	ZE	NS
42	ZE	ZE	ZE	PO	NS
43	ZE	ZE	PO	NE	ZE
44	ZE	ZE	PO	ZE	PS
45	ZE	ZE	PO	PO	PS
46	ZE	PO	NE	NE	NM
47	ZE	PO	NE	ZE	NS
48	ZE	PO	NE	PO	ZE
49	ZE	PO	ZE	NE	NM
50	ZE	PO	ZE	ZE	NS

51	ZE	PO	ZE	PO	ZE
52	ZE	PO	PO	NE	ZE
53	ZE	PO	PO	ZE	PS
54	ZE	PO	PO	PO	PS
55	PO	NE	NE	NE	NM
56	PO	NE	NE	ZE	NM
57	PO	NE	NE	PO	NS
58	PO	NE	ZE	NE	NS
59	PO	NE	ZE	ZE	ZE
60	PO	NE	ZE	PO	ZE
61	PO	NE	PO	NE	PS
62	PO	NE	PO	ZE	PS
63	PO	NE	PO	PO	PM
64	PO	ZE	NE	NE	NB
65	PO	ZE	NE	ZE	NM
66	PO	ZE	NE	PO	NM
67	PO	ZE	ZE	NE	NS
68	PO	ZE	ZE	ZE	ZE
69	PO	ZE	ZE	PO	PS
70	PO	ZE	PO	NE	PS
71	PO	ZE	PO	ZE	PM
72	PO	ZE	PO	PO	PM
73	PO	PO	NE	NE	NM
74	PO	PO	NE	ZE	NS
75	PO	PO	NE	PO	ZE
76	PO	PO	ZE	NE	NM
77	PO	PO	ZE	ZE	NS

78	PO	PO	ZE	PO	ZE
79	PO	PO	PO	NE	PS
80	PO	PO	PO	PS	PM
81	PO	PO	PO	PO	PB

Ta ch n:

- Ph ng pháp suy di n m MAX – MIN.
- Ph ng pháp gi i m t ng có tr ng s .



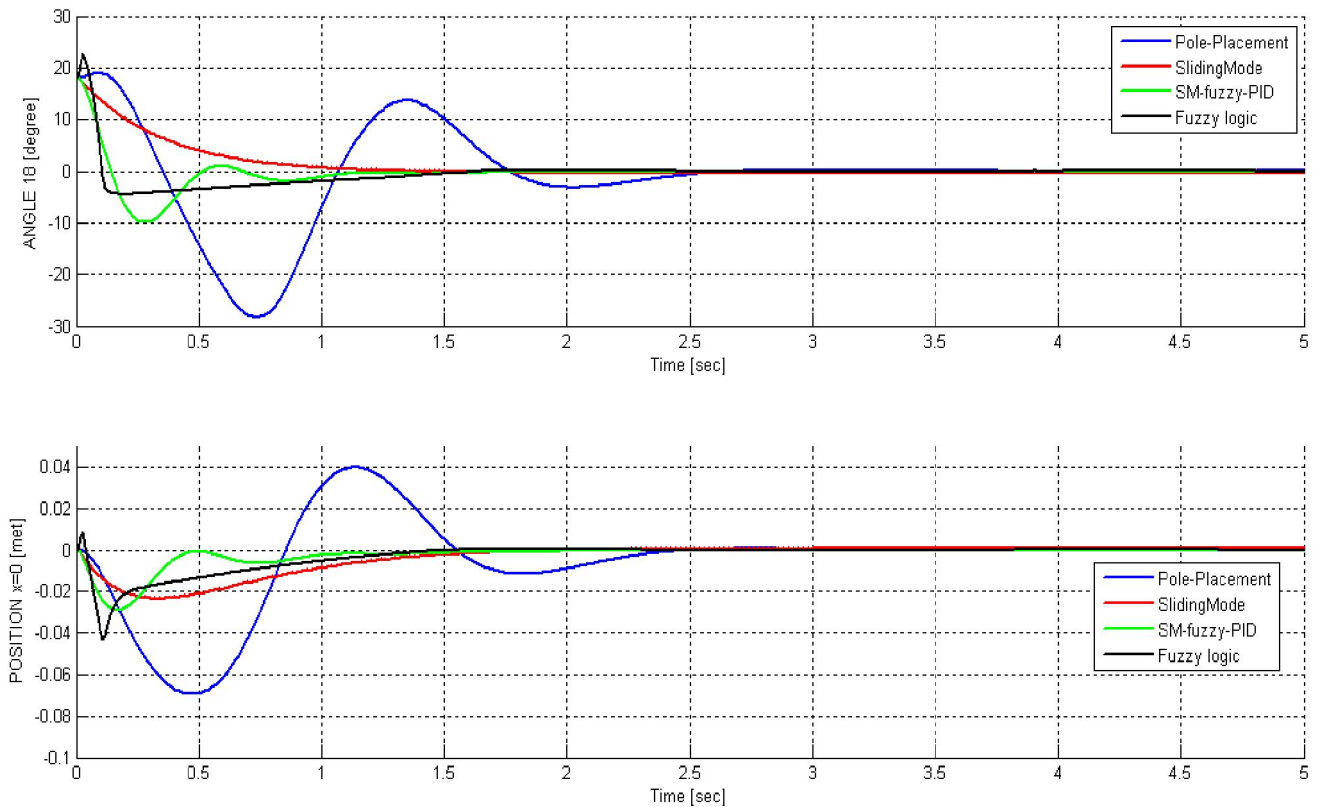
Hình 2.11 : S mô ph ng i u khi n m h nê m ng c

PHẦN III

CÁC KỸ THUẬT ĐIỀU KHIỂN

3.1. Kỹ thuật mô phỏng

3.1.1. Điều khiển vị trí khi nghiêng góc θ nhỏ



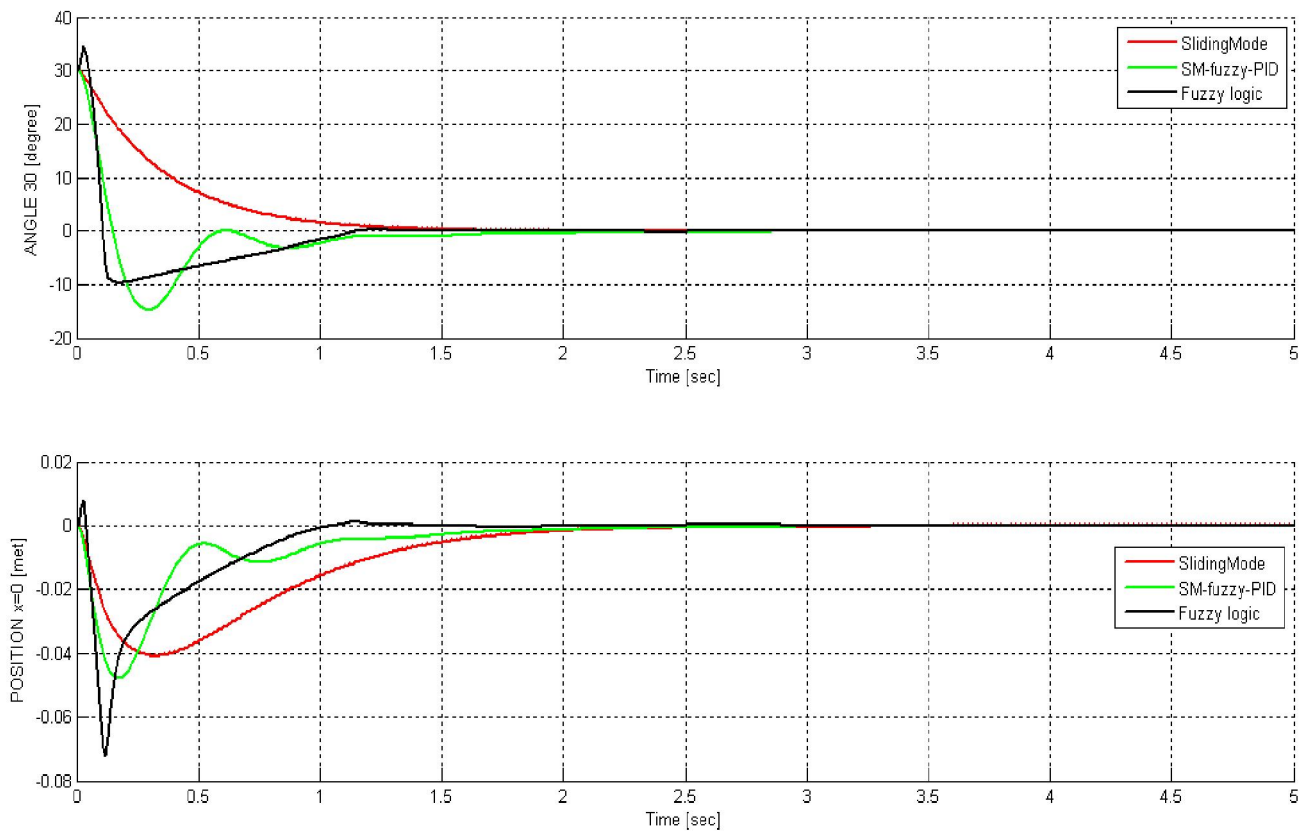
Hình 3.1: So sánh 3 bộ điều khiển vị trí $\theta \leq 18^\circ$

Cả 4 bộ điều khiển đều cho kết quả đáp ứng khá tốt, có thể điều khiển hệ cân bằng con nệm ngả.

- Bộ điều khiển t-c: đáp ứng có dao động lớn, v-t-l cao, thời gian xác lập chậm, không điều khiển được góc nghiêng lớn.
- Bộ điều khiển trượt: đáp ứng điều khiển không v-t-l, điều khiển được góc nghiêng lớn, tuy nhiên đáp ứng điều khiển cho thấy con nệm s-ho-t-ng ch-m. Tín hiệu u-b chattering.
- Bộ điều khiển m: thời gian xác lập nhanh, điều khiển được góc nghiêng lớn, tuy nhiên đáp ứng điều khiển cho thấy khi ho-t-ng con nệm s-b-gi-t-m-nh.

- Biểu đồ khi không thể áp dụng Trượt-Mô-đun-PID: biểu đồ khi góc nghiêng lớn, thời gian xác lập nhanh, vượt lộn xộn, và khi có hiện tượng chattering.

3.1.2. Biểu đồ biểu diễn khi góc nghiêng θ lớn



Hình 3.2: So sánh 3 biểu đồ khi vận tốc $\theta > 18^\circ$

Biểu đồ góc nghiêng $\theta > 18^\circ$ thì biểu đồ khi không còn biểu đồ cân bằng con nệm con a. Biểu đồ này có thể ghi thích như sau: vì khi thí nghiệm biểu đồ khi không thể áp dụng tính toán phi tuyến của hệ thống thì lần đầu tiên cân bằng $\sin(\theta) \approx 0; \cos(\theta) \approx 1$, cho nên khi góc nghiêng θ lớn ($\theta > 18^\circ$) thì hệ phương trình không gian trạng thái suy ra từ phương trình (2.11) không còn mô tả chính xác tính toán của vị trí con nệm con c, biểu đồ dẫn đến hệ thống tính toán pháp tuyến sẽ không còn phù hợp vì mô hình phi tuyến của con nệm con a. Hình 3.2 chỉ còn so

sánh áp ng i u khi n con nê m ng c khi s d ng b i u khi n tr t và b i u khi n m v i góc nghiêng ban u $\theta = 30^\circ$.

hình 3.2 ta th y r ng b i u khi n M (Fuzzy logic) cho k t qu áp ng i u khi n góc nghiêng con nê m t th n b i u khi n Tr t-M -PID (SM-Fuzzy-PID), th i gian xác l p nhanh h n, tuy nhiên áp ng tín hi u i u khi n v trí v t n ng có v t l l n h n b i u khi n Tr t-M -PID (SM-Fuzzy-PID).

B i u khi n Tr t-M -PID (SM-Fuzzy-PID) ã kh c ph c c hoàn toàn hi n t ng chattering.

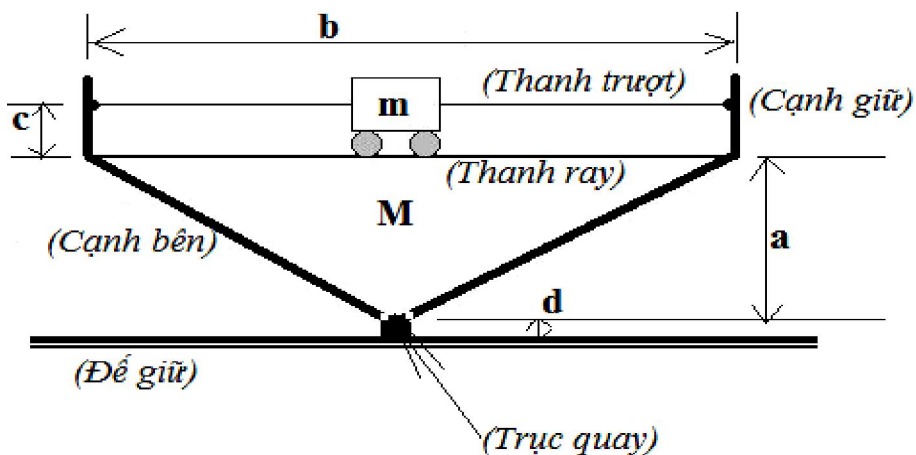
B ng 3.1: So sánh các b i u khi n

B i u khi n	Th i gian xác l p	Th i gian t ng	v t l
1. t c c	2.5 s	0.4 s	100 %
2. Tr t	1.7 s	1.5 s	0 %
3. Tr t – M –PID	1.7 s	0.2 s	50 %
4. M	1.2 s	0.15 s	30 %

3.2. K t qu thí công th c nghi m

3.2.1. Thi t k mô hình c khí

Ph n c khí c a mô hình c thí t k đ a tr n mô hình minh h a sau:



Hình 3.3: S thí t k mô hình h con nê m ng c

Bảng 3.2: Thông số thiết kế mô hình con nôm ng c thí nghiệm

Thông số	Ý nghĩa	Giá trị
a	Khoảng cách từ “trục quay” đến “thanh ray”	0.12 (m)
b	Chiều dài của “thanh ray” và “thanh trượt”	0.7 (m)
c	Khoảng cách giữa “thanh ray” và “thanh trượt”	0.05 -:- 0.1 (m)
d	Khoảng cách từ “trục quay” đến “gối”	0.11 (m)
M	Khối lượng của toàn khung của h nôm (không kể khối lượng của con ch y và ng c DC)	2 -:- 3 (kg) (t i a: 3 kg)
m	Khối lượng của con ch y (bao gồm cả khối lượng ng c DC gắn trên con ch y)	0.6 (kg)

Mô hình h con nôm ng c có các thành phần chính:

- V t n ng và c c u thanh trượt v t n ng : gồm v t n ng b ng kim lo i, thanh trượt giúp v t n ng trượt trên b m t c a con nôm mà không b r i ra ngoài khi di chuyển.
- C c u truy n ng gồm 1 motor DC-Servo, i n áp cung c p t 12VDC đến 24VDC, k t h p v i v t n ng t o thành “con ch y” giúp v t n ng di chuyển c trên b m t c a con nôm.
- Khung của h nôm (“c nh bên”, “c nh gi ”) làm b ng nh a c ng hay mica có dày phù h p ($t \geq 8 \text{ mm}$ tr lên) – và có màu tr ng s a.
- “Thanh trượt” là c u trúc tr t, làm b ng v t li u s t-thép, có b m t tr n “con ch y” có th tr t d dàng. (Tỉ t di n hình ch nh t hay tròn u c).
- “Thanh ray” là v t li u s t hay thép hay nhôm, có “gai” hay “rãnh” t o bám, dùng bánh xe của “con ch y” có th ch y qua l i trên ó.

- Nguồn DC nối vào trên “con chày”, đầu trục nguồn cố định vào mặt bánh xe (đường kính 2:-3 cm) “con chày” có thể di chuyển linh hoạt trên “thanh ray”.
- “Trục quay” là thanh sắt-thép có tiết diện tròn, đường kính khoảng 8 :-10 (mm).
- Giá làm bằng gỗ, có chiều dài lớn hơn chiều dài của “thanh trượt” - khoảng 0.8(m) và chiều rộng khoảng 0.2(m). “Giá” phải cố định cho mô hình không bị “rung-lắc” ngay khi hệ thống hoạt động.

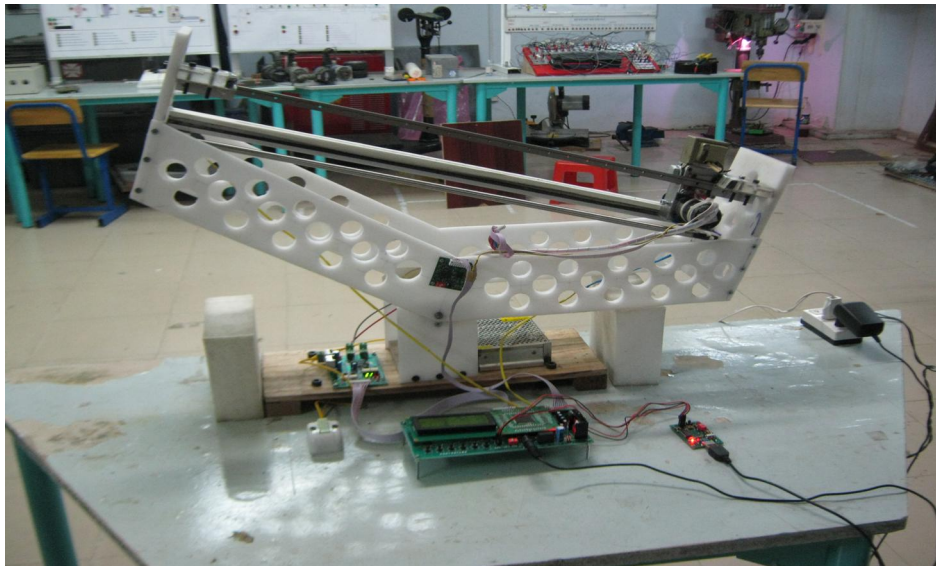
3.2.2. Hình ảnh mô hình con nện bằng cơ cấu cân bằng thí nghiệm



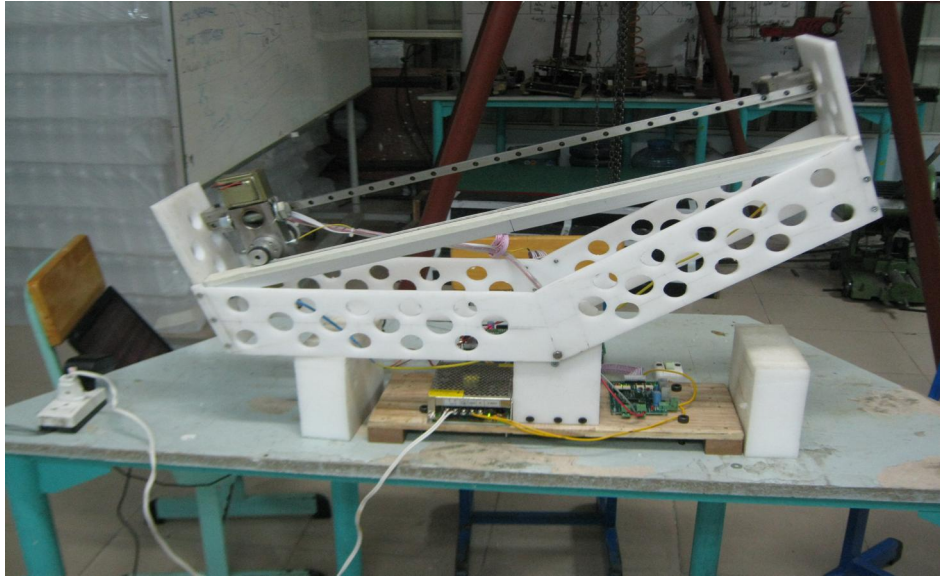
Hình 3.4 (a)



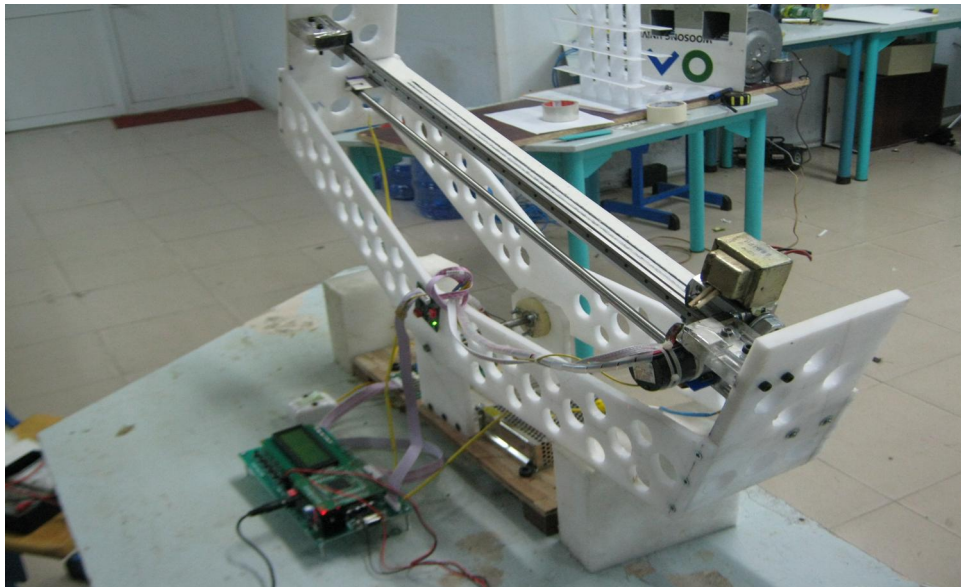
Hình 3.4 (b)



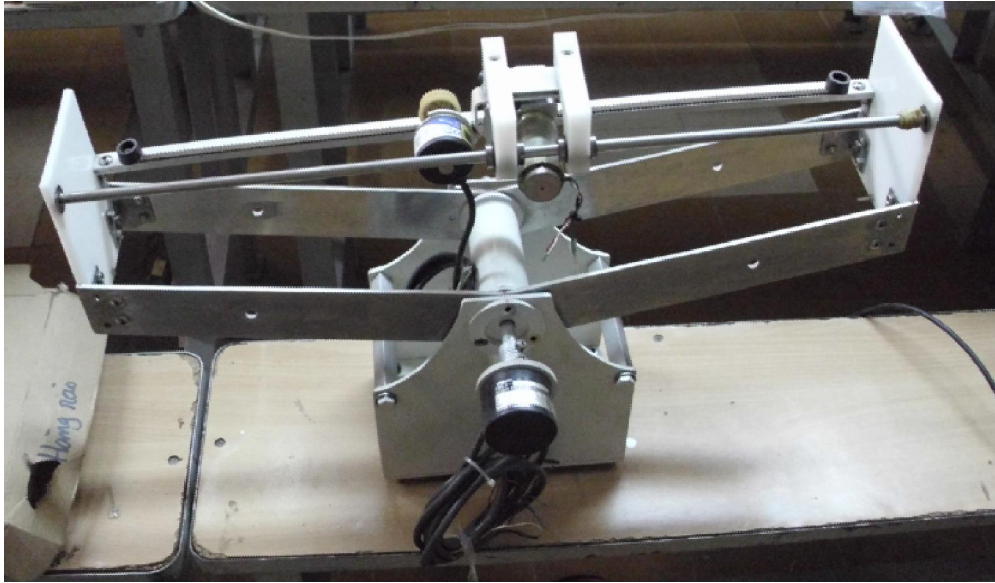
Hình 3.4 (c)



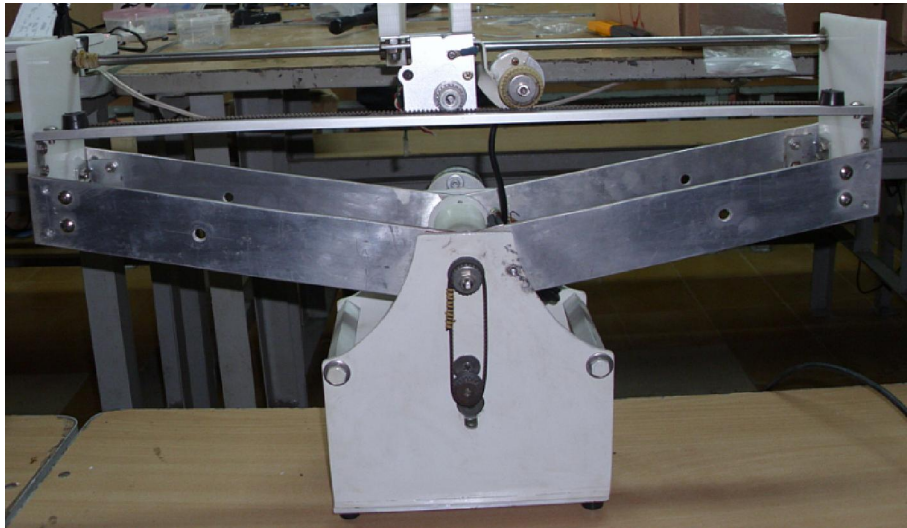
Hình 3.4 (d)



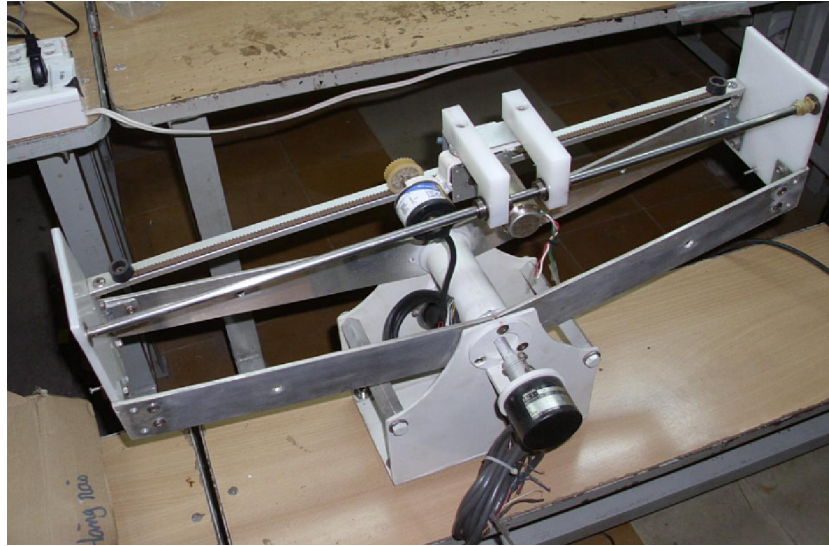
Hình 3.4 (e)



Hình 3.4 (f)



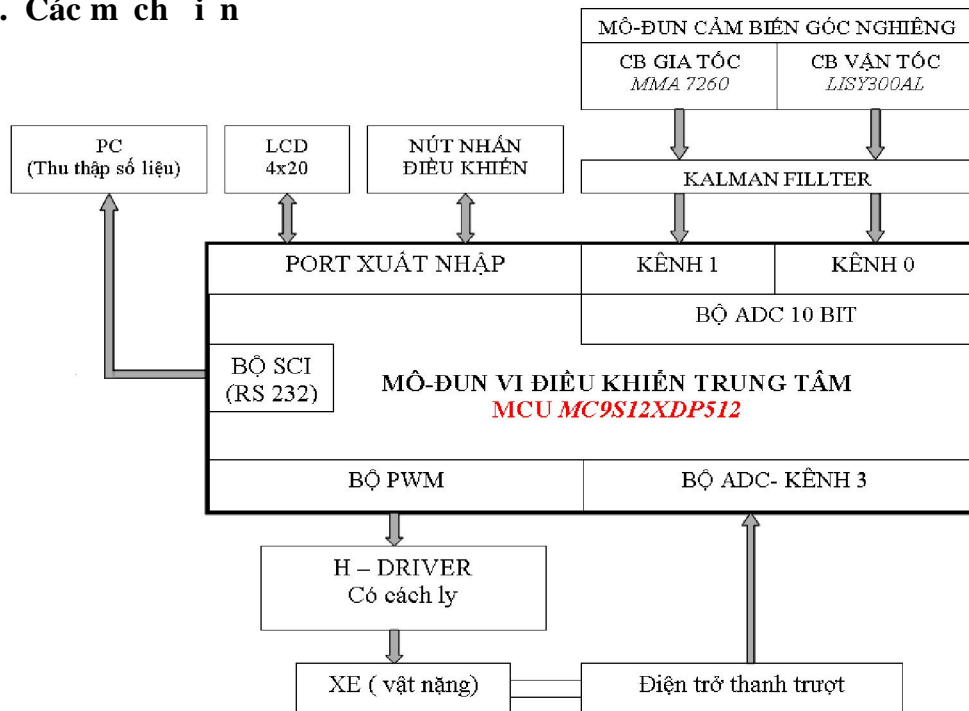
Hình 3.4 (g)



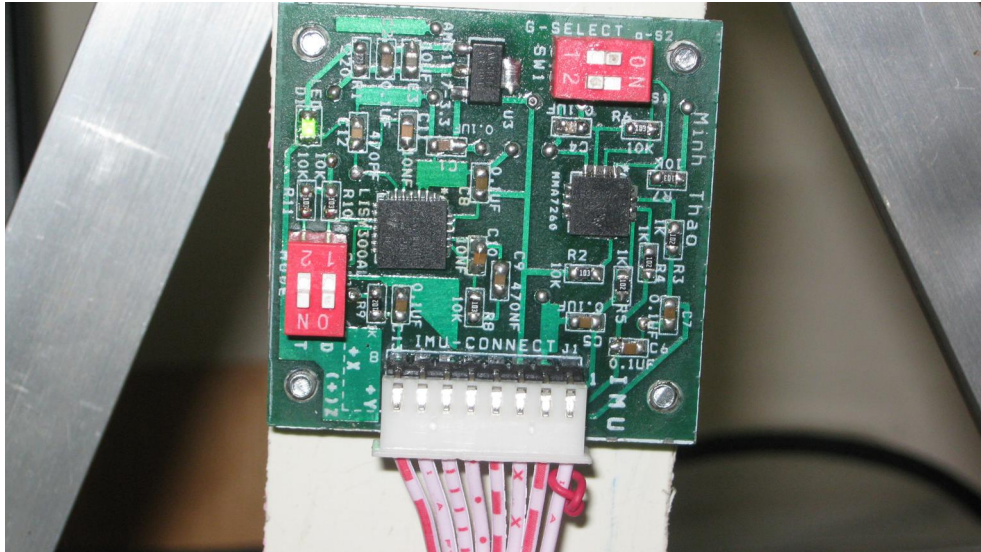
Hình 3.4 (h)

Hình 3.4 (a),(b),(c),(d),(e),(f),(g),(h): Các Mô hình con nôm ng c th c nghi m ã thi công

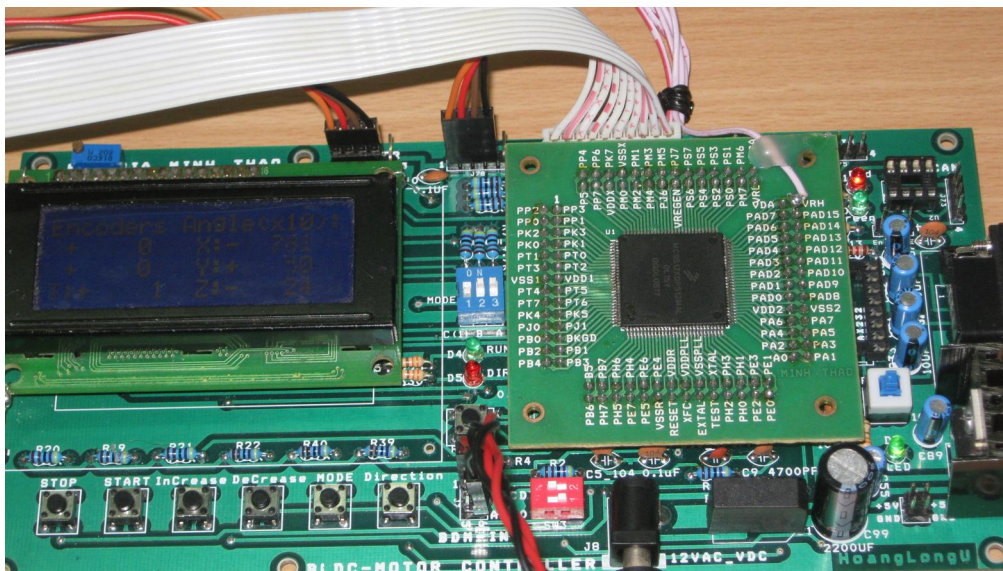
3.2.3. Các m ch i n



Hình 3.5: S k h i b i u khi n



Hình 3.6: Board mạch microcontroller góc nghiêng



Hình 3.7: Board mạch microcontroller

✓ **NH N XÉT:**

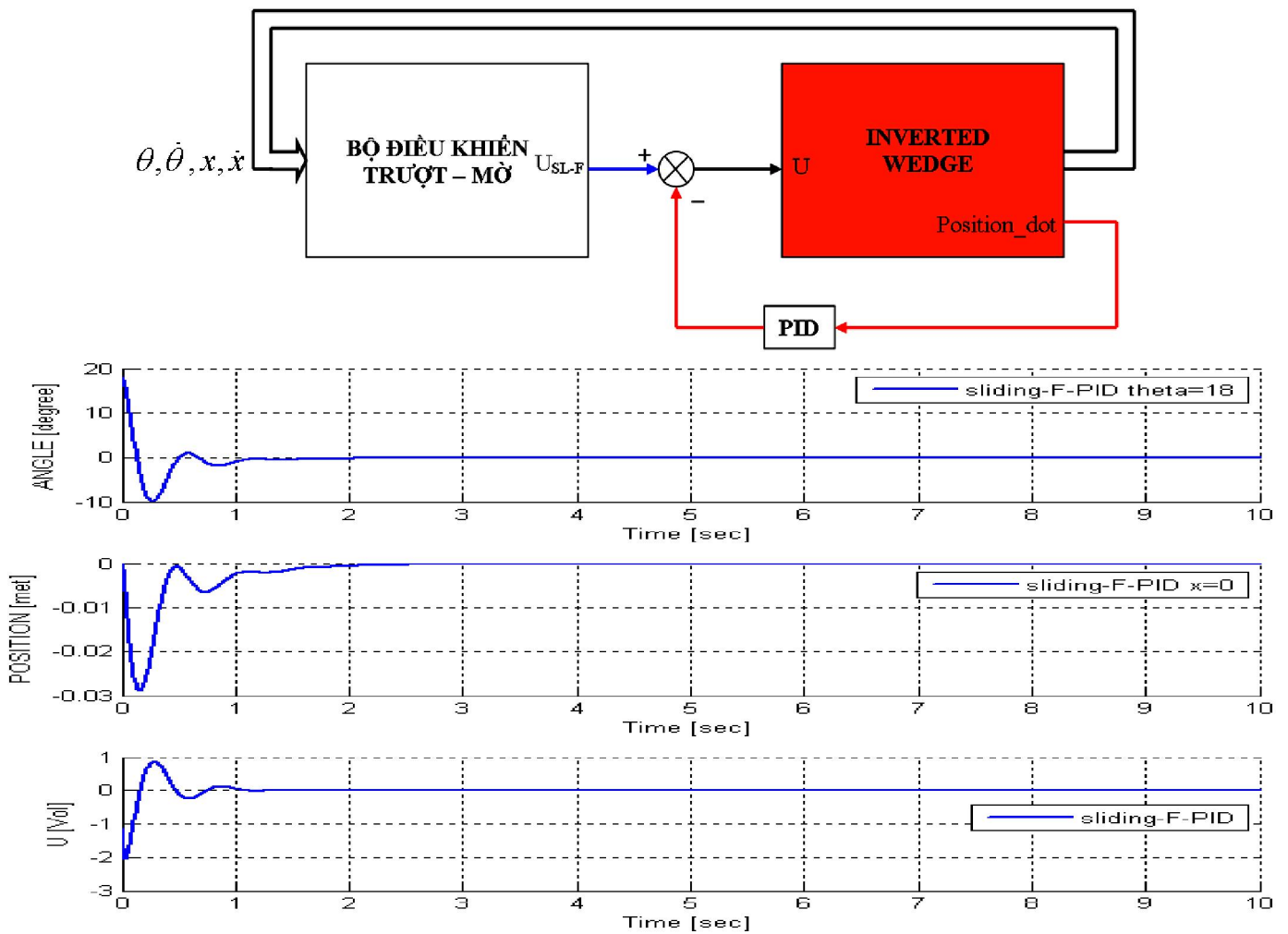
- Hình con nệm ng c là m t mô hình có tính phi tuyến phức tạp và r t khó i u khi n. Trong quá trình nghiên c u, tác gi ã thi t k , thi công và c i ti n l n l t 3 mô hình c khí con nệm ng c th c nghi m, các mô hình c khí và các board m ch i n c thi t k ch c ch n, b n, p.
 - + Mô hình 1 (hình 3.4a-b): thi t k ch c ch n b ng v t li u nh a, tuy nhiên mô hình quá n ng, vi c s d ng ng c DC k t h p dây ai kéo v t n ng làm h th ng rung l c m nh r t khó i u khi n cân b ng.
 - + Mô hình 2 (hình 3.4c-d-e): thi t k ch c ch n, mô hình s d ng ng c DC t o thành xe di chuy n trên m t ngang c a con nệm ng c t o thành v t n ng gi cân b ng con nệm. Tuy nhiên mô hình v n còn khá n ng làm quán tính l n, gây khó kh n cho vi c i u khi n.
 - + Mô hình 3 (hình 3.4f-g-h): thi t k ch c ch n b ng nhôm, mô hình g n nh . c thi t k cho vi c i u khi n b ng Card Real-Time giao ti p máy tính và Matlab/Simulink.
- Mô hình th c nghi m ã hoàn thành và ã ho t ng, tuy nhiên v n còn x y ra l i khâu thi công c khí làm cho vi c l p trình i u khi n r t khó kh n, và mô hình ã ho t ng nh ng không c t t nh mong mu n.

PHẦN IV

KẾT QUẢ THỰC NGHIỆM

4.1. Kết quả thực nghiệm

Tác giả đã thiết kế mô phỏng thành công bộ điều khiển trượt-M-PID nhằm khắc phục hiện tượng chattering của bộ điều khiển trượt.



Hình 3.8: kết quả thực nghiệm của bộ điều khiển trượt-M-PID

4.2. Kiến nghị

Hệ con nôm ngữ là hình thức phi tuyến phức tạp, rất khó hiểu khi nói, và đòi hỏi phải có thiết kế kỹ thuật chính xác cao, sử dụng linh kiện hiện đại. Hơn nữa, tài liệu tham khảo về phần thi công thực nghiệm là rất ít, vì thế đã gây rất nhiều khó khăn cho ngành thiết kế nghiên cứu.

- Tiếp tục nghiên cứu tìm kiếm những nhà hình thức.
- Nghiên cứu các tính mô hình các kỹ thuật con nôm ngữ.
- Nghiên cứu hiểu khi nói con nôm ngữ để cân bằng trên các ứng dụng Card Real-Time giao tiếp máy tính và Matlab/Simulink.

PH N V**BÁO CÁO KINH PHÍ ã S D NG TRONG TÀI***B ng 4.1: Báo cáo kinh phí ã s d ng trong tài*

STT	N i Dung	D toán	Quy t toán
I	Thuê khoán chuyên môn	10.000.000	10.000.000
1	Gia công c khí	10.000.000	10.000.000
II	Nguyên v t li u, n ng l ng	52.600.000	52.600.000
1	V t li u kim lo i	600.000	600.000
2	Thi t b truy n ng	3.000.000	3.000.000
3	ng c DC	6.000.000	6.000.000
4	Encoder	9.000.000	9.000.000
5	C m bi n góc	26.000.000	26.000.000
6	Board ng	200.000	200.000
7	i n tr	100.000	100.000
8	Diode	200.000	200.000
9	T i n	100.000	100.000
10	IC s	200.000	200.000
11	IC vi i u khi n	7.000.000	7.000.000
12	LCD 16x2	200.000	200.000
III	Xây d ng s a ch a nh	1.000.000	1.000.000
1	S a ch a sai sót khâu thi t k và thi công	1.000.000	1.000.000
IV	Chi khác	17.200.000	15.700.000
1	Xây d ng c ng chi ti t	700.000	700.000
2	Qu n lý c s	6.000.000	6.000.000
3	Chi phí nghi m thu c p khoa	1.520.000	0
4	Chi phí nghi m thu c p tr ng	2.500.000	2,150,000
5	n loát tài li u, v n phòng ph m	500.000	500.000
6	Ph c p ch nhi m T	6.000.000	6.000.000
	T ng C ng (I + II + III +IV)	80.800.000	79.300.300

TÀI LI U THAM KH O

- [1] Nguyễn Thị Ph ng Hà (2009), *Lý thuy t i u khi n hi n i*, NXB H Qu c Gia, Thành Ph H Chí Minh.
- [2] Nguyễn Thị Ph ng Hà, Huỳnh Thái Hoàng (2003), *Lý thuy t i u khi n t ng*, NXB H Qu c Gia, Thành Ph H Chí Minh.
- [3] Huỳnh Thái Hoàng (2006), *H th ng i u khi n thông minh*, NXB H Qu c Gia, Thành Ph H Chí Minh.
- [4] Đặng Hoài Ngh a (2007), *i u khi n h th ng a bi n*, NXB H Qu c Gia, Thành Ph H Chí Minh.
- [5] Nguyễn Doãn Ph c, Phan Xuân Minh (1997), *Lý thuy t i u khi n m*, NXB Khoa h c và K thu t.
- [6] Nguyễn Doãn Ph c, Phan Xuân Minh, Hán Thành Trung (2003), *Lý thuy t i u khi n phi tuy n*, NXB Khoa h c và K thu t.
- [7] Nguyễn Phùng Quang (2006), *Matlab và Simulink dành cho k s i u khi n t ng*, NXB Khoa h c và K thu t.
- [8] Nguyễn c Thành (2004), *Matlab Và ng D ng Trong i u Khi n*, NXB H Qu c Gia, Thành Ph H Chí Minh.
- [9] Võ Bá Thi (2009), *Nghiên c u gi i thu t i u khi n h con nê m ng c*, Luận v n t t nghi p th c s t ng hoá, i h c Bách Khoa, TP. H Chí Minh, Vi t Nam.
- [10] Chun-Hsien Tsai, Hung-Yuan Chung, Fang-Ming Yu, Jan (2004), “Neuro-Sliding Mode Control With Its Applications to Seesaw Systems”, *Neural Networks*, IEEE Transactions on, pp. 124 – 134.
- [11] Dianwei Qian, Jianqiang Yi, Dongbin Zhao (2007), “Hierarchical Sliding Mode Control to Swing up a Pendubot”, *IEEE - American Control Conference, ACC '07*, pp. 5254 – 5259.

- [12] Hagan M.T., Latino C.D., Misawa E., Young G. (2002), “An interdisciplinary control systems laboratory”, *Control Applications*, IEEE International Conference on, Proceedings of the 1996, pp. 403 – 408.
- [13] Jeng-Hann Li, Tzoo-Hseng S. Li, Ting-Han Ou, July (2003), “Design and Implementation of Fuzzy Sliding-Mode Controller for a Wedge Balancing System”, *Journal of Intelligent and Robotic Systems*, Volume 37 Issue 3, pp. 285-306.
- [14] J. Petri , Z. Situm (2003), "Pneumatic Inverted Wedge", *Symposium on Advances in Control Education Oulu, TACE2003 - 6th IFAC* , Finland, pp.1-7.
- [15] Kazuo Tanaka, Hua O.Wang (2001), *Fuzzy Control Systems Design and Analysis*, John Wiley & Sons, Inc.
- [16] Moore M.L., Musacchio J.T., Passino K.M. (2002), “Genetic Adaptive Control for an Inverted Wedge: Experiments and Comparative Analyses”, *IEEE - American Control Conference*, Proceedings of the 1999 (vol 1), pp. 400 – 404.
- [17] Shing-Jen Wu, Cheng-Tao Wu, Yung-Yi chiou, Chin-Teng Lin, Yi-Nung Chung (2006), “Balancing Control of Sliding Inverted-Wedge System: classical-method-based compensation”, *Systems Man and Cybernetics*, IEEE International Conference on, SMC '06, pp. 1349 – 1354.

Trang web

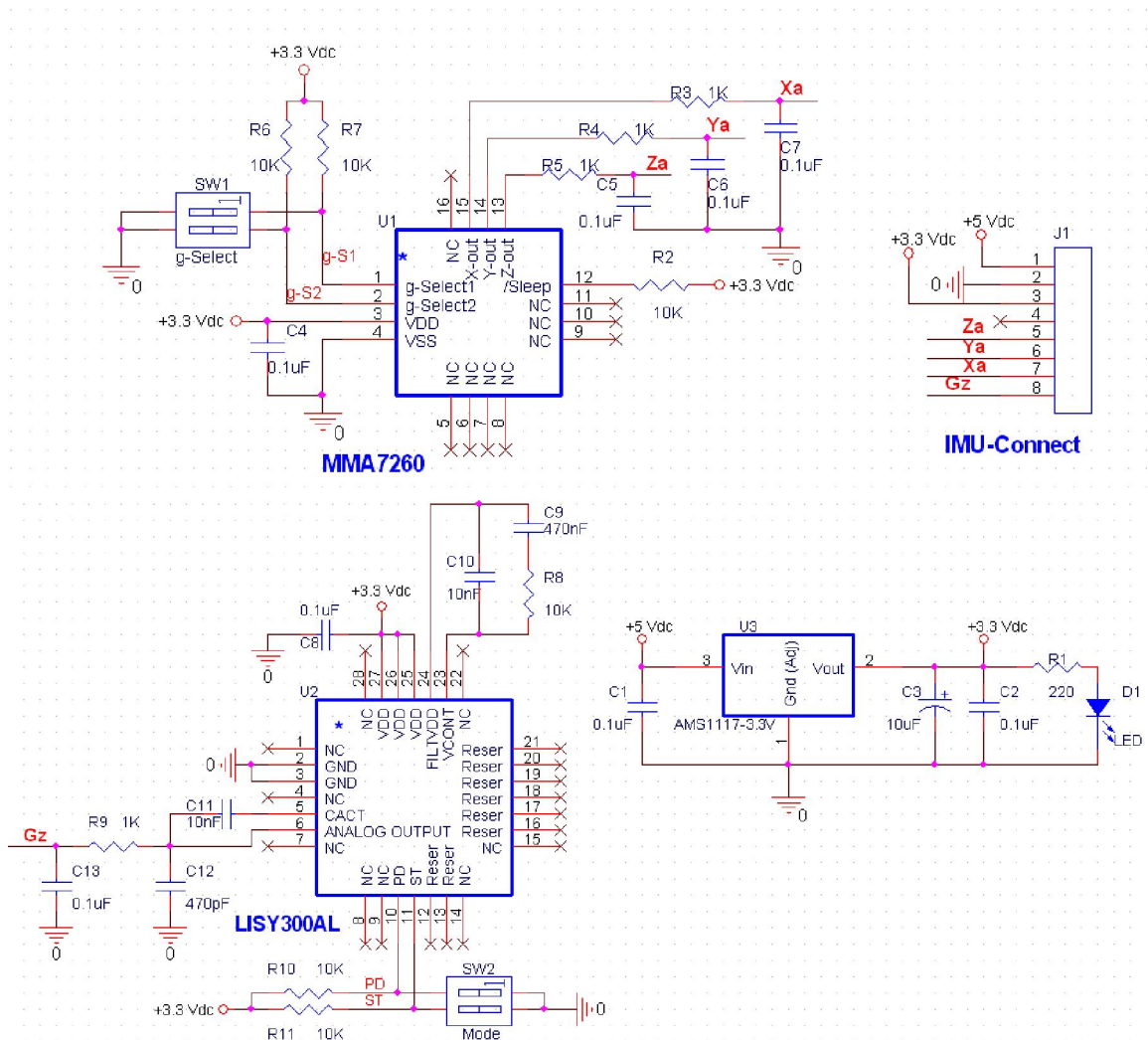
- [18] http://www.fsb.unizg.hr/acg/inv_wedge_model.html
- [19] <http://www.freescale.com>
- [20] <http://www.unizg.hr>
- [21] <http://www-srt.upb.de/Rundgang/pendel.html>
- [22] http://www.quanser.com/english/html/challenges/fs_chall_linear_flash.htm
- [23] http://www.fsb.unizg.hr/acg/inv_wedge_model.html

PH L C 1

CÁC S M CH I N

1. M ch c m bi n góc nghiêng

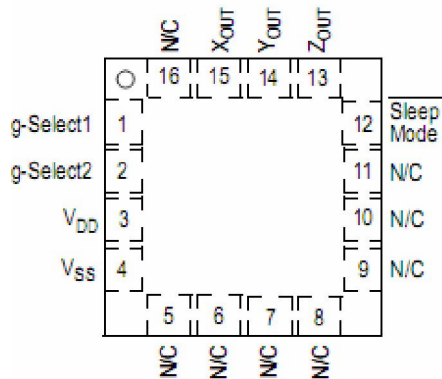
M ch c m bi n góc nghiêng là s k t h p gi a m t c m bi n gia t c góc MMA7260 c a h ăng FreeScale và m t c m bi n v n t c góc LISY300AL c a h ăng ST-Electronics .



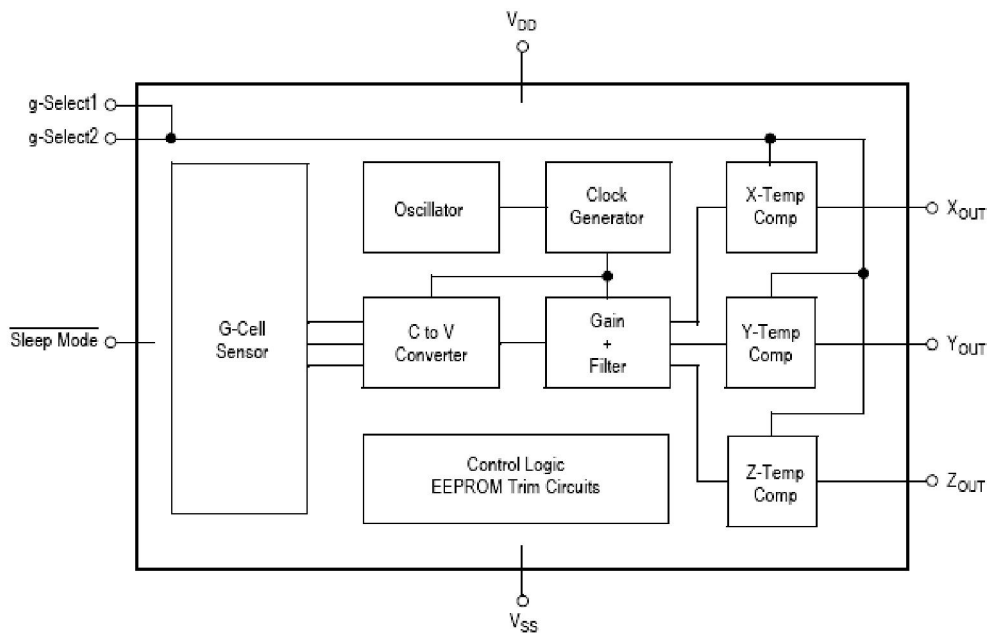
Ø C m bi n gia t c góc MMA72600 c a h ăng FreeScale, có các c i m sau:

- Có thể chọn các nhạy ứng với m (1.5g/2g/4g/6g).
- Dòng tiêu thụ 500uA và sleep mode là 3uA.
- Nhạy cao 800mv/g với ứng với m 1.5g.
- Thời gian khởi động nhanh.
- Tích hợp sẵn bộ lọc thông thấp, và có khả năng chống nhiễu cao.

Top View



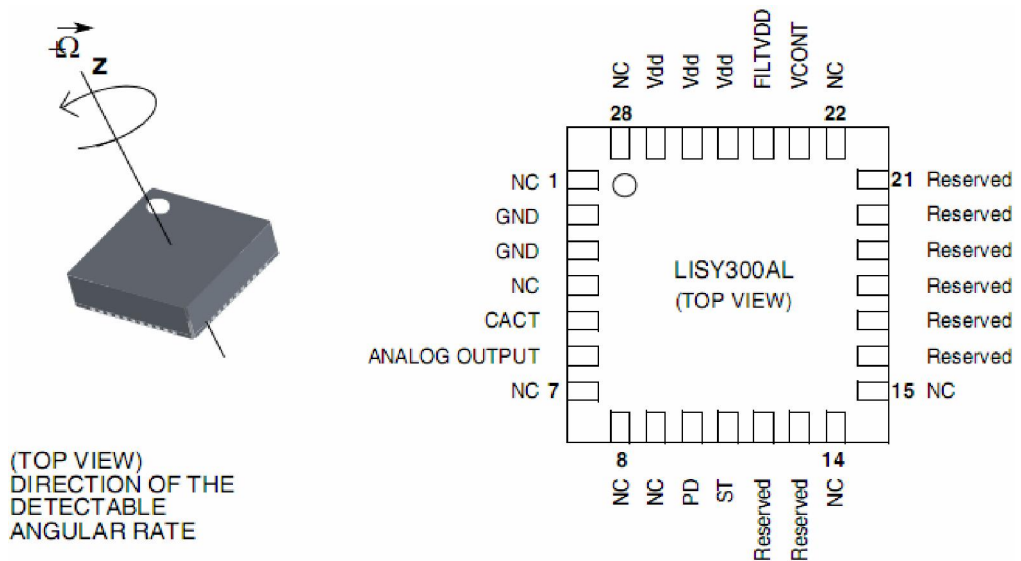
- G1-G2: chọn mức độ nhạy.
- SLEEP: chọn chế độ.
- Xout, Yout, Zout: ngõ ra các trục tọa độ XYZ.



nh y và t m o c a c m b i n c xác nh nh b ng sau

g-Select2	g-Select1	g-Range	Sensitivity
0	0	1.5g	800mV/g
0	1	2g	600mV/g
1	0	4g	300mV/g
1	1	6g	200mV/g

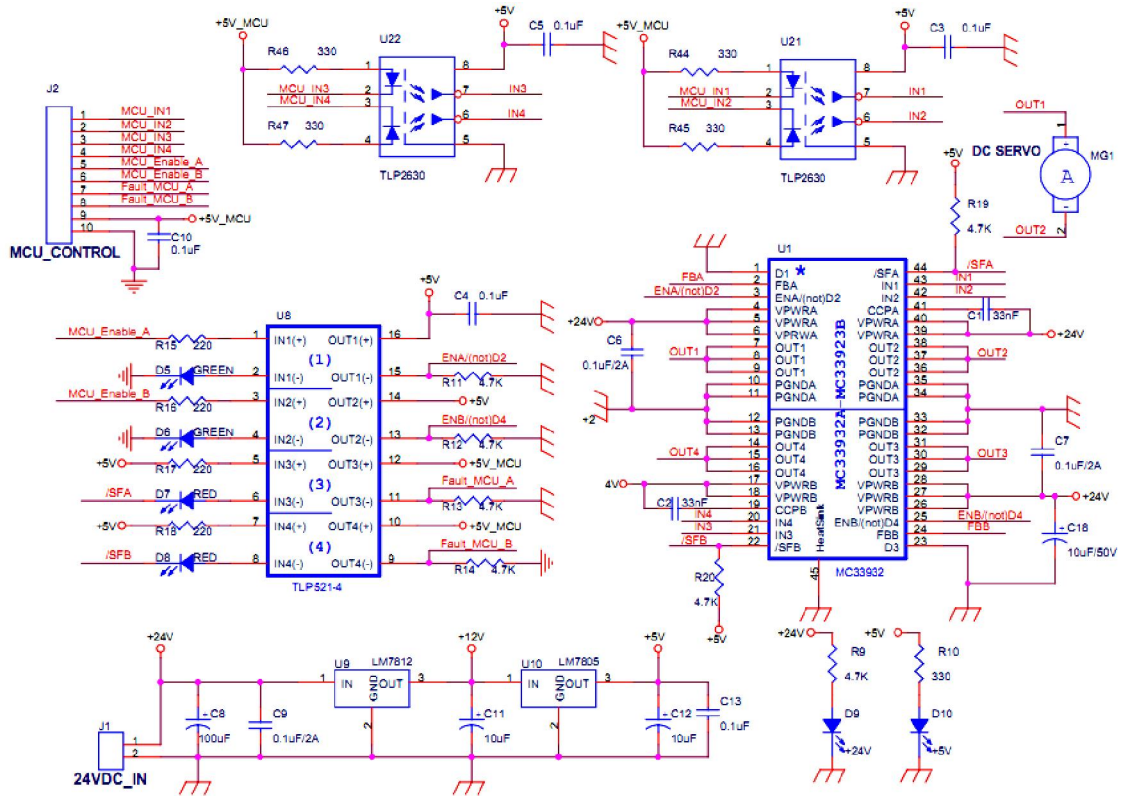
Ø c tính c a c m b i n v n t c góc LISY300AL, hãng ST-Electronics.



- Ho t ng ngu n t 2.7V n 3.6V. T m nhi t ho t ng: -40°C n $+85^{\circ}\text{C}$.
- Tiêu th i n n ng th p.
- Tích h p b gi m áp.
- o v n t c góc quanh tr c z.
- Toàn t m o là : $\pm 300^{\circ}/\text{s}$, kh n ng o l ng là -3 dB, b ng thông là 88 Hz.

- Ngõ ra là tín hiệu tương tự.
- Độ nhạy là : 3.3 mV/°/s.
- Tích hợp bộ lọc thông thấp.
- Tích hợp mô-đun kiểm tra. Khả năng chống nhiễu cao.

2. Mạch uH công suất iu khi nng c DC có cách ly



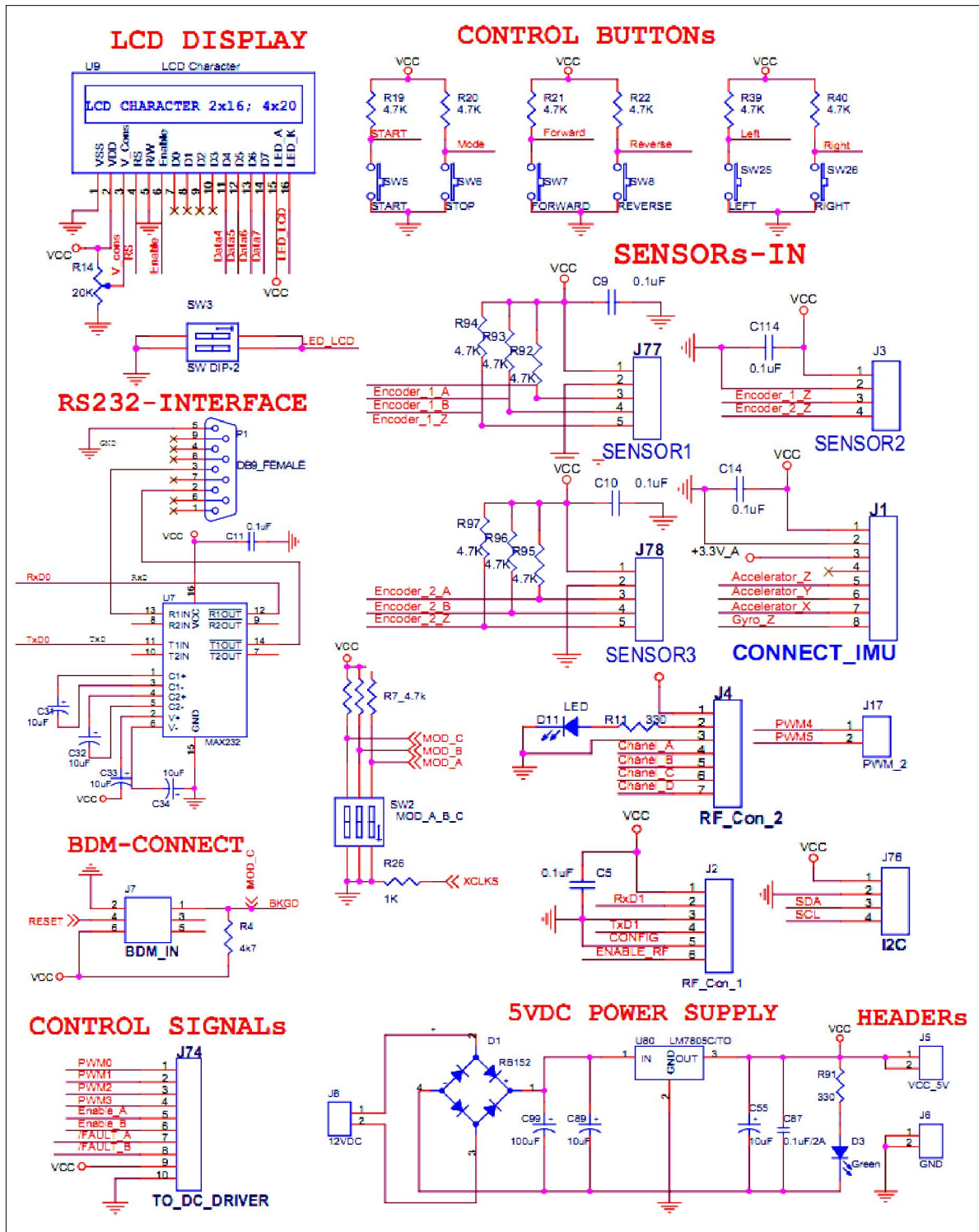
- IC MC33932 chứa 2 kênh uH độc lập với nhau, mỗi kênh uH có chức năng như sau:
 - Tầm áp hoạt động là : 8Vdc đến 28Vdc và vị trí ngưỡng dòng liên tục, 5V đến 40V và vị trí ngưỡng dòng gián đoạn.
 - Trường thích vị mức tín hiệu iu khi n TTL (3V , 5V).

- Có bo v ng n m ch ngõ ra, quá dòng, quá nhi t.
 - T m nhi t ho t ng (v i t n nhi t c có s n) là t -40°C n 125°C .
 - Dòng i n t i trung bình là 5A , dòng i n t i nh là 8A..
 - T n s i u r ng xung t i a là 20 Khz.
- M ch ng l c này c cách ly v i m ch vi i u khi n trung tâm b ng hai opto t c cao TLP2630 (cách ly cho các tín hi u i u r ng xung) và opto TLP521 (cách ly các tín hi u i u khi n cho phép IC MC33932 c tích c c,...).

3. M ch vi i u khi n trung tâm

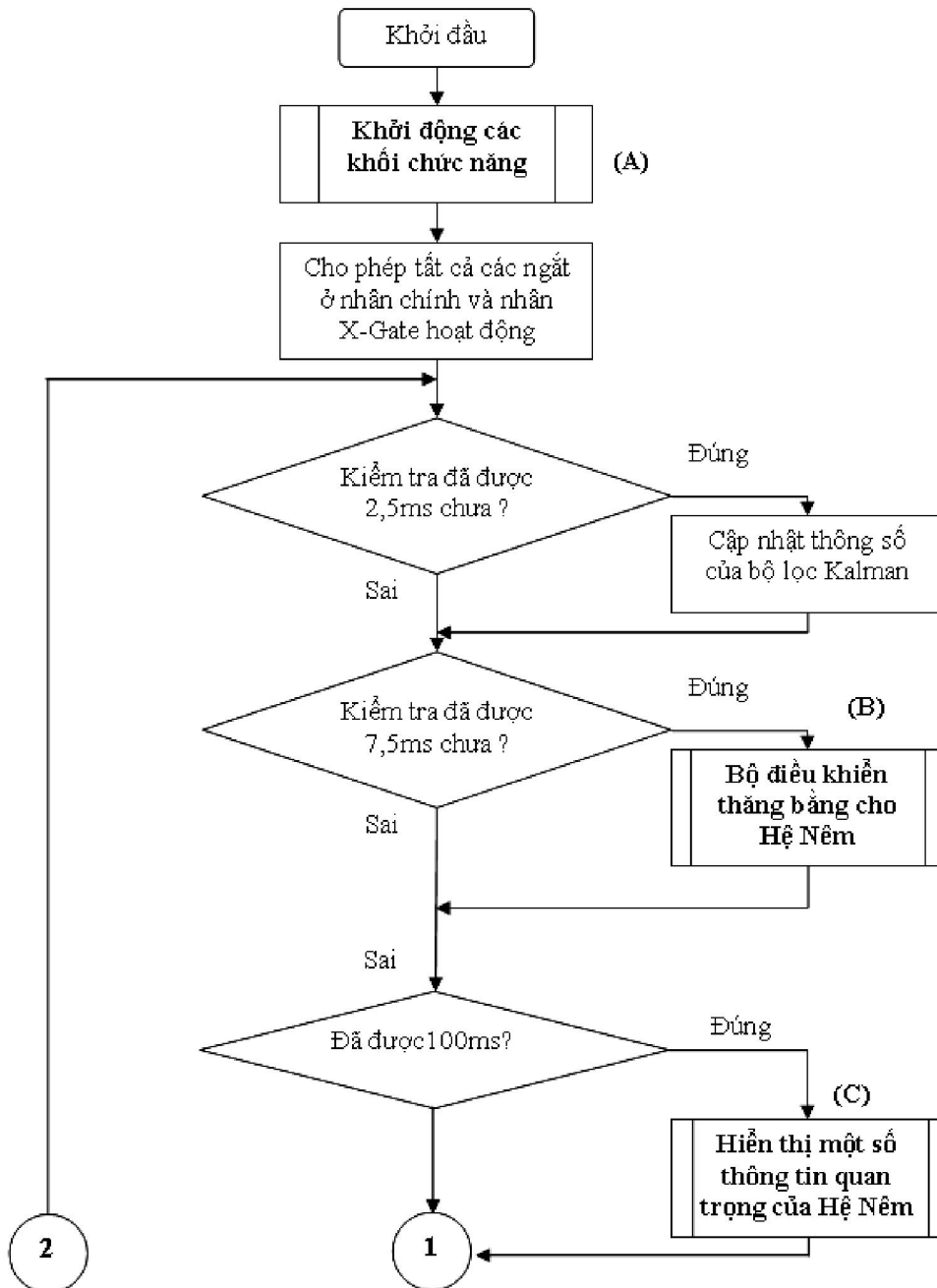
Vi i u khi n s d ng có tên là MC9S12DP512 c a hãng FreeScale Semiconductor, thu c dòng D c a h HCS12, là dòng c phát tri n cho các ng d ng trong l nh v c thu c công ngh ô tô và trong công nghi p. Có y các ch c n ng c b n chung c a h vi i u khi n 16-bit HCS12 (Motorola).

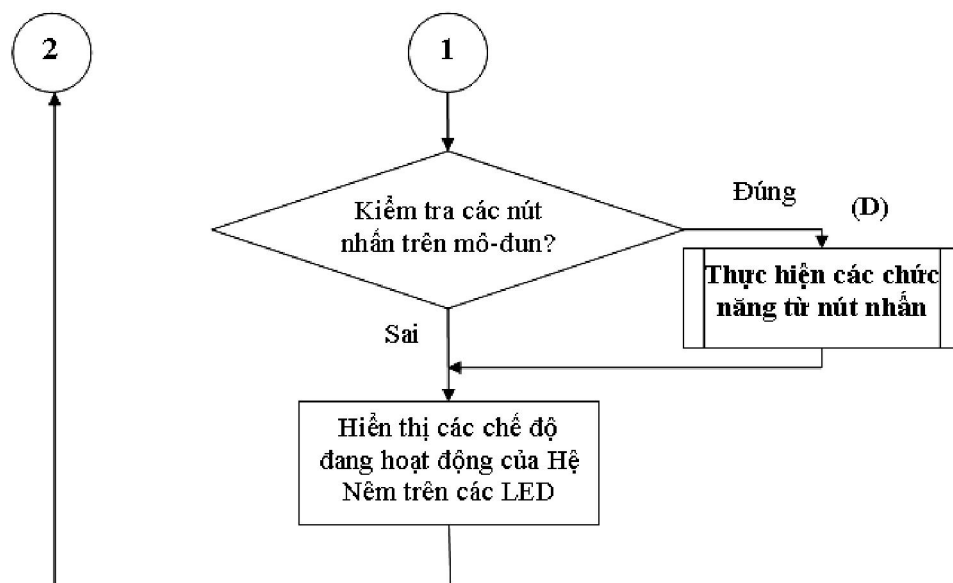
- c óng gói theo ki u chân LQFP – có 112 chân, t m nhi t là: -40 n 150 C.
- T n s xung nh p bus n i t i a là 32 Mhz, có tích h p b PLL.
- B nh : 14 KB b nh RAM ; 4 KB b nh EEPROM ; 512 KB b nh FLASH.
- 2 b giao ti p n i ti p b t ng b (SCI), 3 b giao ti p n i ti p ng b (SPI), 1 b IIC.
- 5 b CAN, h tr c 2 chu n V2.0A và V2.0B; 1 b giao ti p BDLC (theo chu n J1850).
- 8 kênh ch c n ng th i gian nh : B t gi ngõ vào, so sánh ngõ ra, m xung-s ki n c ng, ng t th i gian th c, WatchDog ...
- 2 b ADC -16 kênh , phân gi i10-bit ; 8 kênh trong mô- un i u r ng xung (PWM).



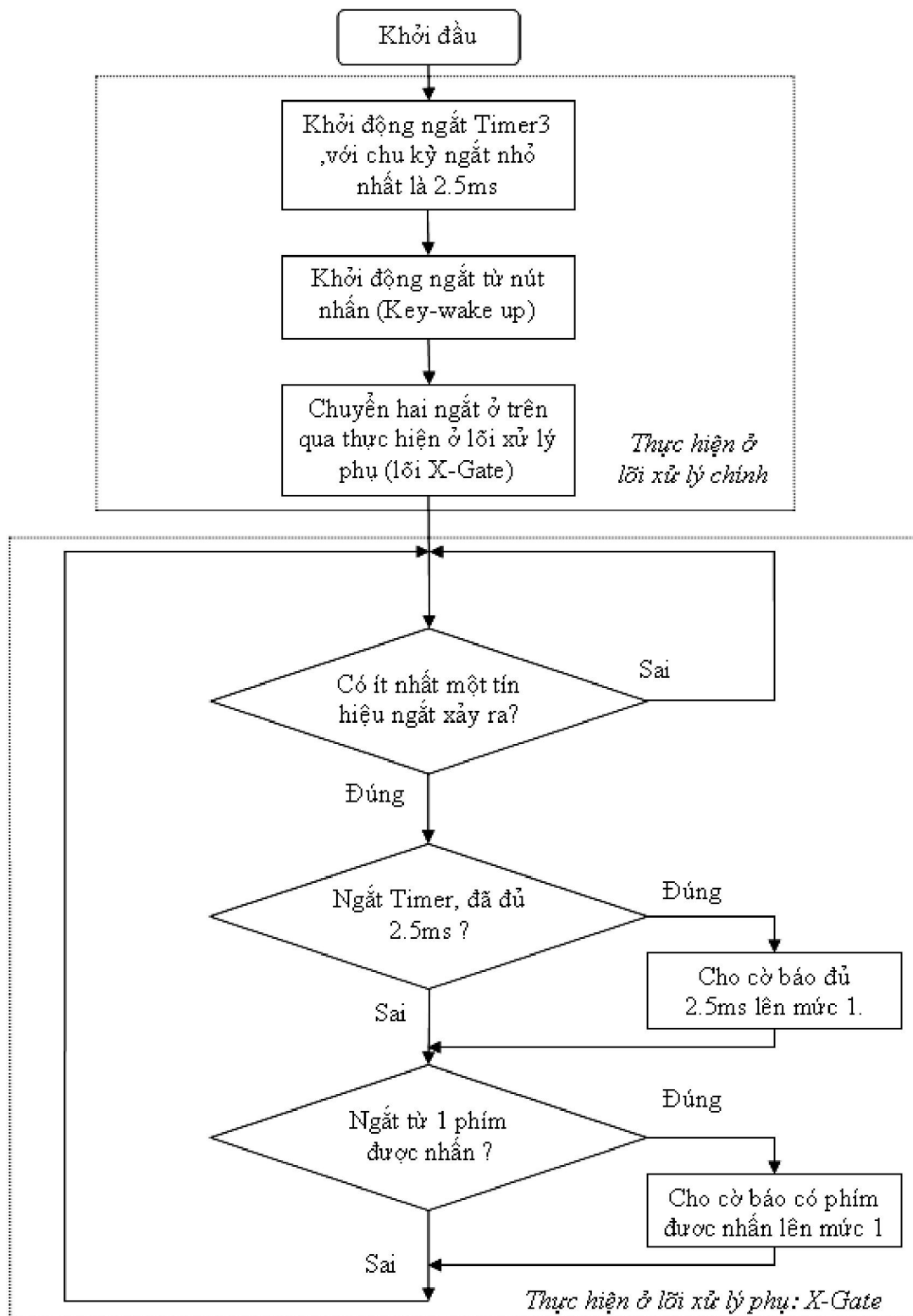
PHẦN 2 CHƯƠNG TRÌNH I U KHI N

1. Giới thiệu chương trình chính





2. **Giới thiệu thuật toán trình ngát**



3. Chương trình i u khi n

```

/*
1> CLOSE-LOOP with Capture Encoder's signals --> measure Motor's Speed
   Used A-B signals, had not used Z-signal yet.

2> MC33932 Driver for two DC-Servo Motors

3> Use Kalman Filter to predict and measure angle: V_ref(ADC) = 3.3 Volt --> OK
   (angle_Y)

*/

#include <hidef.h> /* common defines and macros */
#include <mc9s12xdp512.h> /* derivative information */
#pragma LINK_INFO DERIVATIVE "mc9s12xdp512"

#include <string.h>
#include <math.h>
#include "xgate.h"

#include "pll_2.h" // for X-GATE , 40Mhz
#include "lcd.h"
#include "SCI0.H"
#include "Timer.h"
#include "ADC.h"

//----- GLOBAL VARAIBLE -----//
unsigned char OK_flag=0,Key, LCD_flag=0, mode=0, flag_5ms, flag_250ms,
count_turn=0, count_forward=0, count_reverse=0
,START_flag, increase_flag, decrease_flag, mode_flag,
direction_flag_1,direction_flag_2, count_4
,reverse_flag=0, forward_flag=0, turn_flag=0, direction_flag=0, stop_flag=0;

long Torque_1, Torque_2, speed_1_10ms, speed_2_10ms, count_stop, j;
float position_X, position_X_center, speed_wheel_1, u_signal, u_angle;
float pulses_1, pulses_2, omega_degree, angle_error_old, angle_degree, angle_error,
set_speed, sigma_position_error
, speed_error, position_error, position_error_old, set_angle, set_position,
count_pulse_1, count_pulse_2, u_position;

```

```

/**----- Gains of PID Controller for ANGLE -----**//
float Kp = 15;      // For Angle
float Ki = 0;      // For Sigma (Angle)
float Kd = 0.5;    // For Omega (Angle)

/**----- PID Controller for POSITION: PARAMATERS -----**//
float Kp_X = 15 ;   // For Position
float Ki_X = 0 ;
float Kd_X = 0.5 ;

float sigma_error;
long u_desire;
//-----//

//----- IW's paramaters -----//
long Encoder_revol = 1000 ;    // 2*1000 pulses/s = 2000 pulses/s
int Kb = 1 ;                  // Wheel(1) <--> Motor(8), Kb = 125mm/12.5mm = 10
float d_wheel = 0.027 ;      // 2(cm) = 0.2(m)
float g=9.81;                //
float pi = 3.1416;           // pi
//-----//
//*****-----**//

/**----- Constant of Kalman Filter -----**//
float offset_acc_X = 517;    // 516 (calibed) <=> 1.65 Volt/3.3 Volt ; 3.3V <-->
1023 , so that 1.65V <--> 512 (10-bit ADC)
float offset_acc_Y = 505;    // <===== NOTE
float offset_acc_Z = 517;
float offset_gyro= 500;     // 3.3V <--> 1023 , so that 1.65V <--> 502 (calibed) ;
(10-bit ADC)

float scale_acc= 186 ;      // Sensity is 0.6V/g at 2g: 3.3V <--> 1023, so that 0.6V
<--> 186 (10-bit ADC)
float scale_rot= 1.023;     // 3.3mV/degree/sec
// <--> 1.023/(pi/180) = 1.023 * 57.3 ( /rad/sec)
static float dt = 0.0025;   // Sampling-Time is 2.5ms
static float R_angle = 0.08; // <<===== NOTE
static float Q_angle = 0.003;
static float Q_gyro = 0.001;

```



```

//----- Global Kalman filter 's variables -----//
static float P[2][2]={{1,0},{0,1}};
static float angle=0;
static float q_bias=0;
static float rate=0;

float X, Y, Z, ROT_Z, mes_X, mes_Z, angle_m, q_m ;
// -----//

/* this variable definition is to demonstrate how to share data between XGATE and
S12X */
#pragma DATA_SEG SHARED_DATA
volatile int shared_counter; /* volatile because both cores are accessing it. */

//-----Variable use for both MCU & XGATE <<-----
char button_flag, flag_2_5ms, flag_7_5ms, flag_10ms, count, count_2, count_3 ;
//-----//

#pragma DATA_SEG DEFAULT
//-----//

/***** Macros
*****/

#define ROUTE_INTERRUPT(vec_adr, cfdata) \
    INT_CFADDR= (vec_adr) & 0xF0; \
    INT_CFDATA_ARR[((vec_adr) & 0x0F) >> 1]= (cfdata)

#define SCI0_VEC 0xD6 // vector address SCI0= $xxD6 <<-----

#define TIC0_VEC 0xEE // vector address TIC0= $xEE <<-----
#define TIC1_VEC 0xEC // vector address TIC1= $xEC <<-----
#define TIC2_VEC 0xEA // vector address TIC2= $xEA <<-----
#define TIC5_VEC 0xE4 // vector address TIC5= $xE4 <<-----
#define TIC6_VEC 0xE2 // vector address TIC6= $xE2 <<-----
#define TIC7_VEC 0xE0 // vector address TIC7= $xE0 <<-----

#define T0C3_VEC 0xE8 // vector address T0C3= $xE8 <<-----
#define T0C4_VEC 0xE6 // vector address T0C4= $xE6 <<-----

#define PTH_VEC 0xCC // vector address PTH= $xE8 <<-----

```

```

//-----//

/**----- Set up XGATE -----**//
#define SOFTWARETRIGGER0_VEC 0x72 /* vector address= 2 * channel id */
static void SetupXGATE(void) {
    /* initialize the XGATE vector block and
       set the XGVBR register to its start address */
    XGVBR= (unsigned int)(void*__far)(XGATE_VectorTable -
XGATE_VECTOR_OFFSET);

    /* switch software trigger 0 interrupt to XGATE */
    ROUTE_INTERRUPT(SOFTWARETRIGGER0_VEC, 0x81); /* RQST=1 and
PRIO=1 */

    ROUTE_INTERRUPT(T0C3_VEC, 0x81); // switch T0C3 interrupt to XGATE <<-
-----
    ROUTE_INTERRUPT(T0C4_VEC, 0x81); // switch T0C4 interrupt to XGATE <<-
-----
    ROUTE_INTERRUPT(PTH_VEC, 0x81); // switch PTH interrupt to XGATE <<--
-----

    /* enable XGATE mode and interrupts */
    XGMCTL= 0xFBC1; /* XGE | XGFRZ | XGIE */

    /* force execution of software trigger 0 handler */
    XGSWT= 0x0101;
}
//-----//

/* *****
* KALMAN FILTER: output is value of ANGLE *
*****/

/*----- Pitch gyro measurement -----*/
void state_update( const float q_m)
{
    /* Unbias our gyro */
    float q = q_m - q_bias;

    /* Compute the derivative of the covariance matrix
    *

```

```

* Pdot = A*P + P*A' + Q
*
* We've hand computed the expansion of A = [ 0 -1, 0 0 ] multiplied
* by P and P multiplied by A' = [ 0 0, -1, 0 ]. This is then added
* to the diagonal elements of Q, which are Q_angle and Q_gyro.
*/
float Pdot0 = Q_angle - P[0][1] - P[1][0]; /* 0,0 */
float Pdot1 = -P[1][1]; /* 0,1 */
float Pdot2 = -P[1][1]; /* 1,0 */
float Pdot3 = Q_gyro; /* 1,1 */
/* Store our unbiased gyro estimate */
rate = q;

/* Update our angle estimate
* angle += angle_dot * dt
* += (gyro - gyro_bias) * dt
* += q * dt
*/
angle += q * dt;

/* Update the covariance matrix */
P[0][0] += Pdot0 * dt;
P[0][1] += Pdot1 * dt;
P[1][0] += Pdot2 * dt;
P[1][1] += Pdot3 * dt;
}
/*-----*/

/*----- Update value of ANGLE in Kalman Filter -----*/

void kalman_update(const float angle_m) // X acc
{
/* Compute our measured angle and the error in our estimate */
float angle_err = angle_m - angle;

/*
* C_0 shows how the state measurement directly relates to
* the state estimate.
*
* The C_1 shows that the state measurement does not relate
* to the gyro bias estimate. We don't actually use this, so
* we comment it out.
*/

```

```

*/
float  C_0 = 1;

/*
* PCt<2,1> = P<2,2> * C'<2,1>, which we use twice. This makes
* it worthwhile to precompute and store the two values.
* Note that C[0,1] = C_1 is zero, so we do not compute that
* term.
*/
float  PCt_0 = C_0 * P[0][0];          /* + C_1 * P[0][1] = 0 */
float  PCt_1 = C_0 * P[1][0];          /* + C_1 * P[1][1] = 0 */

/* Compute the error estimate. From the Kalman filter paper:
*
* E = C P C' + R
*
* Dimensionally,
*
* E<1,1> = C<1,2> P<2,2> C'<2,1> + R<1,1>
*
* Again, note that C_1 is zero, so we do not compute the term.
*/
float  E = R_angle + C_0 * PCt_0;      /* + C_1 * PCt_1 = 0 */

/*
* Compute the Kalman filter gains. From the Kalman paper:
*
* K = P C' inv(E)
*
* Dimensionally:
*
* K<2,1> = P<2,2> C'<2,1> inv(E)<1,1>
*
* Luckily, E is <1,1>, so the inverse of E is just 1/E.
*/
float  K_0 = PCt_0 / E;
float  K_1 = PCt_1 / E;

/*
* Update covariance matrix. Again, from the Kalman filter paper:
*
* P = P - K C P

```

```

*
* Dimensionally:
*
*  $P_{\langle 2,2 \rangle} = K_{\langle 2,1 \rangle} C_{\langle 1,2 \rangle} P_{\langle 2,2 \rangle}$ 
*
* We first compute  $t_{\langle 1,2 \rangle} = C P$ . Note that:
*
*  $t_{[0,0]} = C_{[0,0]} * P_{[0,0]} + C_{[0,1]} * P_{[1,0]}$ 
*
* But, since  $C_{-1}$  is zero, we have:
*
*  $t_{[0,0]} = C_{[0,0]} * P_{[0,0]} = PC_{t[0,0]}$ 
*
* This saves us a floating point multiply.
*/
float  t_0 = PCt_0; /* C_0 * P[0][0] + C_1 * P[1][0] */
float  t_1 = C_0 * P[0][1]; /* + C_1 * P[1][1] = 0 */

P[0][0] -= K_0 * t_0;
P[0][1] -= K_0 * t_1;
P[1][0] -= K_1 * t_0;
P[1][1] -= K_1 * t_1;

/*
* Update our state estimate. Again, from the Kalman paper:
*
*  $X += K * err$ 
*
* And, dimensionally,
*
*  $X_{\langle 2 \rangle} = X_{\langle 2 \rangle} + K_{\langle 2,1 \rangle} * err_{\langle 1,1 \rangle}$ 
*
*  $err$  is a measurement of the difference in the measured state
* and the estimate state. In our case, it is just the difference
* between the two accelerometer measured angle and our estimated
* angle.
*/
angle += K_0 * angle_err;
q_bias += K_1 * angle_err;

//----- Limitation -----//
// -90 <= angle <= 90 (degree)

```

```

if (angle > (pi/2)) angle = pi/2 ;
else if (angle < (-pi/2)) angle=-pi/2 ;
//-----//
}
/***** END OF KALMAN FILTER *****/

// -----PWM_Init-----//
/* Enable PWM channel 0->5 - Make 8Khz Frequency square , Left-Aligned

Note: Channel 0-1-4-5: use Clock A and Clcok SA
      Channel 2-3-6-7: use Clock B and Clock SB

period = PWMPERX * 2^(PCKA(B)+1) * PWMSCLA(B) (@@@) (when:
PWMCLK_PCLKX = 1: choose Clock SA(B) )
      200 * 2 * 20 = 8000
      ( Bus Clock is 40Mhz --> period = Bus_clock / f_square = 40Mhz / 5Khz = 8000 )

      Duty_cycle = PWMDTYx / PWMPERx
*/

void PWM_Init(void){
//----- Common -----//
  PWMCTL = 0x00 ;          // 8 Separate 8-bit PWM Channels
  PWMPRCLK = 0x00 ;       // PCKA = 0 ; PCKB = 0 :use for 8-Channels
  PWMSCLA = 20 ;          // for (@@@)
  PWMSCLB = 20 ;          // for (@@@)
//-----//

//----- PWM0 -----//
  PWMCLK_PCLK0 = 1 ;      // use Clock SA for PWM0-Channel
  PWMPOL_PPOL0 = 1 ;      // Polarity=1 --> start with High-Signal in PWM0
  PWMCAE_CAE0 = 0 ;       // PWM0 is Left-Aligned
  PWMPER0 = 200 ;         // Periode0 = 200 --> 5KHz of Square_Wave (@@@)
  PWME_PWME0 = 1 ;        // Enable PWM0
//-----//

//----- PWM1 -----//
  PWMCLK_PCLK1 = 1 ;      // use Clock SA for PWM0-Channel
  PWMPOL_PPOL1 = 1 ;      // Polarity=1 --> start with High-Signal in PWM1
  PWMCAE_CAE1 = 0 ;       // PWM1 is Left-Aligned
  PWMPER1 = 200 ;         // Periode0 = 200 --> 5KHz of Square_Wave (@@@)

```

```

PWME_PWME1 = 1 ;          // Enable PWM1
//-----//

//----- PWM2 -----//
PWMCLK_PCLK4 = 1 ;        // use Clock SB for PWM2-Channel
PWMPOL_PPOL4 = 1 ;        // Polarity=1 --> start with High-Signal in PWM2
PWMCAE_CAE4 = 0 ;        // PWM2 is Left-Aligned
PWMPER4 = 200 ;          // Periode2 = 200 --> 5KHz of Square_Wave (@@@)
PWME_PWME4 = 1 ;        // Enable PWM2
//-----//

//----- PWM3 -----//
PWMCLK_PCLK3 = 1 ;        // use Clock SB for PWM3-Channel
PWMPOL_PPOL3 = 1 ;        // Polarity=1 --> start with High-Signal in PWM3
PWMCAE_CAE3 = 0 ;        // PWM0 is Left-Aligned
PWMPER3 = 200 ;          // Periode3 = 200 --> 5KHz of Square_Wave (@@@)
PWME_PWME3 = 1 ;        // Enable PWM3
//-----//

//----- Initial Value -----//
PWMDTY0 = 0 ;            // 0% Hiduty is IN-1 of MC33932-A
PWMDTY1 = 0 ;            // 0% Hiduty is IN-2 of MC33932-A
PWMDTY4 = 0 ;            // 0% Hiduty is IN-3 of MC33932-B
PWMDTY3 = 0 ;            // 0% Hiduty is IN-4 of MC33932-B
//-----//
}
//-----//

//----- TOC3_Init -----//
// TOC3 Init
void TOC3_Init(void){
  TSCR1 = 0x80; // Enable TCNT, 40MHz boot mode
  TSCR2 = 0x04; // divide by 16 TCNT prescale.+ Disable TOF Interrupt

/* Bottom three bits of TSCR2 (PR2,PR1,PR0) determine TCNT period
   divide fast(24MHz)      slow (4MHz)      fastest: HCS12X(40Mhz)
000 1   42ns TOF 2.73ms  250ns TOF 16.384ms      25ns
001 2   84ns TOF 5.46ms  500ns TOF 32.768ms
010 4  167ns TOF 10.9ms  1us TOF 65.536ms
011 8  333ns TOF 21.8ms  2us TOF 131.072ms      200ns

```

```

100 16 667ns TOF 43.7ms 4us TOF 262.144ns 400ns <<-----
---

101 32 1.33us TOF 87.4ms 8us TOF 524.288ms 800ns
110 64 2.67us TOF 174.8ms 16us TOF 1.048576s
111 128 5.33us TOF 349.5ms 32us TOF 2.097152s */

TIOS = TIOS | 0x08 ; // PT3 is OutPut Compare : 0b00001000
TCTL2 = TCTL2 & ~0xC0 ; // Disconnected at pin PT3: 0b11000000
TFLG1 |= 0x08; // Clear Flag : 0b00001000
TIE = TIE | 0x08 ; // Enable TOC3 Interrupt : 0b00001000
TC3 = TCNT+6250; // At first : 6250 * 400ns = 2.5ms
}
//-----//

//----- TIC_Init -----//
void TIC_Init(void){
    TIOS &= ~0b11100111 ; // PT0-1-2 and PT5-6-7 are Input Captures : for
HALLs and Encoders
    TCTL4 |= 0b00010101 ; // PT0-1-2: Capture on Rising edge -> Encoders of
Motor 1
    TCTL3 |= 0b01010100 ; // PT5-6-7: Capture on Rising edge -> Encoders of
Motor 2
    TFLG1 |= 0b11000011 ; // Clear Flags in PT0-1-2 and PT5-6-7
    TIE |= 0b10000001 ; // Enabled interrupt in PT0 and PT7
}
//-----//

/**----- Encoder of MOTOR 1 -----**//
//----- TIC0_Interrupt -----//
// For Encoder's A signal of (Motor1)
interrupt 8 void TIC0hndlr(void){ // LEFT-MOTOR ( B_signal is faster than
A_signal)
    char B_signal ;
    B_signal = PTT_PTT1 ; // get B_signal in PT1

    if (B_signal == 0){ // A is Raising-Pulse, B is Low --> Reverse , pulse-
        direction_flag_1 = 1 ;
        count_pulse_1 -= 1 ; // increase count-pulse
    } else if (B_signal == 1){ // A is Raising-Pulse, B is High--> Forward , pulse++
        direction_flag_1 = 0 ;

```



```

    count_pulse_1 += 1 ;          // decrease count-pulse
}
TFLG1 |= 0b00000001 ;          // Clear Flag in PT0
}
//-----//

/**----- Encoder of Motor 2 -----**//
//----- Interrupt TIC7 - Encoder's A -----//
// interrupt when capture a raising pulse-A. (Motor 2)

interrupt 15 void TIC7hndlr(void){ // RIGHT-MOTOR ( A_signal is faster than
B_signal)
char B_signal ;
B_signal = PTT_PTT6 ;          // get B_signal in PT6

if (B_signal == 0){           // A is Raising-Pulse, B is Low --> Forward , pulse++
    direction_flag_2 = 0 ;
    count_pulse_2 += 1 ;      // increase count-pulse
} else if (B_signal == 1){    // A is Raising-Pulse, B is High--> Reverse , pulse--
    direction_flag_2 = 1 ;
    count_pulse_2 -= 1 ;      // decrease count-pulse
}

TFLG1 |= 0b10000000 ;        // Clear Flag in PT7
}

//-----//

//----- CONTROL DC SERVO -----//
void control_2_motors(void){
// Left-Motor : PWM0-PWM1
// Right-Motor: PWM2-PWM3

if (Torque_1 > 195) Torque_1 = 195 ;          // Limitation - 80% PWM
else if (Torque_1 < -195) Torque_1 = -195 ;    // Limitation - 80% PWM

if (Torque_2 > 195) Torque_2 = 195 ;          // Limitation - 80% PWM
else if (Torque_2 < -195) Torque_2 = -195 ;    // Limitation - 80% PWM

//----- Doesn't Turn -----//
//---- PWM : HAI CAU-H PHAI DUOC DIEU KHIEN PHAI GIONG NHAU

```

```

//----- LEFT-Motor -----//
if (Torque_1 >= 0){ // FORWARD
  PWMDTY0 = 0; // IN2-->OUT2: 0(volt) - Motor 1 (Left) PWMDTY1 =
(int)(Torque_1); // IN1-->OUT1: PWM
  PWMDTY1 = (int)(Torque_1); // IN3-->OUT3: PWM
}else{ // REVERSE
  PWMDTY1 = 0; // IN1-->OUT1: 24(volt) - Motor 1 (Left)
  PWMDTY0 = (int)((-1)*Torque_1); // IN2-->OUT2: PWM (24 --> 0: Volt)
}

//----- RIGHT-Motor -----//
if (Torque_2 >= 0){ // FORWARD
  PWMDTY4 = 0; // IN4-->OUT4: 0(volt) - Motor 2 (Right) ; PWM2 <==>
PWM4
  PWMDTY3 = (int)(Torque_2); // IN3-->OUT3: PWM
}else{ // REVERSE
  PWMDTY3 = 0; // IN3-->OUT3: 24(volt) - Motor 2 (Right) PWM2 <==>
PWM4
  PWMDTY4 = (int)((-1)*Torque_2); // IN4-->OUT4: PWM (24 --> 0: Volt)
}

}

//-----//

//-----//

/***** SCAN - KEY - WAKE UP *****/
void KeyBoard(void){
  PIFH |= 0xFF; // Clear Flag
  Key = PTH & 0xFF; // Receive 8 bit: PTH[0-7]
}

void Key_Init(void){
  asm(" sei"); // make atomic
  TIOS |= 0x10; // PT4 as OC4 : 0b00010000
  TCTL1 = TCTL1 & ~0x03; // Disconnected at pin PT4
  TIE &= 0xEF; // Disarm OC4 : 0b11101111

  DDRH = 0x00; // PTH[0-7] are key inputs
  PPSH = 0x00; // Falling edge at PTH[0-7]
  PIEH |= 0xFF; // Enable Interrupts PTH[0-7]
  PIFH |= 0xFF; // Clear Flag

```

```

Keyboard();    // Initial read
}

//-----//

//***** MAIN
//*****//

void main(void) {

    PLL_40_Init();    // 40Mhz - Bus Clock
    TOC3_Init();     // Enable TOC3 - 20ms Interrupt
    SCIO_Init_40(115200); // Enable SCIO at 115200 - 40 Mhz Bus
    PWM_Init();      // Enable PWM
    TIC_Init();      // Enable Timer Input Capture for Hall-Signals and Encoder-
Signals
    ADC_Init();      // 10-bit ADC , right-justified
    Key_Init();      // Key Wakeup
    LCD_Init();      // Enable LCD_module

    SetupXGATE();    // Enable XGATE with 80Mhz-Bus Clock
    //-----//

    LCD_goto(2,0);
    writeLCD_String("A:",2);
    LCD_goto(2,12);
    writeLCD_String("X:",2);

    DDRB = 0xFF ;    // Output
    PORTB = 0x07 ;   // Two-LEDs are OFF, LED (PB.0) is ON
    PORTB_PB0 = 1;

    DDRM |= 0x03 ;   // PM0-PM1 are output
    PTM |= 0x03 ;    // Enable MC33932-A+B

    DDRJ &= 0x3F ;   // 0x3F = 0b0011.1111 --> PJ.6-7 are inputs for sensor.

    /* ----- RF-module Connection -----
    PTS.7 <--> Channel A : Forward >>
    PTS.6 <--> Channel B : Reverse <<
    PTS.5 <--> Channel C : Stop    ||

```

```

    PTS.4 <--> Channel D : Turn Left-Right
*/
DDRS &= 0x0F ;          // 0xF3 = 0b0000.1111 --> PTS.4-7 are inputs for RF

set_angle = 0 ;        // Set value: Stand up and no run.
set_speed = 0 ;
set_position = 0;

EnableInterrupts;
    //-----//

/**----- LOOP -----**//
for(;;) {

/**----- T_sampling = 2.5ms -----**//
if (flag_2_5ms == 1){ // 2.5 ms

//----- Read ADC's value to calculate Angle-Y -----//
    ROT_Z = ADC_In(0x81);          // Gyro-Z-Value is ADC-Channel 1
    Y = ADC_In(0x83);             // Accelerometer-Y-value is ADC-Channel
3
    Z = ADC_In(0x84);             // "Channel-4" is position (Z)

    q_m= (ROT_Z - offset_gyro)/(scale_rot*57.3); // [scale_rot/degree/sec] <-->
[scale_rot/(pi/180)]= [scale_rot*57.3] (/rad/sec)

//-----//

    state_update(q_m);           // Update some parameters of KALMAN
FILTER

/**----- 3 * 2.5ms = 5ms for CONTROLLER -----
-*/
    if ((count_3==1)||count_3==4) { //if ((count==1)||count==4) { // T=7.5ms
(f=133 Hz)

    angle_m= asinf((Y - offset_acc_Y)/scale_acc); // Arc_sin --> Y_angle: is the
IMPORTANT ANGLE

//----- Limitation -----//
// -90(deg) <= angle_m <= 90(deg)
if (angle_m > (pi/2)) angle_m = pi/2 ;

```

```

else if (angle_m < (-pi/2)) angle_m=-pi/2 ;
//-----//

kalman_update(angle_m);           // KALMAN FILTER --> Output:
"ANGLE" (exactly)

/*----- PID - POSITION CONTROLLER -----
-----*/

position_X = (58*Z)/1023 ;         // (cm) "LEFT-ALIGN"
position_X_center = position_X - 29 ; // (cm) "CENTER-ALIGN"

position_error = set_position - position_X_center ;           // error

sigma_position_error += position_error ;                     // intergal of error
if (sigma_position_error > 120) sigma_position_error=120 ;    // Limitation
else if (sigma_position_error < -120) sigma_position_error=-120 ;

speed_error = position_error - position_error_old;
position_error_old = position_error;

u_position = (Kp_X*position_error + 0.5*Ki_X*sigma_position_error +
(Kd_X*1)*speed_error) ; // u_position (1)

/*-----*/

/*----- PID - ANGLE CONTROLLER -----
-----*/

angle_degree = angle * 57.3 ;           // Angle (radian) --> Angle (degree)
angle_error = set_angle - angle_degree ; // Error (radian)
sigma_error += angle_error ;           // Sigma (radian)
if (sigma_error > 120) sigma_error=120 ; // Limitation
else if (sigma_error < -120) sigma_error=-120 ;

omega_degree = angle_error - angle_error_old ;           // Omega "speed"
(radian/sec)
angle_error_old = angle_error;           // UPDATE

u_angle = (Kp*angle_error+ 0.5*Ki*sigma_error + (Kd*1)*omega_degree) ; //
u_angle (2)

```

```

    u_signal = u_angle + u_position;          // NOTE: u_signal --> Sarturation -->
u_out_final

    //----- Limitation -----//
    if (u_signal > 195) u_signal = 195 ;
    else if (u_signal < -195) u_signal = -195 ;
    /*-----*//

    if(fabsf(angle_degree) > 30) {           // Stop Motor if |Angle_degree| > 20
(degree)
        u_signal = 0 ;
        sigma_error = 0;
        sigma_position_error = 0;
        Torque_1 = 0;
        Torque_2 = 0;
    }

                                        // NOTE

    Torque_1 = u_signal ;
    Torque_2 = Torque_1;
    control_2_motors() ;                  // Control 2 DC-Servo --> MOTORS
}
/**----- END OF CONTROLLER -----
**//

//----- 101 * 2.5ms = 500ms =0.5 (sec) -----//
if (count_2 == 101){                    // For update value to LCD
    flag_250ms = 1;
    count_2 = 0;
}

flag_2_5ms = 0;                          // Reset 2.5ms-flag
}

/**----- End T_sampling (Ts=2.5ms)-----
**//

/*----- Control by RF-Module -----*/
/* ----- RF-module Connection -----
PTS.7 <--> Channel A : Forward >>
PTS.6 <--> Channel B : Reverse <<
PTS.5 <--> Channel C : Stop ||
PTS.4 <--> Channel D : Turn Left-Right

```

```
*/  
  
/*  
if(PTS_PTS7==1){      // FORWARD (Channel D )  
    while(PTS_PTS7==1) ;  
    count_forward += 1 ;  
    if(set_angle==0){  
        set_angle = 0.03 ; // +0.015(rad) --> Go Forward  
        set_speed = 0.03 ;  
        LCD_goto(4,9);  
        writeLCD_String("F1",2);  
    }  
    if((set_angle<0)||count_forward>=2){  
        set_angle += 0.01 ; // +0.015(rad) --> Go Forward  
        set_speed += 0.01 ;  
  
        if(set_angle>=0.15){  
            set_angle = 0.15 ;  
            set_speed = 0.15 ;  
            count_forward = 0;  
        }  
    }  
    forward_flag = 1;  
    reverse_flag = 0;  
    stop_flag = 0;  
    turn_flag = 0 ;  
}  
  
if(PTS_PTS6==1){      // REVERSE (Channel C)  
    while(PTS_PTS6==1) ;  
    count_reverse += 1 ;  
    if(set_angle==0){  
        set_angle = -0.03 ; // +0.015(rad) --> Go Forward  
        set_speed = -0.03 ;  
        LCD_goto(4,9);  
        writeLCD_String("R1",2);  
    }  
    if((set_angle>0)||count_reverse>=2){  
        set_angle -= 0.01 ; // +0.015(rad) --> Go Forward  
        set_speed -= 0.01 ;  
  
        if(set_angle<=-0.15) {
```

```

        set_angle = -0.15 ;
        set_speed = -0.15 ;
        count_reverse = 0;
    }
}
forward_flag = 0;
reverse_flag = 1;
stop_flag = 0;
turn_flag = 0 ;
}

if(PTS_PTS5==1){      // STOP   (Channel B)
    while(PTS_PTS5==1) ;
    stop_flag = 1 ;
    LCD_goto(4,9);
    writeLCD_String("S ",2);
}

if(PTS_PTS4==1){      // TURN   (Channel A)
    while(PTS_PTS4==1) ;
    reverse_flag = 0;
    forward_flag = 0;
    turn_flag = 1 ;
    LCD_goto(4,9);
    writeLCD_String("T ",2);
}

*/

//-----//

/*----- STOP FLAG -----*/
// STOP ROBOT
if(stop_flag == 1){
    count_stop += 1 ;
    if((count_stop%2000)==0){          // Time For Delay
        if((forward_flag==1)||(turn_flag==1)){
            set_angle -= 0.01;
            set_speed -= 0.01;
            if(set_angle<=-0.027){
                j = 0;
                while(j<=7000)

```



```

    {
        j++;
    }
    j = 0;
    set_angle = 0;
    set_speed = 0;
    count_stop = 0;
    turn_flag=0;
    forward_flag = 0;
    stop_flag = 0;
}
}
if(reverse_flag==1){
    set_angle += 0.01;
    set_speed += 0.01;
    if(set_angle>=+0.027){
        j = 0;
        while(j<=7000)
        {
            j++;
        }
        j = 0;
        set_angle = 0;
        set_speed = 0;
        count_stop = 0;
        reverse_flag = 0;
        turn_flag=0;
        stop_flag = 0;
    }
}
}
}

/*----- PRESS BUTTON -----*/
if(button_flag == 1){

    Keyboard();    // Read keys

    if ( Key == 0xE3){ // 0b11100011 ("MODE") - PH.4
        START_flag = 1 ; // Start --> Choose MODE for Tune Gains: (1) K1-K2-C1-Ka ,
        (2) Kp_X - Pd_X
        mode += 1;
    }
}

```

```

if(mode == 8) mode = 1;
LCD_goto(3,9);
if(mode == 1) writeLCD_String("Kp_A",4);    // Mode
if(mode == 2) writeLCD_String("Ki_A",4);
if(mode == 3) writeLCD_String("Kd_A",4);
if(mode == 4) writeLCD_String("Kp_X",4);
if(mode == 5) writeLCD_String("Ki_X",4);
if(mode == 6) writeLCD_String("Kd_X",4);
if(mode == 7) writeLCD_String("SetA",4);
}

else if( Key ==0xD3){ // 0b11010011 ("CLEAR") - PH.5
  u_signal = 0 ;    // RESET
  Torque_1 = 0;
  Torque_2 = 0;
  sigma_position_error = 0;
  sigma_error = 0;
  angle_error = 0;
  position_error = 0;
}

else if( Key == 0xB3){ // 0b10110011 - PH.6
  increase_flag = 1 ; // Increase

  if(mode == 1) Kp += 0.25 ;
  if(mode == 2) Ki += 0.05 ;
  if(mode == 3) Kd += 0.05 ;
  if(mode == 4) Kp_X += 0.25 ;
  if(mode == 5) Ki_X += 0.05 ;
  if(mode == 6) Kd_X += 0.05 ;
  if(mode == 7) {
    set_angle += 0.005 ;
    if (set_angle < -0.05) set_angle=-0.05;
    else if (set_angle > 0.05) set_angle=0.05;
  }
}

else if( Key == 0x73){ // 0b01110011 - PH.7
  decrease_flag = 1; // Decrease

  if(mode == 1) Kp -= 0.25 ;
  if(mode == 2) Ki -= 0.05 ;

```

```

if(mode == 3) Kd -= 0.05 ;
if(mode == 4) Kp_X -= 0.25 ;
if(mode == 5) Ki_X -= 0.05 ;
if(mode == 6) Kd_X -= 0.05 ;
if(mode == 7) {
    set_angle -= 0.005 ;
    if (set_angle < -0.05) set_angle=-0.05;
    else if (set_angle > 0.05) set_angle=0.05;
}
}

else if( Key == 0xF2){ // 0b11110010 - PH.0 ("OK/SAVE)
    offset_acc_Y += 0.25 ; // Increase "offset" of angle
}
else if( Key == 0xF1){ // 0b11110001 - PH.1 ("TURN")
    offset_acc_Y -= 0.25 ; // Decrease "offset" of angle
}
button_flag = 0; // Reset Flag
}
/*-----*/

/*----- Update value to LCD -----*/
if(flag_250ms == 1){
    count_4 += 1;

    if(count_4==1){
// LCD_goto(1,2);
// write_float(offset_acc_Y*10); // Y_offset
LCD_goto(1,1);
write_float(angle_degree*10); // ANGLE-Y: mutiplied by 10 <<----- Use it
LCD_goto(1,7);
write_float(position_X_center); // Kd_X*100
LCD_goto(1,14);
write_float(u_signal); // U_signal (Control-Signal)
}

if(count_4==2){
LCD_goto(2,2);
write_float(Kp*10); // K1*10
LCD_goto(3,2);
write_float(Ki*100); // K2*100
LCD_goto(4,2);

```

```

write_float(Kd*100);          // K2*100

LCD_goto(2,14);
write_float(Kp_X*10);        // C1*100
LCD_goto(3,14);
write_float(Ki_X*100);      // Kp_X
LCD_goto(4,14);
write_float(Kd_X*100);      // Kp_X
count_4 = 0;
}

if(turn_flag==1){
    PORTB_PB0 = 1 ;          // OFF
    if(direction_flag==1){
        PORTB_PB1 = 0;      // Led On
        PORTB_PB2 = 1-PORTB_PB2; // Leg Toggle
    }
    if(direction_flag==0){
        PORTB_PB2 = 0;      // Led On
        PORTB_PB1 = 1-PORTB_PB1; // Leg Toggle
    }
}

flag_250ms = 0;             // reset the Flag
}

/*
if(turn_flag==0){           // Don't turn
if(set_angle==0){
    PORTB_PB1 = 1;
    PORTB_PB2 = 1;
    PORTB_PB0 = 1 ;
}
if(set_angle<0){
    PORTB_PB1 = 1;
    PORTB_PB2 = 0;
    if(set_angle<=-0.04){
        PORTB_PB0 = 0 ;
    }else PORTB_PB0 = 1;
}
}
if(set_angle>0){
    PORTB_PB1 = 0;
}
}

```

```
PORTB_PB2 = 1;
if(set_angle>=0.04){
    PORTB_PB0 = 0 ;
}else PORTB_PB0 = 1;
}
}
*/

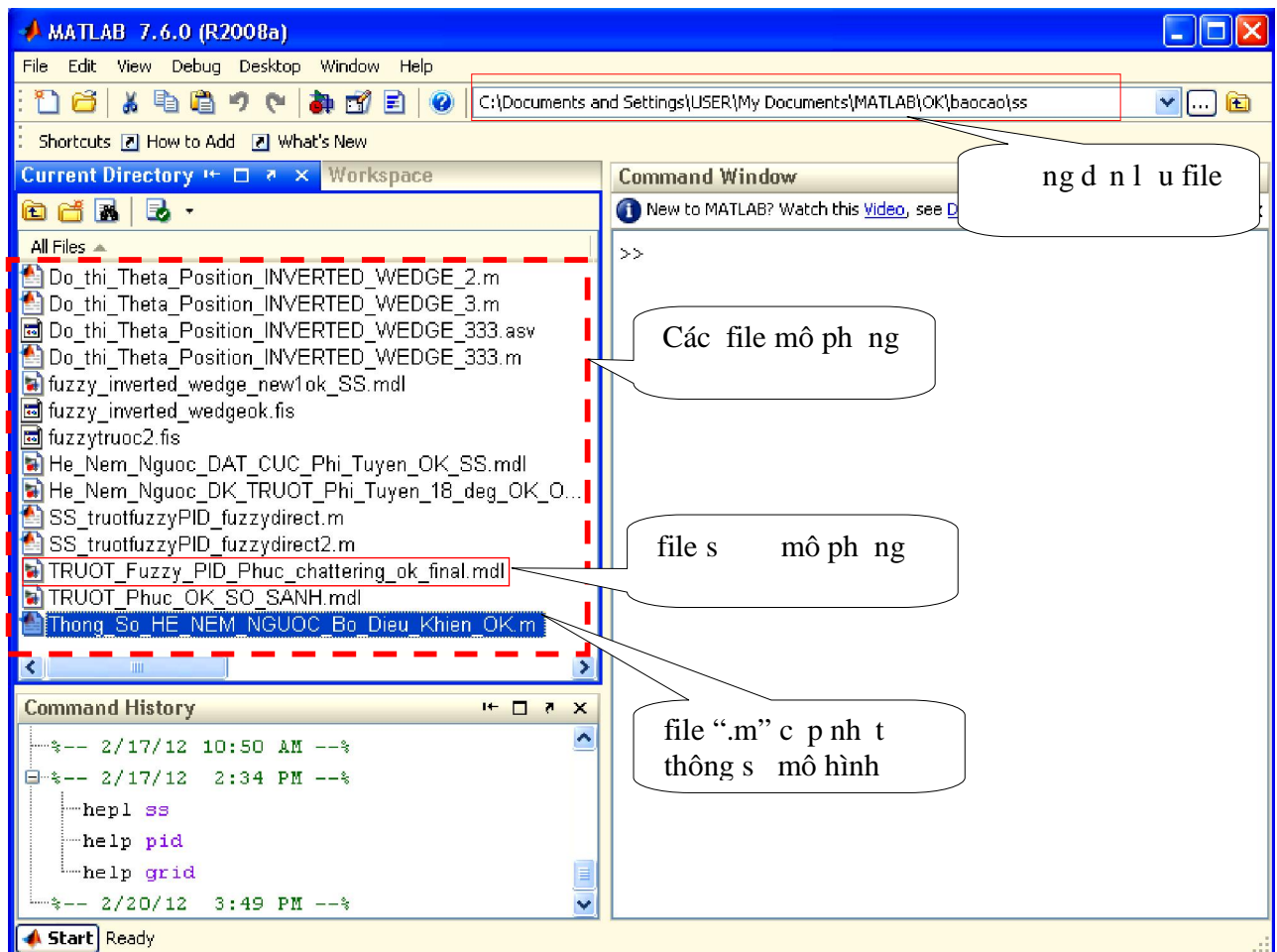
/*-----*/
} // end of "For"
} // end of "void_main"

//-----The END -----//
/******/
```

PHẦN 3 HƯỚNG DẪN DÙNG SIMULINK TRONG VIỆC THIẾT KẾ

1. Sử dụng kỹ thuật mô phỏng

- **Bước 1:** khởi tạo mô hình Matlab/Simulink. Chọn đường dẫn để tải các file mô phỏng (các file mô phỏng tập chung trong một thư mục).
- **Bước 2:** mở và chạy file chương trình mô phỏng, thể hiện kết quả như sau:
 - + Chạy file “*Thong_So_HE_NEM_NGUOC_Bo_Dieu_Khien_OK.m*” để nhập các thông số sử dụng trong mô phỏng.



- + Chọn các file “. *mdl” thể hiện số mô phỏng. Gồm các file:
 - “*fuzzy_inverted_wedge_newIok_SS.mdl*”
 - “*He_Nem_Nguoc_DAT_CUC_Phi_Tuyen_OK_SS.mdl*”
 - “*TRUOT_Phuc_OK_SO_SANH.mdl*”
 - “*TRUOT_Fuzzy_PID_Phuc_chattering_ok_final.mdl*”
- + Double Click vào “Scope” hiển thị kết quả mô phỏng.

2. Sơ đồ mô hình con nhện ng c th c nghi m

✓ So n th o và n p ch ng trình i u khi n:

- Cài t và kh i ng ph n m m “Freescale Codewarrior”: ch y file “*CW_HC12_v5.1_SPECIAL*” tiến hành cài t.
- G n b n p ch ng trình vào máy tính và k t n i cáp v i board i u khi n t i ngõ n p ch ng trình.
- T c a s ch ng trình “Freescale Codewarrior”: Ch n file / open / “ n i l u ch ng trình”/ “ tên file”.
- Sau khi m file ch ng trình, ta có th tiến hành so n th o, ch nh s a, sau ó nh p ch n “ debug” trên thanh công c biên d ch và n p ch ng trình vào chíp.

The screenshot shows the Freescale CodeWarrior IDE interface. The top menu bar includes File, Edit, View, Search, Project, Processor Expert, Device Initialization, Window, and Help. The toolbar contains various icons for file operations and development tools. The left pane shows a project tree for 'X-Gate.mcp' with a table of files and their code/data sizes. The right pane shows the source code for 'main_NEM_NGUOC_FINAL.c'.

File	Code	Data
Rescued Items	0	0
Includes	0	0
Libs	54535	4052
Project Settings	0	0
Sources	0	0
adc.c	0	0
ADC.h	0	0
datapage.c	0	0
LCD.c	0	0
lcd.h	0	0
PLL_2.c	0	0
pll_2.h	0	0
SCIO.C	0	0
SCIO.H	0	0
Timer.C	0	0
Timer.h	0	0
xgate.h	0	0
xgate.cxgate	0	0
main_NEM_NGUOC_FINAL.c	0	0

Code Editor Content:

```
1> C...-LOOP with Capture Encoder's signa  
Us... A-B signals, had not used Z-signal  
2  
3  
4  
/*  
... two DC-Servo Motors  
... to predict and measur  
*/  
  
#include <hidef.h> /* common defines a  
#include <mc9s12xdp512.h> /* derivative  
#pragma LINK_INFO DERIVATIVE "mc9s12xdp512"  
  
#include <string.h>  
#include <math.h>  
#include "xgate.h"  
  
#include "pll_2.h" // for X-GATE  
#include "lcd.h"  
#include "SCIO.H"  
#include "Timer.h"  
#include "ADC.h"  
  
//----- GLOBAL VARIAIBLE -----  
unsigned char OK_flag=0,Key, LCD_flag=0, mc  
START_flag, increase_flag, de  
reverse_flag=0, forward_flag=
```

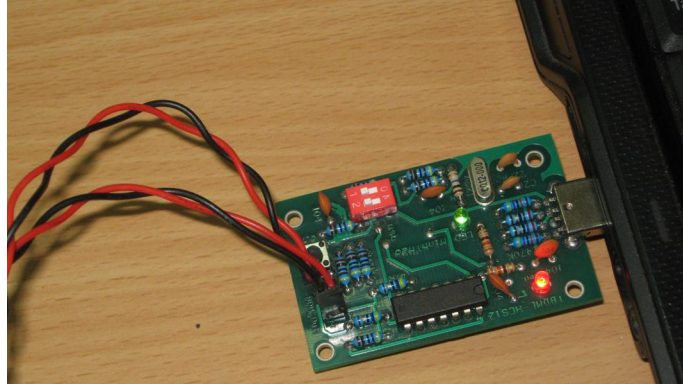
Nút chèn ng debug

Project chương trình

Vùng số n th o chương trình

✓ **Ch ý th c nghi m**

- Sau khi ã n p ch ng trnh vào bên trong Chip, tháo b n p ra kh i board i u khi n.



- B c công t c ngu n i n c p cho Motor và H-driver ch ý mô hình th c nghi m (bo m ch i u khi n c c p ngu n b ng adaptor riêng).

M C L C

TRANG

L I C M N

DANH M C CÁC HÌNH V , TH

DANH M C CÁC B NG

DANH M C CÁC KÝ HI U, CÁC CH VI T T T

PH N I: T NG QUAN TÀI	1
1.1. S c n thi t c a tài.....	1
1.2. Các công trình nghiên c u liên quan.....	2
1.2.1. <i>M t s mô hình con nê m ng c t cân b ng</i>	2
1.2.2. <i>M t s báo cáo nghiên c u khoa h c có liên quan</i>	6
1.3. N i dung nghiên c u.....	8
1.4. M c tiêu c a tài.....	8
1.5. Quy trình th c hi n.....	9
PH N II: PH NG PHÁP VÀ K THU T TH CHI N	10
2.1. Nguyên lý ho t ng c a con nê m ng c t cân b ng.....	10
2.2. Mô hình hoá h con nê m ng c.....	11
2.2.2. <i>Kh o sát mô hình h th ng con nê m ng c</i>	11
2.2.3. <i>Các ph ng trình bi n tr ng thái c a h th ng</i>	12
2.2.4. <i>Tuy n tính hoá h th ng t i i m cân b ng</i>	16
2.2.4. <i>B ng thông s c a h con nê m ng c t cân b ng</i>	18
2.3. B i u khi n t c c.....	19
2.4. B i u khi n tr t.....	22
2.5. B i u khi n k th p Tr t- M -PID.....	25
2.6. B i u khi n m.....	28
2.6.1. <i>Phân tích thi t k b i u khi n m</i>	28
2.6.2. <i>Thi t k b i u khi n m</i>	29
PH N III: CÁC K T QU T C	37

3.1. K t qu mô ph ng.....	37
3.1.1. <i>i v i i u khi n góc nghiêng θ nh</i>	37
3.1.2. <i>i v i i u khi n góc nghiêng $\theta l n$</i>	38
3.2. K t qu thi công th c nghi m.....	39
3.2.1. <i>Thi tk mô hình c khí</i>	39
3.2.2. <i>Hình nh mô hình con nêm ng c t cân b ng ã thi công</i>	41
3.2.3. <i>Các m ch i n</i>	45
PH NIV: K T QU N IB T VÀ KI N NGH	48
4.1. K t qu n i b c	48
4.2. Ki n ngh	49
PH NV: BÁO CÁO KINH PHÍ ã S D NG TRONG TÀI	50
TÀI LI U THAM KH O	51
PH L C 1: CÁC S M CH I N	53
PH L C 2: CH NG TRÌNH I U KHI N	60
PH L C 3: H NG D NS D NG S NPH MC A TÀI	86