

**Trường Đại học Công nghiệp Tp. Hồ Chí Minh**  
**Khoa Công nghệ Điện tử**  
-----

**Bài giảng**

**THÍ NGHIỆM ĐIỀU KHIỂN TỰ ĐỘNG**

**Biên soạn: Th.S Huỳnh Minh Ngọc**

**LƯU HÀNH NỘI BỘ**  
**2008**

# MỤC LỤC

Lời nói đầu

**Bài 1: Khảo sát hệ thống điều khiển tự động dùng MATLAB ..... 5**

1.1. Mục tiêu

1.2. Nội dung

1.2.1. Phần mềm MATLAB và cách sử dụng

1.2.2. Hàm truyền đạt, phương pháp biến trạng thái

1.2.3. Đáp ứng của hệ thống

1.2.4. Đặc tính của hệ thống điều khiển.

1.2.5. Phân tích ở miền tần số.

1.2.6. Phân tích quỹ đạo nghiệm số.

1.3. Thí nghiệm

1.3.1. Hàm truyền đạt, phương pháp biến trạng thái

1.3.2. Khảo sát hệ thống tự động dùng giản đồ Bode và Nyquist.

1.3.3. Khảo sát hệ thống bằng phương pháp quỹ đạo nghiệm số

1.4. Kiểm tra đánh giá

1.5. Phần mềm ACSYS2002

**Bài 2: Thiết kế hệ thống điều khiển tự động dùng MATLAB ..... 44**

2.1. Mục tiêu

2.2. Nội dung

2.2.1. Công cụ Sisotool

2.2.2. Thiết kế hệ thống điều khiển dùng phương pháp quỹ đạo nghiệm số bằng Matlab.

2.2.3. Thiết kế ở miền tần số

2.2.4. Chuyển từ mô hình liên tục theo thời gian ra rời rạc và ngược lại.

2.3. Thí nghiệm

2.3.1. Thiết kế bộ hiệu chỉnh sớm pha

2.3.2. Thiết kế bộ hiệu chỉnh trễ pha.

2.3.3. Thiết kế bộ hiệu chỉnh sớm trễ pha.

2.3.4. Thiết kế bộ điều khiển PID

2.4. Kiểm tra đánh giá

**Bài 3: Khảo sát và mô phỏng hệ thống tự động dùng SIMULINK ..... 75**

3.1. Mục tiêu

3.2. Nội dung

3.2.1. Giới thiệu về Simulink.

Các bước tiến hành mô phỏng.

3.3. Thí nghiệm

3.3.1. Khảo sát hệ thống ổn định nhiệt độ.

3.3.2. Khảo sát mô hình hệ thống điều khiển tốc độ và vị trí động cơ DC.

3.4. Kiểm tra đánh giá

<b>Bài 4: Hệ thống điều khiển nhiệt độ .....</b>	<b>91</b>
4.1.Mục tiêu	
4.2.Nội dung	
4.2.1.Sơ đồ khối, nguyên lí hoạt động của hệ thống điều khiển nhiệt độ.	
4.2.2.Hàm truyền đạt lò điện và mô hình của Ziegler-Nichols	
4.2.3.Nguyên lý điều khiển ON-AFF (dùng khâu role có trễ)	
4.2.4.Nguyên lý điều khiển tuyến tính	
4.3. Thí nghiệm	
4.3.1. Đo quá trình quá độ hệ hở, đầu vào hàm nấc	
4.3.2. Điều khiển ON-OFF	
4.3.3. Điều khiển tuyến tính.	
4.4. Kiểm tra đánh giá	
4.5.Điều khiển nhiệt độ bằng các phương pháp điều khiển cổ điển.	
<b>Bài 5: Hệ thống điều khiển tốc độ và vị trí động cơ một chiều .....</b>	<b>99</b>
5.1.Mục tiêu	
5.2.Nội dung	
5.2.1. Phần điều khiển tốc độ.	
5.2.2. Phần điều khiển vị trí.	
5.2.3. Truyền động điện động cơ DC điều chỉnh áp phản ứng	
5.3.Thí nghiệm	
5.4.Kiểm tra đánh giá.	
<b>Bài 6: Điều khiển số động cơ một chiều .....</b>	<b>109</b>
6.1.Mục tiêu	
6.2.Nội dung	
6.3.Thí nghiệm	
6.4.Hệ thống điều khiển tốc độ động cơ	
6.5.Nhận xét đánh giá.	
<b>Bài 7: Hệ thống điều khiển mực nước .....</b>	<b>111</b>
7.1.Mục tiêu	
7.2.Nội dung	
7.3.Thí nghiệm	
7.4.Nhận xét đánh giá.	
Phụ lục .....	115
Tài liệu tham khảo .....	118

## LỜI NÓI ĐẦU

Thí nghiệm điều khiển tự động là môn học minh họa và bổ sung phần thực tế cho giáo trình Lý thuyết điều khiển tự động. Sinh viên tiếp cận công cụ Matlab để khảo sát và thiết kế hệ thống tự động và khảo sát các hệ thống điều khiển tự động thực tế. Điều khiển tự động ngày nay có mặt trong các lĩnh vực điện -điện tử, cơ khí, hóa, giao thông, qui trình sản xuất ở nhà máy.

Giáo trình gồm có bảy bài thí nghiệm:

1. Khảo sát hệ thống tự động dùng Matlab.
2. Thiết kế hệ thống tự động dùng Matlab.
3. Khảo sát và mô phỏng hệ thống tự động dùng Simulink.
4. Hệ thống điều khiển nhiệt độ.
5. Hệ thống điều khiển vị trí và tốc độ động cơ DC.
6. Hệ thống điều khiển số động cơ một chiều.
7. Hệ thống điều khiển mực nước.

Giáo trình dùng để giảng dạy cho sinh viên ngành Điện tử tự động hệ đại học. Chắc hẳn giáo trình còn nhiều thiếu sót. Tác giả chân thành cảm ơn các ý kiến đóng góp của các thầy cô giáo trong bộ môn Điều khiển tự động, các đồng nghiệp, và bạn đọc để giáo trình ngày hoàn thiện hơn. Thư góp ý xin gửi về bộ môn Điều khiển tự động, Khoa Công nghệ Điện tử, trường Đại học Công nghiệp Tp. HCM, ĐT:8940390.

Tp. HCM ngày 15-7-2008

Tác giả

Th.S Huỳnh Minh Ngọc

## Bài 1

# KHẢO SÁT HỆ THỐNG TỰ ĐỘNG DÙNG MATLAB

### 1.1. Mục tiêu:

- Căn bản Matlab: giới thiệu,mô tả toán học hệ thống dùng hàm truyền đạt, phương pháp biến trạng thái, đặc tính động học, khảo sát tính ổn định hệ, sai số xác lập.
- Phân tích ở miền tần số: Biểu đồ Bode, Nyquist.
- Khảo sát hệ dùng phương pháp quỹ đạo nghiệm số .

### 1.2. Nội dung

#### 1.2.1.Phần mềm MATLAB và cách sử dụng

##### 1.2.1.1.Giới thiệu về Matlab

Matlab được phát triển bởi công ty Math Works Inc, là một chương trình phân tích ma trận, thiết kế điều khiển, nhận dạng hệ thống và đồ thị kỹ thuật.

Gõ đường dẫn : MATLAB khi đó sẽ xuất hiện dấu nhắc ">>" hoặc double-click vào biểu tượng Matlab trên màn hình nền.

Matlab cũng có khả năng thi hành một dãy lệnh chứa trong một tập tin, tập tin file.M. Một chương trình có thể được viết và ghi lại ở dạng ASCII với tên tập tin có phần mở rộng ra trong thư mục mà Matlab đang chạy. Tên tập tin phải ở dạng chữ thường. Dạng tập tin file.M này được coi như là tập tin hay tập lệnh nguyên bản. Để chạy chương trình tại dấu nhắc ">>"đánh tên tập tin không có phần mở rộng .m.

##### 1.2.1.2.Lệnh và Biến số:

Lệnh có dạng ở hình 1.1. Matlab dùng phép gán để dấu « = » chỉ ra rằng gán biểu thức vào một biến.

>>biến= biểu thức

Dạng lệnh Matlab

>>: dấu nhắc lệnh

Biểu thức nhập vào mà không có tên biến thì Matlab tính toán và kết quả được nhớ và hiển thị sau từ ans. Kết quả một biểu thức gán vào một tên biến để tiện cho việc sử dụng khác. Tên biến có thể đến 19 ký tự (bao gồm chữ và số), tuy nhiên ký tự đầu tiên của một tên biến phải bắt đầu bằng một chữ.

Chuỗi ký tự:

Một chuỗi ký tự nối tiếp nhau trong các câu sau được gọi là một chuỗi ký tự hoặc biến câu.

>>C='Good'

Kết quả sẽ là :

C=Good

Bảng 1.1. Các phép toán học

+	Cộng
-	Trừ
*	Nhân
/	Chia
^	lũy thừa

### 1.2.1.3. Các phép tính toán vector:

Một vector bậc n là một dãy hàng hay dãy cột của n số hạng. Để biểu diễn vector cột ta nhập : các phần tử nằm trong dấu ngoặc vuông [], cách nhau bởi dấu chấm phẩy (;). Ví dụ :

```
>>X=[2;-4;8]
```

Kết quả là :

```
X=
     2
    -4
     8
```

Nếu các phần tử được phân cách bởi các dấu phẩy hoặc khoảng trắng thì đó là vector hàng. Ví dụ :

```
>>R=[tan(pi/4) sqrt(9) -5]
```

Kết quả sẽ là :

```
R=
     1.0000     3.0000    -5.0000
```

Chuyển vị một vector. Ví dụ :

```
>>Y=R'
```

sẽ cho ra :

```
Y=
     1.0000
     3.0000
    -5.0000
```

Các vector có cùng kích thước có thể cộng hoặc trừ. Ví dụ :

```
>>P=5*R sẽ cho ra kết quả sau :
```

```
P=
     5.0000    15.0000   -25.0000
```

Toán tử \* thực hiện phép tính nhân tương ứng các phần tử với nhau. Ví dụ : với X và Y ở trên ta có :

```
>>E=X*Y
```

Kết quả sẽ là :

```
E=
     2
    -12
    -40
```

Tích vô hướng của 2 vector X và Y là số vô hướng được xác định bởi  $\sum_{i=1}^n X_i.Y_i$ . Nếu X và Y

là hai vector cột được xác định ở trên thì tích vô hướng của chúng là :  $S=X'*Y$ .

Kết quả sẽ là:

```
S=
    -50
```

Có các hàm của Matlab để tính chuẩn của vector. Ví dụ tính Norm Euclid

```
>>N=norm(X)
```

Đưa ra kết quả

```
N=
     9.1652
```

Góc giữa hai vector X và Y được xác định bằng  $X.Y/(ModunX. ModunY)$ . Dòng lệnh:

```
» theta=acos(X'*Y/(norm(X)*norm(Y)))
```

Kết quả là :

```
theta =
     2.7444
```

trong đó theta được đo bằng radian.

Biểu diễn một vector zero. Ví dụ:

» Z=zeros(1,4)

Z =

0 0 0 0

Trong MATLAB(:) có thể sử dụng để tính vector hàng. Ví dụ như :

» x=1:8

cho ra một vector hàng của các số nguyên từ 1 đến 8.

x =

1 2 3 4 5 6 7 8

#### 1.2.1.4. Ma trận

Trong MATLAB một ma trận được tạo bởi một dãy số trong ngoặc vuông. Các phần tử trong mỗi hàng được phân biệt bởi các khoảng trống hoặc dấu phẩy. Dấu chấm phẩy được dùng để kết thúc một hàng. Ví dụ:

» A=[6 1 2;-1 8 3;2 4 9]

Kết quả là :

A =

6 1 2  
-1 8 3  
2 4 9

Một cột hoặc hàng của một ma trận có thể được ký hiệu bằng (:). Ví dụ:

» r3=A(3,:)

Kết quả là :

r3 =

2 4 9

Tương tự A(:,2) biểu thị tất cả các phần tử của cột thứ 2 trong A.

» A(:,2)

ans =

1  
8  
4

Cộng 2 ma trận A và B. Ví dụ :

» A=[6 1 2;-1 8 3;2 4 9]

A =

6 1 2  
-1 8 3  
2 4 9

» B=[1 2 3;-2 4 6;1 3 7]

B =

1 2 3  
-2 4 6  
1 3 7

» D=A+B

D =

```

7   3   5
-3  12   9
3   7  16
    
```

Trừ hai ma trận A và B. Ví dụ :

» C=A-B

C =

```

5  -1  -1
1   4  -3
1   1   2
    
```

Nhân hai ma trận A và B từ phép tính A\*B nếu chúng tương thích. Ví dụ :

» E=A\*B

E =

```

6  22  38
-14 39  66
3  47  93
    
```

Hai ký hiệu được sử dụng để chia ma trận. A\B tương ứng  $A^{-1}*B$  và A/B tương ứng với  $A*B^{-1}$ . Ví dụ:

AX=B

$$\begin{bmatrix} 4 & -2 & -10 \\ 2 & 10 & -12 \\ -4 & -6 & 16 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} -10 \\ 32 \\ -16 \end{bmatrix}$$

» A=[4 -2 -10;2 10 -12;-4 -6 16]

A =

```

4  -2  -10
2  10  -12
-4  -6   16
    
```

» B=[-10;32;-16]

B =

```

-10
32
-16
    
```

» X=A\B

Kết quả là :

X =

```

2.0000
4.0000
1.0000
    
```



Ngoài ra còn rất nhiều hàm toán học khác trong file.m.

Nên sử dụng chức năng **inv** để xác định nghịch đảo của ma trận A rồi sau đó xác định ma trận X.

» A=[4 -2 -10;2 10 -12;-4 -6 16]

A =

```

4  -2  -10
2   10 -12
-4  -6   16

```

» B=[-10;32;-16]

B =

```

-10
 32
-16

```

» C=inv(A)

C =

```

2.2000  2.3000  3.1000
0.4000  0.6000  0.7000
0.7000  0.8000  1.1000

```

» X=C\*B

X =

```

2
4
1

```

### 1.2.1.5. Giá trị riêng:

Nếu ma trận A là một ma trận có (nxn) phần tử, thì có n số  $\lambda$  thoả mãn  $Ax=\lambda x$  là giá trị riêng của A. Chúng tìm được bằng cách sử dụng lệnh **eig(A)**. Giá trị riêng và vector riêng của A cũng có thể tìm được bằng lệnh **[X,D]=eig(A)**. Các phần tử trên đường chéo chính của ma trận chéo D là các  $\lambda$ , còn các cột của ma trận X là các vector riêng thoả mãn  $AX=XD$ . Ví dụ: Tìm giá trị riêng và vector riêng của ma trận A cho bởi:

» A=[0 1 -1;-6 -11 6;-6 -11 5]

A =

```

0   1  -1
-6 -11  6
-6 -11  5

```

» [X,D]=eig(A)

X =

```
0.7071 -0.2182 -0.0921
0.0000 -0.4364 -0.5523
0.7071 -0.8729 -0.8285
```

D =

```
-1.0000    0    0
    0 -2.0000    0
    0    0 -3.0000
```

### 1.2.1.6.Số phức :

Hầu hết các phép tính số phức đều có thể sử dụng được trong chương trình Matlab. Số ảo  $\sqrt{-1}$  được ngầm định trước bởi hai biến số I và j trong chương trình. Nếu i và j được sử dụng cho các giá trị khác thì ta phải định nghĩa phần ảo như sau:

» j=sqrt(-1)

j =

0 + 1.0000i

hoặc :

» i=sqrt(-1)

i =

0 + 1.0000i

Ví dụ: Tính  $Z_c \cosh g + \sinh g/Z_c$ , với  $Z_c=200+ i300$  và  $g=0.02 +j1.5$

» i=sqrt(-1)

i =

0 + 1.0000i

» Zc=200+300\*i;

» g=0.02+1.5\*j;

» V=Zc\*cosh(g)+sinh(g)/Zc

Kết quả là: :

V =

8.1672 +25.2172i

### 1.2.1.7. Đồ thị :

Chương trình MATLAB có thể tạo ra các loại đồ thị 2-D, 3-D, đường log, semilog, đồ thị cực, đồ thị khối và đường viền trên máy vẽ, máy in kim, máy in laser. Một số đồ thị 2-D có thể vẽ bằng các lệnh **plot**, **loglog**, **semilogx**, **semilogy**, **polar('text')**, **ylabel('text')** và **text('text')** có thể được sử dụng để đặt tên và thêm các chú thích trên đồ thị. Cú pháp của các câu lệnh trên bao gồm các ký hiệu (.,+,\* ,0,x) và màu sắc (r,h,g,w). Ví dụ: dòng lệnh sau

```

» plot(t,y1,'r',t,y2,'+b')
sẽ cho màu đỏ trên đường cong thứ nhất và màu xanh trên đường cong thứ hai.
Lệnh plot(x,y)- vẽ đồ thị y theo x.
Ví dụ : Vẽ đồ thị quan hệ x-y với nhiều biến khác nhau :
X 0 1 2 3 4 5 6 7 8 9 10 11 12
Y 0 0.5 1 2 4 7 11 14 15.5 16 16 16 16
» x=[0 1 2 3 4 5 6 7 8 9 10 11 12];
» y=[0 0.5 1 2 4 7 11 14 15.5 16 16 16 16];
» plot(x,y)
» grid
» meta EX18
    
```

Đồ thị x-y của ví dụ.

Lệnh meta cho phép mở file EX18.MET và ghi đồ thị vào đó.

### **Đồ thị mặt lưới 3 chiều :**

Lệnh mesh(Z) tạo một đồ thị 3 chiều của các phần tử trong ma trận Z. Bề mặt mặt lưới được định nghĩa bởi tọa độ Z của các điểm ở trên một ô lưới trong mặt phẳng X-Y. Biểu đồ được hình thành bởi sự liên kết các điểm gần kề với các đường thẳng. Meshdom làm biến đổi phạm vi định rõ bởi vector x và y vào trong các dãy X và Y.

Ví dụ : Để có đồ thị 3D của hàm Bessel  $j_0 \sqrt{x^2 + y^2}$  trong khoảng  $-12 < x < 12$ ,  $-12 < y < 12$ .

Ta có các dòng lệnh sau:

```

» clf;% thay cho clg
» [x,y]=meshgrid(-12:.6:12,-12:.6:12);% thay cho meshdom
» r=sqrt(x^2+y^2);
» z=bessel(0,r);
» m=[-45 60];
» mesh(z,m)
    
```

### **1.2.1.8. Script -T ập tin .m**

Matlab cho phép thực hiện một chuỗi lệnh trong một tập tin . Tập tin này gọi là M-file vì tên có dạng filename.m. Một scripts là một loại M-file. Một script là một chuỗi các lệnh thông thường và hàm được dùng ở đầu nhắc lệnh.

Hộp công cụ hệ thống điều khiển (Control systems Toolbox) là tập hợp các hàm để mô hình hóa, phân tích và thiết kế hệ thống điều khiển. Nó cung cấp các công cụ cổ điển như là vẽ Bode, biểu đồ Nichols, quỹ đạo nghiệm số, cũng như các kỹ thuật hiện đại như không gian trạng thái LQG và thiết kế đặt cực.

Thí dụ :

```

>>alpha=50;
>>plotdata
plotdata.m
%Day la script ve ham y=sin(alpha*t)
%Gia tri alpha phai ton tai trong workspace truoac de kich hoat script.
T=[0:0.01:1];
Y=sin(alpha*t);
Plot(t,y)
Xlabel('Time(sec)')
Ylabel('y(t)=sin(\alpha t)')
Grid on
    
```

Matlab có các lệnh vòng lặp như for, while và logic như là if. Cách sử dụng chúng tương tự trong ngôn ngữ Pascal hay C.

**Lệnh điều kiện:** if và switch

Cú pháp:

If term command [elseif term command...] [else command] end

Switch term case term command[...] [otherwise command] end

Trong đó term là điều kiện, command là lệnh.

Thí dụ:

```
>> test=5;
```

```
>> if test<=2;a=2,elseif test<=5; a=5,else a=10, end;
```

a =

5

```
>> switch test case 2;a=2,case {3,4,5};a=5,otherwise a=10,end;
```

a =

5

Trong cả hai trường hợp trên, các lệnh co được ngăn cách bởi dấu ( ;) và dấu (,).

Trong các Scripts, thường ta hay viết mỗi lệnh con trong một dòng riêng. Thí dụ :

```
>> if test<=2,
```

```
    a=2;
```

```
elseif test<=5,
```

```
    a=5;
```

```
else a=10;
```

```
end;
```

```
>> a
```

a =

5

**Vòng lặp và logic:** for và while

Bằng vòng lặp ta có thể thực hiện lặp lại nhiều lần một số lệnh nhất định. Cú pháp:

For variable=term command end

While term command end

Trong cả hai trường hợp lệnh break đều có tác dụng kết thúc vòng lặp.

Thí dụ:

```
>> for k=0:1, k^2, end;
```

ans =

0

ans =

1

```
>> n=1;
```

```
>> while 1,n=n+1;m=n^2;if m>10, break;end;end
```

m =

4

m =

9

m =

16

>>

**Bài tập :**

1.Cho hai ma trận sau :

$$A = \begin{bmatrix} 4 & 2\pi \\ 6j & 10 + \sqrt{2}j \end{bmatrix}$$

$$B = \begin{bmatrix} 6j & -13\pi \\ \pi & 16 \end{bmatrix}$$

Dùng Matlab tính toán sau :

- a)A+B                    b)AB
- c)A<sup>2</sup>                    d)A'
- e)B<sup>-1</sup>                    f)B'A'
- g) A<sup>2</sup> + B<sup>2</sup> -AB

2.Cho hệ phương trình đại số tuyến tính :

$$\begin{aligned} 5x+6y+10z &= 4 \\ -3x+14z &= 10 \\ -7y+21z &= 0 \end{aligned}$$

Xác định giá trị x,y, và z để tập phương trình đại số thỏa mãn.

3.Viết một Matlab Script (Tập tin .m) để vẽ hàm sau :

$$y(x) = \frac{4}{\pi} \cos \omega x + \frac{4}{9\pi} \cos 3\omega x$$

trong đó  $\omega$  là ngõ vào biến ở dòng lệnh. Đặt nhãn trục x là time(sec) và trục y là y(x).

**1.2.2.Hàm truyền đạt, phương pháp biến trạng thái**

**1.2.2.1.Hàm truyền đạt :**

**Khai báo hàm truyền :**

-Sử dụng lệnh tf(num, den) : Nhập đa thức tử số num và đa thức mẫu số den dưới dạng vector tham số của s theo trình tự số mũ của s bé dần.

>> h=tf([2 -3], [1 1])

Transfer function:

$$2s - 3$$

$$\frac{\text{-----}}{s + 1}$$

>>

-Khai báo dưới dạng hàm hữu tỷ của s. Trước hết ta phải khai báo s là biến mô hình TF, sau đó nhập hàm truyền đạt dưới dạng hàm hữu tỷ của s.

>> s=tf('s')

Transfer function:

s

>> h=(s+2)/(s^2+5\*s+4)

Transfer function:

s + 2

-----  
s^2 + 5 s + 4

>>

**Hàm truyền:**

Nếu P là một vector hàng chứa các hệ số của đa thức, thì hàm **roots(P)** sẽ cho ta một vector cột các phần tử của nó là nghiệm của đa thức.

Ví dụ: Tìm nghiệm của đa thức sau:

$$S^6 + 9s^5 + 31.25s^4 + 61.25s^3 + 67.75s^2 + 14.75s + 15$$

Nhập:

» P=[1 9 31.25 61.25 67.75 14.75 15]

P =

1.0000 9.0000 31.2500 61.2500 67.7500 14.7500 15.0000

» R=roots(P)

Được :

R =

-4.0000  
-3.0000  
-1.0000 + 2.0000i  
-1.0000 - 2.0000i  
0.0000 + 0.5000i  
0.0000 - 0.5000i

Nếu r là vector cột chứa các nghiệm của đa thức thì hàm **poly(r)** cho vector hàng, các phần tử là hệ số của đa thức.

Ví dụ: Nghiệm của đa thức là : -1,-2,-3+4i,-3-4i. Hãy xác định phương trình đa thức.

» i=sqrt(-1);

» R=[-1 -2 -3+4\*i -3-4\*i]

R =

-1.0000 -2.0000 -3.0000 + 4.0000i -3.0000 - 4.0000i

» P=poly(R)

P =

1 9 45 87 50

Từ đó phương trình đa thức là;

$$S^4 + 9s^3 + 45s^2 + 87s + 50 = 0$$

**Nghiệm và zero của hàm truyền:**

Ta sử dụng hàm **tf2zp** để tìm zero, nghiệm và độ lợi của hàm truyền.

Ví dụ: tìm nghiệm và zero của hàm truyền sau:

$$H(s) = \frac{(s^3 + 11s^2 + 30s)}{(s^4 + 9s^3 + 45s^2 + 87s + 50)}$$

Nhập :

```
» num=[1 11 30 0]
```

num =

```
1 11 30 0
```

```
» den=[1 9 45 87 50]
```

den =

```
1 9 45 87 50
```

```
» [z,p,k]=tf2zp(num,den)
```

Ta có :

z =

```
0
-6
-5
```

p =

```
-3.0000 + 4.0000i
-3.0000 - 4.0000i
-2.0000
-1.0000
```

k =

```
1
```

Hàm **zp2tf** dùng để xác định hàm truyền từ nghiệm và xem hệ số khuếch đại của hệ thống.

```
» z=[-6; -5; 0];k=1
```

```
» i=sqrt(-1);
```

```
» p=[-3+4*i;-3-4*i;-2;-1];
```

```
» [num,den]=zp2tf(z,p,k)
```

num =

```
0 1 11 30 0
```

den =

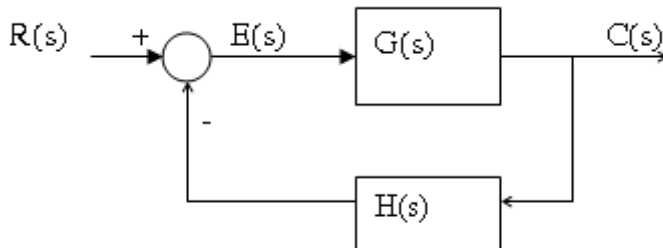
1 9 45 87 50

Lệnh đơn giản hàm truyền : minreal

Tính hàm truyền của hệ thống nối tiếp: lệnh series.

Tính hàm truyền của hệ song song: lệnh parallel.

Cho hệ thống hồi tiếp âm như sau:



Hình 1.1

Hồi tiếp âm :  $G_k = G/(1+G.H)$

Hồi tiếp dương :  $G_k = G/(1-G.H)$

$G_k = \text{feedback}(G,H)$  tính hàm truyền hệ thống hồi tiếp âm.

$G_k = \text{feedback}(G,H,+1)$  tính hàm truyền hệ thống hồi tiếp dương

Thí dụ:

```
>> G=tf([1 1],[1 3 2])
```

Transfer function:

```

s + 1
-----
s^2 + 3 s + 2
```

```
>> H=tf(1,[1 5])
```

Transfer function:

```

1
-----
s + 5
```

```
>> Gk=feedback(G,H) % Hàm truyền kín hệ hồi tiếp âm.
```

Transfer function:

```

s^2 + 6 s + 5
-----
s^3 + 8 s^2 + 18 s + 11
```

```
>> Gk=feedback(G,H,+1) % hàm truyền kín hệ hồi tiếp dương
```

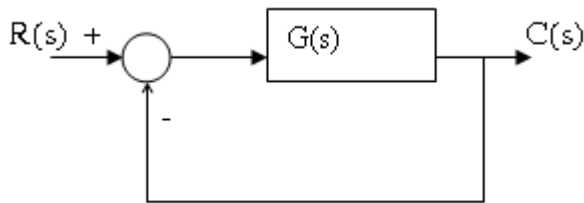
Transfer function:

```

s^2 + 6 s + 5
-----
s^3 + 8 s^2 + 16 s + 9
```

```
>> Gk=feedback(G,1) %hàm truyền kín hệ hồi tiếp âm đơn vị.
```





Hình 1.2

Transfer function:

$$\frac{s + 1}{s^2 + 4s + 3}$$

$$s^2 + 4s + 3$$

### 1.2.2.2. Phương pháp biến trạng thái:

Phương trình vi phân:

$$\frac{dx}{dt} = Ax(t) + Bu(t)$$

$$y = Cx(t)$$

là hệ phương trình vi phân bậc nhất của hệ thống và  $x$  gọi là biến trạng thái. Ưu điểm của phương pháp biến trạng thái là các phương trình mô tả hệ được giải dễ dàng nhờ máy tính. Ngoài ra phương pháp biến trạng thái có thể mở rộng để khảo sát hệ phi tuyến. Ta lập các phương trình biến trạng thái từ một phương trình vi phân bậc  $n$  bằng cách chỉ định các biến trạng thái thích hợp.

Ví dụ: lập phương trình biến trạng thái cho phương trình vi phân dưới đây:

$$2\frac{d^3y}{dt^3} + 4\frac{d^2y}{dt^2} + 6\frac{dy}{dt} + 8y = 10u(t)$$

Ta sử dụng hàm **ode2phv.m** để chuyển phương trình vi phân bậc  $n$  thành phương trình biến trạng thái.

Cú pháp :  $[A,B,C]=ode2phv(ai,k)$

»  $ai=[2\ 4\ 6\ 8]$

$ai =$

2 4 6 8

»  $k=10$

$k =$

10

»  $[A,B,C]=ode2phv(ai,k)$

### Hàm truyền-mô hình biến trạng thái :

Hàm  $[A,B,C]=tf2ss(num,den)$  sẽ biến đổi hàm truyền thành mô hình biến trạng thái.

Ví dụ : Cho hàm truyền  $\frac{s^2 + 7s + 2}{s^3 + 9s^2 + 26s + 24}$

»  $num=[1\ 7\ 2]$

$num =$

1 7 2

» den=[1 9 26 24]

den =

1 9 26 24

» [A,B,C,D]=tf2ss(num,den)

A =

-9 -26 -24  
1 0 0  
0 1 0

B =

1  
0  
0

C =

1 7 2

D =

0

**Mô hình biến trạng thái-Hàm truyền:**

Cho hệ  $dx/dt=Ax + Bu$

$$Y=Cx + Du$$

Áp dụng biến đổi Laplace suy ra  $G(s)=C(sI-A)^{-1}B +D$

Hàm [num,den]=ss2tf(A,B,C,D,i) sẽ biến đổi phương trình biến trạng thái thành hàm truyền.

Ví dụ : Hệ thống được biểu diễn bằng phương trình biến trạng thái sau :

$$dx/dt=Ax+Bu$$

$$A=\begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \\ -1 & -2 & -3 \end{bmatrix}$$

$$B=\begin{bmatrix} 10 \\ 0 \\ 0 \end{bmatrix}$$

$$y=[1 \ 0 \ 0]x$$

Tìm hàm truyền  $G(s)=Y(s)/X(s)$

» A=[0 1 0;0 1 1;-1 -2 -3]

A =

0 1 0  
0 1 1  
-1 -2 -3

» B=[10; 0;0]

B =

10  
0  
0

» C=[1 0 0]

C =

1 0 0

» D=[0]

D =

0

» [num,den]=ss2tf(A,B,C,D,1)

num =

0 10.0000 20.0000 -10.0000

den =

1.0000 2.0000 -1.0000 1.0000

Do đó hàm truyền là :

$$G(s)=(10s^2 + 20s -10)/(s^3 +2s^2 -s +1)$$

### 1.2.3.Đáp ứng của hệ thống

#### Hàm truyền chuẩn của hệ thống bậc 2:

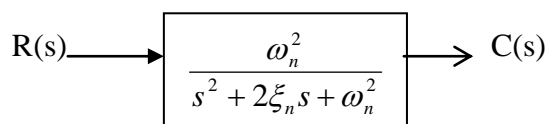
$$G(s) = \frac{\omega_n^2}{s^2 + 2\xi_n s + \omega_n^2}$$

#### 1.2.3.1. Đáp ứng miền thời gian :

Ta có thể dùng hàm `c=stepzwn(z,ω,R,a,T,t)` để tính đáp ứng của hệ . Ở đây z- thông số suy giảm, ωn-tần số tự nhiên, R-biên độ hàm bậc thang. Đối với hệ bậc 2 thì a=0, T=0 còn t-khoảng thời gian cần khảo sát. Đáp ứng của hệ suy giảm nhanh và có vọt lố cũng được khảo sát bằng hàm trên.. Các hàm **c=impulse(num,den)**, **c=step(num,den,t)** và **c=lsim(num,den,u,t)** cũng có thể được sử dụng để khảo sát đáp ứng chuyển tiếp hệ thống.

Ví dụ: Xác định đáp ứng bậc thang của hệ thống với  $\xi=0.6$  và  $\omega_n=5$ .

$$G(s)=(\omega_n^2)/(s^2 + 2\xi_n s + \omega_n^2)$$



Cách 1:

```
Num=25;
Den=[1 6 25];
Gs=tf(num,den)
T=0:0.02:2;
C=step(gs,t);
Plot(t,c)
Xlabel('t-sec'),ylabel('c(t)'),grid,pause
```

Cách 2:

```
>> den=[1 6 25]
```

den =

```
1 6 25
```

```
>> t=0:0.02:2;
```

```
>> c=step(num,den,t);
```

```
>> plot(t,c);
```

```
>> xlabel('t-sec'),ylabel('c(t)');
```

```
>> grid,pause
```

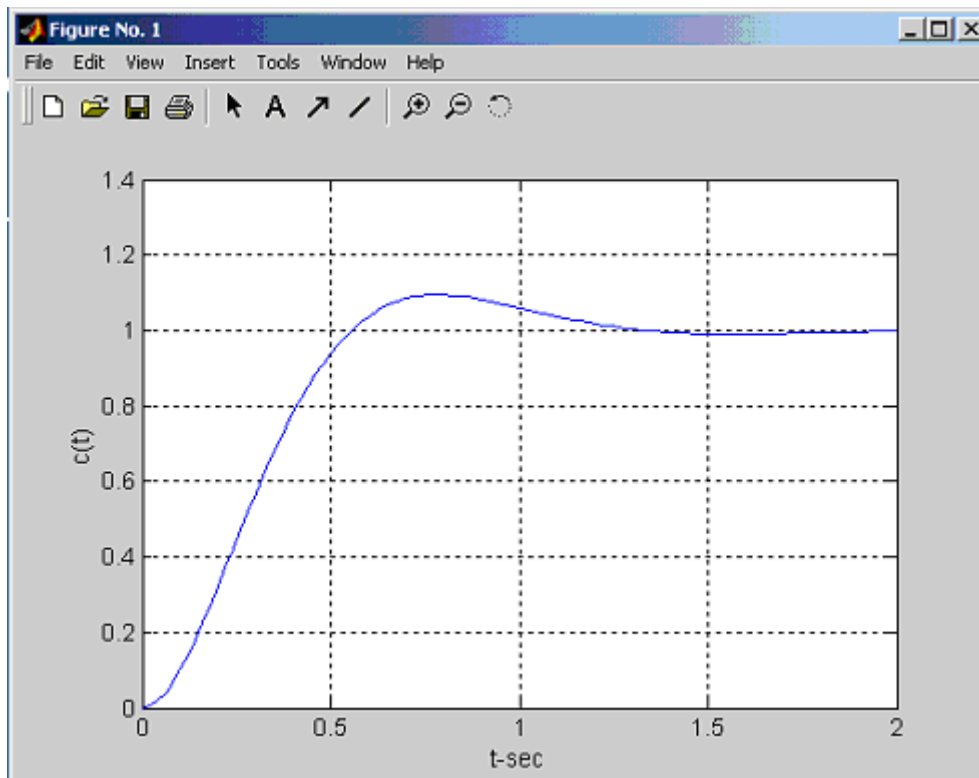
```
>>
```

Kết quả là :

$T_p=0.776667$  Phần trăm vọt lô=9.47783

$T_r=0.373333$

$T_s=1.18667$



Hình 1.3.Đáp ứng thời gian

Lệnh step: Cho hàm truyền đạt của hệ sau:  $G(s) = \frac{30}{s^2 + 4s + 30}$ . Lệnh Matlab vẽ đáp ứng nấc là :

```
num=30;den=[1 4 30]; G=tf(num,den)
```

step(G)

Lệnh impulse: vẽ đáp ứng xung

Cho hàm truyền đạt của hệ sau:  $G(s) = \frac{30}{s^2 + 4s + 30}$ . Lệnh Matlab vẽ đáp ứng xung là :

```
num=30;den=[1 4 30]; G=tf(num,den)
impulse(G)
```

### 1.2.3.2. Đáp ứng tần số của hệ thống:

Khi  $\xi < 0.707$  thì tần số cộng hưởng  $\omega_r$  được tính bởi:

$$\omega_r = \omega_n \sqrt{1 - 2\xi^2} \quad (1)$$

Biên độ cộng hưởng được ký hiệu  $M_{p\omega}$  và bằng :

$$M_{p\omega} = \frac{1}{2\xi\sqrt{1-\xi^2}} \quad (2)$$

Đáp ứng tần số của hệ thống được xác định bằng hàm  $g = \text{freqs}(\text{num}, \text{den}, w)$ . Đối với hệ thống bậc 2 với  $\xi < 1$  thì phương trình (1) và (2) có thể kết hợp để tính  $\omega_r, M_{p\omega}$ . Tuy nhiên, ta có thể sử dụng hàm  $\text{frqspec}(w, \text{mag})$  để xác định  $\omega_r, M_{p\omega}, \omega_B$ .

Ví dụ: Cho hệ thống được mô tả bởi hàm truyền vòng kín

$$G(s) = \frac{4}{s^2 + 2s + 4}$$

Xác định đáp ứng tần số với các thông tin  $\omega_r, M_{p\omega}, \omega_B$ .

Cách giải:

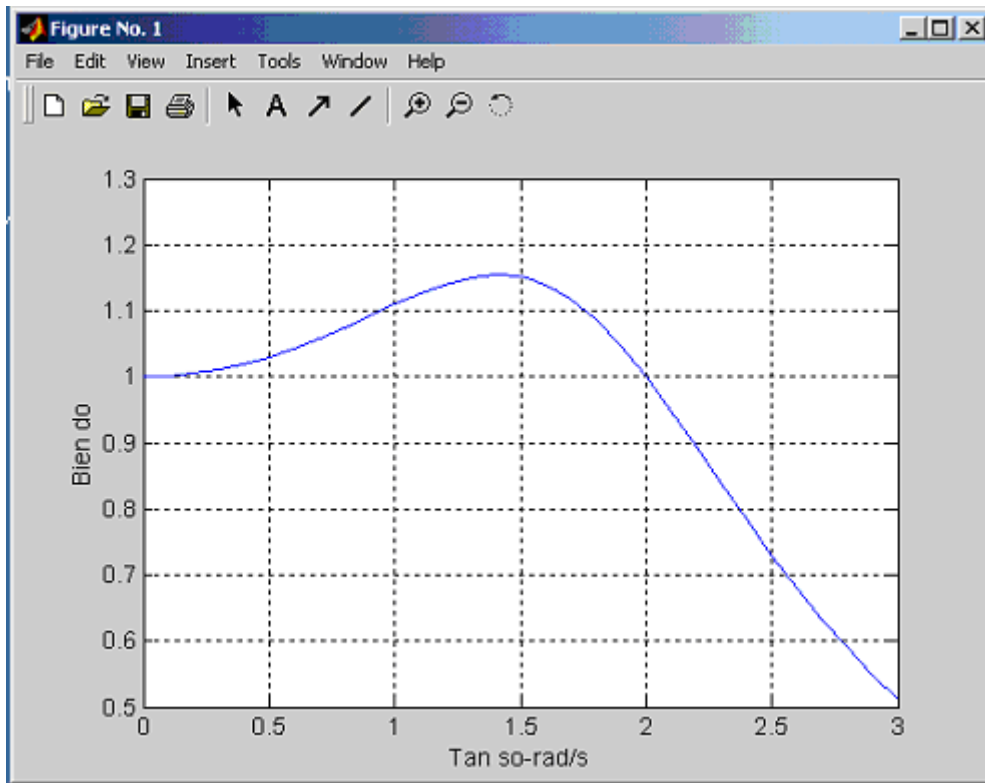
```
>> num=4;
>> den=[1 2 4];
>> w=0:0.1:3;
>> g=freqs(num,den,w);
>> mag=abs(g);
>> frqspec(w,mag)

>> plot(w,mag);
>> ylabel('Biên độ'),xlabel('Tần số-rad/s')
>> grid
>>
```

Kết quả là:

$\omega_r=1.41, M_{p\omega}=1.15, \omega_B=2.55$

Kết quả được biểu diễn trên hình:



Hình 1.4. Đáp ứng biên độ tần số

**1.2.4. Đặc tính của hệ thống điều khiển:**

**1.2.4.1. Tiêu chuẩn ổn định Routh-Hurwitz:**

Hàm roots(a) được xây dựng dựa vào tiêu chuẩn Routh-Hurwitz sẽ giúp khảo sát tính ổn định tuyệt đối của hệ.

Ví dụ: Cho hệ với phương trình đặc tính:

$$S^4 + 10s^3 + 35s^2 + 50s + 24 = 0$$

» a=[1 10 35 50 24]

a =

1 10 35 50 24

» routh(a)

Hệ ổn định.

**1.2.4.2. Sai số xác lập:**

Hai hàm errorzp(z,p,k) và errortf(num,den) được viết để tính toán sai số xác lập của hệ thống khi tín hiệu vào là tín hiệu bậc thang đơn vị, dốc đơn vị và parabolic đơn vị. Hàm errorzp(z,p,k) tính được sai số xác lập khi hệ thống được biểu diễn bằng các zero, các nghiệm và độ lợi. Ở đây z là vector cột chứa các zero của hàm truyền, p là vector cột bao gồm các nghiệm cực và k là độ lợi. Nếu bậc của tử số nhỏ hơn mẫu số, tức là n < m thì ta có n-m số inf(ω) để chiều cao của 2 vector p và z như nhau. Hàm errortf(num,den) tìm được sai số xác lập khi hệ thống được mô tả bởi hàm truyền.

Ví dụ: Xác định hằng số sai số và sai số xác lập của hệ:

$$G(s) = \frac{10(s + 4)}{s(s + 1)(s + 2)(s + 5)}$$

» k=10

k =

10

```
» z=[-4;inf;inf;inf]
```

```
z =
```

```
-4
Inf
Inf
Inf
```

```
» p=[0;-1;-2;-5]
```

```
p =
```

```
0
-1
-2
-5
```

```
» errorzp(z,p,k)
```

Ví dụ :  $G(s)=10/(s^2 +14s+50)$

```
>>num=10
```

```
>>den=[1 14 50]
```

```
>>errortf(num,den)
```

### 1.2.5.Phân tích ở miền tần số:

#### 1.2.5.1. Biểu đồ Bode

Hàm [mag, phase]= bode(num,den,ω) cho ta biết biên độ và góc pha của hàm truyền trong dãy tần số ω đang khảo sát.

Ví dụ 1: Vẽ biểu đồ Bode cho hệ thống điều khiển hồi tiếp đơn vị với hàm truyền hở.

$$GH(s) = \frac{K}{s(s+2)(s+50)} = \frac{K}{s^3 + 52s^2 + 100s}$$

Cho K=1300 chúng ta sử dụng dòng lệnh sau:

```
>> clf
```

```
>> k=1300;
```

```
>> num=[k];
```

```
>> den=[1 52 100 0];
```

```
>> w=logspace(-1,2);
```

```
>> [mag,phase]=bode(num,den,w);
```

```
>> dB=20*log10(mag);
```

```
>> subplot(211),semilogx(w,dB)
```

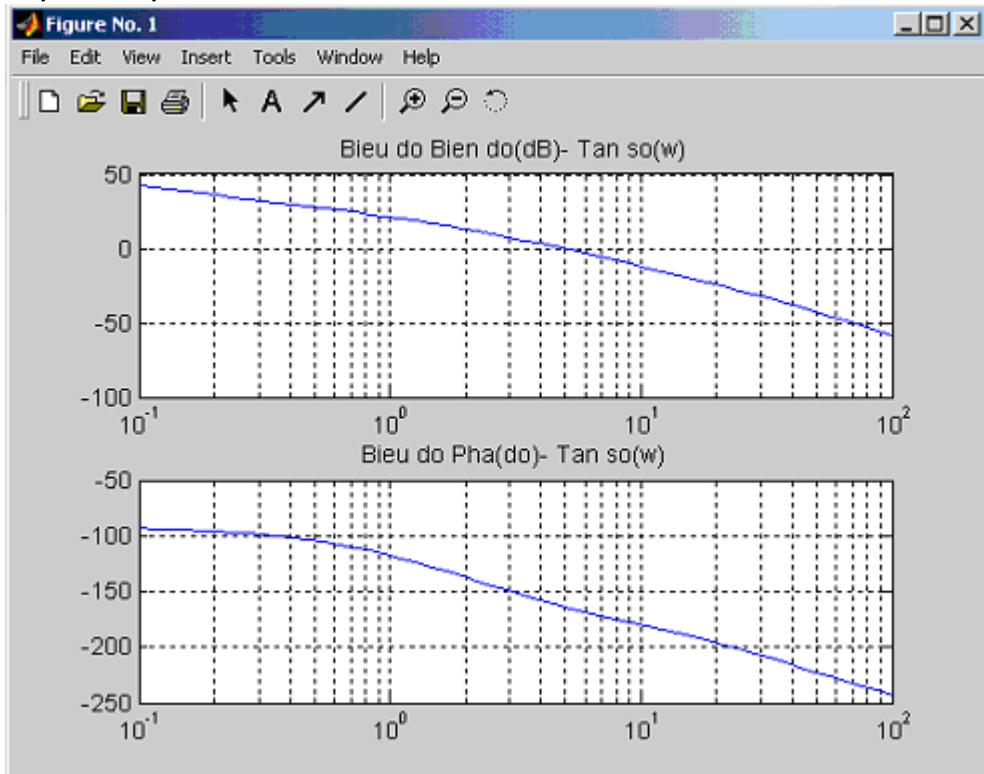
```
>> title('Bieu do Bien do(dB)- Tan so(w)'),grid
```

```
>> subplot(212),semilogx(w,phase)
```

```
>> title('Bieu do Pha(do)- Tan so(w)'),grid
```

```
>>
```

Kết quả được thể hiện trên hình sau:



Hình 1.5. Đáp ứng biên độ-tần số và pha-tần số

### 1.2.5.2. Biểu đồ nghiệm (Biểu đồ Nyquist)

Hàm  $[Re, Im]=Nyquist(num,den,\omega)$  của MATLAB cho biết phần thực và phần ảo của hàm truyền khi tần số thay đổi trong khoảng được chỉ định.

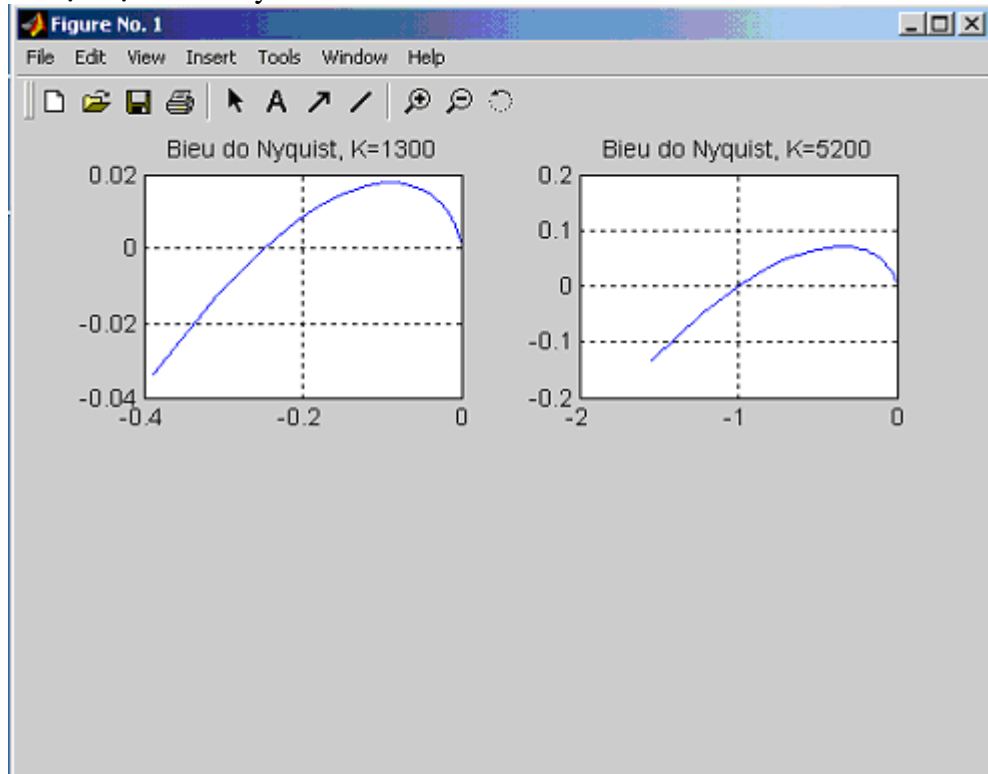
Ví dụ 2: Vẽ biểu đồ nghiệm Nyquist cho hệ thống của ví dụ 1 với  $K=1300$  và  $K=2500$ .

Lệnh Matlab:

```
>> clf
>> k1=1300;
>> k2=5200;
>> w=8:1:80;
>> num1=[k1];
>> num2=[k2];
>> den=[1 52 100 0];
>> [Re1, Im1]=Nyquist(num1,den,w);
>> [Re2, Im2]=Nyquist(num2,den,w);
>> subplot(221),plot(Re1,Im1);
>> title('Bieu do Nyquist, K=1300')
>> grid,subplot(222),plot(Re2,Im2)
>> title('Bieu do Nyquist, K=5200')
>> grid
>>
```



Đồ thị được trình bày ở hình vẽ sau :



Hình 1.6. Biểu đồ Nyquist

### 1.2.5.3. Độ ổn định tương đối

#### Độ lợi và pha tới hạn

Hàm  $[G_m, P_m, \omega_{pc}, \omega_{gc}] = \text{margin}(\text{mag}, \text{phase}, w)$  có thể được sử dụng với hàm Bode để ước lượng độ lợi và pha tới hạn  $\omega_{pc}$  và  $\omega_{gc}$ .

Ví dụ: Trong ví dụ 1 cho  $K=1300$  hãy tính độ lợi tới hạn  $\omega_{pc}$  và pha tới hạn  $\omega_{gc}$ :

Tập tin **doloi\_pha.m**

% Tính do loi toi han, Wpc va pha toi han, Wgc

k=1300;

num=[k];

den=[1 52 100 0];

w=.1:.1:20;

[mag,phase]=bode(num,den,w);

[Gm,Pm,wpc,wgc]=margin(mag,phase,w);

fprintf('Do loi toi han=%7.3g',Gm),fprintf('Do loi cat w=%7.3g\n',wgc)

fprintf('Pha toi han=%7.3g',Pm),fprintf('Pha cat w=%7.3g\n\n',wpc)

Kết quả thực hiện :

>> doloi\_pha

Do loi toi han= 4Do loi cat w= 4.89

Pha toi han= 16.6Pha cat w= 10

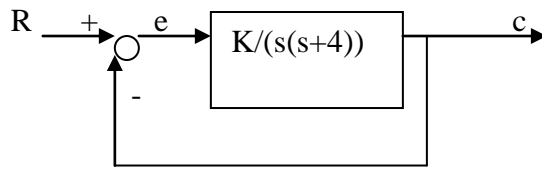
>>

### 1.2.6. Phân tích quỹ đạo nghiệm số

Xem tài liệu [1] về 11 quy tắc vẽ quỹ đạo nghiệm số.

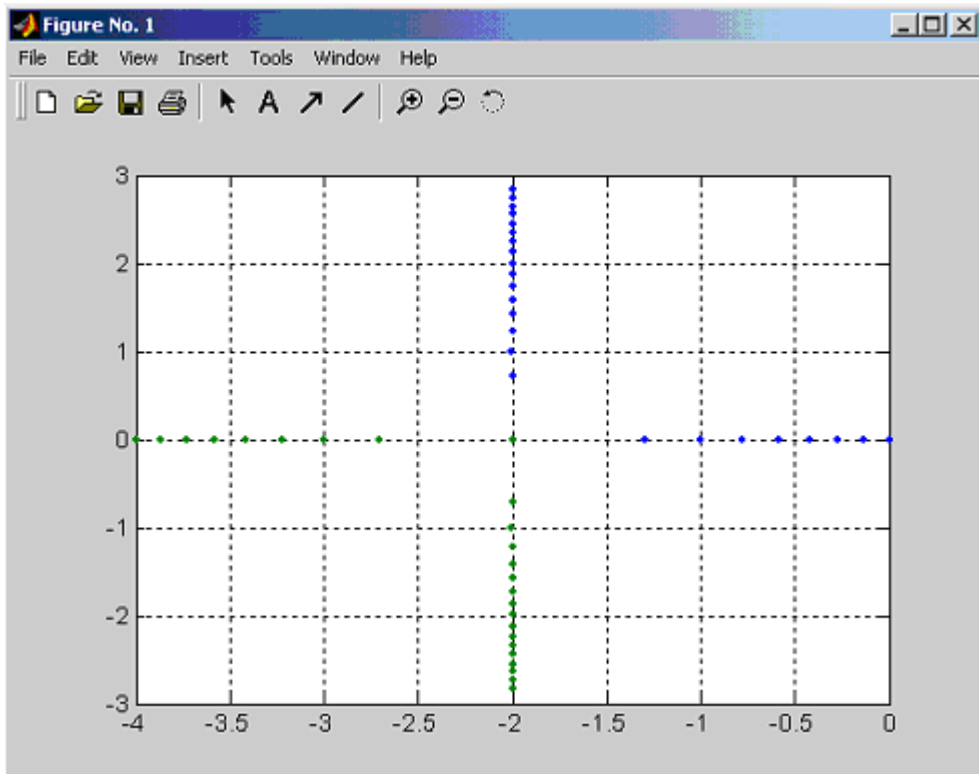
Hàm rlocus : vẽ quỹ đạo nghiệm

Thí dụ : Tìm quỹ đạo nghiệm số ( $K > 0$ ) cho hệ thống đơn giản



Hàm truyền vòng hở:  $G(s)H(s) = K/(s(s+4))$

- » axis([-6,0,-3,3])
- » axis('square')
- » K=0:0.5:12;
- » num=1
- » den=[1 4 0]
- » r=rlocus(num,den,K);
- » plot(r, 'o')
- » grid



Hình 1.7. Quỹ đạo nghiệm số

### 1.3.Thí nghiệm:

1.3.1.Hàm truyền đạt, phương pháp biến trạng thái:

**Mục đích:** biểu diễn hệ bằng hàm truyền và hệ phương trình trạng thái.

**Thí nghiệm:**

1.Cho hàm truyền  $\frac{10}{s^3 + 8.5s^2 + 20.5s + 15}$ . Biểu diễn hệ dùng hàm truyền và hệ phương trình biến trạng thái.

Lệnh Matlab:

» num=[10]

» den=[1 8.5 20.5 15]

» G=tf(num,den)

» [A,B,C,D]=tf2ss(num,den)

A =

```
-8.5000 -20.5000 -15.0000
 1.0000    0    0
    0  1.0000    0
```

B =

```
1
0
0
```

C =

```
0  0  10
```

D =

```
0
```

2. Hệ thống được biểu diễn bằng phương trình biến trạng thái sau:

$$\frac{dx}{dt}=Ax+Bu$$

```
A=[ 0  1  0
    64.4  0 -16
    0  0 -100]
```

```
B=[0
    0
    100]
```

```
y=[1 0 0]x
```

Tìm hàm truyền  $G(s)=Y(s)/X(s)$

» A=[0 1 0;64.4 0 -16;0 0 -100]

» B=[0; 0;100]

» C=[1 0 0]

» D=[0]

» [num,den]=ss2tf(A,B,C,D,1)

num =

```
1.0e+003 *
    0  0.0000  0.0000 -1.6000
```

den =

1.0e+003 \*

0.0010 0.1000 -0.0644 -6.4400

Do đó hàm truyền là :

$$G(s) = (-1600) / (s^3 + 100s^2 - 64.4s - 6440)$$

### 3. **Lệnh ss:**

Lệnh Matlab tạo ra hệ thống mô tả bởi phương trình trạng thái có các ma trận a,b,c,d : a=[0 1;-3 -2]; b=[0;1]; c=[1 0]; d=0

>>PTTT=ss(a,b,c,d)

Biến đổi mô tả toán học từ dạng phương trình trạng thái về dạng hàm truyền : a=[0 1;-3 -2]; b=[0;1]; c=[1 0]; d=0;. Lệnh Matlab là:

>>PTTT=ss(a,b,c,d)

>>g=tf(PTTT)

4. **Dùng tiêu chuẩn Routh-Hurwitz xét ổn định** các hệ thống cho 3 hàm truyền được khảo sát trong thí nghiệm.

$$\text{Hệ thống 1: } G(s) = \frac{20}{s(1+0,1s)(1+0,5s)}$$

$$\text{Hệ thống 2: } G(s) = \frac{K(s+5)}{s(s+2)(s+1)} \text{ khi } K=1.$$

$$\text{Hệ thống 3: } G(s) = \frac{K(s+1)}{s(s+3)(s^2+8s+20)} \text{ khi } K=1.$$

### 1.3.2. Khảo sát hệ thống tự động dùng giản đồ Bode và Nyquist

#### 1.3.2.1. Khảo sát hệ thống tự động dùng giản đồ Bode:

**Mục đích:** Vẽ biểu đồ biên độ và pha theo tần số hàm truyền hệ hở của hệ thống phản hồi âm đơn vị. Trên đồ thị đo độ dự trữ pha, biên độ suy ra sự ổn định và chất lượng hệ thống kín.

#### **Thí nghiệm :**

1. Khảo sát hệ thống có hàm truyền hở:

$$G(s) = \frac{20}{s(1+0,1s)(1+0,5s)}$$

$$\text{Ta có: } G(s) = \frac{20}{(0,05s^3 + 0,6s^2 + s)}$$

Hàm Bode (num,den) vẽ bode biên và bode pha.

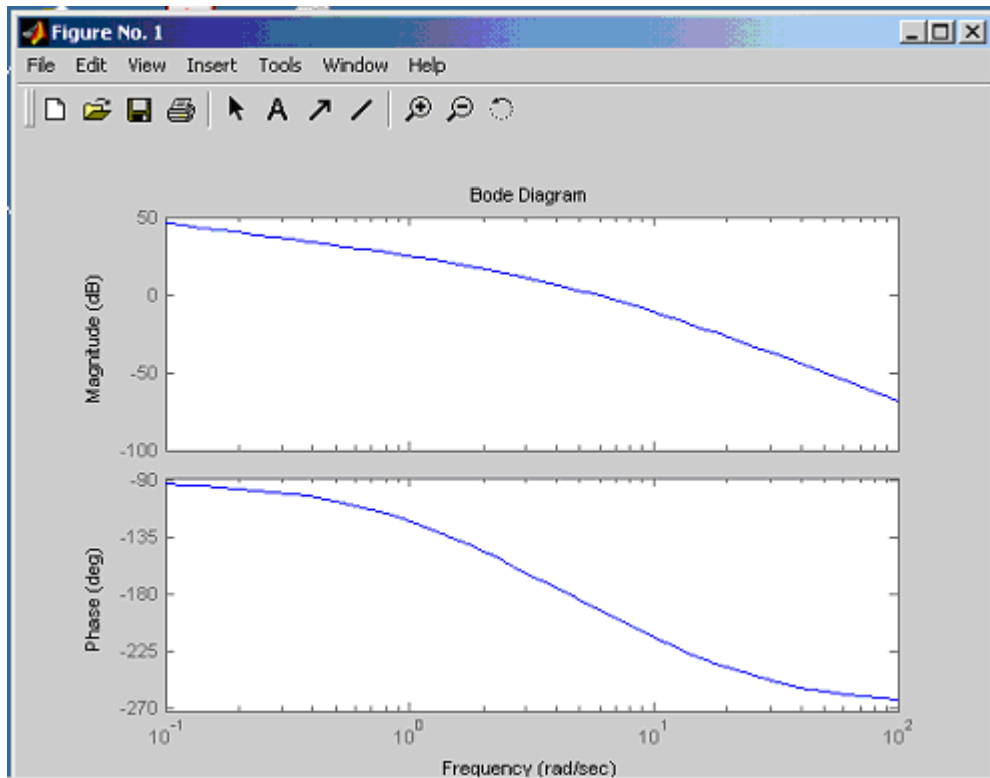
Lệnh Matlab:

>> num=[20]

>> den=[0.05 0.6 1 0]

>> bode(num,den)

>>



Hình 1.8. Biểu đồ Bode

2. Cho hệ có hàm truyền là :

$$G(s) = (10s^2 + 20s - 10) / (s^3 + 2s^2 - s + 1)$$

Lệnh Matlab:

```
>> num=[10 20 -10]
```

```
>> den=[1 2 -1 1]
```

```
» bode(num,den)
```

3. Cho hàm truyền  $G(s) = \frac{s+4}{s^2+3s+2}$ , lệnh MATLAB vẽ đồ thị biên độ-tần số và pha-tần số

là:

```
num=[1 4]
```

```
den=[1 3 2]
```

```
bode(num,den)
```

4. Xét hệ sau:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -25 & -4 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 25 \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Hệ có một ngõ vào  $u$  và một ngõ ra  $y$ . Bằng cách dùng lệnh `bode(A,B,C,D)` của Matlab vẽ biểu đồ bode biên độ và pha.

Lệnh Matlab:

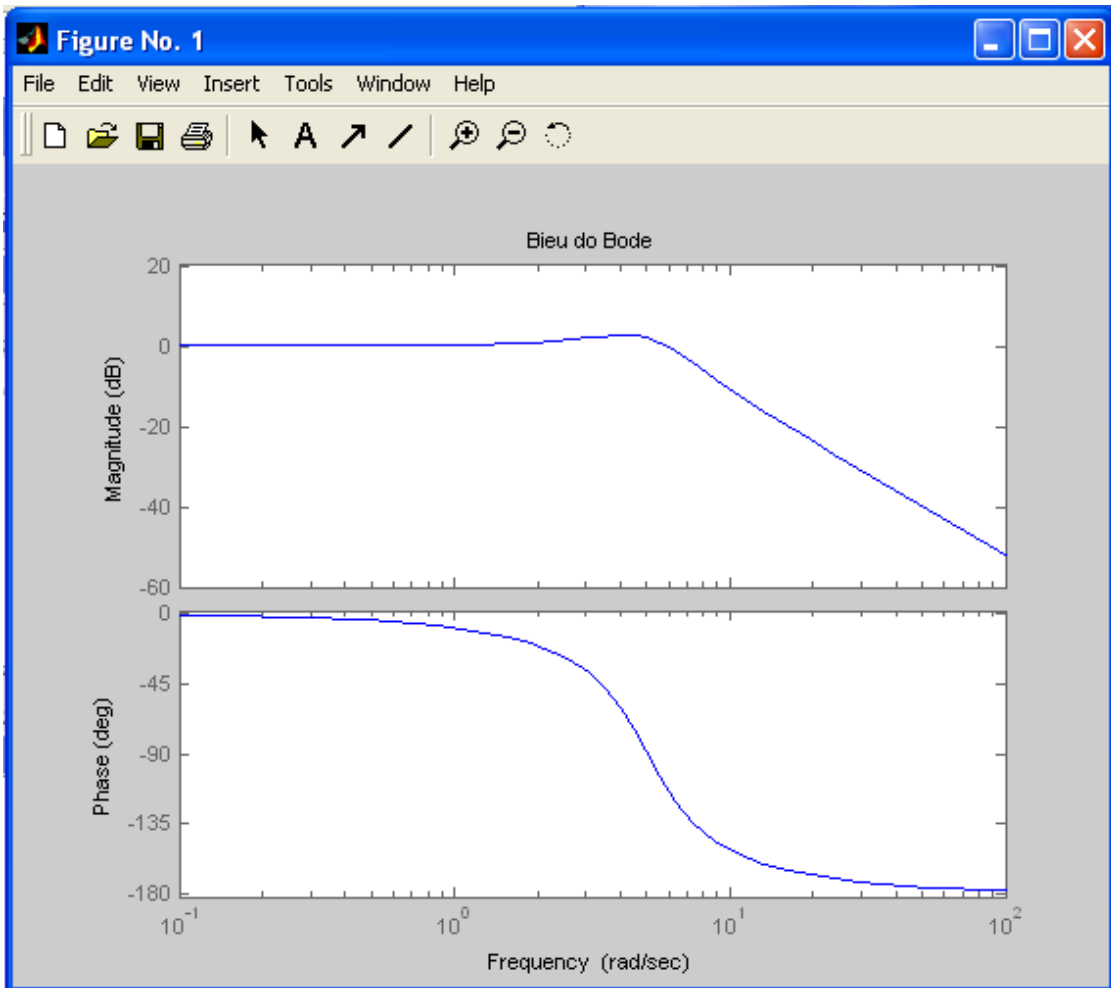
```
>> A=[0 1;-25 -4];
```

```
>> B=[0;25];
```

```
>> C=[1 0];
```

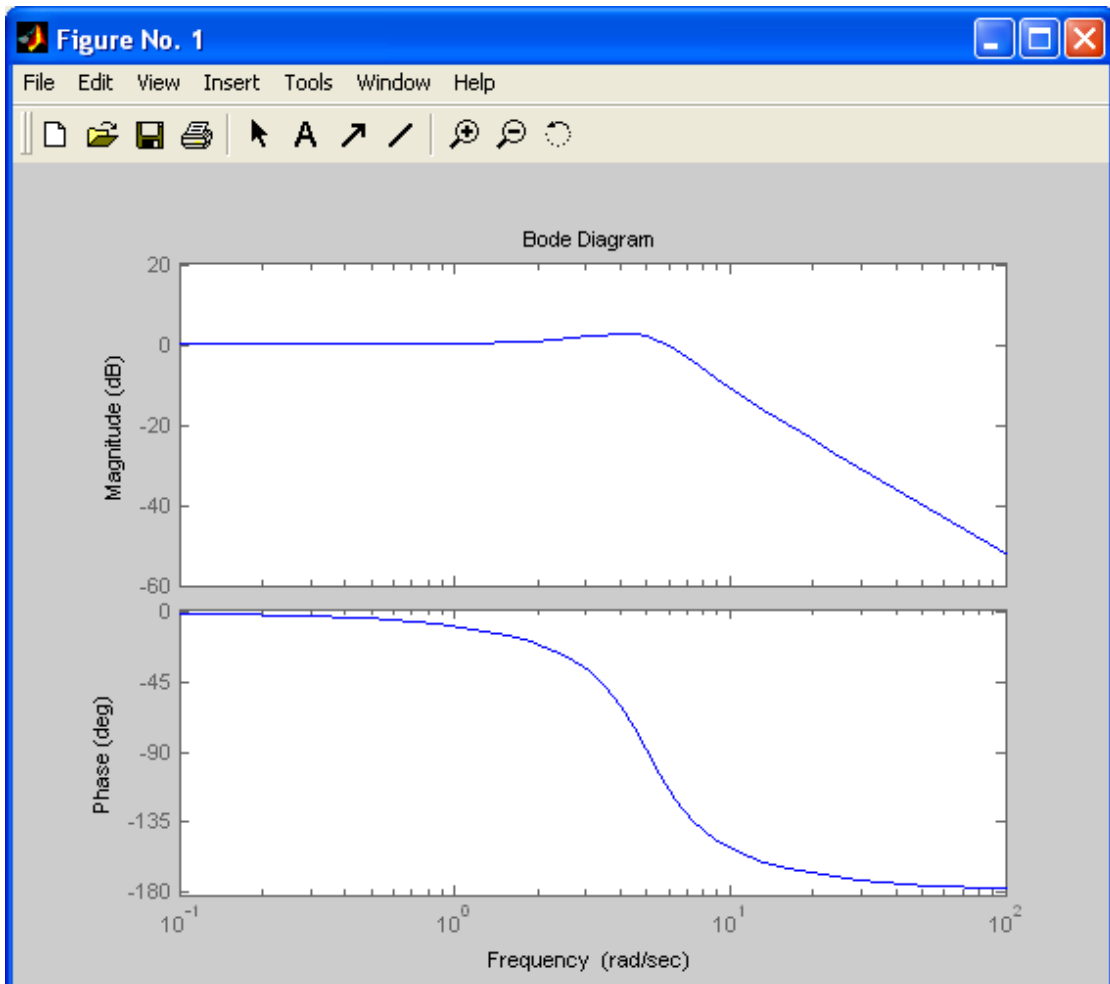
```
>> D=[0];
```

```
>> bode(A,B,C,D)  
>> title('Bieu do Bode')
```



Hình 1.9. Biểu đồ Bode

Nếu ta thay lệnh `bode(A,B,C,D)` bằng lệnh `bode(A,B,C,D,1)` thì MATLAB tạo ra giản đồ bode giống như ví dụ trên.



Hình 1.10

### 1.3.2.2. Khảo sát hệ thống tự động dùng giản đồ Nyquist

**Mục đích:** tương tự như biểu đồ Bode, từ bản in biểu đồ Nyquist của hàm truyền hệ hở của hệ thống phản hồi âm đơn vị, ta cũng có thể đo độ dự trữ về pha, biên độ suy ra sự ổn định chất lượng của hệ thống kín.

**Thí nghiệm:**

1. Xét ổn định hệ thống các hàm truyền hở:

$$G(s) = \frac{K(s+5)}{s(s+2)(s+1)} \quad \text{với } K=1$$

Lệnh Nyquist: vẽ biểu đồ Nyquist

2. Cho hàm truyền đạt của hệ sau:  $G(s) = \frac{30}{s^2 + 4s + 30}$ . Vẽ biểu đồ Nyquist.

Lệnh Matlab:

```
num=30;den=[1 4 30]; G=tf(num,den)
nyquist(G)
```

### 1.3.3. Khảo sát hệ thống dùng phương pháp quỹ đạo nghiệm số:

**Mục đích:**

-Khảo sát đặc tính của hệ thống tuyến tính có hệ số khuếch đại K thay đổi, tìm giá trị K giới hạn K<sub>gh</sub> để hệ thống ổn định. K<sub>gh</sub> là hệ số khuếch đại để hệ thống có nghiệm số nằm trên trục ảo.

-Cũng trong phần này, Ta tính chọn K để hệ thống có cặp nghiệm số không chẵn (các cực của hàm truyền hệ kín ảnh hưởng chủ yếu đến đặc tính quá độ) có thông số cho trước.

Với mỗi hệ thống, ta đều vẽ đặc tính quá độ hệ thống kín với đầu vào hàm nấc để kiểm tra.

**Thí nghiệm:**

1. Vẽ quỹ đạo nghiệm số (QĐNS) của hệ có hàm truyền hở:

$$G(s) = \frac{K(s+1)}{s(s+3)(s^2+8s+20)}, 0 \leq K \leq \infty$$

Tìm Kgh và giá trị K để quá trình quá độ của hệ thống kín có tính dao động với tần số dao động riêng  $\omega_n = 4$ .

Hàm rlocus(num,den,K) dùng để vẽ quỹ đạo nghiệm số.

2. Xem xét hệ thống mà hàm truyền vòng hở  $G(s)H(s)$  là :

$$G(s)H(s) = \frac{K}{s(s+0.5)(s^2+0.6s+10)}$$

$$= \frac{K}{s^4 + 1.1s^3 + 10.3s^2 + 5s}$$

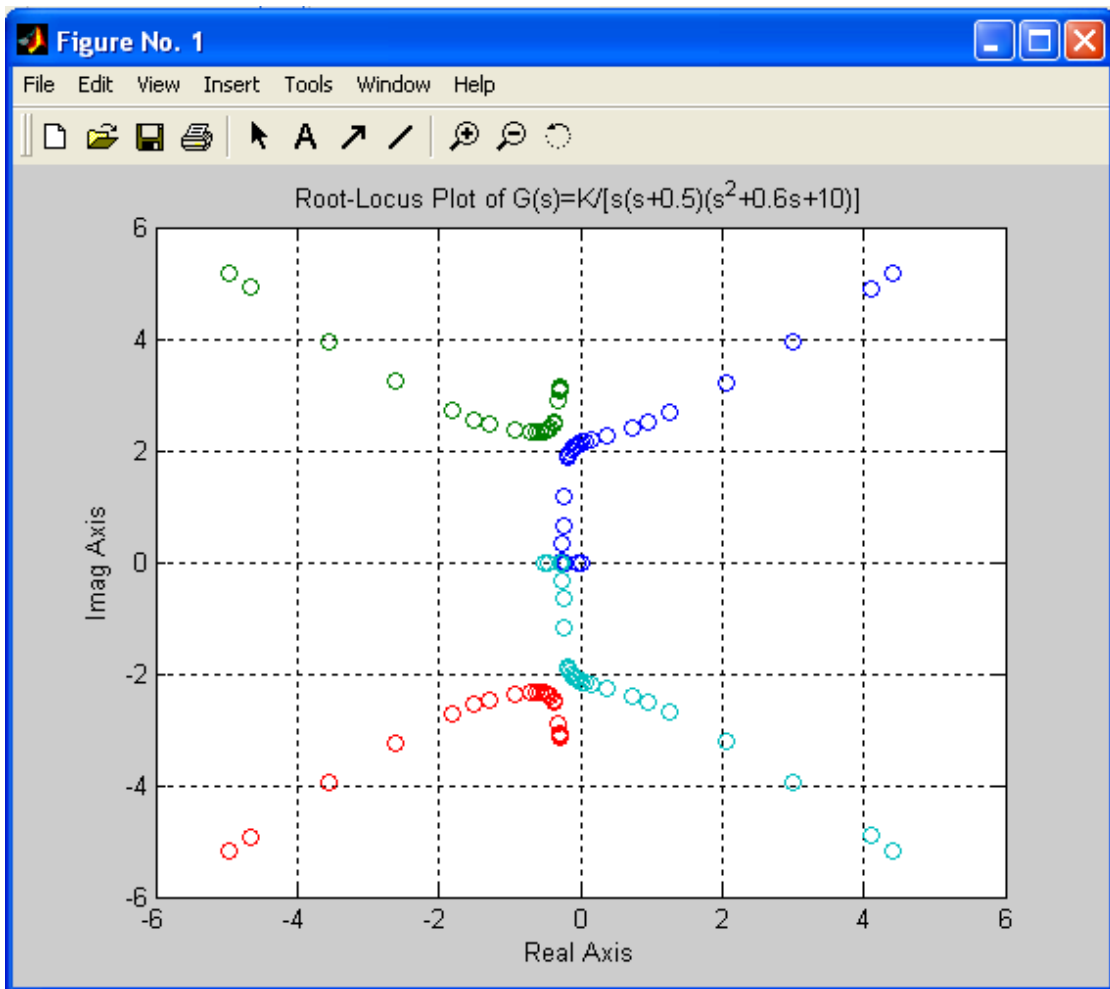
Viết lệnh Matlab vẽ quỹ đạo nghiệm.

Không có zero vòng hở, cực vòng hở là  $s=-0.3+j3.1480$ ,  $s=-0.3-j3.1480$ ,  $s=-0.5$  và  $s=0$ .

Lệnh Matlab :

```
>> num=[0 0 0 0 1];
>> den=[1 1.1 10.3 5 0];
>> r=rlocus(num,den);
>> plot(r,'o')
>> v=[-6 6 -6 6];axis(v)
>> grid
>> title('Root-Locus Plot of G(s)=K/[s(s+0.5)(s^2+0.6s+10)]')
>> xlabel('Real Axis')
>> ylabel('Imag Axis')
>>
```





Hình 1.11. Quỹ đạo nghiệm số bài 2.

3. Xem xét hệ sau:

$$\dot{x} = Ax + Bu$$

$$y = Cx + Du$$

$$u = y - r$$

trong bài này ta vẽ quỹ đạo nghiệm trong không gian trạng thái. Trong ví dụ trên cho các ma trận A, B, C, và D như sau:

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -160 & -56 & -14 \end{bmatrix}, B = \begin{bmatrix} 0 \\ 1 \\ -14 \end{bmatrix}$$

$$C = [1 \ 0 \ 0], D = [0]$$

Dùng Matlab vẽ quỹ đạo nghiệm.

Lệnh Matlab:

Lệnh rlocus(A,B,C,D) vẽ quỹ đạo nghiệm. Lệnh này tương đương với lệnh rlocus(num,den) trong đó num, den đạt được từ lệnh sau:

$$[\text{num}, \text{den}] = \text{ss2tf}(A, B, C, D)$$

như sau:

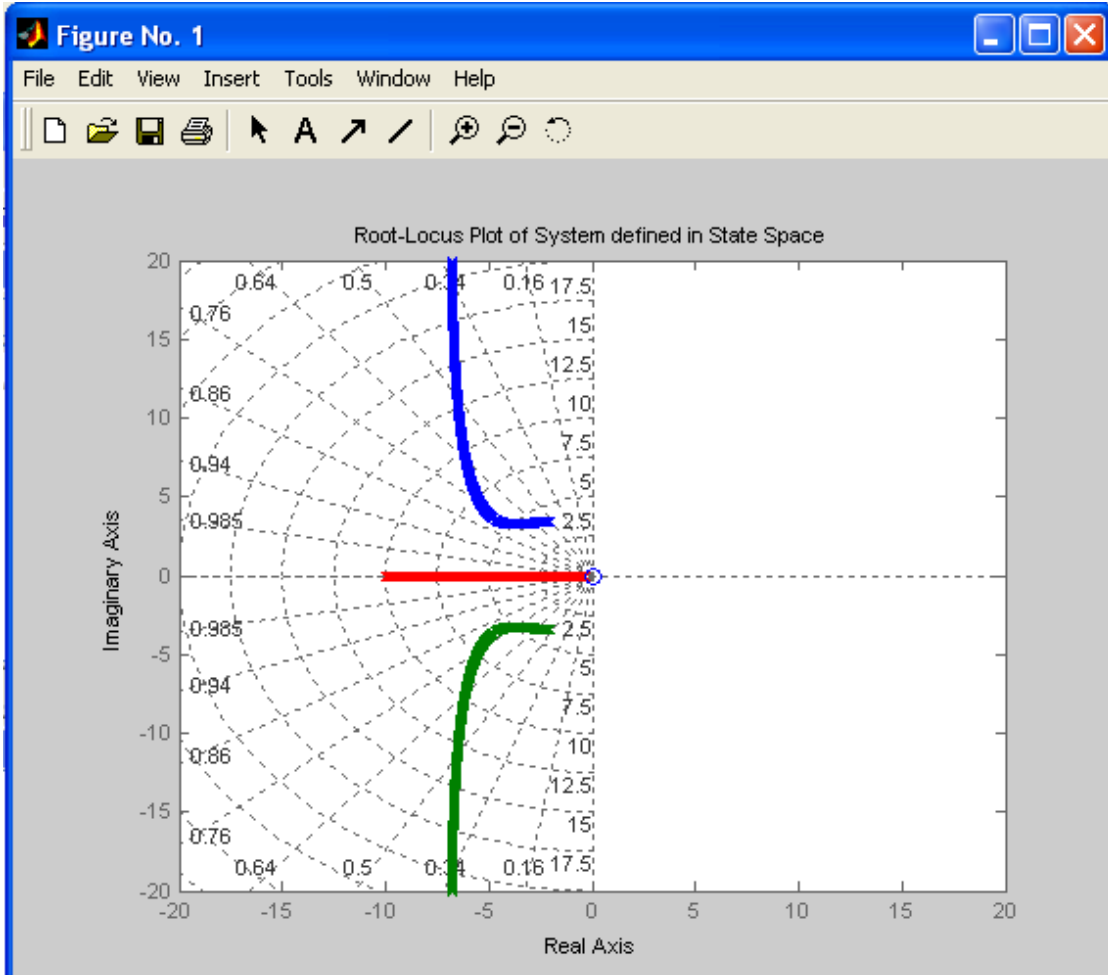
$$\text{num} = [0 \ 0 \ 1 \ 0]$$

$$\text{den} = [1 \ 14 \ 56 \ 160]$$

Lệnh Matlab:

```
>> A=[0 1 0;0 0 1;-160 -56 -14];
```

```
>> B=[0;1;-14];  
>> C=[1 0 0];  
>> D=[0];  
  
>> K=0:0.1:400;  
>> rlocus(A,B,C,D,K);  
>> v=[-20 20 -20 20];axis(v)  
>> grid  
>> title('Root-Locus Plot of System defined in State Space')  
>>
```



Hình 1.12. Quỹ đạo nghiệm số bài 3.

#### 1.4. Kiểm tra đánh giá

## 1.5. Phần mềm ACSYS2002

Phần này mô tả và giới thiệu công cụ (toolbox) ACSYS (Automatic Control Systems).

### 1.5.1. Cài đặt ACSYS

Đầu tiên người sử dụng phải làm theo bước 1 và bước 2 trước khi sử dụng ACSYS:

1. Tạo thư mục C:\ACSYS2002.

2. Sao chép tất cả các tập MATLAB và ảnh từ thư mục ACSYS2002 trong CDROM đi kèm sách vào thư mục C:\ACSYS2002.

Đề chạy ACSYS2002, đơn giản khởi động MATLAB R12 hay cao hơn (Matlab version 6.0 hay cao hơn 6.1, 6.5, 7.0).

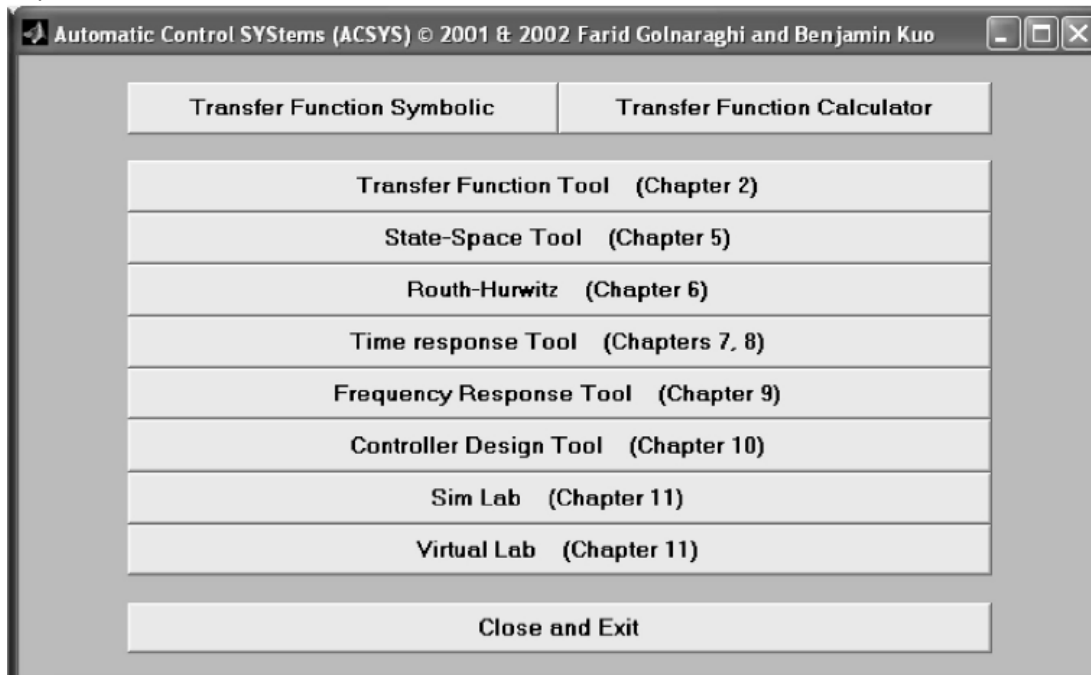
Từ dấu nhắc MATLAB, gõ lệnh:

```
>>cd C:\ACSYS2002
```

Bạn có thể sử dụng Matlab Directory Browser để chuyển đến thư mục ACSYS2002. Từ dấu nhắc ">>" trong cửa sổ lệnh Matlab, ta gõ :

```
>>acsys (hay ACSYS).
```

Hình 1.13 xuất hiện. Dựa trên việc nhấn chuột vào bất cứ nút nào, công cụ tương ứng xuất hiện.



Hình 1.13 : Cửa sổ chính toolbox Automatic Control Systems(ACSYS )

### 1.5.2. Mô tả phần mềm:

Công cụ ACSYS gồm có một số m-files và GUI (graphical user interfaces) để phân tích 10 hàm truyền kỹ thuật điều khiển đơn giản.

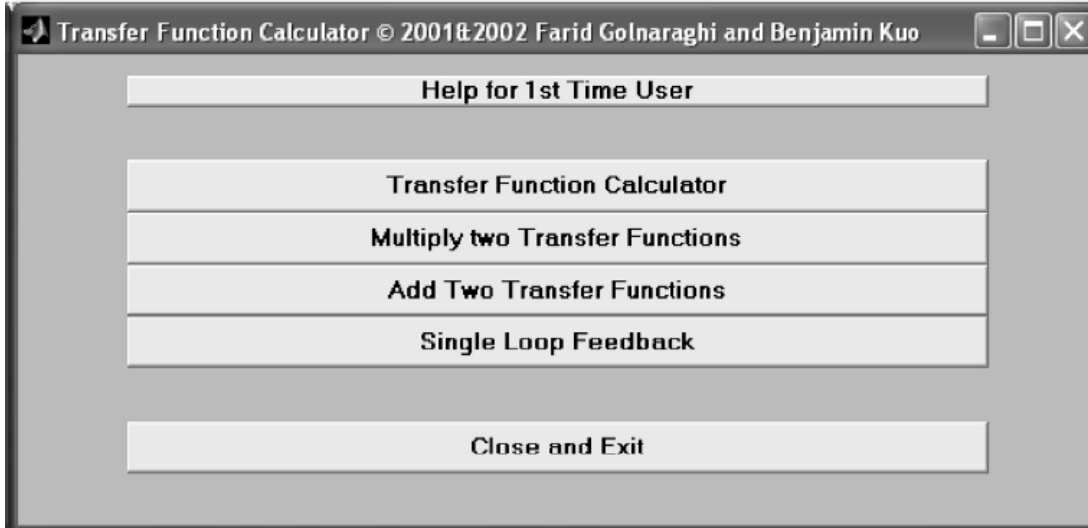
Cụ thể ACSYS gồm có 10 công cụ khác nhau và một công cụ cho chuyển đổi đại lượng kỹ thuật giữa hệ SI và đơn vị khác.

Những thành phần này bao gồm các chủ đề khác nhau trong nghiên cứu hệ thống điều khiển. Chúng bao gồm:

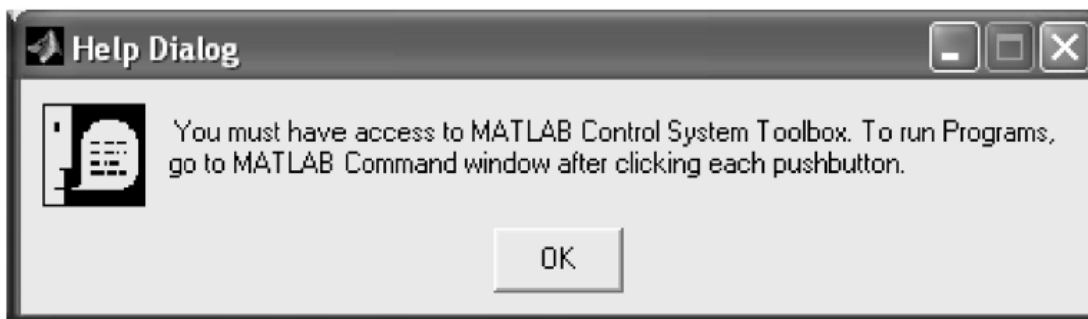
- . Tính toán hàm truyền : tfcal.
- . Mô hình ký hiệu hàm truyền : tfsym.
- . Công cụ hàm truyền : tftool.
- . Công cụ không gian trạng thái : statetool.
- . Công cụ Routh-Hurwitz: stabtool.
- . Công cụ phân tích đáp ứng thời gian: timetool.
- . Công cụ phân tích đáp ứng tần số: freqtool.
- . Công cụ thiết kế bộ điều khiển: controls.

.Thí nghiệm mô phỏng SIM : simlab.

.Thí nghiệm ảo Virtual : virtuallab.



Hình 1.14: Cửa sổ tính toán hàm truyền.



Hình 1.15: Hộp thoại trợ giúp MATLAB cho công cụ tfcal.

Các thành phần ACSYS hay toolbox có thể gọi trực tiếp bằng cách gõ tên ở đầu nhắc lệnh MATLAB.

### 1.5.3.Sử dụng công cụ phân tích không gian trạng thái:

**Mục đích** : sử dụng công cụ phân tích không gian trạng thái để khảo sát hệ phương trình trạng thái, biểu diễn hệ phương trình trạng thái, tính trị riêng và vector riêng, xác định hàm truyền đạt, cực và zero và xét tính điều khiển được và quan sát được.

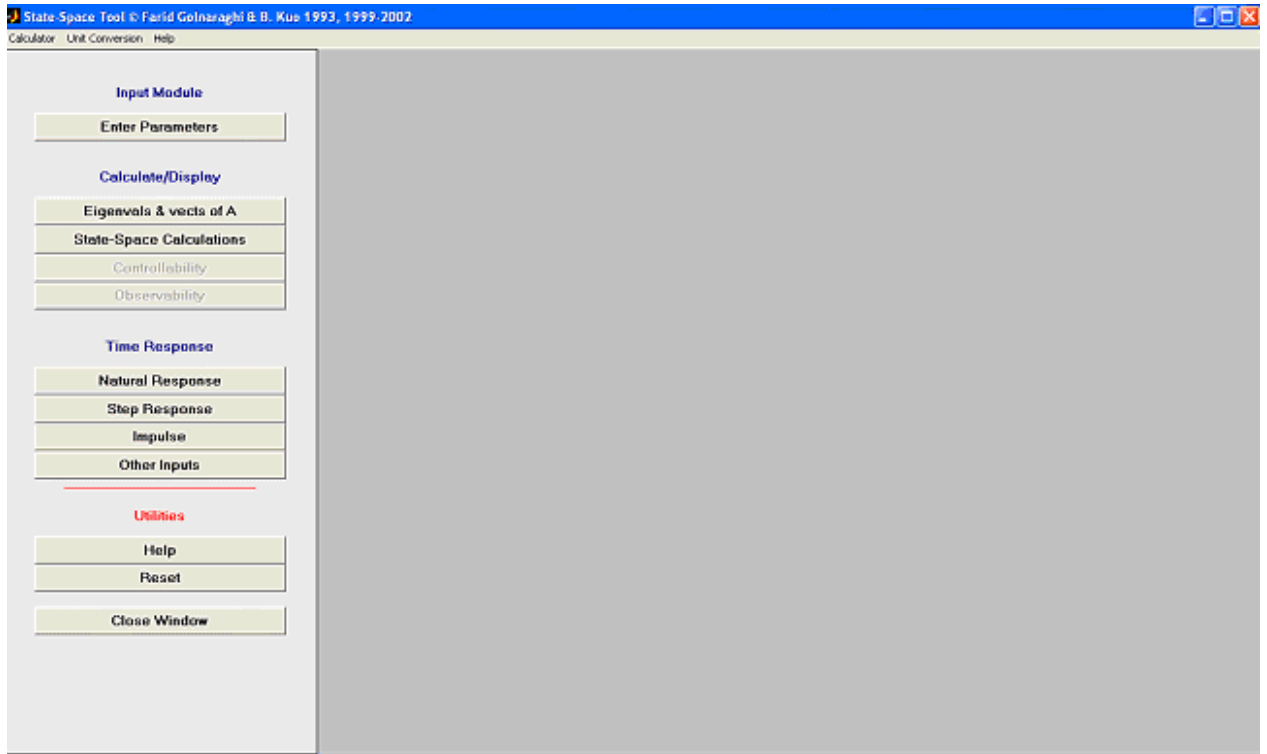
#### **Thí nghiệm:**

Bước 1: Khởi động ACSYS

```
>> acsys
```

Bước 2: Chọn State-Space Tool từ màn hình menu ACSYS

Cửa sổ State-Space Tool xuất hiện:



Hình 1.16

Thí nghiệm : Xét hệ sau:

$$\dot{x}/dt = Ax + Bu$$

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 64.4 & 0 & -16 \\ 0 & 0 & -100 \end{bmatrix}$$

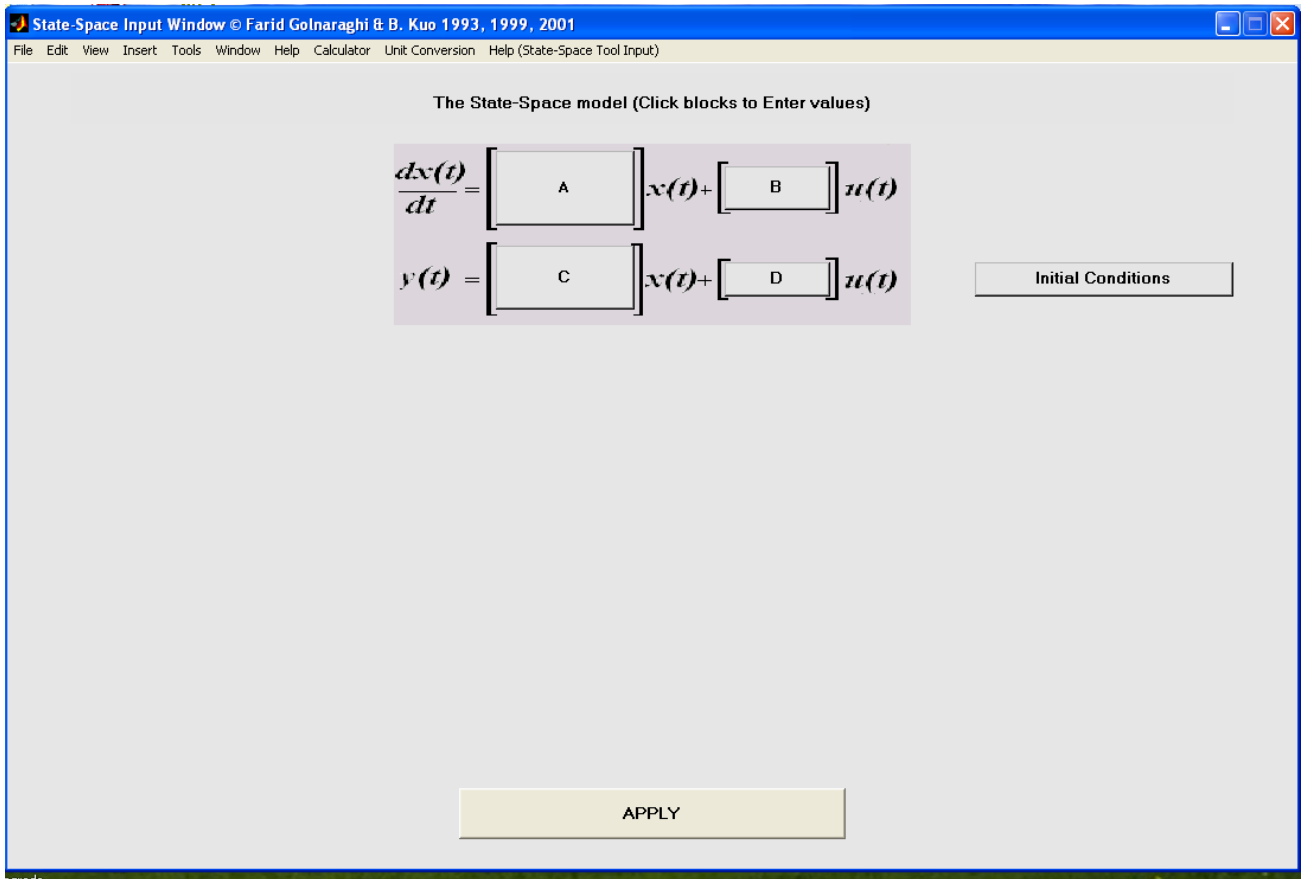
$$B = \begin{bmatrix} 0 \\ 0 \\ 100 \end{bmatrix}$$

$$y = [1 \ 0 \ 0]x$$

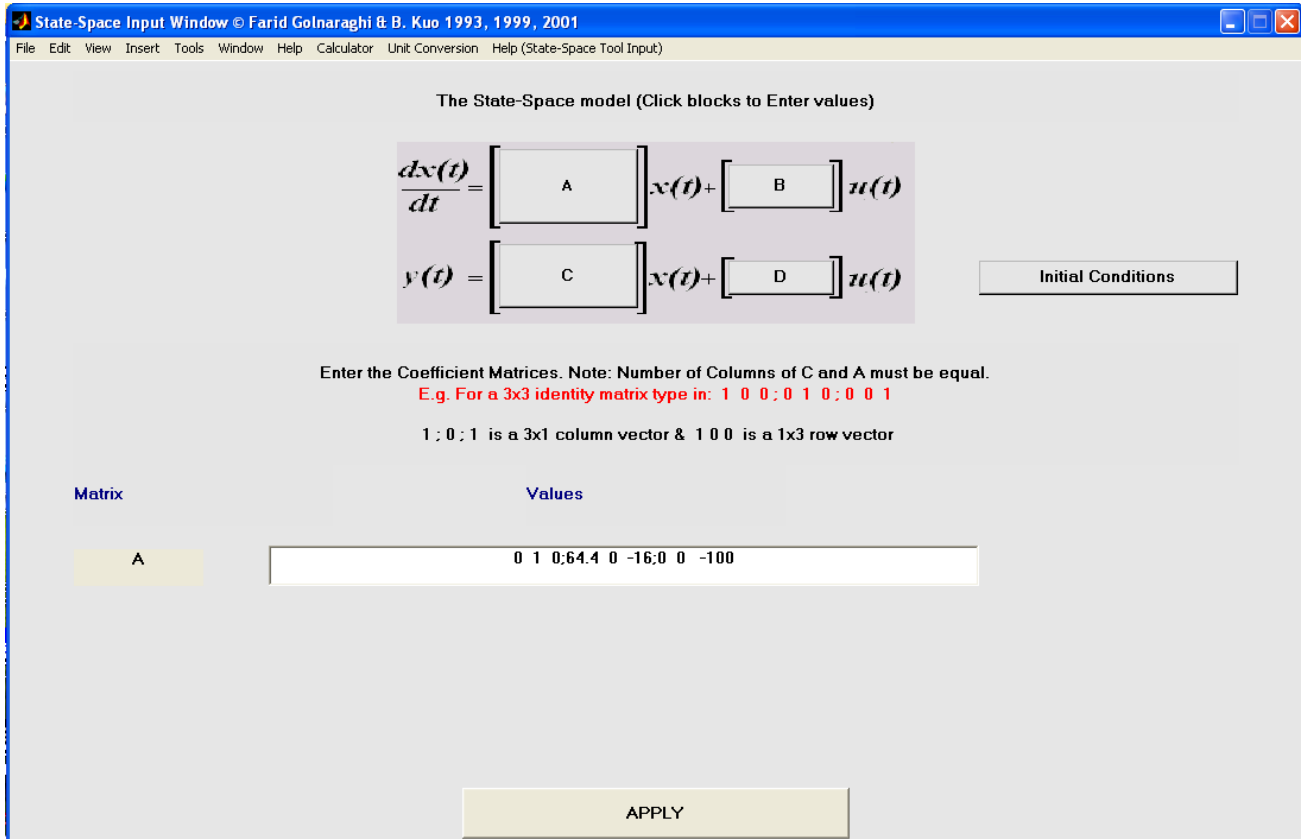
Bước 3: Nhấp chuột vào Enter Parameter để nhập thông số ma trận A và B như sau:

Ma trận A: 0 1 0; 64.4 0 -16; 0 0 -100

Ma trận B : 0 ; 0 ; 100

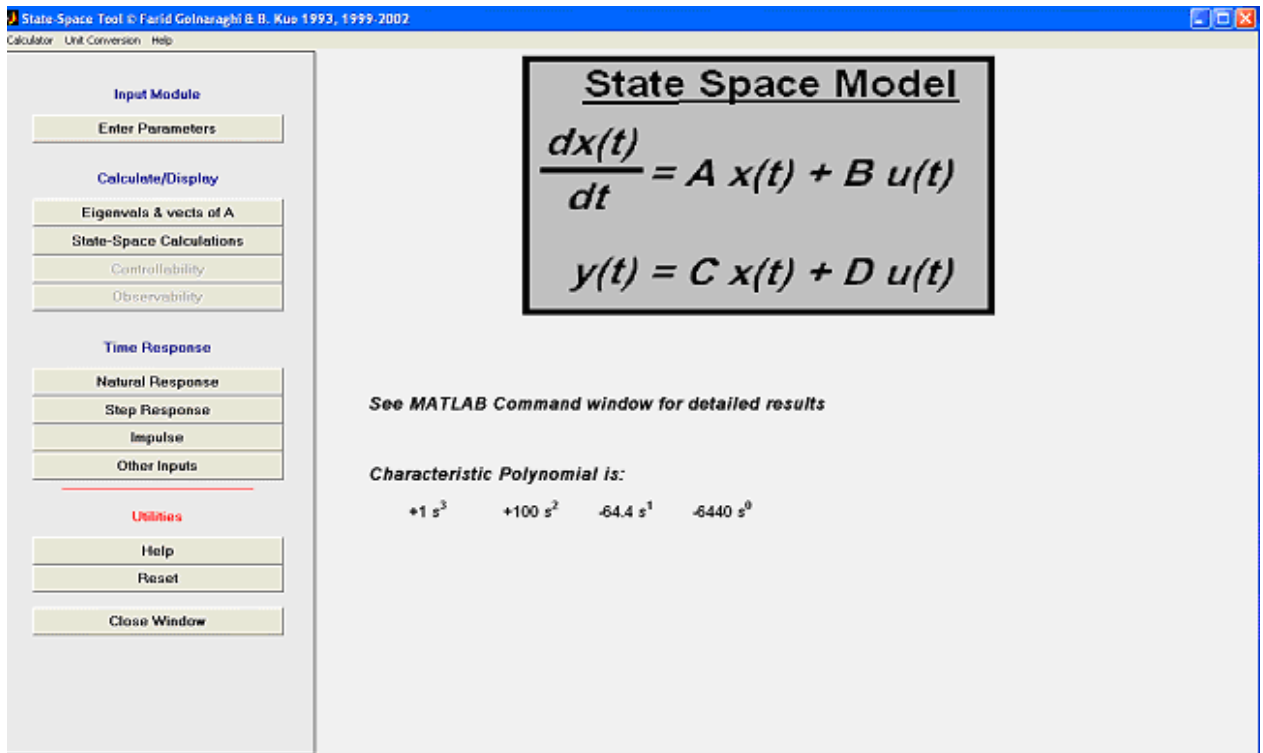


Hình 1.17



Hình 1.18 : Nhập ma trận A trong cửa sổ Input State-Space

Bước 4: Nhập vào “Eigenval & vects of A” (Trị riêng và vector riêng của A”):



Hình 1.19

Cửa sổ lệnh Matlab sau khi ta nhấp vào “Eigenval &vects of A”:  
The A matrix is:

Amat =

$$\begin{bmatrix} 0 & 1.0000 & 0 \\ 64.4000 & 0 & -16.0000 \\ 0 & 0 & -100.0000 \end{bmatrix}$$

Characteristic Polynomial:

ans =

$$s^3 + 100*s^2 - 2265873562520787/35184372088832*s - 6440$$

Eigenvalues of A = diagonal canonical form of A is:

Aeigs =

$$\begin{bmatrix} 8.0250 & 0 & 0 \\ 0 & -8.0250 & 0 \\ 0 & 0 & -100.0000 \end{bmatrix}$$

Eigenvectors are

T =

$$\begin{bmatrix} 0.1237 & -0.1237 & -0.0016 \\ 0.9923 & 0.9923 & 0.1590 \\ 0 & 0 & 0.9873 \end{bmatrix}$$

Bước 5: Nhấp chuột vào State-Space Calculation (tính toán không gian trạng thái). Cửa sổ lệnh Matlab:

State-space model is:

a =

```
      x1  x2  x3
x1    0   1   0
x2 64.4   0 -16
x3    0   0 -100
```

b =

```
      u1
x1    0
x2    0
x3 100
```

c =

```
      x1  x2  x3
y1    1   0   0
```

d =

```
      u1
y1    0
```

Continuous-time model.

Characteristic polynomial:

ans =

$s^3 + 100s^2 - 2265873562520787/35184372088832s - 6440$

Equivalent transfer function model is:

Transfer function:

$4.263e-014 s^2 + 8.527e-014 s - 1600$

-----  
 $s^3 + 100 s^2 - 64.4 s - 6440$

pole, zero form:

Zero/pole/gain:

$4.2633e-014 (s+1.937e008) (s-1.937e008)$

-----  
 $(s+100) (s+8.025) (s-8.025)$

Bước 6: Nhấp chuột vào “Controllability”(tính điều khiển được). Cửa sổ lệnh Matlab:

The controllability matrix [B AB A<sup>2</sup>B ...] is =



Smat =

$$\begin{bmatrix} 0 & 0 & -1600 \\ 0 & -1600 & 160000 \\ 100 & -10000 & 1000000 \end{bmatrix}$$

The system is therefore controllable, rank of S matrix is =

rankS =

3

Mmat =

$$\begin{bmatrix} -64.4000 & 100.0000 & 1.0000 \\ 100.0000 & 1.0000 & 0 \\ 1.0000 & 0 & 0 \end{bmatrix}$$

The controllability canonical form (CCF) Transformation matrix is:

Ptran =

$$\begin{bmatrix} -1600 & 0 & 0 \\ 0 & -1600 & 0 \\ -6440 & 0 & 100 \end{bmatrix}$$

The transformed matrices using CCF are

Abar =

1.0e+003 \*

$$\begin{bmatrix} 0 & 0.0010 & 0 \\ 0 & 0 & 0.0010 \\ 6.4400 & 0.0644 & -0.1000 \end{bmatrix}$$

Bbar =

$$\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

Cbar =

$$\begin{bmatrix} -1600 & 0 & 0 \end{bmatrix}$$

Dbar =

$$0$$

Bước 7: Nhấn chuột vào “Observability”(tính quan sát được). Cửa sổ lệnh Matlab:

The observability matrix (transpose:[C CA CA^2 ...]) is =

Vmat =

```
1.0000    0    0
    0 1.0000    0
64.4000    0 -16.0000
```

The System is therefore observable, rank of V matrix is =

rankV =

3

Mmat =

```
-64.4000 100.0000 1.0000
100.0000 1.0000    0
1.0000    0    0
```

The observability canonical form (OCF) Transformation matrix is:

Qtran =

```
0    0 1.0000
0 1.0000 -100.0000
-0.0625 6.2500 -625.0000
```

The transformed matrices using OCF are

Abar =

1.0e+003 \*

```
0.0000 -0.0000 6.4400
0.0010 -0.0000 0.0644
0 0.0010 -0.1000
```

Bbar =

```
-1600
0
0
```

Cbar =

```
0 0 1
```

Dbar =

0

>>

Tham khảo:

- [1]. B.C. Kuo, M.F. Golnaraghi, Automatic Control Systems, Wiley, 2003
- [2]. Appendix K : ACSYS2002: Description of the Software, của sách Automatic Control Systems, tác giả : B. C. Kuo và M.F. Golnaraghi, 8<sup>th</sup> ed. ,NXB : Wiley, 2003.

## Bài 2

# THIẾT KẾ HỆ THỐNG TỰ ĐỘNG DÙNG MATLAB

### 2.1. Mục tiêu

Trong bài thí nghiệm này sinh viên sẽ tìm hiểu cách thiết kế một bộ hiệu chỉnh sớm trễ pha theo phương pháp dùng QĐNS theo yêu cầu chất lượng cho trước. Matlab hỗ trợ một công cụ rất mạnh dùng để thiết kế hệ thống điều khiển tuyến tính một đầu vào một đầu ra đó là công cụ **sisotool**. Dựa vào công cụ này, sinh viên sẽ thiết kế được bộ hiệu chỉnh sớm trễ pha và nhiều bộ hiệu chỉnh khác một cách dễ dàng nhờ giao diện dễ hiểu, dễ sử dụng và hỗ trợ nhiều chức năng mạnh có trong **sisotool**.

### 2.2. Nội dung

#### 2.2.1. Công cụ Sisotool:

Để thực hiện các yêu cầu trong bài thí nghiệm này, sinh viên cần chuẩn bị kỹ và hiểu rõ trình tự các thao tác tiến hành thiết kế một bộ điều khiển trong Sisotool.

Sinh viên cần tham khảo phần phụ lục của chương 6 trong sách Lý thuyết điều khiển tự động [1] để làm quen và hiểu rõ trình tự các bước thiết kế một bộ điều khiển trong Sisotool. Chú ý, sinh viên phải đọc kỹ phần này để nắm rõ trình tự thiết kế. Nếu sinh viên không hiểu kỹ thì sẽ không thực hiện được các yêu cầu trong bài thí nghiệm này.

Để kích hoạt công cụ Sisotool, từ cửa sổ Command Window gõ lệnh `sisotool`. Tiến hành thao tác từ bước 1 đến bước 3, cửa sổ SISO Design Tool xuất hiện như sau:

Trình tự thiết kế như sau:

Bước 1: Khai báo đối tượng điều khiển

$G = \text{tf}(20, \text{conv}([1 \ 1 \ 0], [1 \ 2]))$

Transfer function:

20

-----  
 $s^3 + 3s^2 + 2s$

$H = \text{tf}(1, 1)$

Bước 2: Kích hoạt SiSotool

Cửa sổ SISO Design Tool xuất hiện

Bước 3: Nhập đối tượng điều khiển vào sisotool.

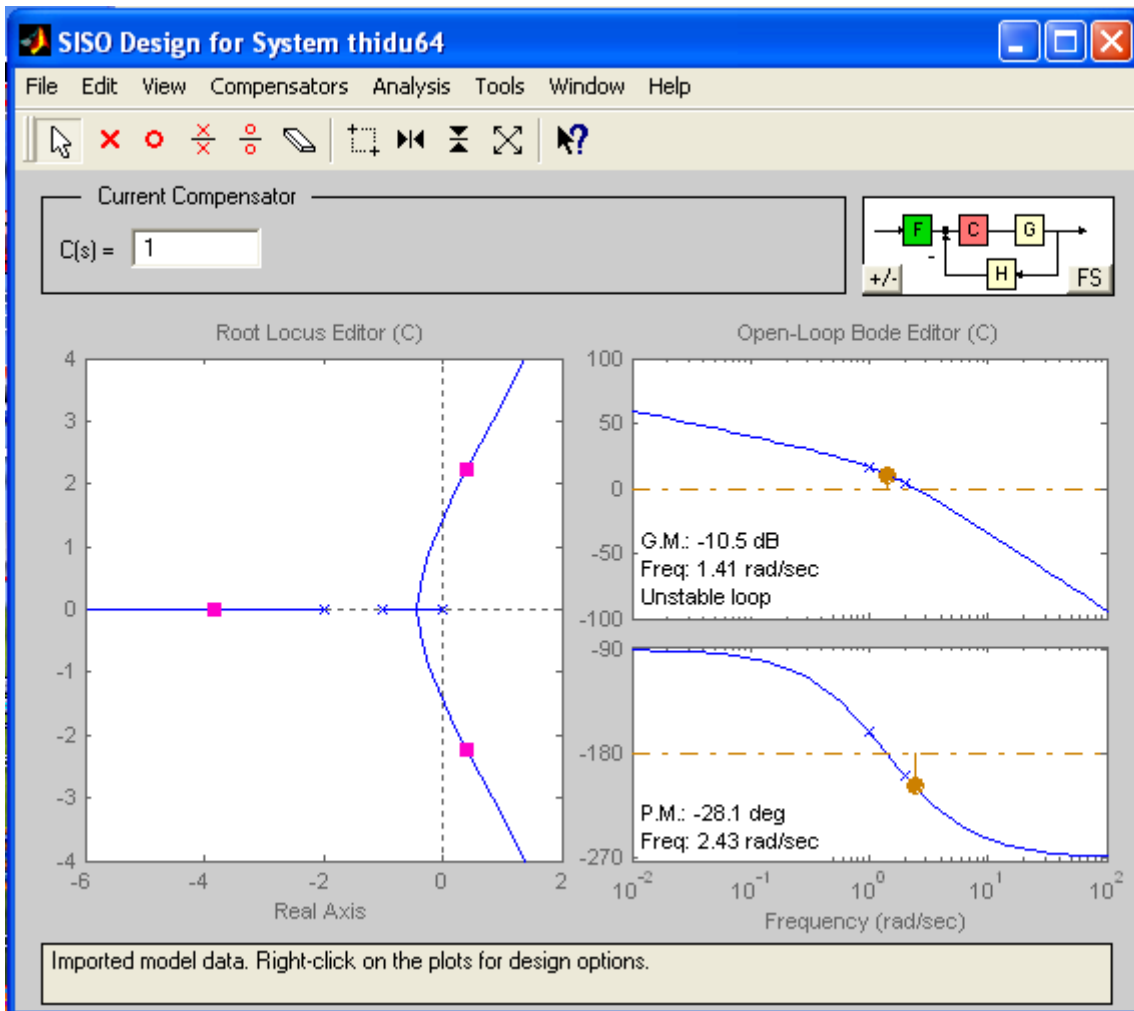
Trong cửa sổ SISO Design Tool chọn [File] → [Import...]. Cửa sổ Import Data xuất hiện.

Đặt tên hệ thống tùy ý, như là `thidu64`. Ban đầu tất cả các khối trong hệ thống điều khiển đều có hàm truyền bằng 1, ta thay đổi đối tượng điều khiển (plant) là  $G$ , cảm biến (sensor) là  $H$ , bộ lọc  $F$  (prefilter) bằng 1, khâu hiệu chỉnh (compensator)  $C$  chưa thiết kế nên cũng bằng 1. Sau khi thực hiện xong nhấp chuột vào nút Ok.

Bước 4: Khảo sát hệ thống trước khi hiệu chỉnh.

Bước 5: Thiết kế khâu hiệu chỉnh.

Bước 6: Kiểm tra lại đáp ứng của hệ thống.



Hình 2.1

Cửa sổ gồm các vùng:

-Vùng hiển thị sơ đồ cấu trúc của hệ thống đang thiết kế. Có thể thay đổi cấu trúc bằng cách kích chuột vào nút [+/-] và [FS] ở góc trái bên dưới. trong bài thí nghiệm này ta sử dụng cấu trúc như hiển thị.

G: đối tượng điều khiển(plant).

H: cảm biến hồi tiếp(sensor).

F: bộ lọc(prefilter).

C: bộ hiệu chỉnh cần thiết kế.

-Hàm truyền của bộ hiệu chỉnh C(s).

-Cửa sổ hiển thị kết quả trong quá trình thao tác.

-Quỹ đạo nghiệm số của hệ thống vòng kín sau khi hiệu chỉnh .

-Biểu đồ Bode biên độ và pha của hệ thống vòng hở sau khi hiệu chỉnh.

### 2.2.2. Thiết kế hệ thống điều khiển dùng phương pháp quỹ đạo nghiệm số bằng Matlab

#### 1. Thiết kế quỹ đạo nghiệm : dùng hàm `sisotool.m` hoặc `rltool.m`

Thí dụ: Cho hệ thống hồi tiếp âm đơn vị có hàm truyền vòng hở

$$G_p(s) = \frac{1}{s(s+1)(s+4)}$$

Dùng `sisotool.m`:

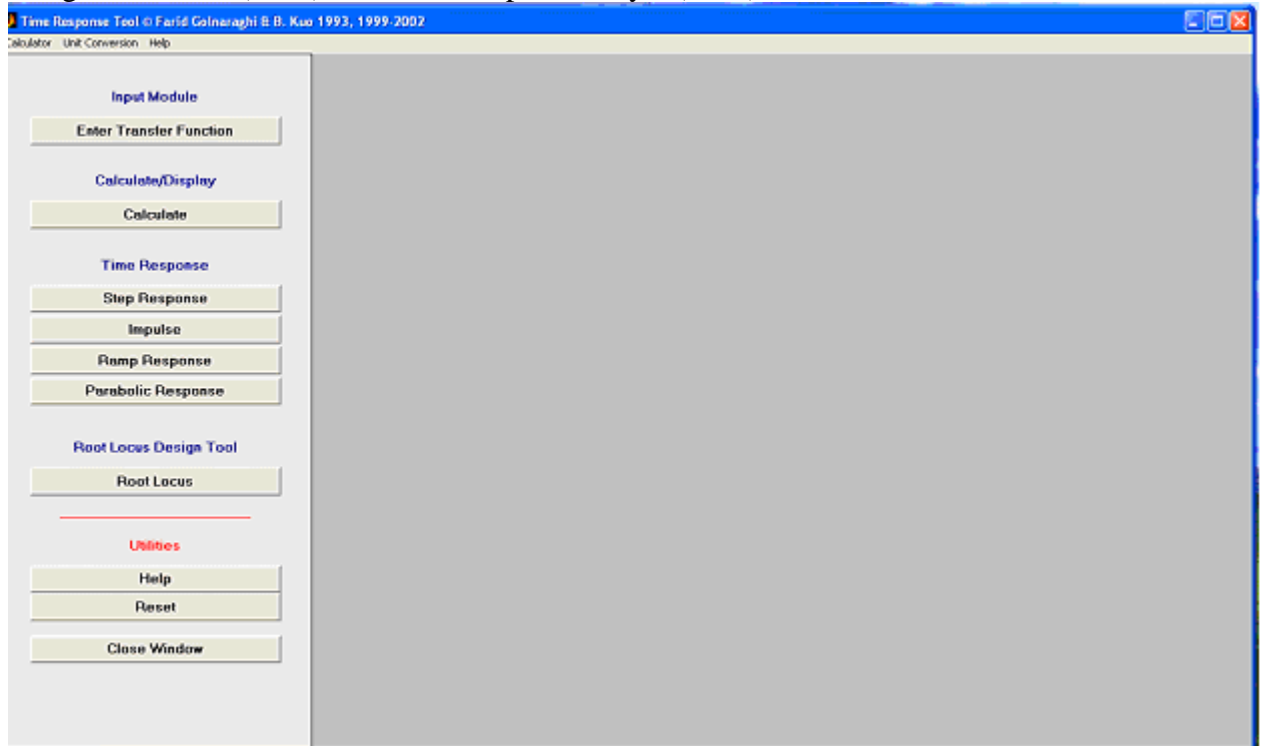
```
>>Num=1;
```

```
>>Den=[1 5 4 0];  
>> sysc=tf(num,den);  
>>sisotool('rlocus', sysc)  
hoặc dùng rltool.m:  
>>Num=1;  
>>Den=[1 5 4 0];  
>> sysc=tf(num,den);  
>>rltool(sysc)
```

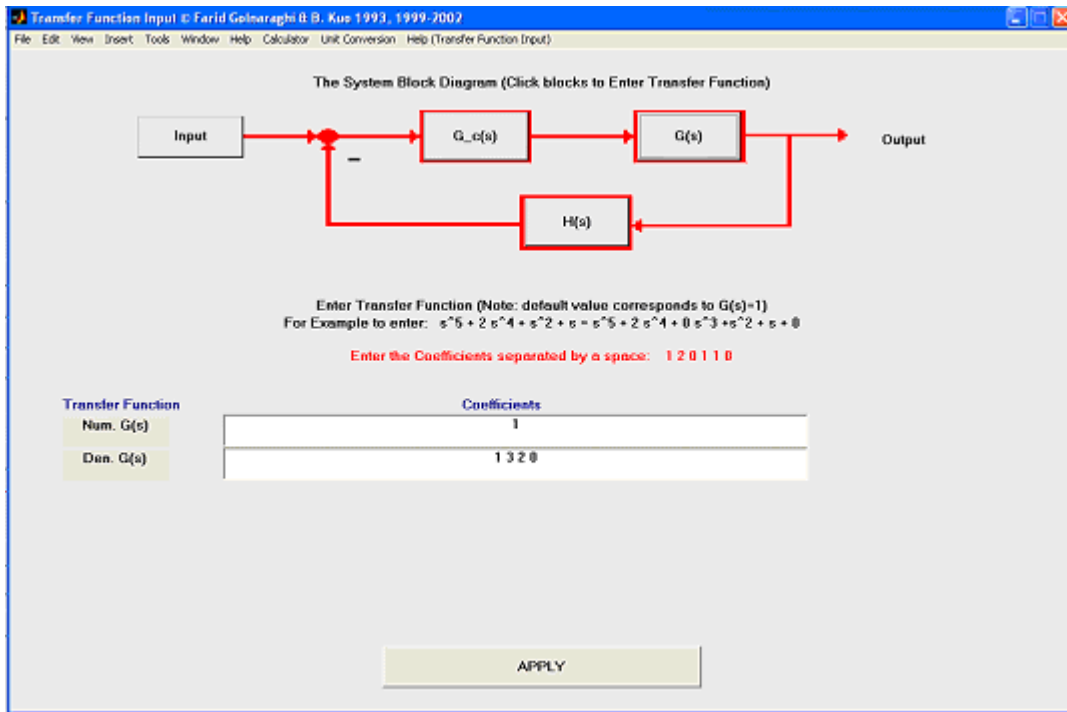
## 2. Quỹ đạo nghiệm số với toolbox ACSYS:

Công cụ timetool: đạt đáp ứng thời gian của hệ thống điều khiển và giới thiệu công cụ thiết kế quỹ đạo nghiệm số. Công cụ timetool có thể dùng để giải các bài toán bao gồm xây dựng quỹ đạo nghiệm số. Bạn có thể dùng công cụ tftool để chuyển hàm truyền từ dạng cực-zero sang dạng đa thức.

Công cụ timetool (H2.2) và cửa sổ nhập hàm truyền(H2.3):



Hình 2.2



Hình 2.3 : Cửa sổ nhập hàm truyền

Xét hàm truyền hở của hệ thống điều khiển vòng kín đơn:

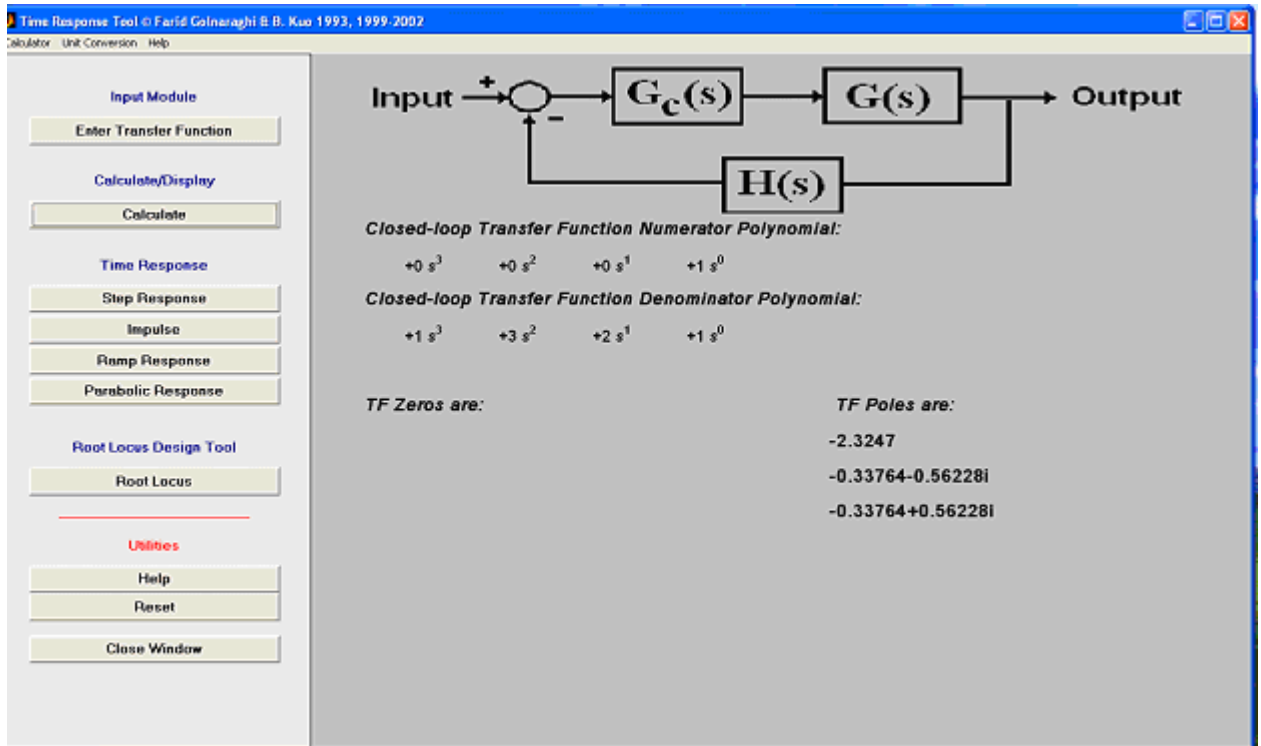
$$G(s)H(s) = \frac{K}{s(s+1)(s+2)} = \frac{K}{s^3 + 3s^2 + 2s}$$

Để tìm quỹ đạo nghiệm, kích hoạt timetool và nhấn vào nút “Enter Transfer Function”. Nhập giá trị của G(s) và H(s). Có thể gán H(s)=K, nghĩa là K=1, trong trường hợp này G(s) là:

$$G(s) = \frac{1}{s^3 + 3s^2 + 2s}$$

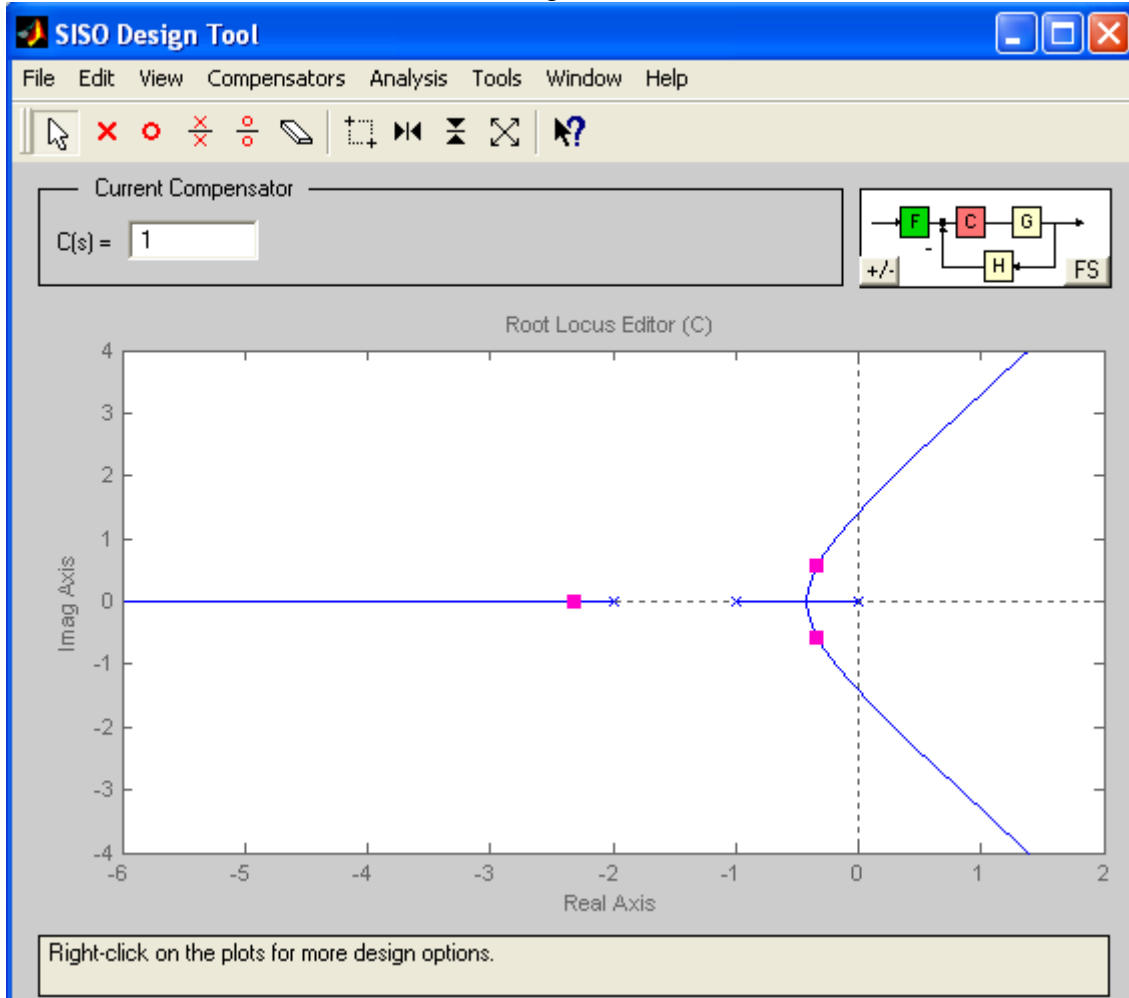
G\_c(s)=1, H(s)=1, K=1. Hình vẽ trên minh họa modul nhập hàm truyền sau khi nhập giá trị G(s). Tiếp theo nhấn nút “APPLY” để trở về cửa sổ Time Response Analysis, nhấn nút “Calculate” để tính hàm truyền.

Để vẽ quỹ đạo nghiệm, ta nhấn “Root Locus”. Hình vuông đồ định nghĩa cực vòng kín, ta có thể xem bằng cách kích hoạt Closed-Loop Pole Viewer từ menu View.



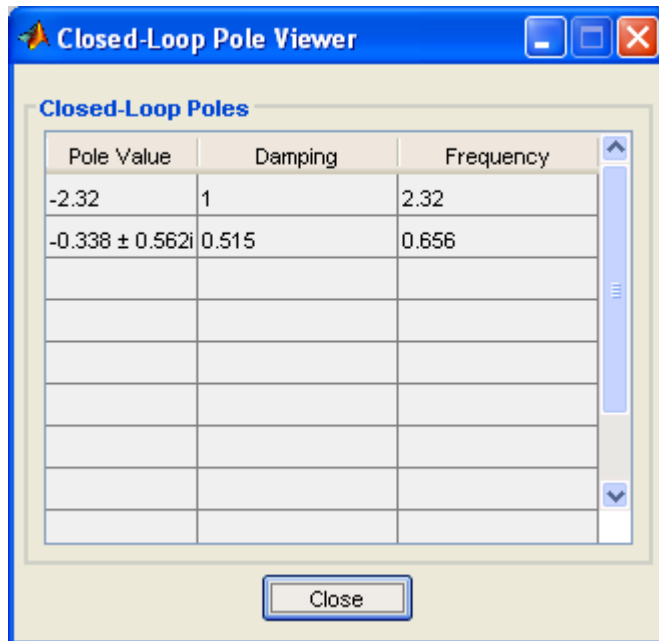
Hình 2.4: Tính toán hàm truyền

Nhấn vào “Root Locus”, kích hoạt SISO design tool của Matlab.



Hình 2.5: Quỹ đạo nghiệm số





Hình 2.6: Cực của hệ thống vòng kín với K=1.

Cửa sổ lệnh Matlab thể hiện hàm truyền :

```
>> acsys
G=
```

Transfer function:

$$\frac{1}{s^3 + 3s^2 + 2s}$$

```
Gc=
```

Transfer function:

$$1$$

```
H=
```

Transfer function:

$$1$$

```
G*G_c ==>open loop
```

Transfer function:

$$\frac{1}{s^3 + 3s^2 + 2s}$$

```
G*G_c*H ==>loop
```

Transfer function:

$$\frac{1}{s^3 + 3s^2 + 2s}$$

```
G*G_c/(1+G*G_c*H) ==> closed loop
```

Transfer function:

$$\frac{1}{s^3 + 3s^2 + 2s + 1}$$

system Coefficients are  
Closed Loop TF in zero/pole format

Zero/pole/gain:

$$\frac{1}{(s+2.325)(s^2 + 0.6753s + 0.4302)}$$

System Zeros are

zeroTF =

Empty matrix: 1-by-0

System Poles are

poleTF =

-2.3247      -0.3376 - 0.5623i   -0.3376 + 0.5623i

>>

### 2.2.3. Thiết kế ở miền tần số

Ví dụ : Vẽ biểu đồ Bode, độ lợi pha tới hạn của hệ thống điều khiển.  
Cho hệ thống hồi tiếp âm đơn vị, có hàm truyền vòng hở

$$G_p(s) = \frac{8}{s(s+1)(s+4)} = \frac{8}{s^3 + 5s^2 + 4s}$$

Dùng hàm sisotool.m để thiết kế tần số.

>> num=1;

>> den=[1 5 4 0];

>> sysc=tf(num,den);

>>sisotool('bode', sysc)

**Thí dụ :**

Cho hàm truyền hở :  $G_p(s) = \frac{1}{s(s+1)(s+4)}$

Lệnh Matlab:

>> num=1

num =  
1

```
>> den=[1 5 4 0]

den =

    1    5    4    0

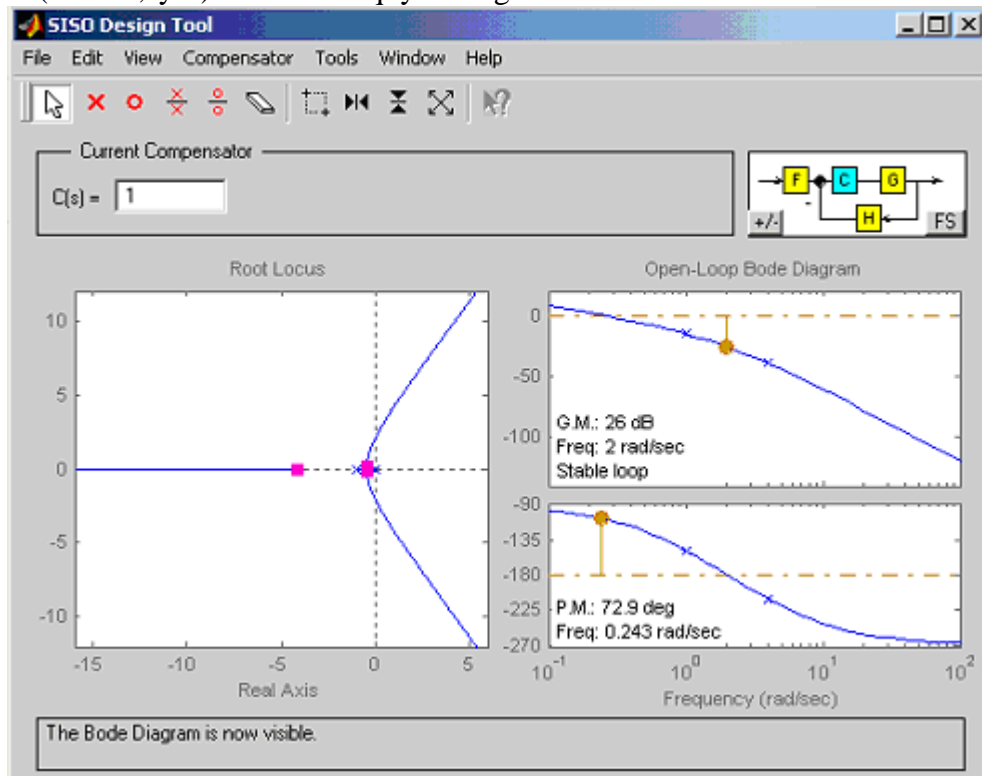
>> sysc=tf(num,den)
```

Transfer function:  

$$\frac{1}{s^3 + 5s^2 + 4s}$$

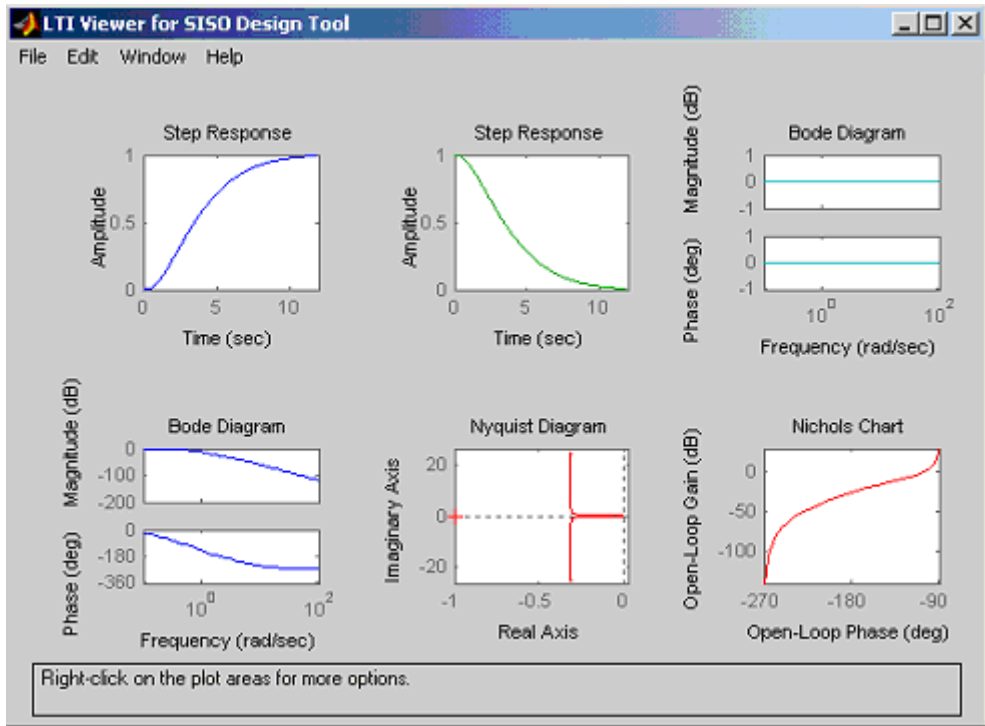
```
>> rltool(sysc) % Thiet ke quy dao nghiem

>>
>> sisotool('bode',sysc) % Thiet ke tan so
>> sisotool('rlocus',sysc) % Thiet ke quy dao nghiem
```



Hình 2.7: Quỹ đạo nghiệm và biểu đồ Bode

Minh họa kết quả :  
 Plant Output(step)  
 Control signal (step)  
 Compensator(Bode)  
 Closed-loop Bode  
 Open-loop Nyquist  
 Open-loop Nichols



Hình 2.8

**2.2.4. Chuyển từ mô hình liên tục theo thời gian ra rời rạc và ngược lại:**

**Hàm c2d.m**

Cú pháp :

$$SYSD = C2D(SYSC, TS, METHOD)$$

chuyển mô hình LTI liên tục thời gian SYSC ra mô hình rời rạc thời gian SYSD với thời gian lấy mẫu TS. Chọn lựa phương pháp rời rạc hóa METHOD.

- 'zoh' Zero-order hold on the inputs.
- 'foh' Linear interpolation of inputs (triangle appx.)
- 'tustin' Bilinear (Tustin) approximation.
- 'prewarp' Tustin approximation with frequency prewarping.  
The critical frequency Wc is specified as fourth input by C2D(SYSC, TS, 'prewarp', Wc).
- 'matched' Matched pole-zero method (for SISO systems only).  
The default is 'zoh' when METHOD is omitted.

Thí dụ:

```
» num=[0 314 0]
```

num =

```
0 314 0
```

```
» den=[1 155.15 197.192]
```

den =

```
1.0000 155.1500 197.1920
```

Chuyển sang hàm truyền G(s)

```
» gs=tf(num,den)
```

Transfer function:

```
314 s
```

$$s^2 + 155.2 s + 197.2$$

Chuyển sang hàm rời rạc bằng biến đổi song tuyến tính:

» c2d(gs,0.1,'tustin')

Transfer function:

$$1.697 z^2 - 1.697$$

$$z^2 - 0.1096 z - 0.6772$$

Sampling time: 0.1

Hàm **d2cm**: đổi từ rời rạc về liên tục

**Biến đổi từ liên tục sang rời rạc và ngược lại dùng Matlab :**

Thí dụ 1 : Cho hệ :  $G_1(s) = \frac{1}{s(s+1)}$  Tìm  $G_1(z)$

Hàm vd2\_14a.m :

numc1=[1];

denc1=[1 1 0];

T=1;

[numz,denz]=c2dm(numc1,denc1,T,'ZOH') ;

printsys(numz,denz,'z')

>> vd2\_14a

num/den =

$$0.36788 z + 0.26424$$

$$z^2 - 1.368 z + 0.3679$$

$$G(z) = \frac{0,369z + 0,2642}{z^2 - 1,368z + 0,3679}$$

Thí dụ 2: Đổi  $G(z)$  sang  $G(s)$

Hàm vd2\_14.m :

numz=[0.369 0.2642];

denz=[ 1 -1.368 0.3679];

T=1;

[numc1,denc1]=d2cm(numz,denz,T,'zoh');

printsys(numc1,denc1)

>> vd2\_14

num/den =

$$0.00078371 s + 1.0017$$

$$s^2 + 0.99994 s - 0.00015819$$

$$G_1(s) = \frac{1}{s(s+1)}$$

### 2.3.Thí nghiệm:

Dùng công cụ Sisotool trong Matlab

Nguyên tắc thiết kế hệ thống điều khiển dùng phương pháp quỹ đạo nghiệm số là dựa vào phân bố vị trí các cực và điểm uốn của hệ thống sau khi hiệu chỉnh:

$$1 + G_c(s).G(s) = 0$$

$$\Leftrightarrow \begin{cases} |G_c(s).G(s)| = 1 \\ \angle G_c(s).G(s) = -180^\circ \end{cases} (*)$$

Ta cần tính toán các thông số của bộ hiệu chỉnh  $G_c(s)$  sao cho đáp ứng ngõ ra của hệ thống đạt được các chất lượng về đáp ứng quá độ và sai số xác lập nhưng phải thỏa mãn điều kiện biên độ và điều kiện pha ở (\*). Chú ý, trong công cụ sisotool thì hàm truyền bộ hiệu chỉnh  $C(s)$  chính là  $G_c(s)$  ở (\*).

#### 2.3.1.Thiết kế bộ hiệu chỉnh sớm pha:

**Mục đích :**

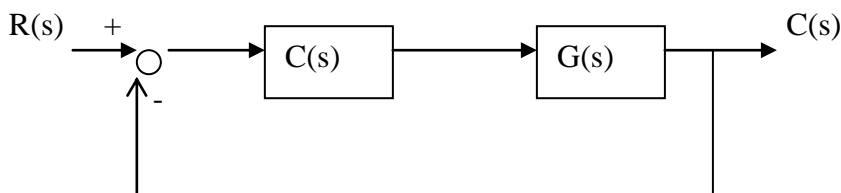
Trong phần này sinh viên sẽ thiết kế bộ hiệu chỉnh sớm pha để hệ thống đạt được các tiêu chuẩn về đáp ứng quá độ như độ vọt lố và thời gian xác lập. Hàm truyền sớm pha có dạng:

$$C(s) = K_c \frac{1 + \alpha Ts}{1 + Ts}, \text{ với } \alpha > 1.$$

Từ các yêu cầu về đáp ứng quá độ ta tìm được vị trí của cặp cực quyết định trên QĐNS. Sau đó ta tính các thông số của bộ hiệu chỉnh  $C(s)$  để sao cho QĐNS của hệ thống sau khi hiệu chỉnh đi qua cặp cực quyết định này.

**Thí nghiệm:**

Cho hệ thống như hình vẽ (H2.9):



Hình 2.9

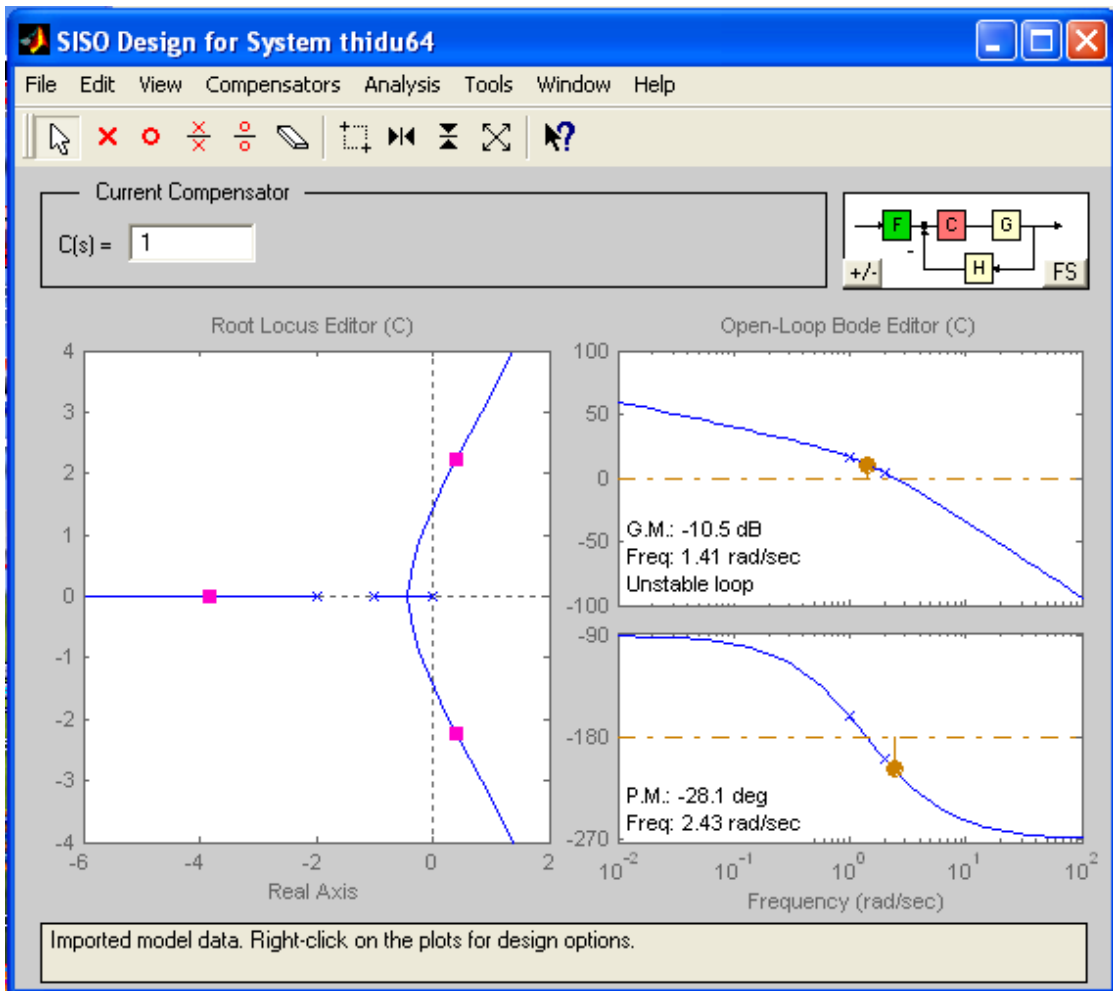
$$G(s) = \frac{20}{s(s+1)(s+2)}.$$

- Dùng công cụ sisotool nhập hàm truyền hệ thống. Dựa vào QĐNS của hệ thống khảo sát hệ thống có ổn định hay không. Giải thích. Vẽ đáp ứng quá độ của hệ thống với đầu vào hàm nấc để minh họa hệ thống có ổn định hay không. Lưu hình vẽ này để so sánh với đáp ứng hệ thống sau khi hiệu chỉnh.
- Thiết kế bộ hiệu chỉnh sớm pha để hệ thống có độ vọt lố POT nhỏ hơn 20% và thời gian xác lập nhỏ hơn 8s. Trình bày rõ quá trình thiết kế này.
- Vẽ đáp ứng quá độ của hệ thống sau khi hiệu chỉnh để chứng minh hệ thống đạt được các yêu cầu ở câu b. Lưu hình vẽ này để viết báo cáo.

**Hướng dẫn :**

Nhập hàm truyền và khởi động siso để import G và H vào sisotool với chú ý:  $G=tf(20,conv([1 1 0],[1 2]))$  và  $H=tf(1,1)$ .

Cửa sổ sisotool hiện ra như sau:



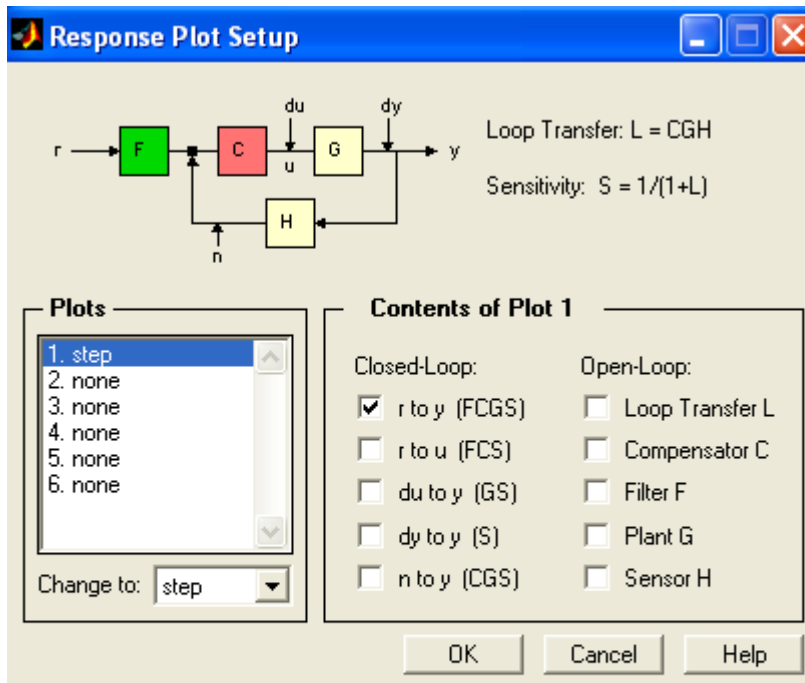
Hình 2.10

Quan sát QĐNS ta thấy phương trình đặc tính vòng kín có 3 nghiệm (đấu vuông màu đỏ):

$$s_1 = -3.84, \quad s_2 = 0.419 + j2.24, \quad s_3 = 0.419 - j2.24$$

Nhận thấy hệ thống có 2 nghiệm  $s_2$  và  $s_3$  nằm bên phải mặt phẳng phức nên hệ thống không ổn định. Ta cũng có thể quan sát trên biểu đồ Bode và nhận thấy  $GM = -10.5 \text{ dB} < 0$  và  $PM = -28.1^\circ < 0$  nên kết luận hệ thống không ổn định. Tuy nhiên, ở phần thí nghiệm này ta dùng phương pháp quỹ đạo nghiệm số nên sẽ không dựa vào biểu đồ Bode để đánh giá mà chỉ dựa vào QĐNS.

Để xem đáp ứng quá độ của hệ thống với đầu vào hàm nấc, vào menu [Analysis] → [Other Loop Responses]. Cửa sổ Response Plot Setup hiện ra. Tiến hành cài đặt các tín hiệu cần vẽ đáp ứng. Ở đây ta chọn như hình bên dưới:



. r to y: vẽ đáp ứng của ngõ ra  $y(t)$  theo tín hiệu đầu vào  $r(t)$ . Đây chính là đáp ứng quá độ cần vẽ.

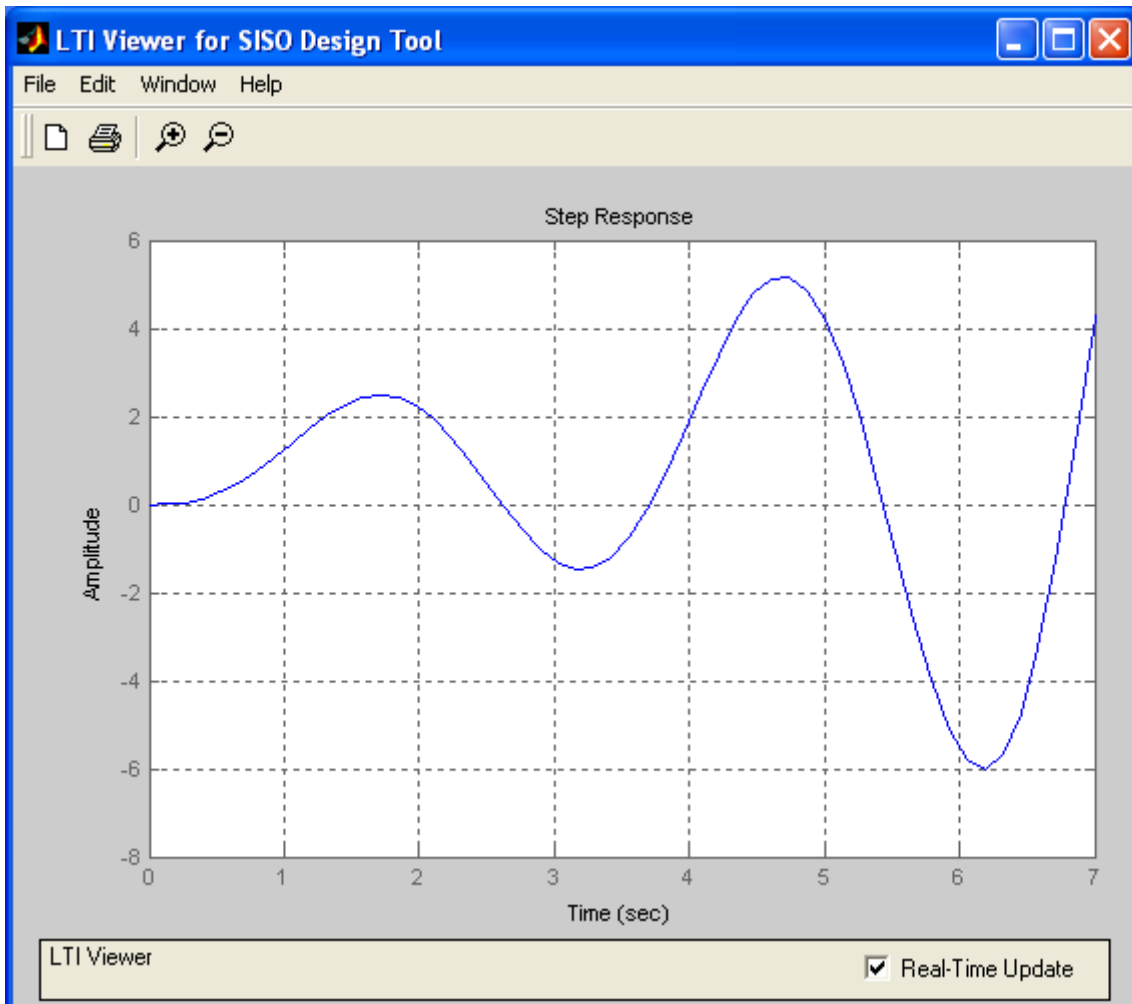
. r to u : vẽ đáp ứng của tín hiệu điều khiển  $u(t)$ .

...

Hình 2.11

Vì cửa sổ LTI Viewer này không hỗ trợ việc lưu hình vẽ nên phải chuyển sang cửa sổ Figure bằng cách vào menu [File]→[Print to Figure]. Cửa sổ Figure hiện ra và sau đó tiến hành lưu hình vẽ.





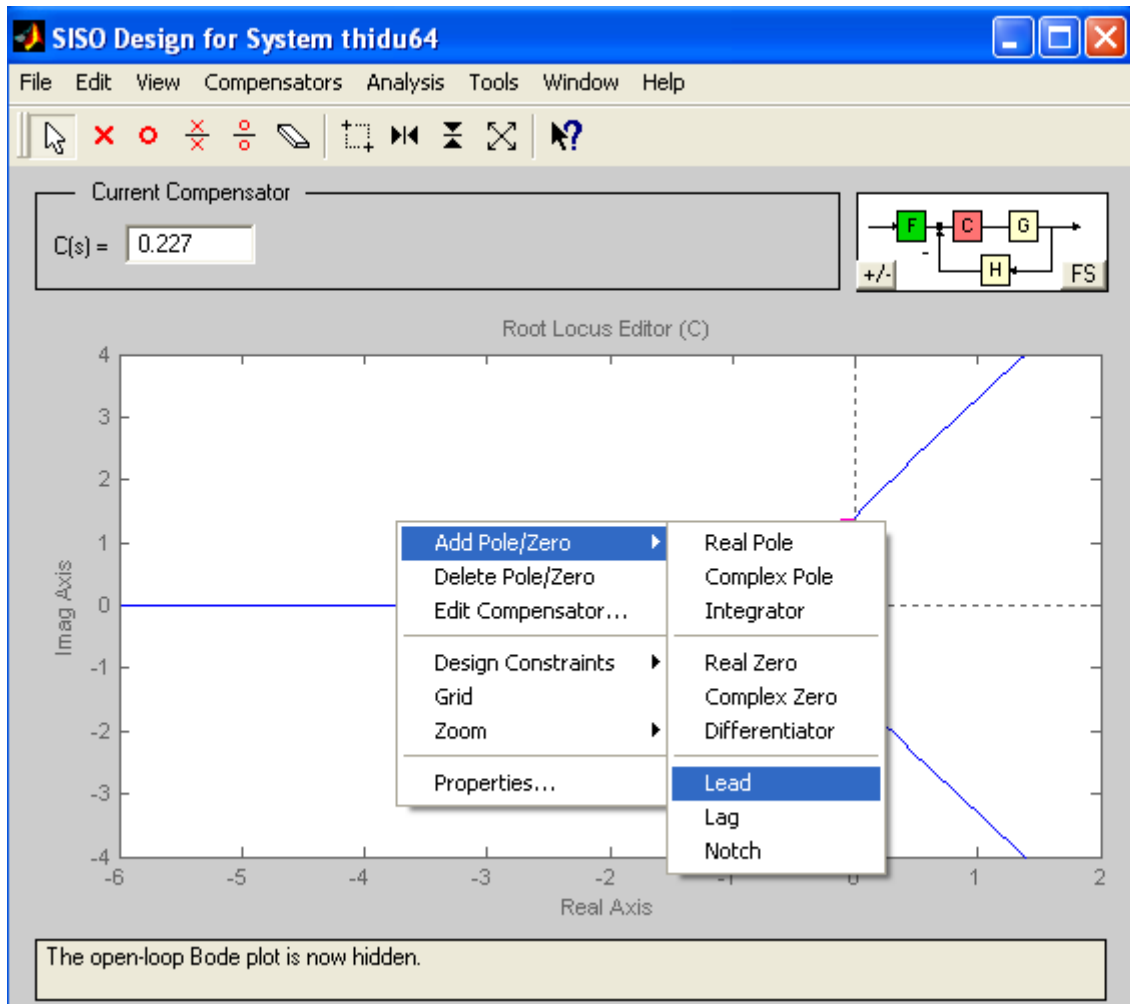
Hình 2.12

Bây giờ thiết kế bộ hiệu chỉnh để hệ thống có  $POT < 20\%$  và  $t_{x1} < 8s$ . Ta quay trở lại với cửa sổ sisotool. Vì trong phần này ta không sử dụng biểu đồ Bode và để mở rộng vùng QĐNS nên ta xóa vùng biểu đồ Bode đi bằng cách vào menu [View] bỏ dấu chọn mục [Open –Loop Bode].

Kích chuột phải vào vùng QĐNS, menu kiểu pop-up xuất hiện:

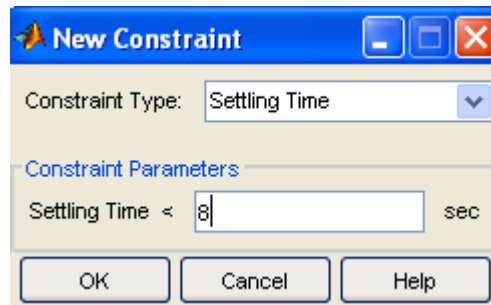
- Lead: bộ hiệu chỉnh sớm pha.
- Lag: bộ hiệu chỉnh trễ pha.
- Notch: bộ hiệu chỉnh sớm trễ pha.
- Delete Pole/Zero: xóa các cực và zero của bộ hiệu chỉnh.
- Edit Compensator...: thay đổi các thông số của bộ hiệu chỉnh.
- Design Constraints: giới hạn vùng thỏa mãn các tiêu chuẩn chất lượng.

Chọn [Add Pole/Zero] → [Lead] để thêm khâu hiệu chỉnh sớm pha vào hệ thống. Nhấp chuột vào một vị trí bất kỳ trên trục thực của QĐNS để xác định vị trí cực và zero của bộ hiệu chỉnh, sisotool sẽ gán tự động vị trí của zero nằm gần gốc tọa độ hơn cực.

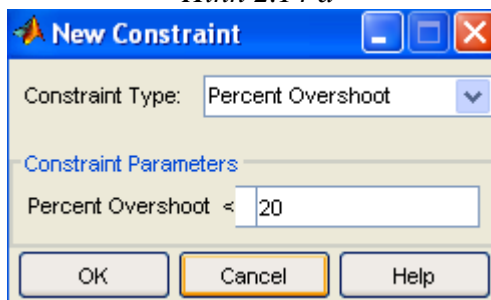


Hình 2.13

Kích chuột phải vào vùng QĐNS, ta chọn [Design Constraints] → New để cài độ vọt lố và thời gian xác lập như sau:

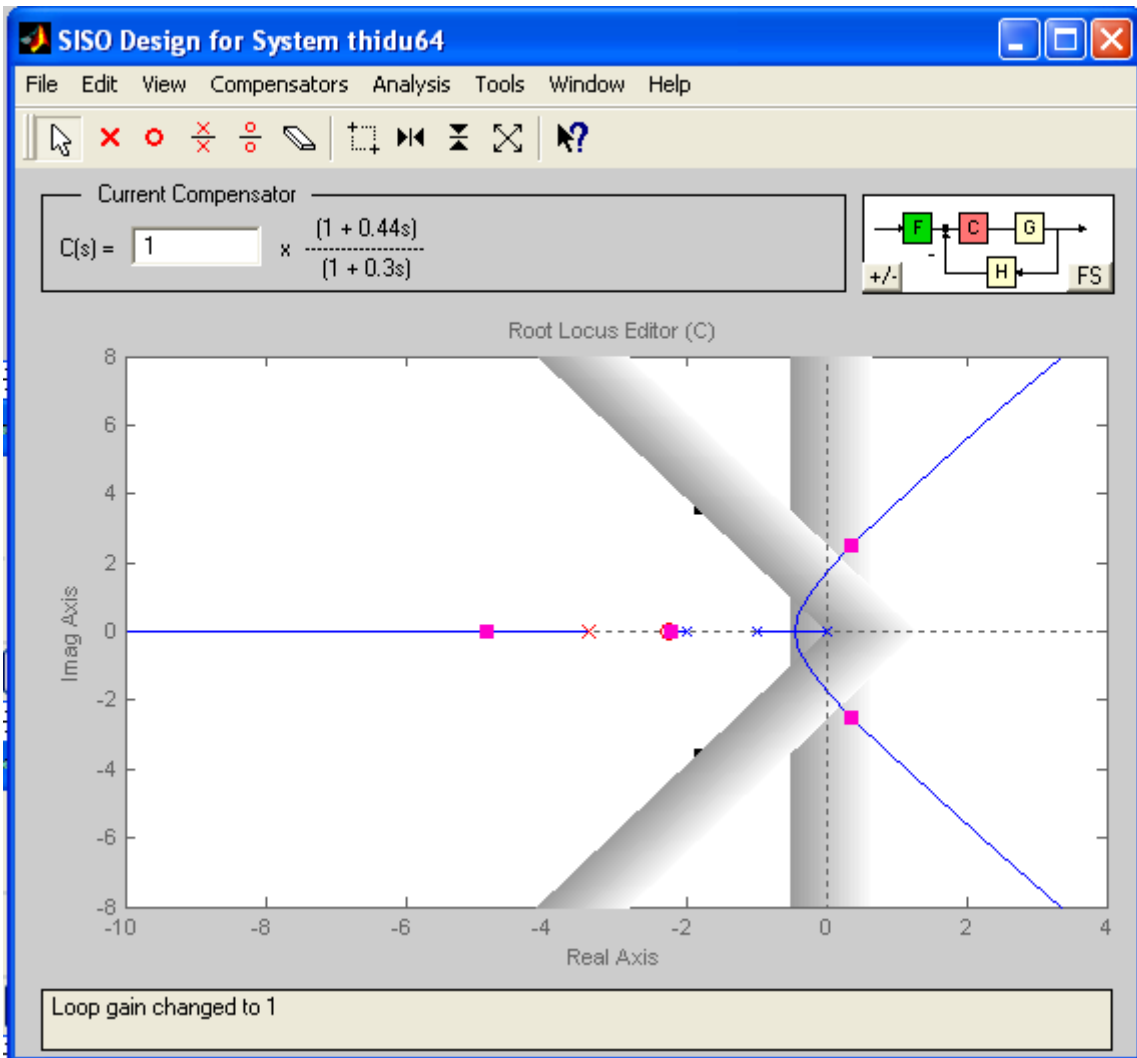


Hình 2.14 a



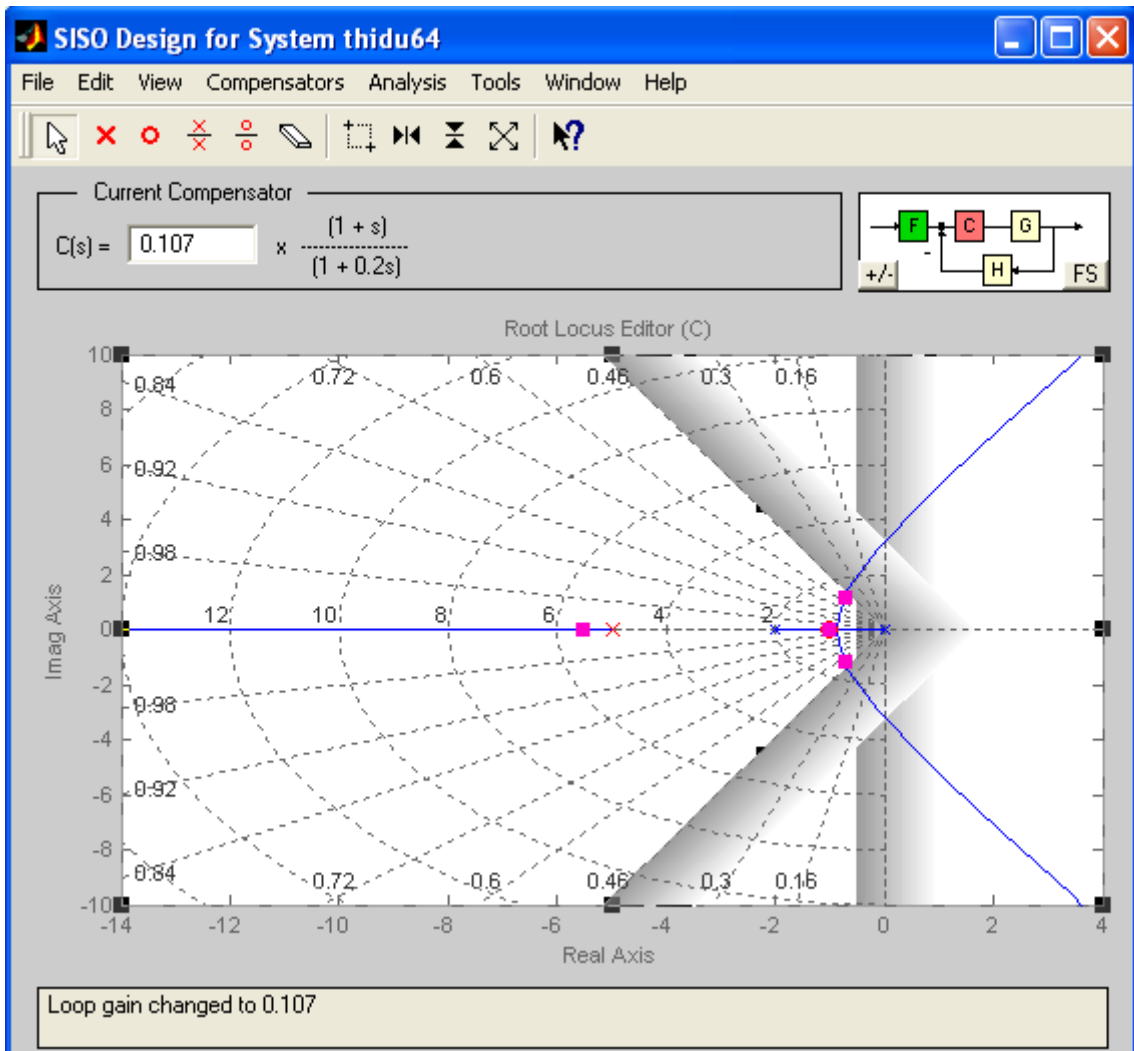
Hình 2.14b

Sau khi tiến hành cài đặt xong, QĐNS lúc này sẽ thay đổi như sau:

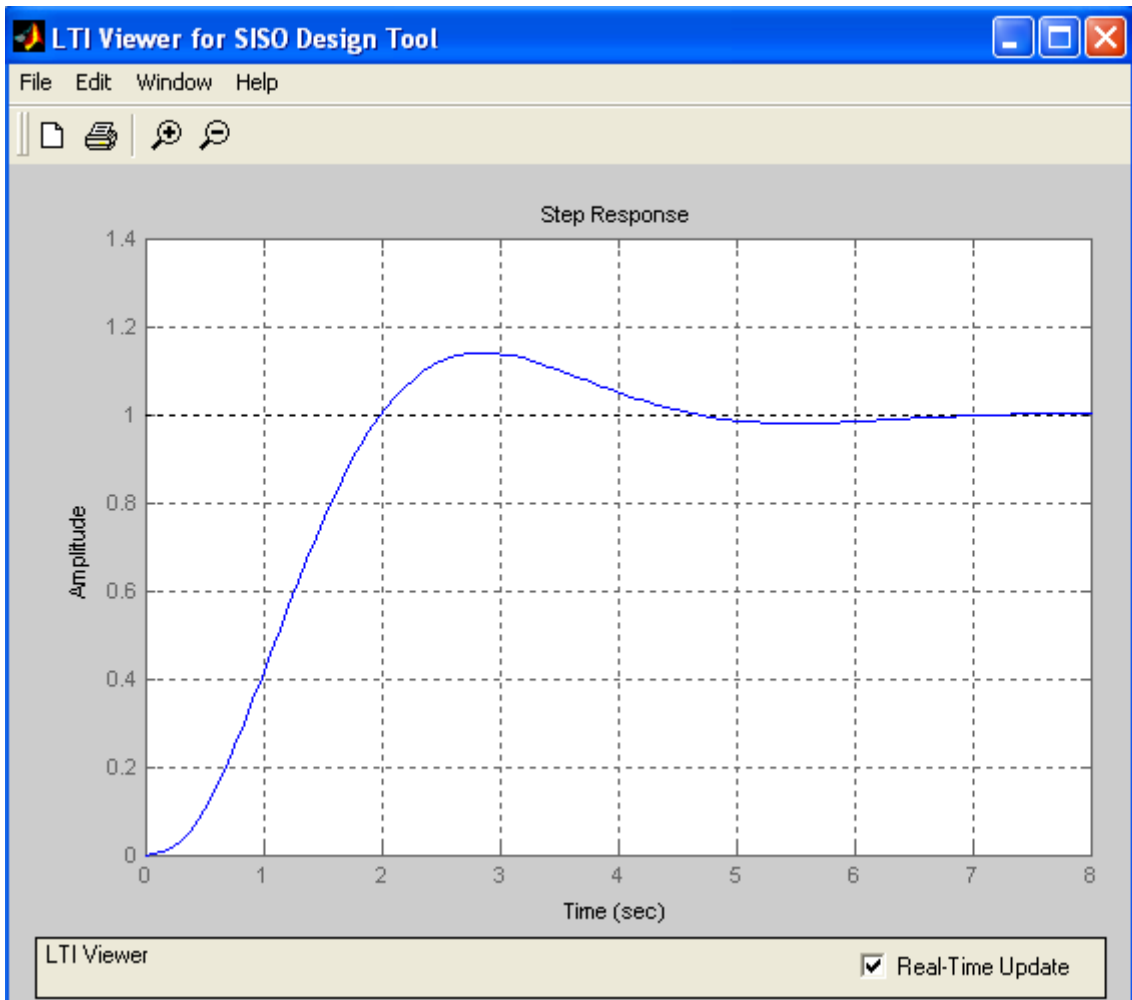


Hình 2.15

Bây giờ ta sẽ di chuyển các cực và zero của bộ hiệu chỉnh  $C(s)$  trên trục thực sao cho nhánh QĐNS(A) kéo vào vùng thỏa mãn thiết kế. Có 2 phương pháp tìm cực và zero của bộ hiệu chỉnh là phương pháp đường phân giác và phương pháp khử cực (sinh viên xem lại sách lý thuyết để hiểu 2 phương pháp này). Ở đây ta chọn phương pháp khử cực vì dễ thao tác và trực quan trên cửa sổ QĐNS.



Hình 2.16



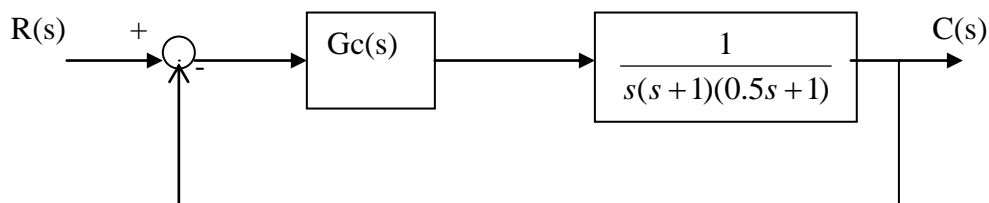
Hình 2.17

**2.3.2. Thiết kế bộ hiệu chỉnh trễ pha:**

Thí dụ: Thiết kế khâu hiệu chỉnh trễ pha dùng phương pháp biểu đồ Bode

Hãy thiết kế khâu hiệu chỉnh trễ pha sao cho hệ thống sau khi hiệu chỉnh có :

$$K_v^* = 5; \phi M^* \geq 40; GM^* \geq 10dB.$$



Hình 2.18

Giải :

Hàm truyền khâu hiệu chỉnh trễ pha cần thiết kế là :

$$G_c(s) = K_c \frac{1 + \alpha Ts}{1 + Ts}, (\alpha < 1)$$

Bước 1: Xác định  $K_C$ .

Hệ số vận tốc của hệ sau khi hiệu chỉnh là:

$$K_v^* = \lim_{s \rightarrow 0} s G_c(s) G(s) = \lim_{s \rightarrow 0} s K_c \frac{1 + \alpha Ts}{1 + Ts} \cdot \frac{1}{s(s+1)(0.5s+1)} = K_c$$

$$\Rightarrow K_c = K_v^* \Rightarrow K_c = 5.$$

Bước 2:

Hàm truyền hệ  $G(s)$ :

>> G1=tf(1,conv([1 1 0],[0.5 1]))

Transfer function:

$$\frac{1}{0.5 s^3 + 1.5 s^2 + s}$$

Đặt  $G_1(s) = K_c G(s) = 5 \cdot \frac{1}{s(s+1)(0,5s+1)}$

$$\Rightarrow G_1(s) = \frac{5}{s(s+1)(0,5s+1)}$$

Bước 3: Xác định tần số cắt mới

Cách 1: Tìm  $\omega_c'$  bằng phương pháp giải tích. Ta có:

$$\varphi_1(\omega_c') = -180^\circ + \Phi M^* + \theta$$

$$\Rightarrow -90^\circ - \arctan(\omega_c') - \arctan(0,5\omega_c') = -180^\circ + 40^\circ + 5^\circ$$

$$\Rightarrow \arctan(\omega_c') + \arctan(0,5\omega_c') = 45^\circ$$

$$\Rightarrow \frac{(\omega_c') + (0,5\omega_c')}{1 - 0,5(\omega_c')^2} = \tan(45^\circ) = 1$$

$$\Rightarrow 0,5(\omega_c')^2 + 1,5(\omega_c') - 1 = 0$$

$$\Rightarrow \omega_c' = 0,56(\text{rad/sec})$$

Cách 2: dựa vào biểu đồ Bode

Ta có :

$$\varphi_1(\omega_c') = -180^\circ + \Phi M^* + \theta$$

$$\Rightarrow \varphi_1(\omega_c') = -180^\circ + 40^\circ + 5^\circ$$

$$\Rightarrow \varphi_1(\omega_c') = -135^\circ$$

Vẽ đường thẳng có hoành độ  $-135^\circ$ . Hoành độ giao điểm của đường thẳng này với biểu đồ Bode về pha  $\varphi_1(\omega)$  chính là giá trị tần số cắt mới. Theo hình vẽ ta có  $\omega_c' = 0,5(\text{rad/sec})$ .

Bước 4:

Cách 1: Tính  $\alpha$  từ điều kiện:

$$|G_1(j\omega_c')| = \frac{1}{\alpha}$$

$$\Rightarrow \left| \frac{5}{s(s+1)(0,5s+1)} \right|_{s=j\omega_c'} = \frac{1}{\alpha}$$

$$\Rightarrow \left| \frac{5}{j0,56(j0,56+1)(0,5 \cdot j0,56+1)} \right| = \frac{1}{\alpha}$$

$$\Rightarrow \frac{5}{0,56(\sqrt{0,56^2+1})(\sqrt{0,28^2+1})} = \frac{1}{\alpha}$$

$$\Rightarrow \frac{5}{0,56 \times 1,146 \times 1,038} = \frac{1}{\alpha} \Rightarrow \alpha = 0,133.$$

Cách 2: Tính  $\alpha$  từ điều kiện:  $L_1(\omega_c') = -20 \lg \alpha$

Dựa vào biểu đồ Bode ta thấy :  $L_1(\omega_c') = 18 \text{dB}$

Suy ra:  $\lg \alpha = -0,9$

$$\alpha = 10^{-0,9}$$

$$\alpha = 0,126$$

Ta thấy giá trị  $\alpha$  sai khác không đáng kể ở hai cách. Ở các bước thiết kế tiếp theo ta chọn  $\alpha = 0,133$ .

Bước 5: Chọn zero của khâu trễ pha

$$\frac{1}{\alpha T} \ll \omega_c = 0,56$$

Chọn  $\frac{1}{\alpha T} = 0,05$

$$\Rightarrow \alpha T = 20$$

Bước 6: Tính thời hằng T

$$\frac{1}{T} = \alpha \frac{1}{\alpha T} = 0,133 \times 0,05 = 0,067$$

$$\Rightarrow T = 150$$

Vậy:  $G_c(s) = 5 \cdot \frac{(20s + 1)}{(150s + 1)}$

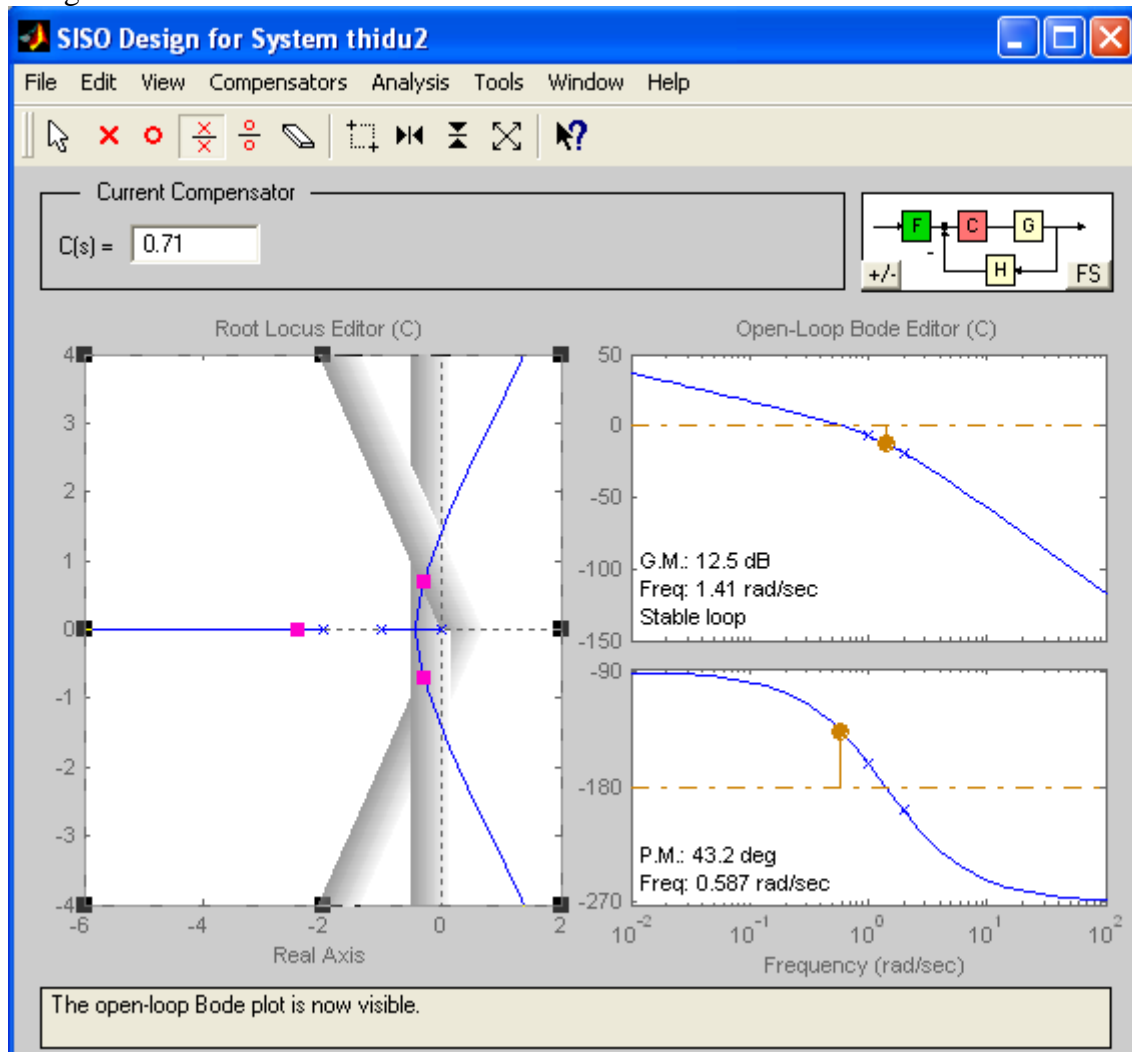
Bước 7: Kiểm tra lại điều kiện biên độ

Dựa vào biểu đồ Bode ta thấy độ dự trữ biên sau khi hiệu chỉnh là :  $GM^* = 10$  dB.

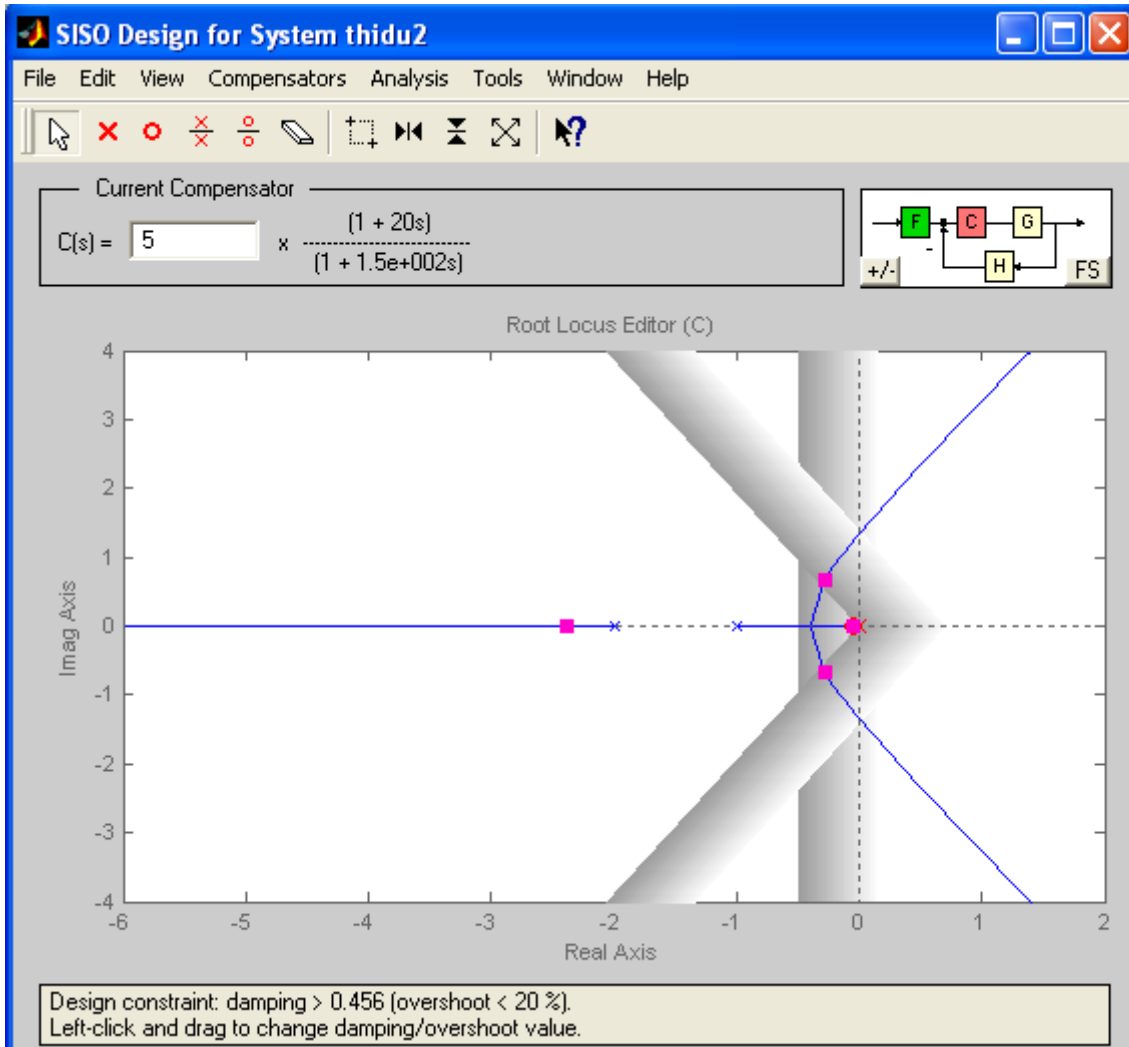
Kết luận: Khâu hiệu chỉnh vừa thiết kế đạt yêu cầu về độ dự trữ biên.

**Kết quả:**

Dùng Sisotool vẽ biểu đồ Bode



Hình 2.19



Hình 2.20

### 2.3.3. Thiết kế bộ điều khiển sớm trễ pha:

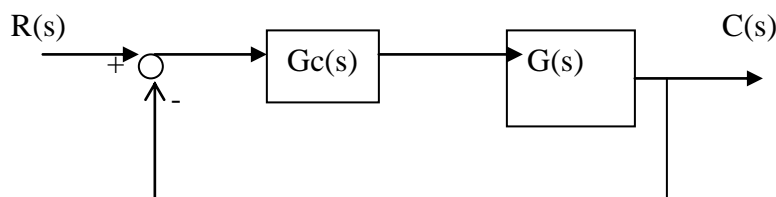
#### Mục đích:

Trong phần này sinh viên sẽ thiết kế bộ hiệu chỉnh sớm trễ pha để hệ thống đạt được các tiêu chuẩn về đáp ứng quá độ và chất lượng xác lập. Hàm truyền sớm trễ pha có dạng:

$$C(s) = K_{c1} \frac{1 + \alpha T_1 s}{1 + T_1 s} \cdot K_{c2} \frac{1 + \beta T_2 s}{1 + T_2 s} \quad (\alpha > 1, \beta < 1)$$

#### Thí nghiệm:

Cho hệ thống như hình vẽ:



$$G(s) = \frac{4}{s(s + 0.5)}$$

- Thiết kế bộ hiệu chỉnh sớm trễ pha để hệ thống có  $\xi=0.5$ ,  $\omega_n=5$  (rad/sec) và hệ số vận tốc  $K_v=80$ . Trình bày rõ quá trình thiết kế.



- b. Vẽ đáp ứng quá độ của hệ thống sau khi hiệu chỉnh với đầu vào hàm dốc để chứng minh hệ thống đạt được các yêu cầu ở câu a. Lưu hình vẽ này để báo cáo.

**Hướng dẫn:**

Trước tiên ta thiết kế bộ hiệu chỉnh sớm pha  $C_1(s) = K_{c1} \frac{1 + \alpha T_1 s}{1 + T_1 s}$  để hệ thống có

$\xi=0.5, \omega_n=5$  cho đối tượng  $G(s)$ . Sau đó, thiết kế bộ hiệu chỉnh trễ pha  $C_2(s) = K_{c2} \frac{1 + \beta T_2 s}{1 + T_2 s}$

cho đối tượng mới  $G1(s)=G(s).C1(s)$ . Chú ý, sau khi thiết kế xong bộ hiệu chỉnh sớm pha ta phải khởi động lại sisotool và nhập lại hàm truyền  $G$  và  $H$  với hàm truyền  $G$  lúc này chính là  $G1(s)$  và  $H=1$ .

với  $\xi=0.5, \omega_n=5 \rightarrow s_{1,2}^* = -2.5 \pm j4.33$

Do đó, khi thiết kế bộ hiệu chỉnh sớm pha  $C1(s)$ , để hệ thống có  $\xi=0.5, \omega_n=5$  ta sẽ di chuyển zero của  $C1(s)$  tới vị trí  $-0.5$  (vị trí cực của  $G(s)$  để khử cực này) và di chuyển cực của  $C1(s)$  (phải cách xa gốc tọa độ hơn zero) sao cho QĐNS đi qua hai nghiệm  $s_{1,2}^*$ . Sau đó dùng chuột di chuyển nghiệm  $s_2$  (dấu vuông màu đỏ) lại vị trí  $s_{1,2}^*$  này.

Kết quả: Hàm truyền khâu sớm pha

$$C1(s)=0.251x(1+2s)/(1+0.2s)$$

**2.3.4. Thi ết kế bộ điều khiển PID bằng ACSYS**

Thí dụ 1: Thiết kế PD

Xét hệ có hàm truyền sau:

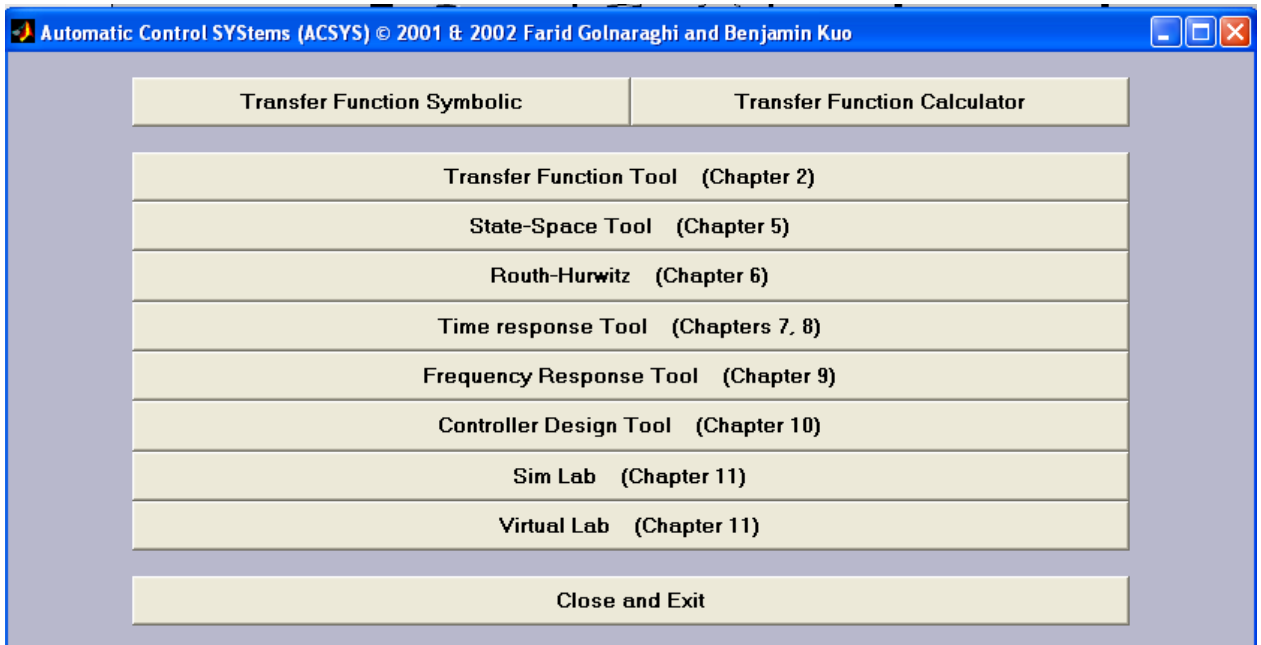
$$G(s) = \frac{4500}{s(s + 361.2)}$$

**Thiết kế miền thời gian :** Thiết kế bộ điều khiển để hệ thống vòng kín thỏa mãn yêu cầu sau:

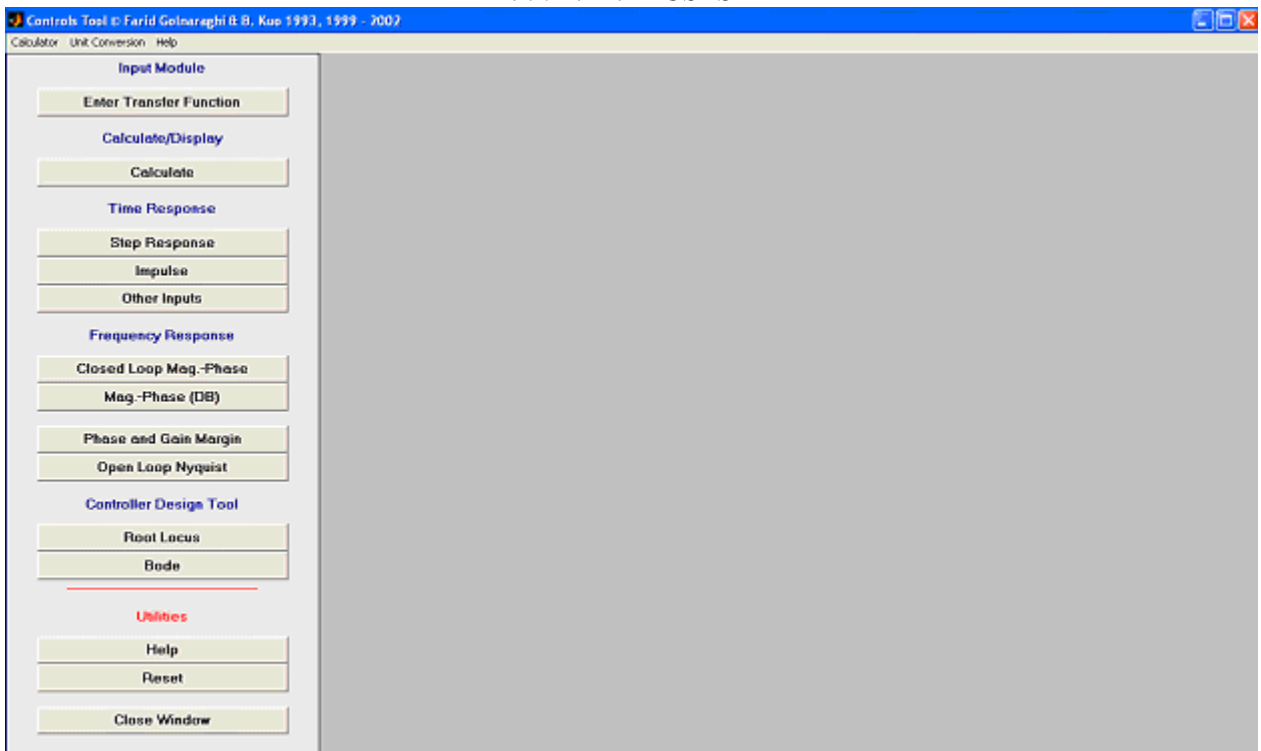
- Sai số xác lập với ngõ vào hàm unit-ramp  $\leq 0.000443$ .
- Vọt lố cực đại  $\leq 5\%$ .
- Thời gian lên :  $t_r \leq 0.005$  sec.
- Thời gian thiết lập  $t_s \leq 0.005$  sec.

Dùng ACSYS

Để xem xét chất lượng của hệ ta bắt đầu dùng bộ điều khiển tỉ lệ. Khởi động ACSYS, kích chọn Controller Design tool bằng cách gõ controls ở dòng lệnh Matlab hay nhấp vào nút trong ACSYS.



Hình 2.21: ACSYS



Hình 2.22: Cửa sổ chính thiết kế bộ điều khiển.

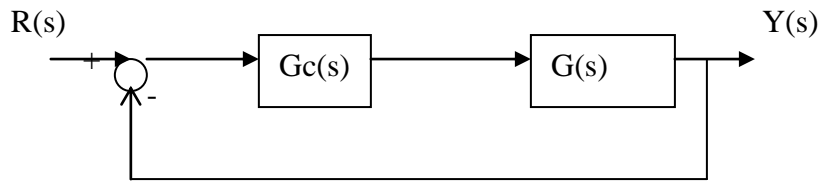
Chọn “Enter Transfer Function” và nhập vào hàm truyền  $G(s)$ :

Num  $G(s)$ : 4500

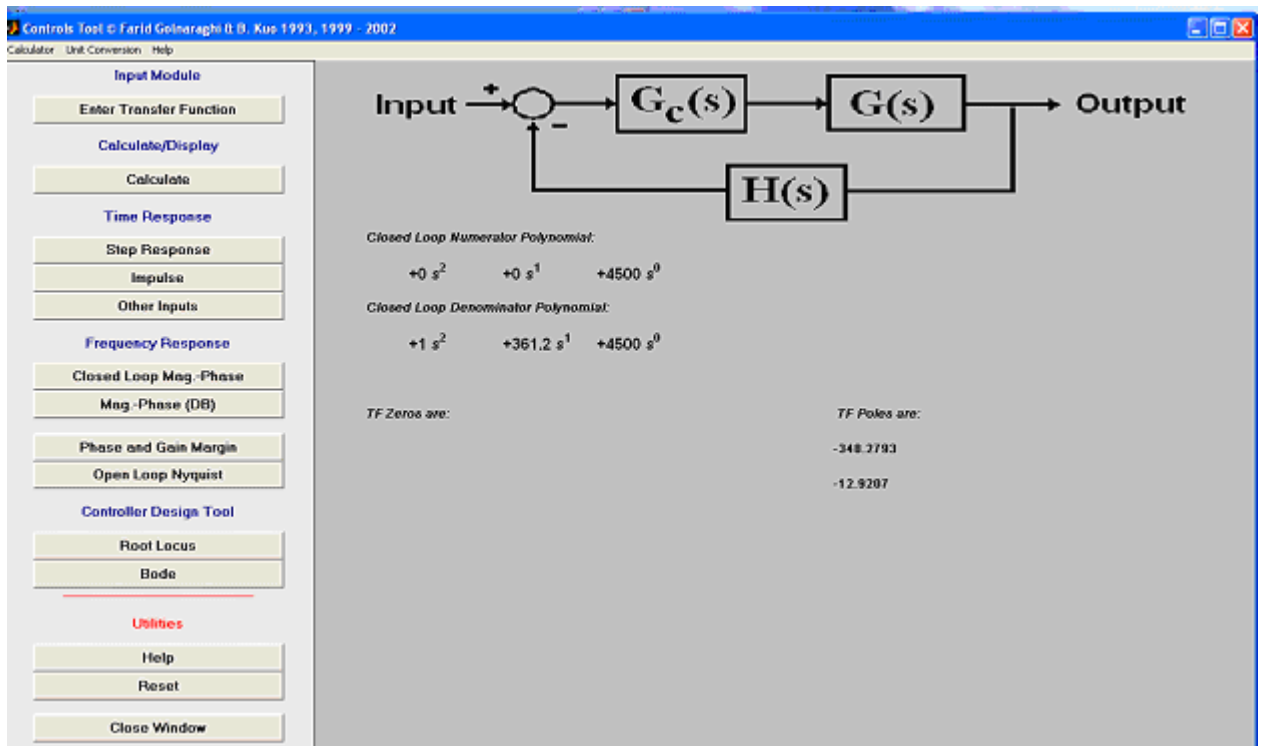
Den  $G(s)$  : 1 361.2 0

Kế tiếp nhập  $G_c(s)=1$  và  $H(s)=1$ .

Nhấp nút “APPLY” trở về cửa sổ chính và nhấn “Calculate” để đánh giá hàm truyền vòng kín. Chú ý rằng bạn luôn tham khảo cửa sổ lệnh Matlab để xem thể hiện chính xác hàm truyền.



Hình 2.23: hệ thống điều khiển hồi tiếp âm đơn vị.



Hình 2.24 : Cửa sổ chính bộ điều khiển minh họa hàm truyền vòng kín.

Cửa sổ lệnh Matlab:

```
>> acsys
```

```
G=
```

```
Transfer function:
```

```
4500
```

```
-----  
s^2 + 361.2 s
```

```
Gc=
```

```
Transfer function:
```

```
1
```

```
H=
```

```
Transfer function:
```

```
1
```

```
G*G_c ==>open loop
```

Transfer function:

4500

-----  
 $s^2 + 361.2 s$

$G*G_c*H ==> \text{loop}$

Transfer function:

4500

-----  
 $s^2 + 361.2 s$

$G*G_c/(1+G*G_c*H) ==> \text{closed loop}$

Transfer function:

4500

-----  
 $s^2 + 361.2 s + 4500$

system Coefficients are  
 Closed Loop TF in zero/pole format

Zero/pole/gain:

4500

-----  
 $(s+348.3) (s+12.92)$

System Zeros are

zeroTF =

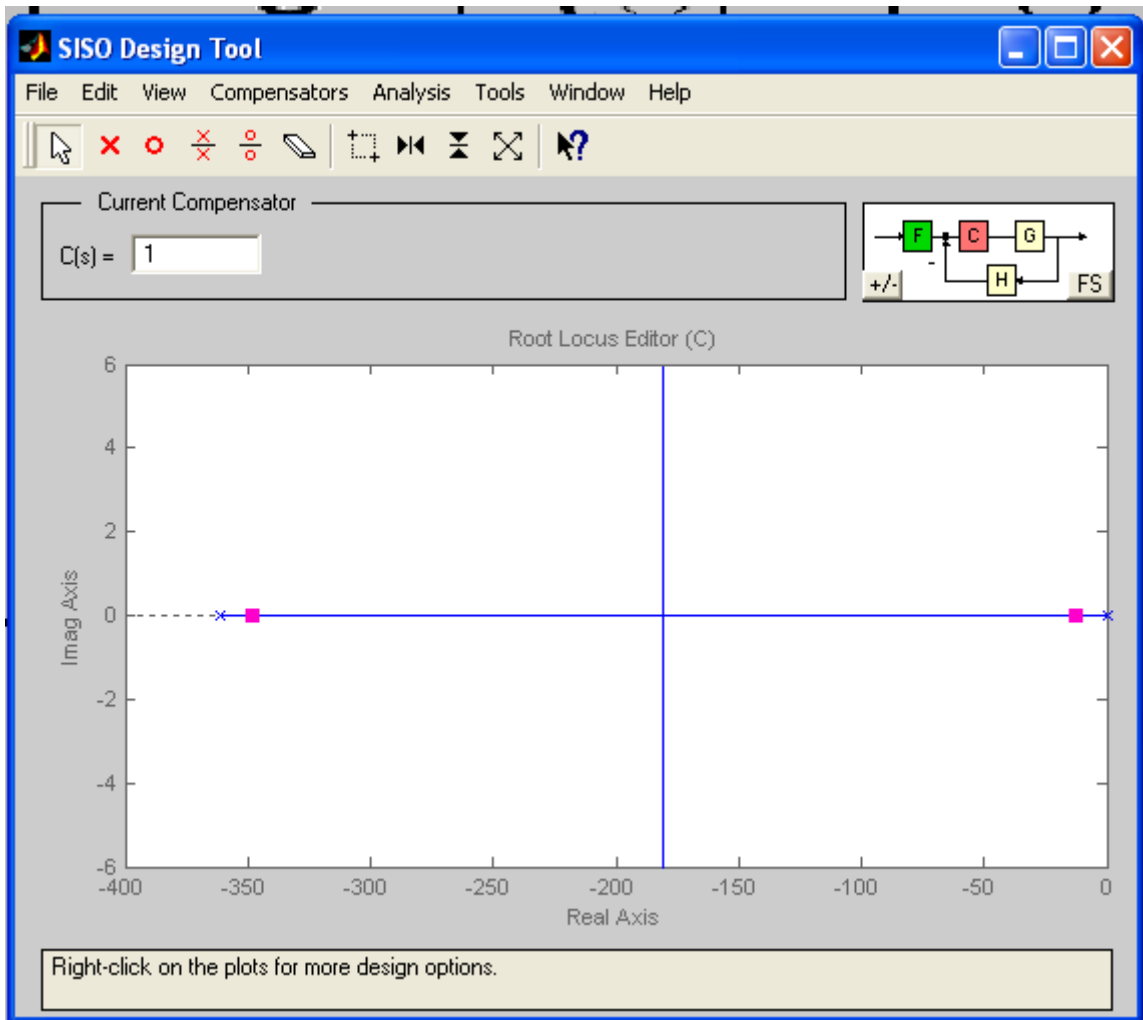
Empty matrix: 1-by-0

System Poles are

poleTF =

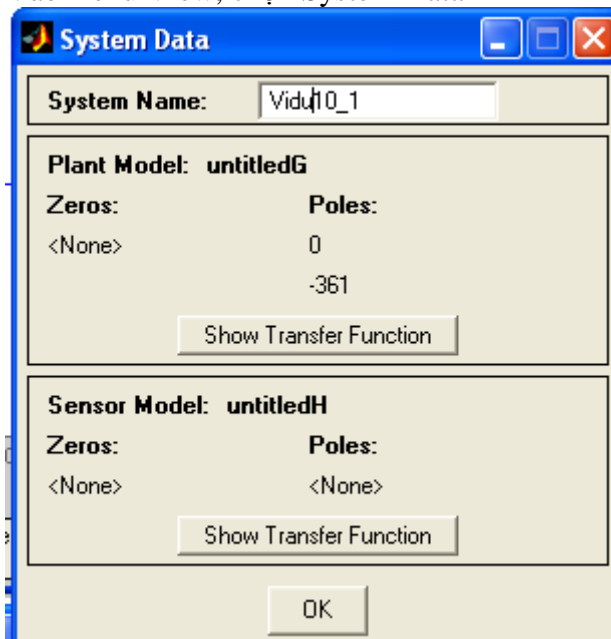
-348.2793 -12.9207

Để xem chất lượng của bộ điều khiển tỉ lệ, ta cần tìm quỹ đạo nghiệm hệ thống. Nhấn vào nút “Root Locus” trong cửa sổ chính điều khiển. Điều này kích hoạt công cụ thiết kế SISO của Matlab.



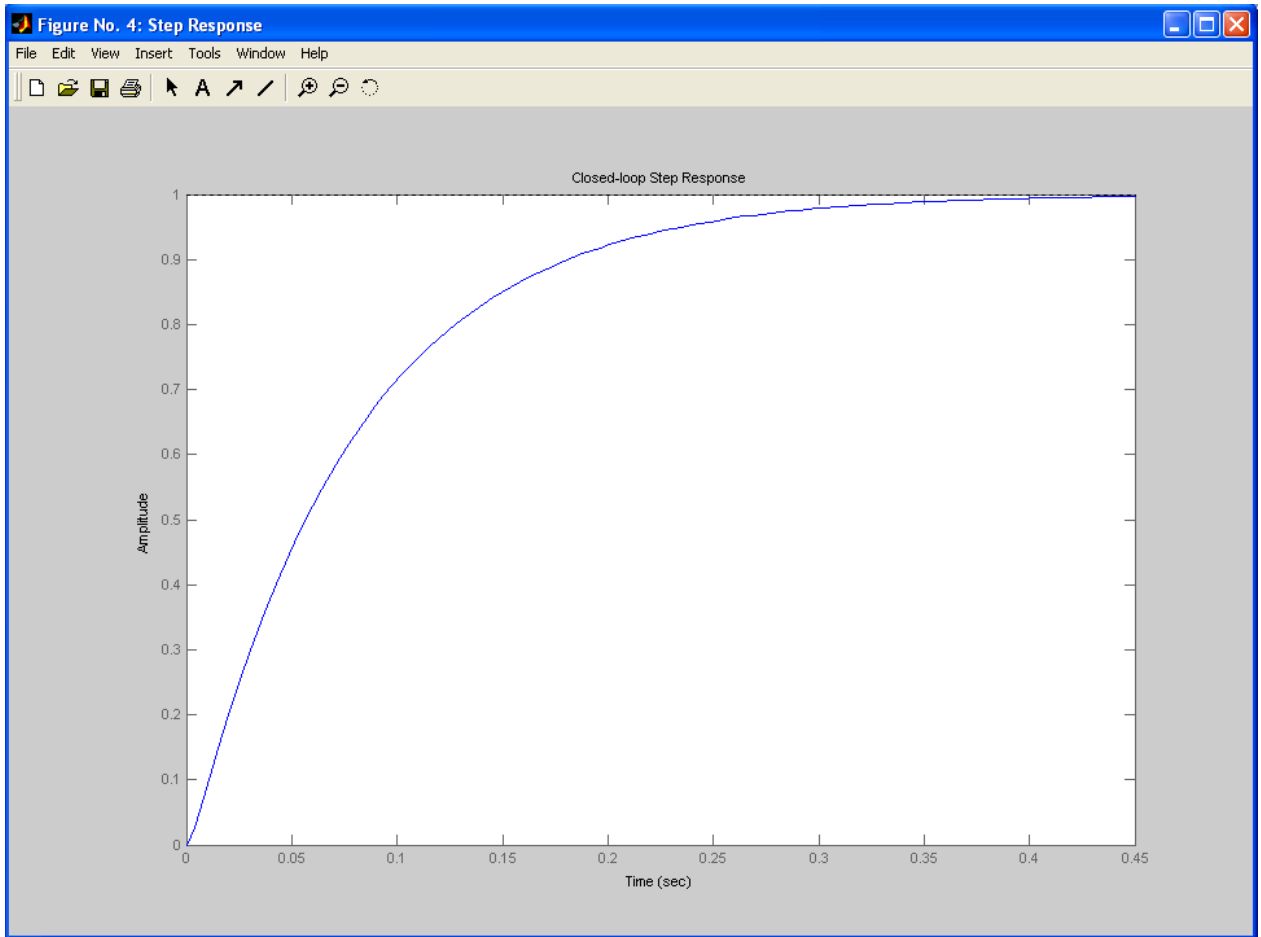
Hình 2.25: Quỹ đạo nghiệm số.

Hình vuông màu đỏ trong hình thể hiện cực của hệ vòng kín với  $K=1$ . Vào menu View, chọn System Data

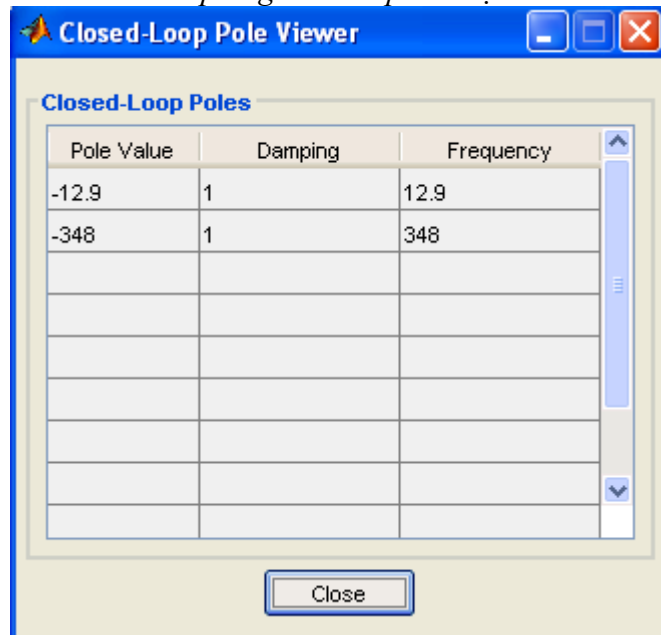


Hình 2.26: cực và zero của G và H.

Chọn Step Response : xem đáp ứng thời gian hệ kín với ngõ vào hàm nấc đơn vị.

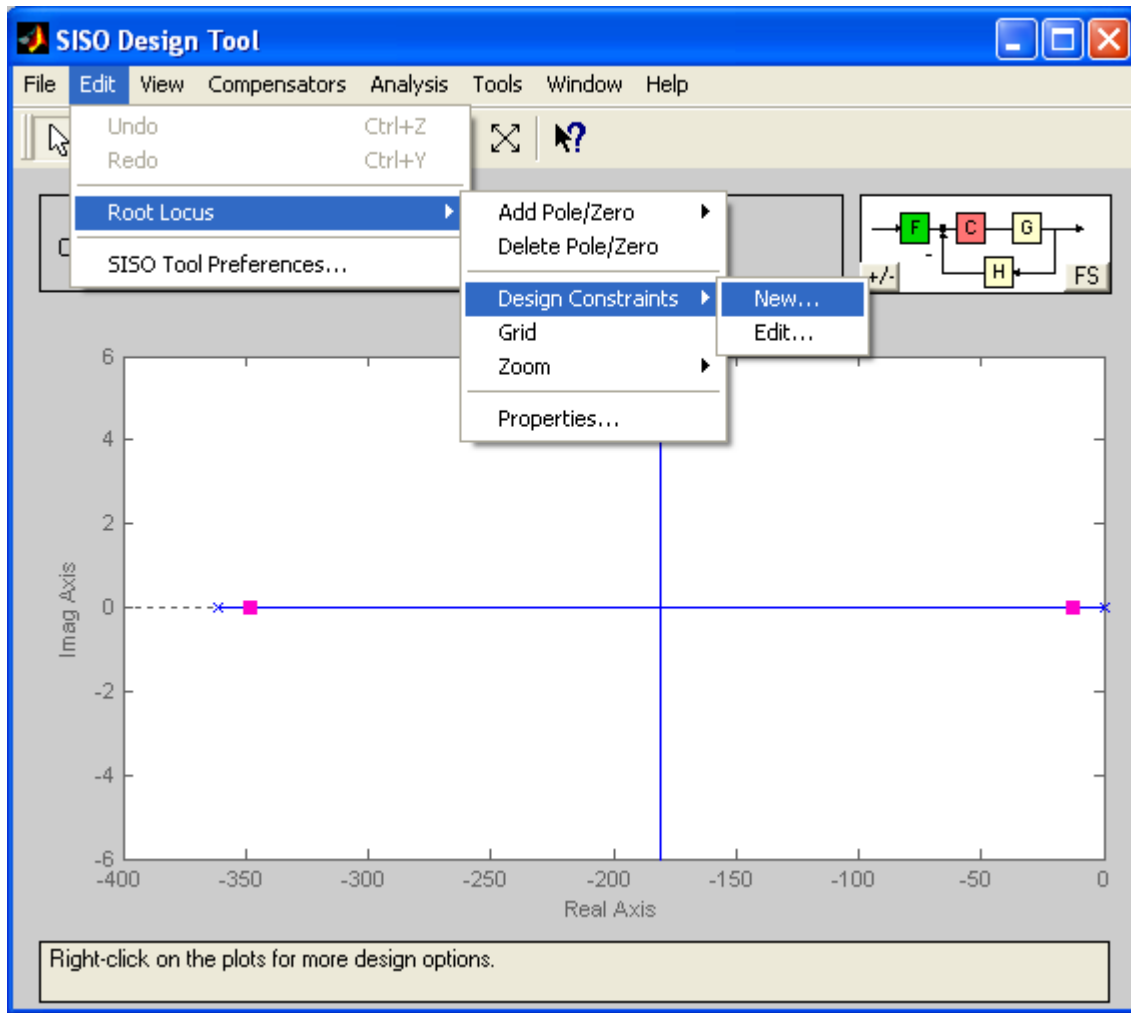


Hình 2.27: Đáp ứng unit-step của hệ kín với  $K=1$ .

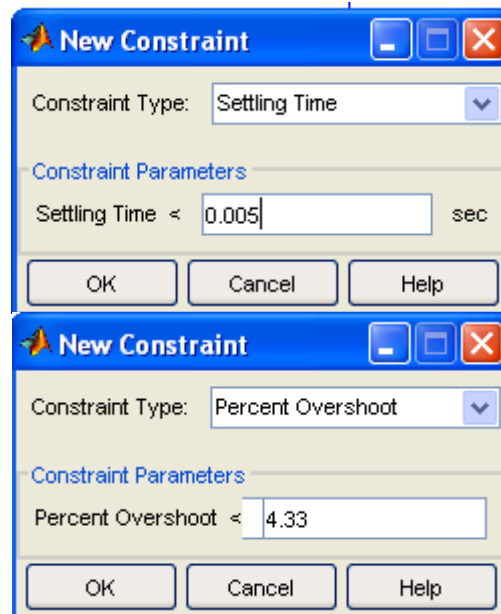


Hình 2.28: cực và zero của hệ với  $K=1$ .

**Tích hợp tiêu chuẩn thiết kế:** Khi bắt đầu thiết kế bộ điều khiển, ta xây dựng tùy chọn tiêu chuẩn thiết kế có sẵn bên trong công cụ SISO Design để thiết lập vùng cực mong muốn trên quỹ đạo nghiệm. Để thêm ràng buộc thiết kế, sử dụng menu Edit và chọn Root Locus, mục New và sau đó chọn tùy chọn Design Constrains như hình sau:



Hình 2.29a



Hình 2.29b: Sử dụng Design Constrains tùy chọn cho ví dụ thiết kế (H2.29a) và Thời gian lên  $< 0.005$  sec và phần trăm vọt lố  $< 4.33$  (H2.29b).

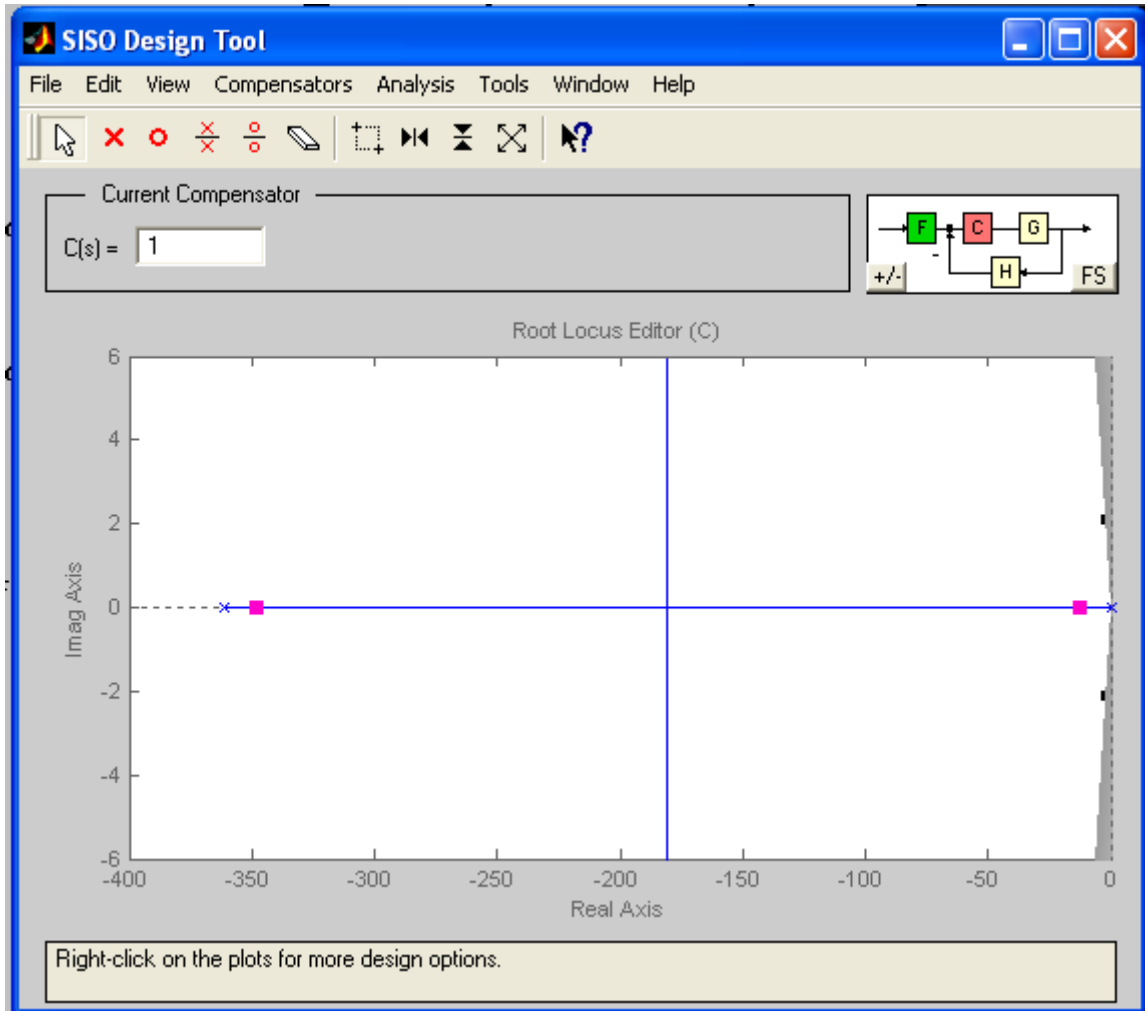
Tùy chọn ràng buộc gồm:

- . Settling time (thời gian thiết lập).
- . Percent Overshoot (phần trăm vọt lố).
- . Damping ratio (tỉ số suy giảm).

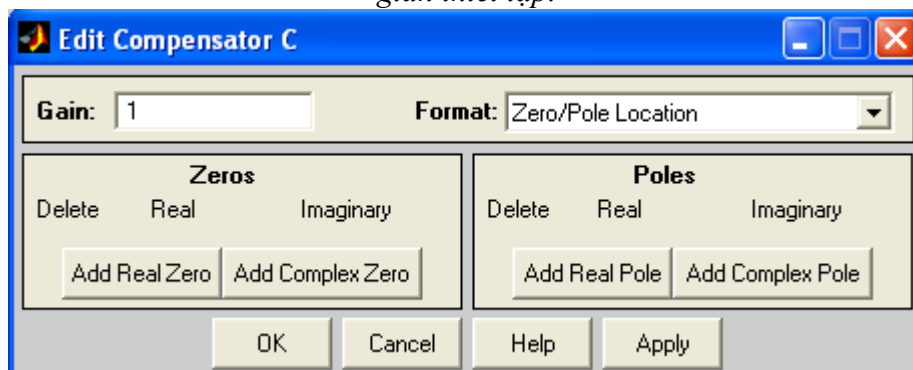
. Natural frequency (tần số tự nhiên).

Trong ví dụ này, ta đưa vào ràng buộc thiết kế là thời gian thiết lập và phần trăm vọt lố.

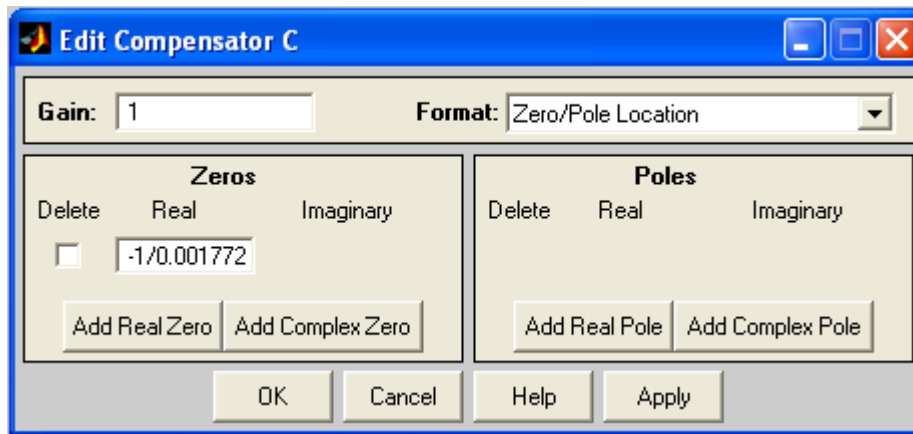
Hình sau minh họa vị trí cực của hệ kín trên quỹ đạo nghiệm sau khi đưa vào ràng buộc thiết kế. Rõ ràng là cực của hệ khi  $K=1$  không nằm trong vùng mong muốn. Không thể dùng bộ điều khiển tỉ lệ để di chuyển cực của hệ kín xa hơn về mặt phẳng trái. Do đó ta dùng bộ điều khiển PD để đạt được điều này, bộ điều khiển PD có zero tại  $z=-1/0.001772$ . Thành phần tỉ lệ  $K_p=1$ , sai số xác lập với ngõ vào unit-ramp là  $ess=0.000443$  với  $K_d=0.001772$ .



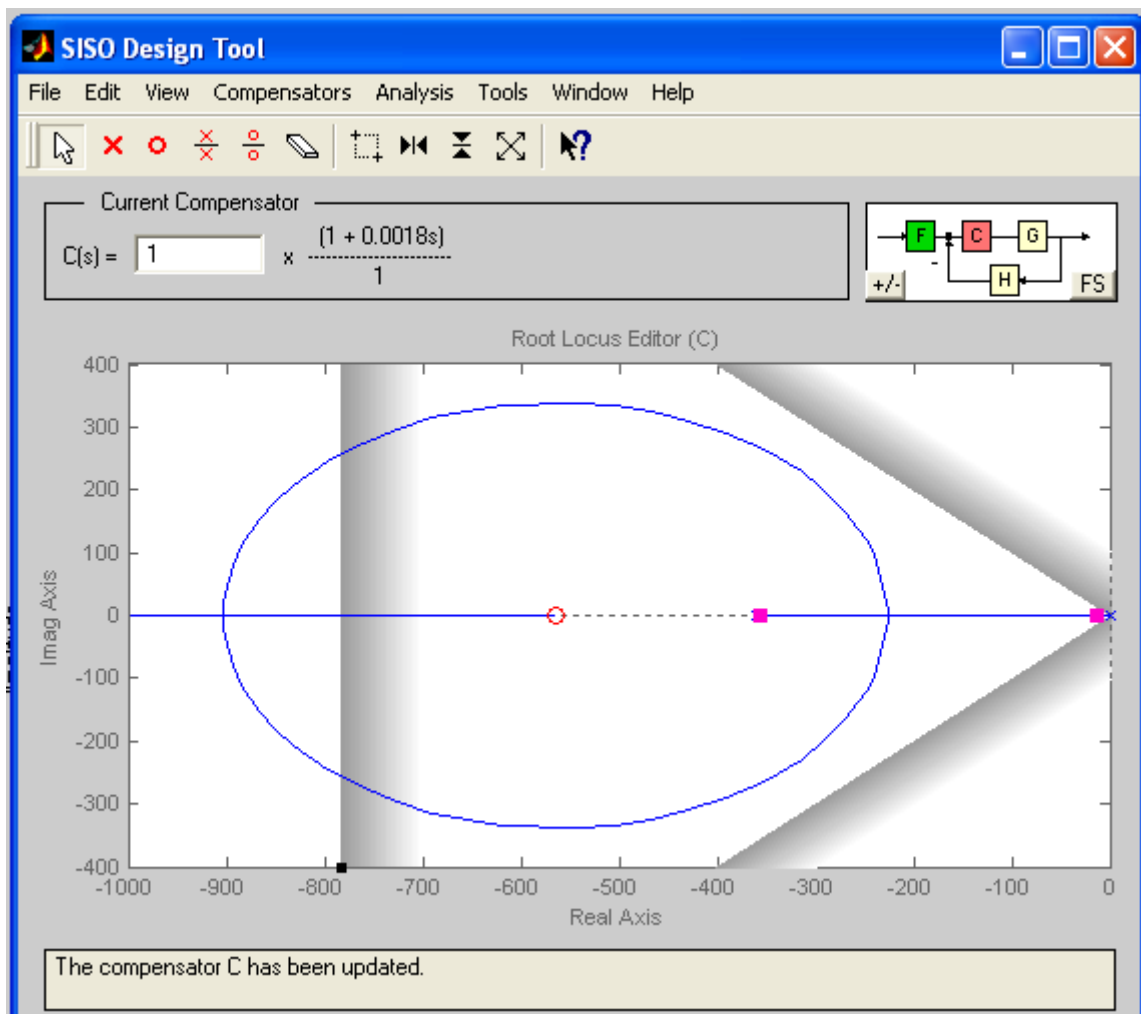
Hình 2.30: Quỹ đạo nghiệm sau khi tích hợp ràng buộc thiết kế là phần trăm vọt lố và thời gian thiết lập.







Hình 2.31 : Sự thêm vào zero của bộ điều khiển  $C(s)$  để tạo ra bộ điều khiển PD Quỹ đạo nghiệm mới xuất hiện như hình sau:



Hình 2.32: Quỹ đạo nghiệm sau khi tích hợp zero trong bộ điều khiển PD ở  $-1/0.001772$ .

**Các bài thực tập:**

1. Thiết kế mẫu:

Mục đích: sinh viên hãy thiết kế bộ bù sớm trễ pha hệ thống sau:

$$G(s) = \frac{A}{s(s+1)(s+2)} \text{ với } A=20$$

Sao cho :  $K_v=10$ , độ d ự tr ữ bi ên=12 dB, độ d ự tr ữ pha là  $50^\circ$ .

Thí nghiệm :

2. Thiết kế hiệu chỉnh sớm trễ pha hệ thống 2:

Có hàm truyền sau :

$$G(s) = \frac{80}{s(0.02s + 1)(0.05s + 1)}$$

Sao cho :  $K_v=80$ , độ dự trữ biên=10 dB, độ dự trữ pha là  $50^\circ$ .

Thí nghiệm :

3. Xem xét mô hình bậc 2 của hệ thống điều khiển hành vi máy bay. Hàm truyền của đối tượng là :

$$G_p(s) = \frac{4500K}{s(s + 361,2)}$$

Thiết kế bộ điều khiển PD với hàm truyền  $G_c(s)=K_p + K_D.s$  để các tiêu chuẩn chất lượng sau thỏa mãn:

- Sai số xác lập với ngõ vào unit-ramp  $\leq 0,001$ .
- Vọt lố cực đại  $\leq 5\%$ .
- Thời gian lên  $t_r \leq 0,005$  sec.
- Thời gian thiết lập  $t_s \leq 0,005$  sec.

#### 2.4. Kiểm tra đánh giá

## Bài 3

# KHẢO SÁT VÀ MÔ PHỎNG HỆ THỐNG TỰ ĐỘNG DÙNG SIMULINK

### 3.1. Mục tiêu

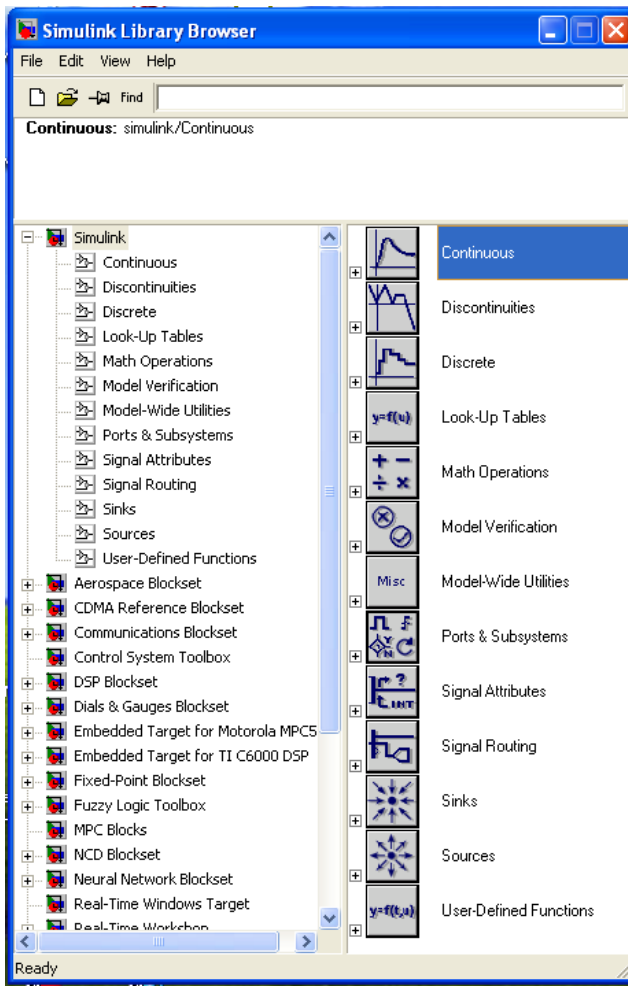
Simulink là một công cụ rất mạnh của Matlab để xây dựng các mô hình một cách trực quan và dễ hiểu. Để mô tả hay xây dựng hệ thống ta chỉ cần liên kết các khối có sẵn trong thư viện của simulink lại với nhau. Sau đó tiến hành mô phỏng hệ thống để xem xét ảnh hưởng của bộ điều khiển đến đáp ứng quá độ của hệ thống và đánh giá chất lượng hệ thống. Sau đó sinh viên thực hiện mô phỏng, khảo sát hệ thống điều khiển nhiệt độ và hệ thống điều khiển tốc độ và vị trí động cơ một chiều.

### 3.2. Nội dung

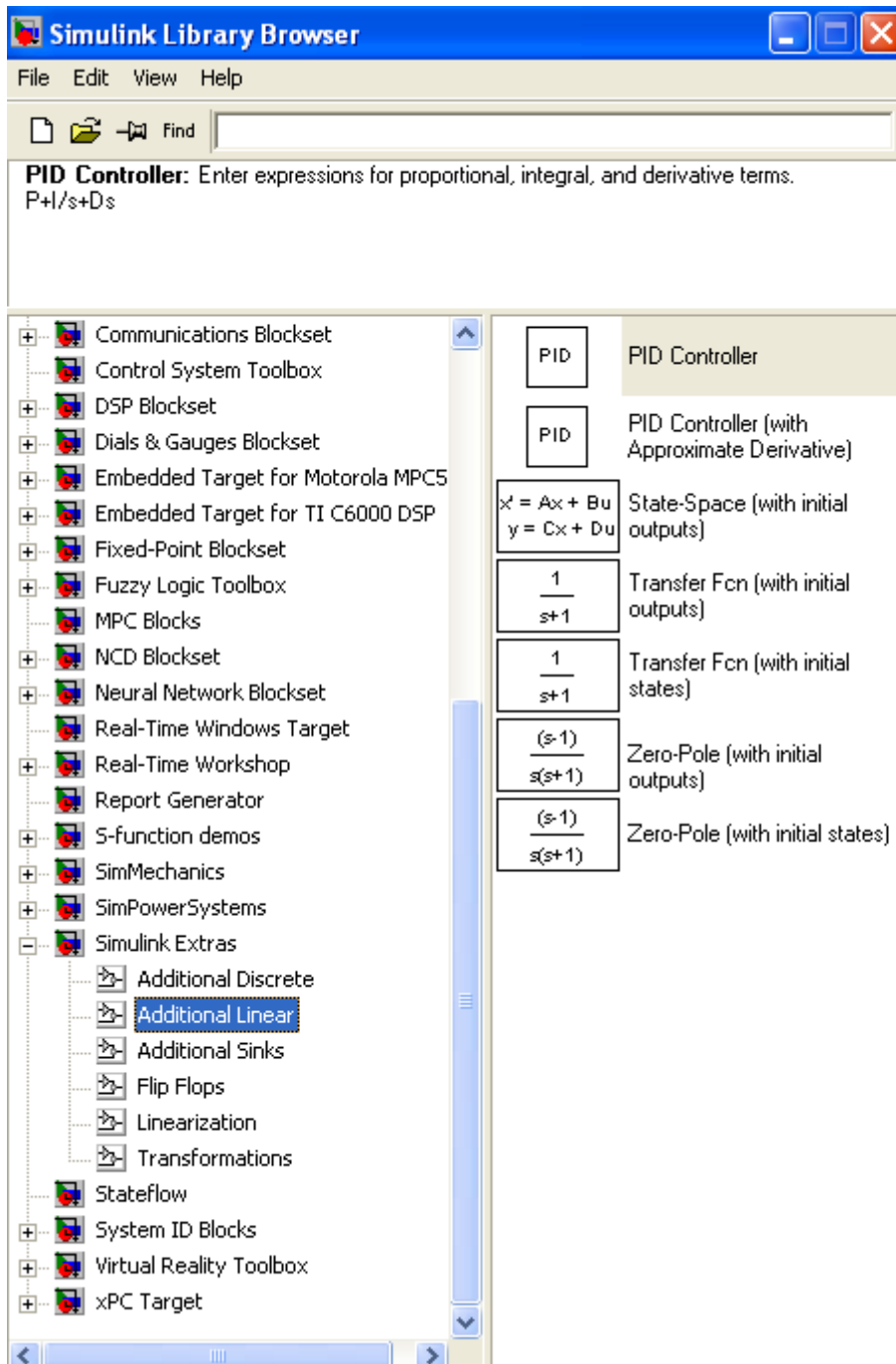
#### 3.2.1. GIỚI THIỆU VỀ Simulink

##### Khởi động Simulink:

- Khởi động Matlab.
  - Gõ lệnh simulink từ dấu nhắc lệnh.
- Cửa sổ đầu tiên được trình bày bởi Simulink.



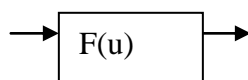
Hình 3.1: Cửa sổ Simulink.



Hình 3.2: Simulink Extras

### Khối chức năng

Khối thư viện Continuous gồm có vi phân, tích phân, hàm truyền, và không gian trạng thái,...  
 Thí dụ : khối Transfer Fcn (hàm truyền)



Fcn

### Tạo mới và soạn thảo lưu đồ tín hiệu:

-Sao chép, di chuyển, đánh dấu, xoá, hệ thống con, nối hai khối, di chuyển đường nối,...

**Simulink** có các thư viện khối chuẩn, được tổ chức thành các khối con theo chức năng.

Các khối thông dụng là :

- . Sources.
- .Sinks.
- .Discrete.
- Continuous.
- .Math Operations.

**Tín hiệu và các loại dữ liệu:**

Đối với Simulink, khái niệm tín hiệu chỉ nhằm vào dữ liệu xuất hiện ở đầu ra của các khối chức năng trong quá trình mô phỏng. Ta tạm hình dung rằng các tín hiệu (các dữ liệu) đó chạy dọc theo đường nối từ đầu ra của khối chức năng này tới đầu vào của các khối chức năng khác mà không tốn thời gian.

Mỗi tín hiệu thuộc sơ đồ cấu trúc Simulink đều được gán một loại số liệu nhất định, và do đó quyết định đến dung lượng bộ nhớ dành cho một tín hiệu. Simulink hỗ trợ tất cả các loại số liệu của Matlab: double, single, int8, uint8, int16, uint16, int32, uint32, v Boolean.

**Thư viện sources và Sinks:**

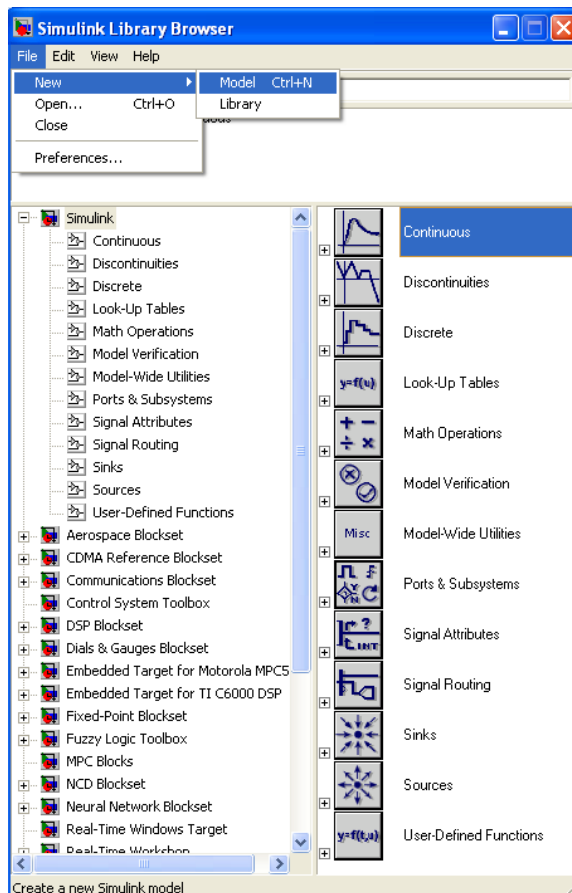
- Sources như là Constant, Step và Ramp, Signal Generator và Pulse Generator,..
- Sinks: Scope, XYGraph, ToWorkspace,..

**Thư viện Math:**

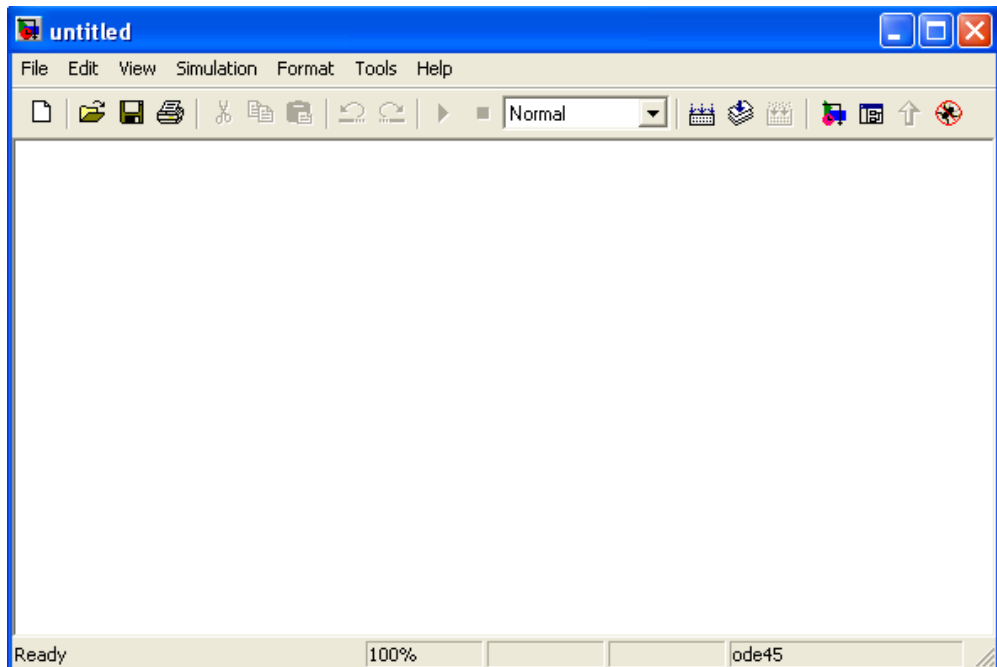
- Sum, Product và Dot Product, Math Function, Gain,..

**3.2.1. Các bước tiến hành mô phỏng:**

Từ cửa sổ thư viện khối, ta tạo mới bằng File, New hay mở file có sẵn File, Open. Untitled được tạo ra đầu tiên và có thể đặt tên lại bằng chọn Save As từ menu File.

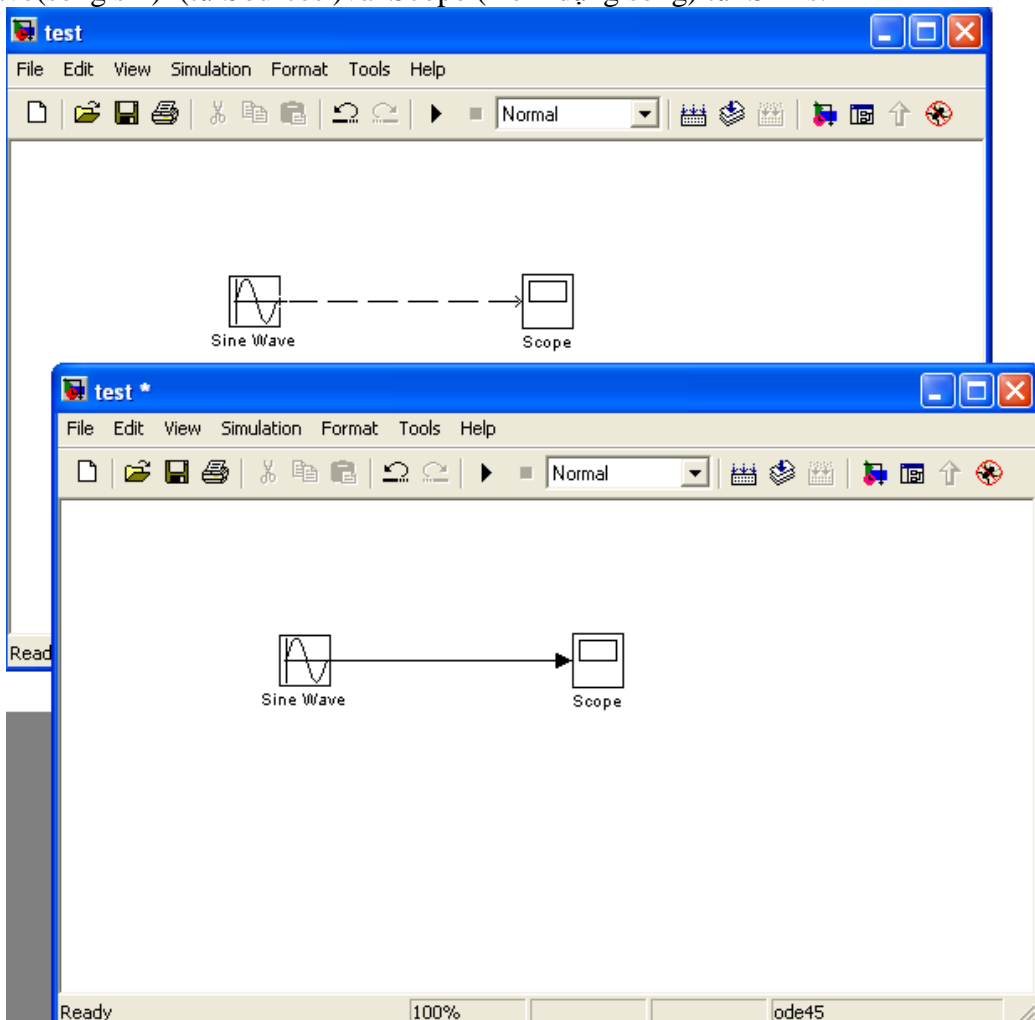


Hình 3.3



Hình 3.4

Để tạo mô hình, ta bấm và thả (drag and drop) các khối từ thư viện khối chuẩn vào mô hình. Trong mô hình đơn giản vừa tạo ra ở trên, ta sẽ định vị và sao chép khối Sine Wave (sóng sin) (từ Sources) và Scope (Xem dạng sóng) từ Sinks.

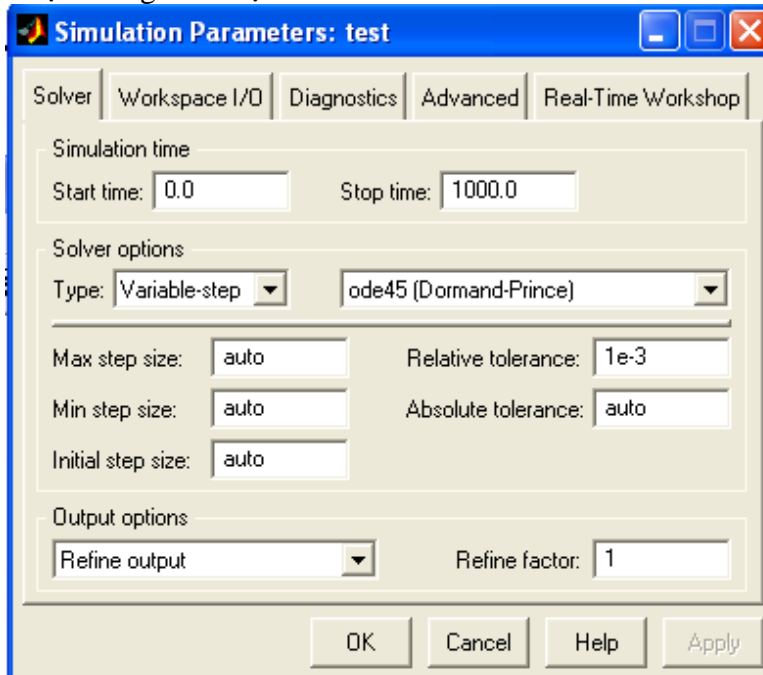


Hình 3.5 và 3.6

Sine Wave 1 là output port, v à Scope là Input port.

Chuẩn bị mô phỏng: khai báo tham số và phương pháp tích phân.

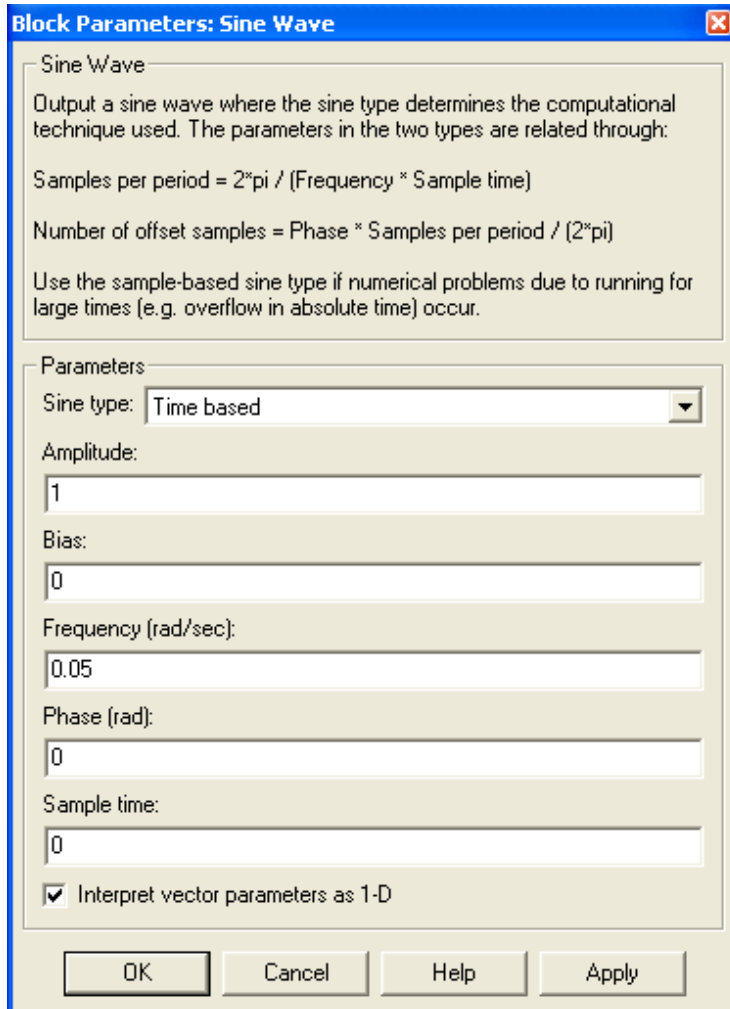
Chọn thông số : chọn Simulation Parameters từ menu Simulation.



Hình 3.7.

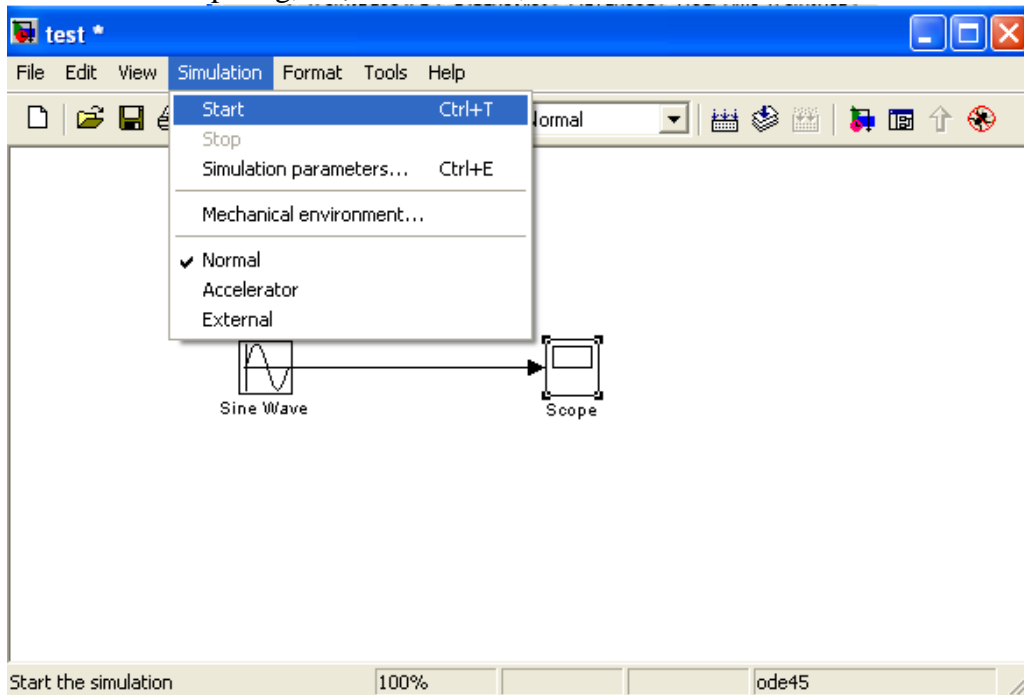
Thời gian mô phỏng bắt đầu từ 0 v à kết thúc là 1000. Nhấn vào nút Ok khi làm xong.

Nhấp đúp chuột vào khối Sine Wave để hiệu chỉnh thông số, chọn tần số là 0.05 rad/sec.



Hình 3.8

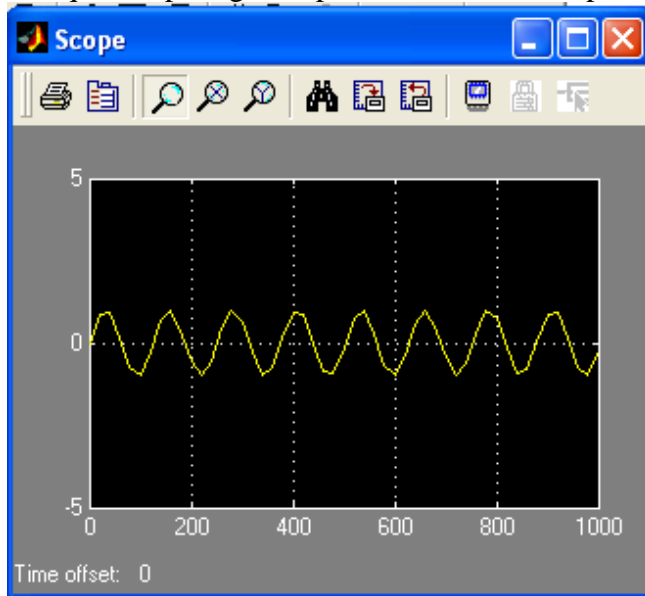
Để bắt đầu mô phỏng, chọn Simulation, start.



Hình 3.9.



Kết quả mô phỏng : nhấp chuột vào khối Scope



Hình 3.10

Để kết thúc Simulink chọn Close từ menu File và trở về dấu nhắc lệnh Matlab để tiếp tục với Matlab. Để kết thúc phiên Simulink mà không rời Matlab, ta chỉ cần đóng tất cả cửa sổ Simulink. Để thoát cả Matlab và Simulink, chọn Exit Matlab.

Trong phân tích và thiết kế hệ thống điều khiển hồi tiếp, Simulink cung cấp tập các khối mô hình hàm truyền và không gian trạng thái. Hình vẽ sau minh họa các thư viện khối Simulink của các mô hình liên quan đến điều khiển như là Continuous và Simulink Extras. Thí dụ bộ điều khiển PID nằm trong Simulink Extras.

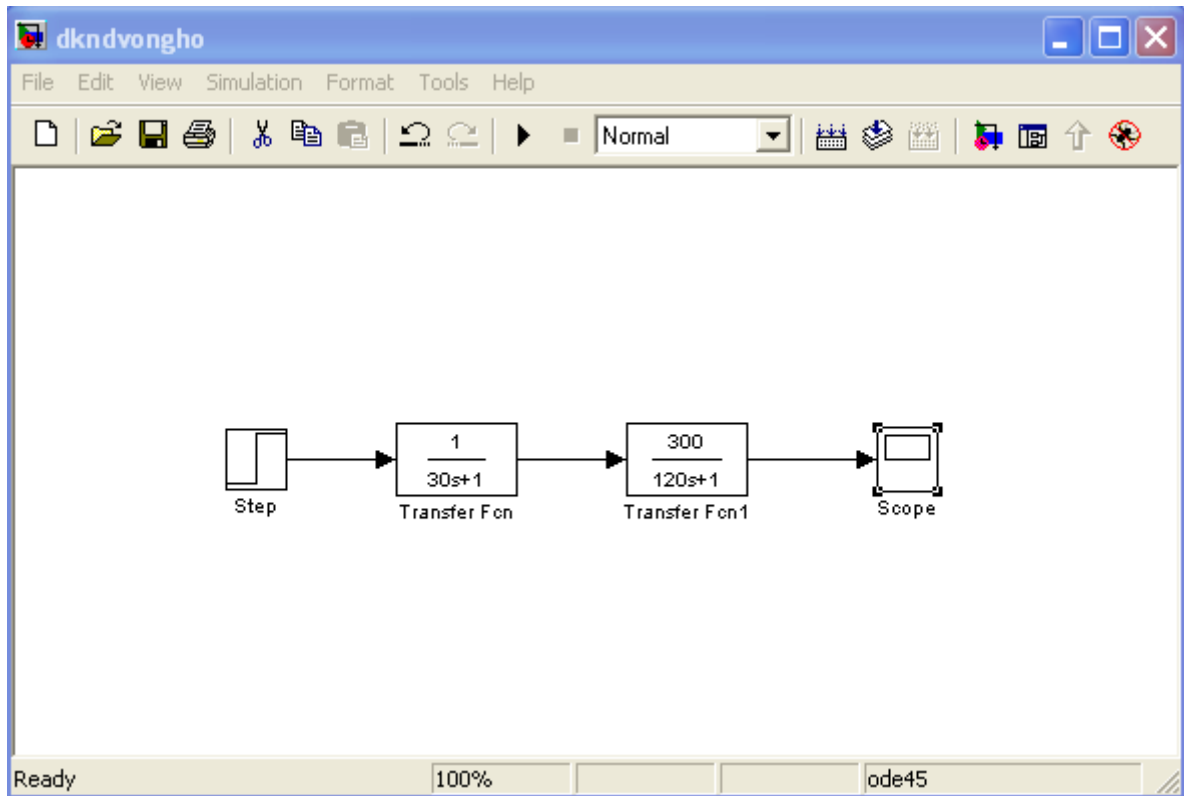
### 3.3. Các bài thực tập:

#### 3.3.1. Khảo sát hệ thống ổn định nhiệt độ

##### 3.3.1.1. Khảo sát hệ hở, nhận dạng hệ thống theo mô hình Ziegler-Nichols:

Mục đích: khảo sát khâu quán tính bậc hai, dùng để so sánh với mô hình Ziegler-Nichols (xem bài 4).

Thí nghiệm: Dùng Simulink xây dựng mô hình hệ thống lò nhiệt vòng hở như sau:



Hình 3.11. Mô hình khảo sát vòng hở.

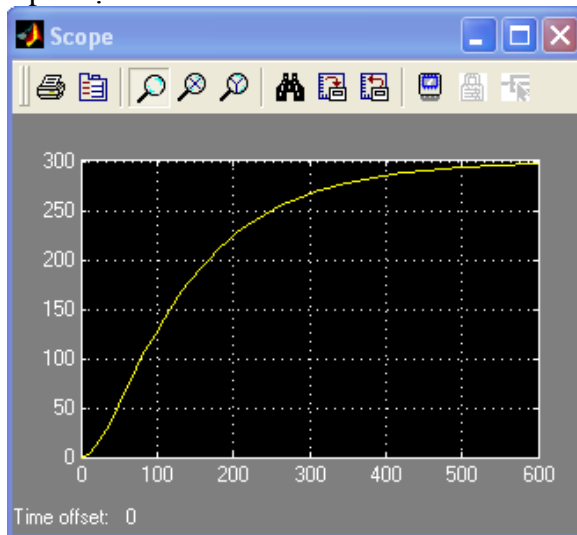
Step: là tín hiệu hàm nấc thể hiện phần trăm công suất cung cấp cho lò nhiệt.  
 Giá trị của hàm nấc từ  $0 \rightarrow 1$  tương ứng công suất cung cấp  $0\% \rightarrow 100\%$ .

Transfer Fcn-Transfer Fcn1: mô hình lò nhiệt tuyến tính hóa.

- Chỉnh giá trị của hàm nấc bằng 1 để công suất cung cấp cho lò là 100% (Step time=0, Initial time=0, Final time=1). Chỉnh thời gian mô phỏng Stop time=600 s. Mô phỏng và vẽ quá trình quá độ của hệ thống trên.
- Trên hình vẽ ở câu trên, vẽ tiếp tuyến tại điểm uốn để tính thông số L và T theo như hướng dẫn trong bài thí nghiệm 4. Chỉ rõ các giá trị này trên hình vẽ. So sánh giá trị L, T vừa tìm được với giá trị của mô hình lò nhiệt tuyến tính hóa.

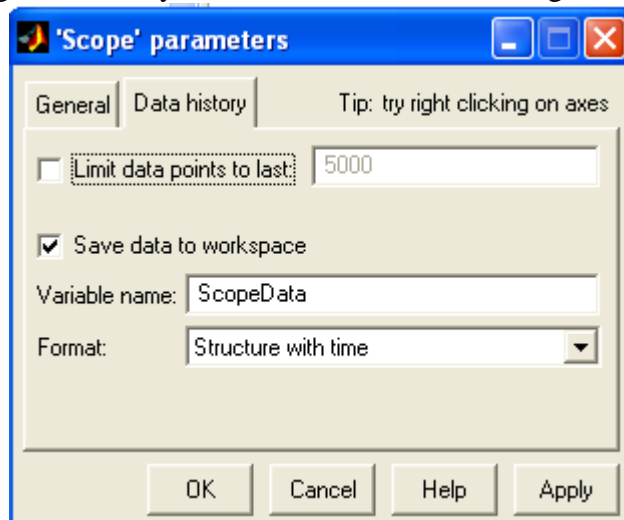
Hướng dẫn:

Sau khi chạy xong mô phỏng, để xem quá trình quá độ của tín hiệu ta nhấp đúp chuột vào khối Scope. Cửa sổ Scope hiện ra như sau:



Hình 3.12

Vì cửa sổ Scope chỉ có thể xem đáp ứng hoặc in trực tiếp ra máy in nhưng không lưu hình vẽ thành file \*.bmp được nên ta phải chuyển Scope này sang cửa sổ Figure để lưu. Thực hiện điều này bằng cách nhấp chuột vào ô Parameters. Cửa sổ Parameters hiện ra, nhấp chuột vào trang Data history và tiến hành cài đặt các thông số như hình bên dưới:



Hình 3.13

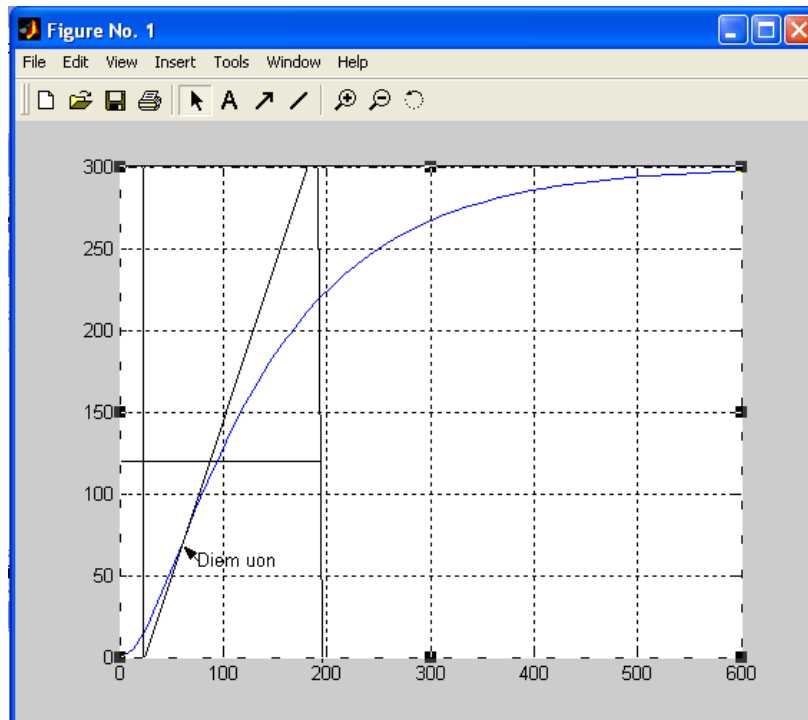
Tiến hành chạy mô phỏng lại để tín hiệu lưu vào biến ScopeData. Chú ý là nếu sau khi khai báo mà không tiến hành mô phỏng lại thì tín hiệu sẽ không lưu vào biến ScopeData mặc dù trên cửa sổ Scope vẫn có hình vẽ.

Sau đó vào cửa sổ Command Window nhập lệnh sau:

```
>> plot(ScopeData.time,ScopeData.signals.values) %ve dap ung
```

```
>> grid on % ke luoi
```

Lúc này cửa sổ Figure hiện ra với hình vẽ giống như hình vẽ ở cửa sổ Scope. Vào menu Insert/Line, Insert/Text để tiến hành kẻ tiếp tuyến và chú thích cho hình vẽ. Kết quả cuối cùng như hình bên dưới.



Hình 3.14

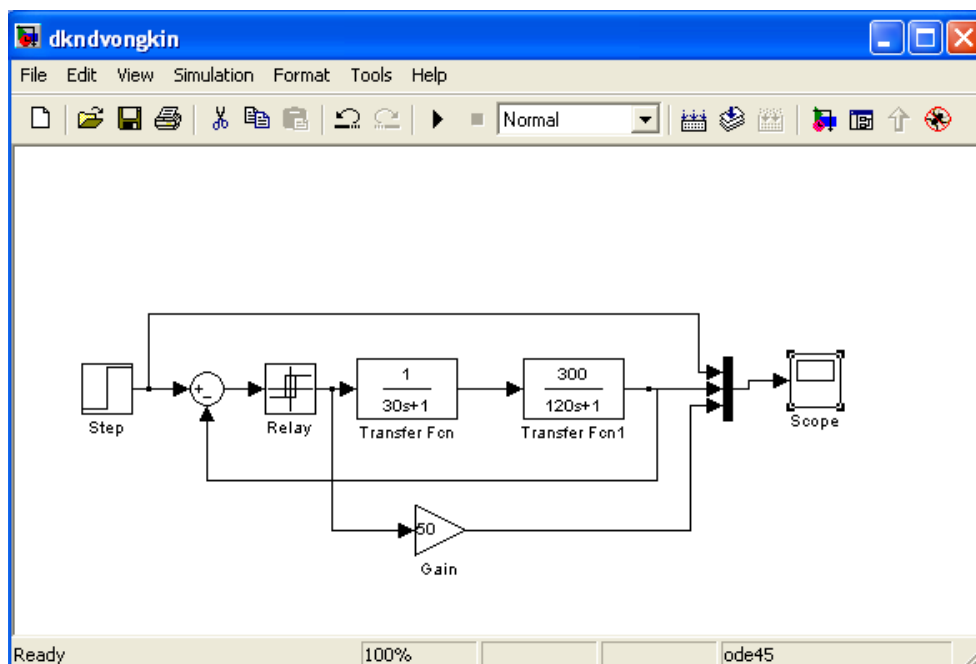
Vào menu [File] → [Export] để lưu thành file \*.bmp như ở bài thí nghiệm 1.

### 3.3.1.2. Khảo sát mô hình hệ thống ổn định nhiệt, điều khiển ON-OFF:

Mục đích: khảo sát hệ thống (HT) ổn định nhiệt độ điều khiển ON-OFF, xét ảnh hưởng của đặc tính rơle có trễ.

Thí nghiệm:

Mô hình được khảo sát tương tự với HT ổn định nhiệt độ của bài 4 (với thông số khác). Sơ đồ khối như sau:



Hình 3.15

Trong đó:

- lò nhiệt được biểu diễn bằng hai khâu quán tính nối tiếp.
- Đầu vào:  $u$  = hàm nấc, đặt 100 (nhiệt độ đặt 1000C).
- Tín hiệu ra đo ở Scope.
- Sum lấy hiệu số tín hiệu đặt và phản hồi.
- Relay là khâu rơle có trễ (Smith trigo), điều khiển On-off..
- Giá trị độ lợi ở khối Gain = 50 dùng để khuếch đại tín hiệu ngõ ra khỏi Relay để quan sát cho rõ. Lưu ý rằng giá trị này không làm thay đổi cấu trúc của hệ thống mà chỉ hỗ trợ việc quan sát tín hiệu.

a. Chính thời gian mô phỏng Stop time = 600s để quan sát được 5 chu kỳ điều khiển. Khảo sát quá trình quá độ của hệ thống với các giá trị của khâu rơle theo bảng sau:

Vùng trễ (Switch on/off point)	Ngõ ra cao (Output when on)	Ngõ ra thấp (Output when off)
+1/-1	1 (Công suất 100%)	0 (công suất 0%)
+5/-5	1 (công suất 100%)	0 (công suất 0%)
+10/-10	1 (công suất 100%)	0 (công suất 0%)
+20/-20	1 (công suất 100%)	0 (công suất 0%)

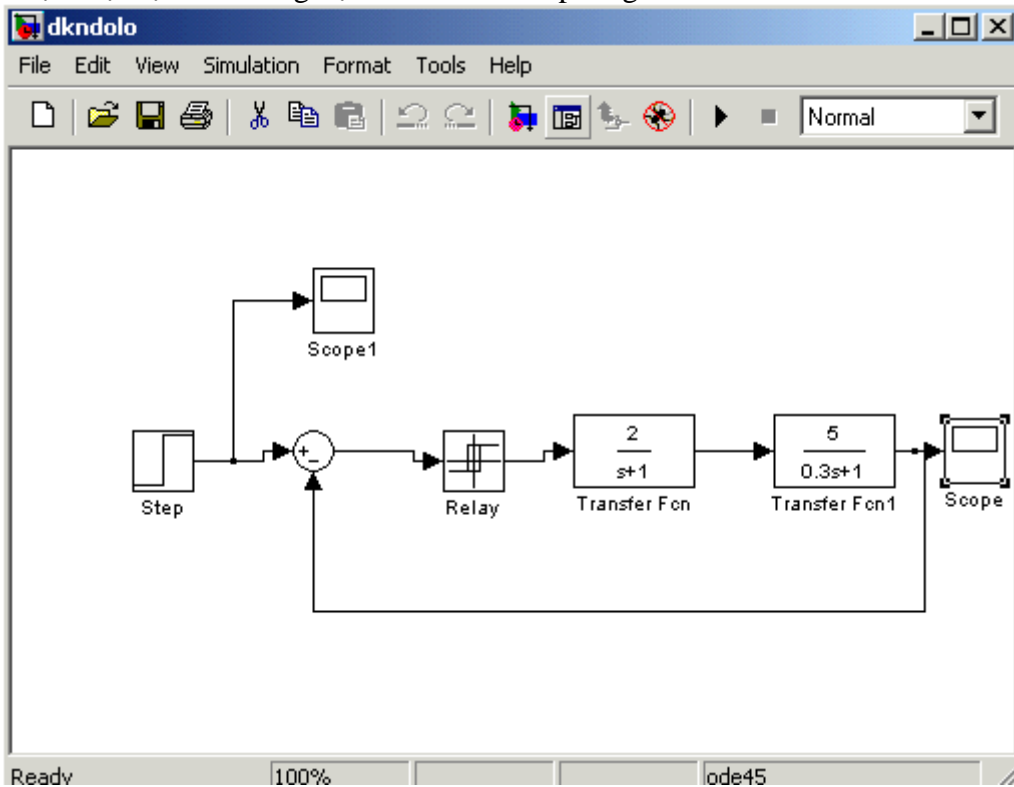
b. Tính sai số ngõ ra so với tín hiệu đặt và thời gian đóng ngắt ứng với các trường hợp của khâu Role ở câu a theo bảng sau:

Vùng trễ	$\Delta e1$	$\Delta e2$	Chu kỳ đóng ngắt (s)
+1/-1			
+5/-5			
+10/-10			
+20/-20			

Nhận xét ảnh hưởng của vùng trễ đến sai số ngõ ra và chu kỳ đóng ngắt của khâu rơle (khoảng thời gian ngõ ra khâu rơle thay đổi 1 chu kỳ).

**Bài tập:**

Thực hiện lại bài thí nghiệm với sơ đồ mô phỏng như sau:



Hình 3.16

**3.3.1.3. Hệ thống điều khiển nhiệt độ hiệu chỉnh Ziegler-Nichols (điều khiển PID):**

Mục đích: Sử dụng hiệu chỉnh Ziegler-Nichols (PID) cho HT điều khiển nhiệt độ đang khảo sát. Từ đó so sánh chất lượng của hệ thống ở hai bộ điều khiển PID với bộ điều khiển On/off.

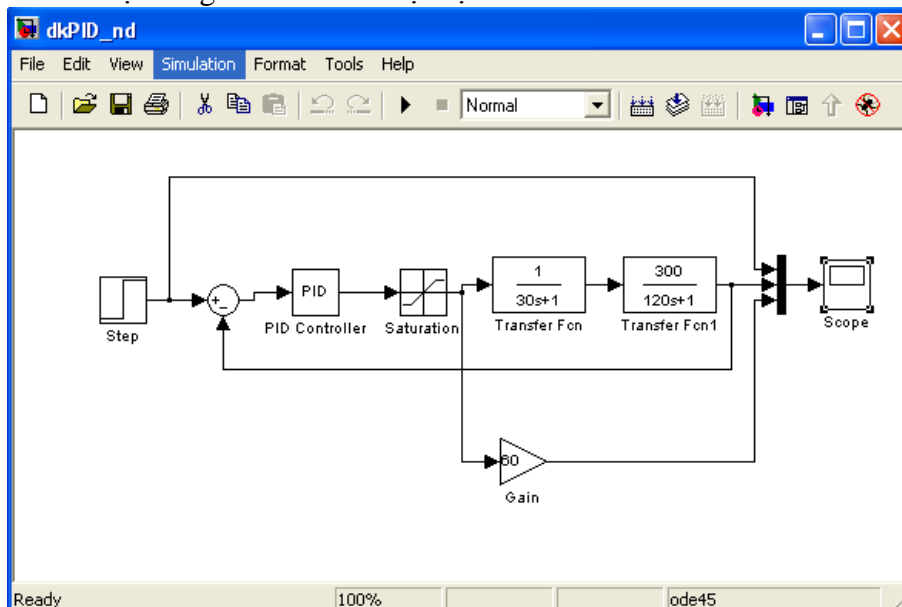
Thí nghiệm:

Bộ hiệu chỉnh có hàm truyền:  $PID = K_p + K_i/s + K_d \cdot s$

Trong đó:  $K_p = (1.2 \cdot T) / LK$ ;  $K_i = K_p / (2 \cdot L)$ ;  $K_d = 0.5 \cdot K_p \cdot L$

Trong đó L, T, K là các giá trị đã tìm được ở phần 3.3.1.1. Chú ý giá trị K đã cho trước ở mô hình hàm truyền lò nhiệt  $K=300$ .

Xây dựng mô hình hệ thống điều khiển nhiệt độ PID như sau:



Hình 3.17

trong đó:

. Tín hiệu đầu vào hàm nấc  $u(t)=100$  (tương trưng nhiệt độ đặt  $100^{\circ}\text{C}$ ).

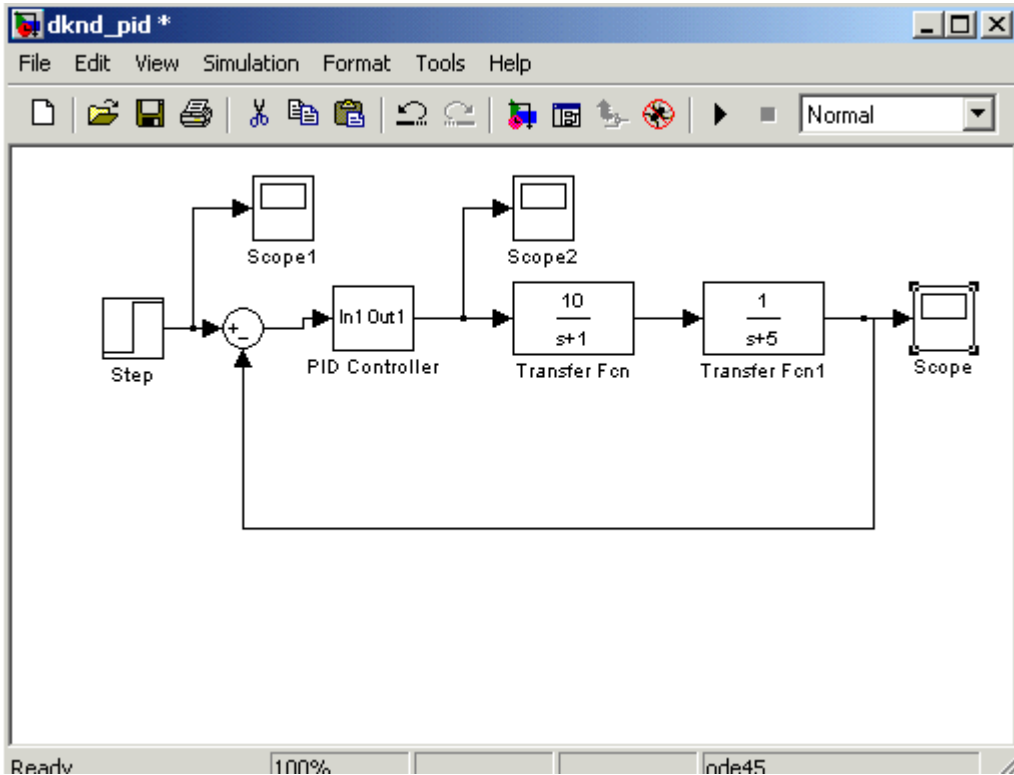
. Khâu bảo hoà Saturation có giới hạn upper limit=1, lower limit=0 (tương trưng ngõ ra bộ điều khiển có công suất cung cấp từ 0% đến 100%).

. Bộ điều khiển PID có thông số cần tính toán.

. Transfer Fcn-Transfer Fcn1 : mô hình lò nhiệt tuyến tính hóa.

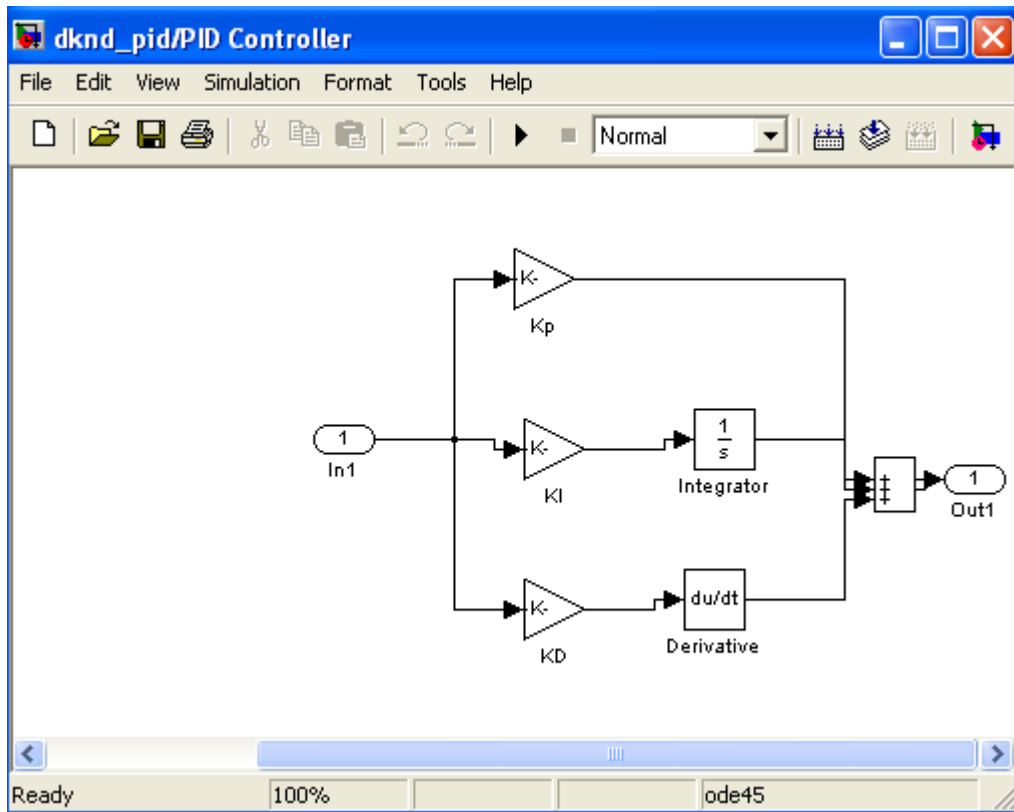
**Bài tập :**

Thực hiện lại bài thí nghiệm theo sơ đồ mô phỏng như sau (H3.18):



Hình 3.18

Bên trong bộ điều khiển PID(H3.19)

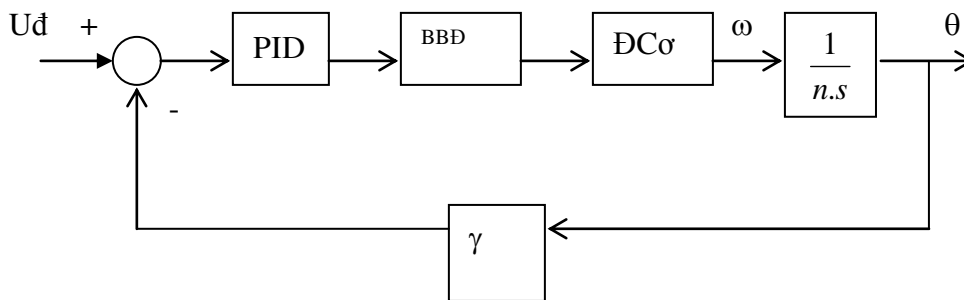


Hình 3.19

### 3.3.2. Khảo sát mô hình hệ thống điều khiển vị trí và tốc độ động cơ DC:

#### 3.3.2.1. HT điều khiển dùng một khâu hiệu chỉnh PID vị trí:

Khảo sát mô hình tuyến tính có sơ đồ khối sau:



Trong đó :  $U_d$  là tín hiệu đặt; PID là bộ hiệu chỉnh; BBĐ là bộ biến đổi cung cấp điện áp thay đổi được cho động cơ một chiều kích từ độc lập ĐC $\sigma$ ;  $\omega, \theta$  là tốc độ trục động cơ và vị trí của tải, quan hệ qua hệ số giảm tốc  $n$ ;  $\gamma$  là hệ số phản hồi vị trí.

a. Khảo sát HT điều khiển vị trí hiệu chỉnh PD có tính đến sự bảo hoà của bộ biến đổi BBĐ với các ngõ vào khác nhau.

b. Khảo sát HT khi hiệu chỉnh PD và PID với ngõ vào hàm dốc.

Sinh viên tham khảo bài thí nghiệm 5 để biết rõ phương trình mô tả động cơ DC. Từ phương trình mô tả động cơ, ta có sơ đồ khối biểu diễn mô hình động cơ như sau:

Trong đó:

. Phần ứng :  $R=1 \text{ Ohm}, L=0.03\text{H} \rightarrow T_{dt}=0.03\text{s}$ .

. Ce: hằng số điện từ,  $C_e=0.2\text{V.s/rad}$ .

. M : moment động cơ,  $M_c$  : moment cản.

U : giá trị điện áp đặt vào động cơ.

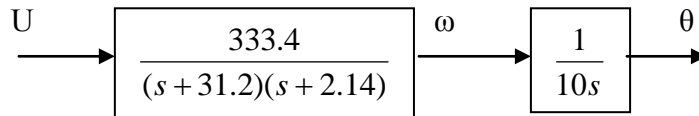
$J$  : moment quán tính của các phần chuyển động,  $J=0.02 \text{ kgm/s}^2$ .

$\Omega$  : tốc độ quay của động cơ (rad/s).

$\theta$  : vị trí góc quay của động cơ (rad).

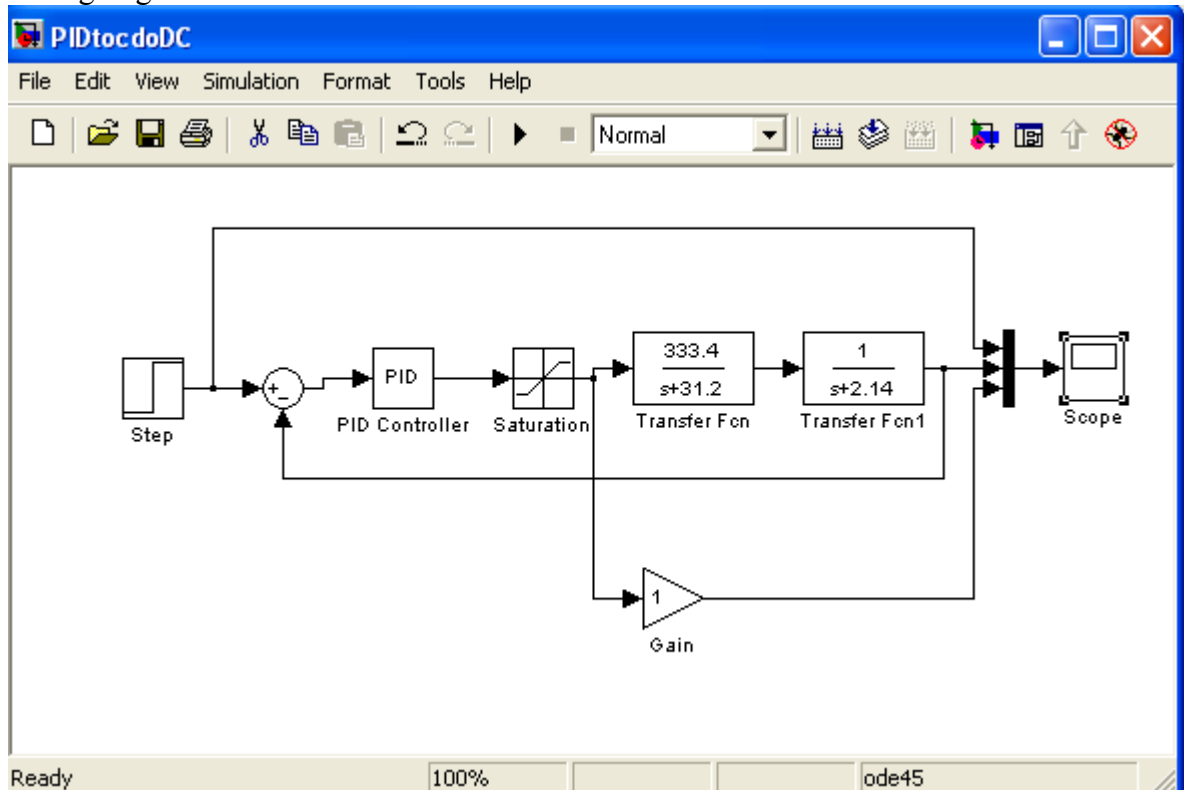
$N$  : tỉ số truyền, trong khảo sát :  $n=10$ .

Với điều kiện không tải  $M_c=0$ , thu gọn sơ đồ khối thành :



### Khảo sát mô hình điều khiển tốc độ động cơ DC :

Tương ứng sơ đồ Simulink:



Hình 3.20

Trong đó :

.Tín hiệu đặt đầu vào hàm nấc  $u(t)=100$  (tương trưng tốc độ đặt 100).

.Khâu bão hòa Saturation có giới hạn là  $+30/-30$  (tương trưng điện áp cung cấp cho phần ứng động cơ từ  $-30V$  đến  $+30V$ ).

.Transfer Fcn-Transfer Fcn1 thể hiện mô hình tốc độ động cơ DC.

- a. Chính thời gian mô phỏng Stop time =10s. Thực hiện khảo sát hệ thống với bộ điều khiển P( $K_i=0, K_d=0$ ) và tính độ vọt lố., sai số xác lập, thời gian xác lập của ngõ ra theo bảng sau:

$K_P$	1	10	20	50	100
POT					
$E_{xl}$					
$t_{xl}$					

Nhận xét chất lượng của hệ thống thay đổi như thế nào khi  $K_p$  thay đổi. Giải thích.



b. Thực hiện khảo sát với bộ điều khiển PI ( $K_p=2, K_d=0$ ) và tính độ vọt lố, sai số xác lập, thời gian xác lập của ngõ ra theo bảng sau:

$K_I$	0.1	0.5	0.8	1	2
POT					
$E_{xl}$					
$t_{xl}$					

Nhận xét chất lượng của hệ thống thay đổi thế nào khi  $K_i$  thay đổi. Giải thích. So sánh chất lượng bộ điều khiển PI với bộ điều khiển P.

c. Thực hiện khảo sát hệ thống với bộ điều khiển PID ( $K_p=2, K_i=2$ ) và tính độ vọt lố, sai số xác lập, thời gian xác lập của ngõ ra theo bảng sau:

$K_D$	0.1	0.2	0.5	1	2
POT					
$E_{xl}$					
$t_{xl}$					

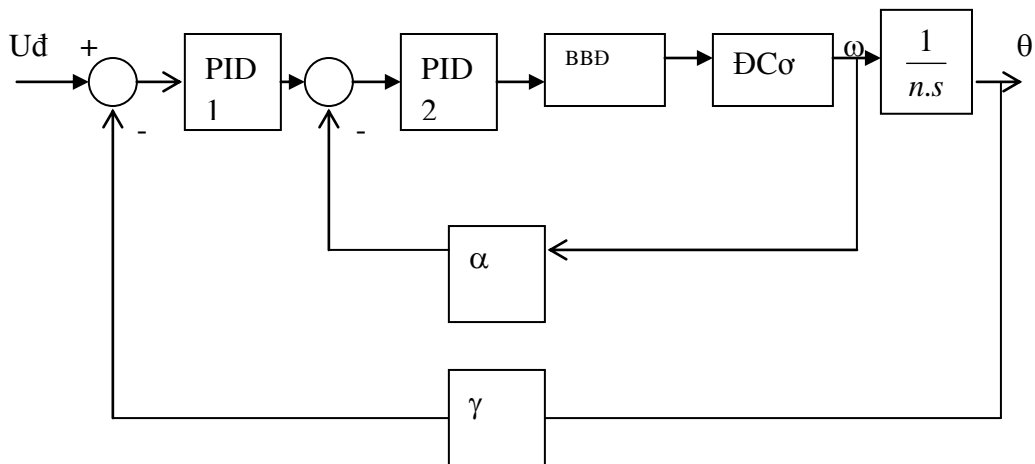
Nhận xét chất lượng của hệ thống thay đổi thế nào khi  $K_d$  thay đổi. Giải thích. So sánh chất lượng bộ điều khiển PID với bộ điều khiển P và PI.

d. Nhận xét ảnh hưởng của các khâu P, I, D lên chất lượng hệ thống.

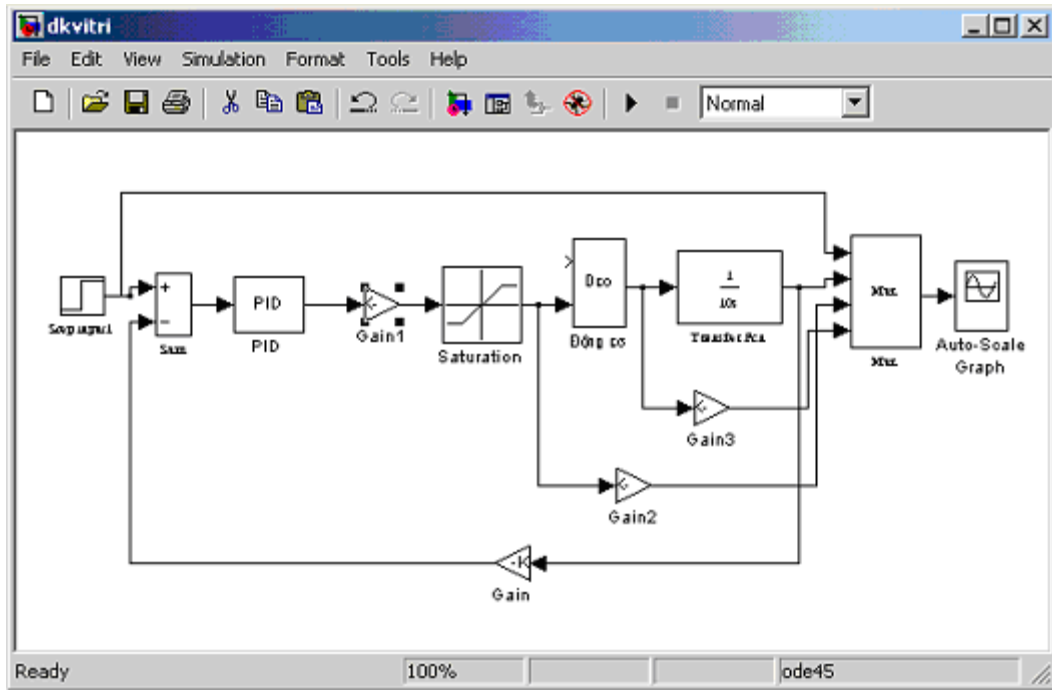
### 3.3.2.2. Hệ thống điều khiển dùng hai hiệu chỉnh PID vị trí và tốc độ:

**Mục đích:** Khảo sát HT điều khiển vị trí hiệu chỉnh PID với ngõ vào hàm dốc có phản hồi tốc độ, so sánh với hệ thống không có phản hồi tốc độ.

Sơ đồ khối như hình sau: trong đó BBĐ và bộ hiệu chỉnh vòng vị trí PID1 bị bão hoà (giới hạn biên độ);  $\alpha$  là hệ số phản hồi tốc độ.



Tương ứng sơ đồ Simulink.



Hình 3.21

### Bài tập ứng dụng

### 3.4. Kiểm tra đánh giá

## Bài 4

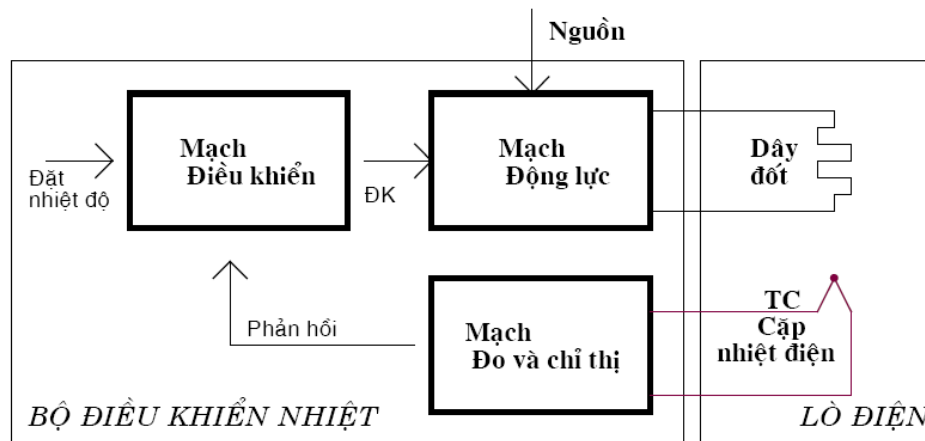
# HỆ THỐNG ĐIỀU KHIỂN NHIỆT ĐỘ

### 4.1. Mục tiêu

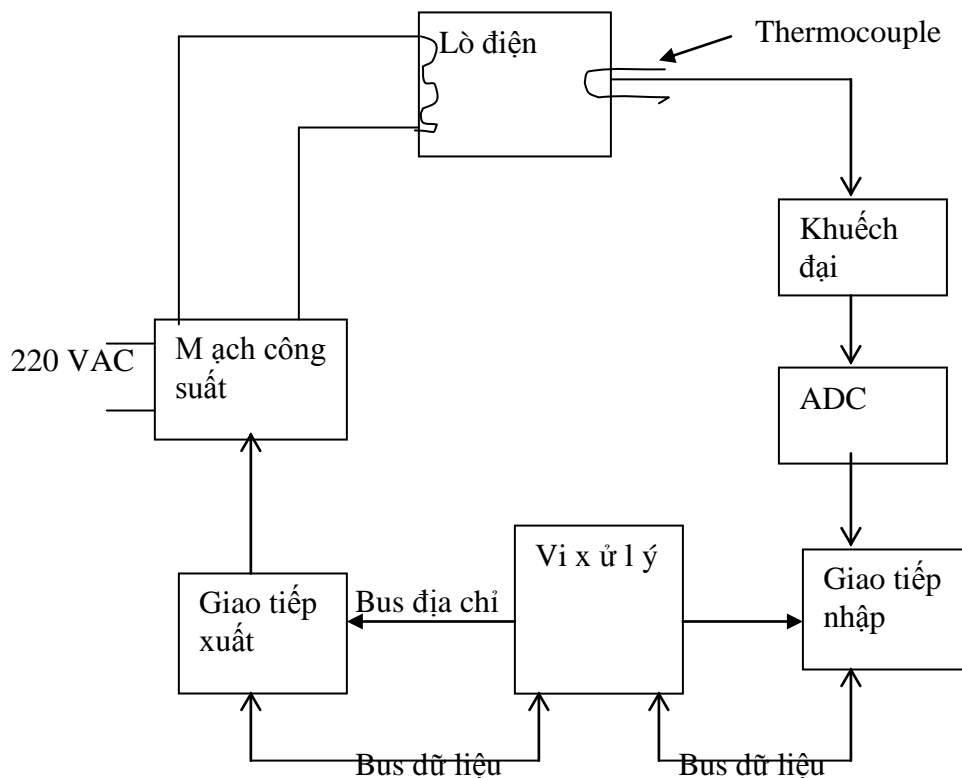
- Làm quen với đối tượng lò điện.
- So sánh nguyên lý điều khiển ON-OFF và tuyến tính trong điều khiển nhiệt độ.

### 4.2. Nội dung

Gồm có bộ điều khiển và lò điện có sơ đồ sau:



Hình 4.1a: Hệ thống điều khiển nhiệt độ



Hình 4.1b : Hệ thống điều khiển nhiệt độ dùng vi xử lý

Lò điện ở đây dùng loại lò nướng dân dụng 220 V đã bị loại bỏ phần điều khiển nhiệt độ.

Bộ điều khiển gồm mạch đo nhiệt độ sử dụng cặp nhiệt điện(Thermo Couple-có điện áp ra thay đổi theo nhiệt độ), mạch điều khiển dùng khuếch đại thuật toán và bộ chấp hành(mạch động lực) dùng TRIAC đóng ngắt nguồn điện lưới cung cấp cho lò khi áp qua zero(zero switching).

Với bộ điều khiển này , có thể chọn một trong hai nguyên tắc điều khiển ON-OFF(dùng rơle có trễ) hay tuyến tính (PID). Trong trường hợp sau, bộ chấp hành hoạt động theo nguyên tắc điều rộng xung, đảm bảo cung cấp công suất cho lò tỉ lệ với tín hiệu điều khiển tương ứng bộ điều khiển là hệ số khuếch đại (khâu P).

**4.2.1. Hàm truyền lò điện và mô hình của Ziegler-Nichols:**

Lò nhiệt có đầu vào là điện áp (hay công suất) cung cấp cho dây điện trở và ngõ ra là nhiệt độ bên trong lò. Để lập hàm truyền lò nhiệt ta phải khảo sát phương trình vi phân mô tả các quan hệ nhiệt độ và năng lượng. Đây là một bài toán phức tạp nếu muốn mô tả chính xác hàm truyền phi tuyến của hệ thống.

Một cách gần đúng, ta có thể xem môi trường nung là đồng chất, đẳng nhiệt. Từ phương trình cân bằng năng lượng : điện năng cung cấp sẽ được dùng để bù vào năng lượng nhiệt truyền ra bên ngoài và tích nhiệt vào môi trường nung, ta tính được hàm truyền lò là bậc nhất, có dạng như sau:

$$P \rightarrow \left[ \frac{K}{Ts + 1} \right] \rightarrow \theta$$

Trong đó :

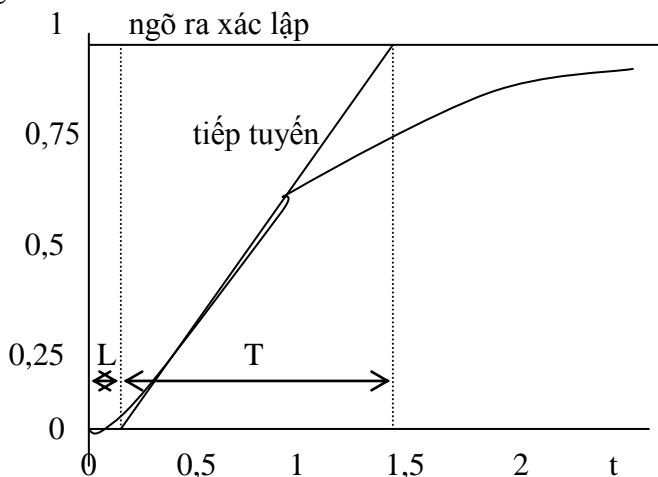
P: công suất cung cấp.

$\theta$ : là độ tăng nhiệt nhiệt độ ngõ ra so với nhiệt độ môi trường.

K: là hệ số tỉ lệ cho biết quan hệ vào ra ở chế độ xác lập.

T: là thời hằng, thể hiện quán tính nhiệt hệ thống.

Mô hình hàm truyền này cho thấy quá trình quá độ với đầu vào hàm nấc có dạng hàm mũ. Thực tế cho thấy mô hình trên chỉ là gần đúng, hệ thống có bậc cao hơn nhưng quá trình quá độ đầu vào hàm nấc vẫn là không vọt lố, có dạng như hình sau khi cho nhiệt độ đầu bằng 0.



Theo Ziegler-Nichols thì một hệ thống như vậy có thể được biểu diễn dưới dạng hàm truyền sau :

$$H(s) = \frac{Ke^{-Ls}}{Ts + 1}$$

bao gồm một khâu quán tính hệ số khuếch đại K và thời hằng T, và khâu trễ thời gian L, các thông số này có thể lấy được khi kẻ tiếp tuyến ở điểm uốn cho đồ thị quá độ hàm nấc như hình vẽ bên. Hệ số khuếch đại K được tính như sau :

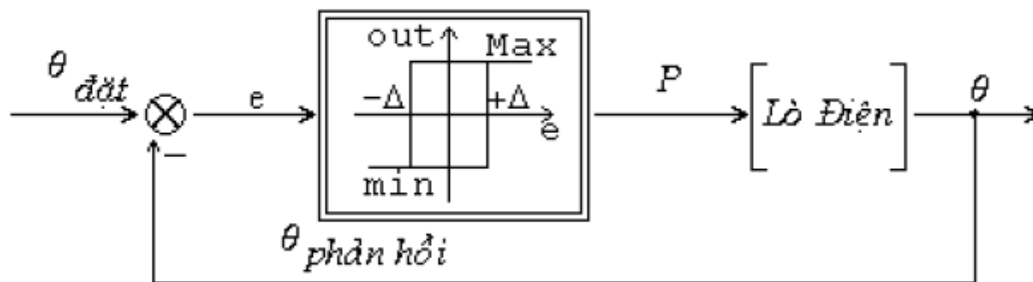
$$K = \frac{\text{Nhiệtdovac lap}}{\% \text{cong suat}}$$

Khi nhiệt độ ban đầu khác không, K được tính từ độ tăng nhiệt độ ngõ ra so với môi trường. Để áp dụng cho hệ tuyến tính, ta lấy khai triển Taylor của  $e^{-LS}$ , hàm truyền trở nên :

$$H(s) = \frac{K}{(Ts + 1)(Ls + 1)}$$

Tóm lại, Ziegler-Nichols xấp xỉ hàm truyền lò với hệ bậc nhất có trễ hay hệ tuyến tính bậc hai, và cho phép tìm hàm truyền bằng thực nghiệm khi vẽ quá trình quá độ hệ thống với ngõ vào là hàm nấc.

#### 4.2.2. Nguyên lý điều khiển ON-OFF (dùng khâu role có trễ) :



Hình 4.2

Phương pháp điều khiển ON-OFF còn được gọi là phương pháp đóng ngắt hay dùng khâu role có trễ : cơ cấu chấp hành sẽ đóng nguồn để cung cấp cho năng lượng ở mức tối đa cho thiết bị tiêu thụ nhiệt nếu nhiệt độ đặt  $\theta_{dait}$  lớn hơn nhiệt độ lò  $\theta_{phan\ hoi}$ , ngược lại mạch điều khiển sẽ ngắt mạch cung cấp năng lượng khi nhiệt độ đặt nhỏ hơn nhiệt độ thực của lò.

Một vùng trễ được đưa vào để hạn chế tần số đóng ngắt như sơ đồ khối trên : nguồn chỉ được đóng khi sai lệch nhiệt độ e lớn hơn một lượng  $\Delta$  và ngắt khi e bé hơn  $\Delta$ . Như vậy, nhiệt độ phản hồi  $\theta_{phan\ hoi}$  sẽ dao động quanh giá trị đặt  $\theta_{dait}$  và  $2\Delta$  còn được gọi là vùng trễ của role.

Khâu role có trễ còn gọi là mạch so sánh Smith trong mạch điện tử, và như vậy  $\Delta$  là giá trị thêm hay ngưỡng.

Điều khiển ON-OFF có ưu điểm là :

- Thiết bị tin cậy, đơn giản, chắc chắn, hệ thống luôn hoạt động được với mọi tải.
- Tính toán thiết kế ít phức tạp và cân chỉnh dễ dàng.

Nhưng có khuyết điểm là sai số xác lập sẽ lớn do hệ chỉ cân bằng động quanh nhiệt độ đặt và thay đổi theo tải. Khuyết điểm này có thể được hạn chế khi giảm vùng trễ bằng cách dùng phần tử đóng ngắt điện tử ở mạch công suất.

Việc điều khiển nhiệt độ với chất lượng cao có thể thực hiện bằng sơ đồ điều khiển tuyến tính với hàm truyền hiệu chỉnh thích hợp.

#### 4.2.3. Nguyên lý điều khiển tuyến tính:

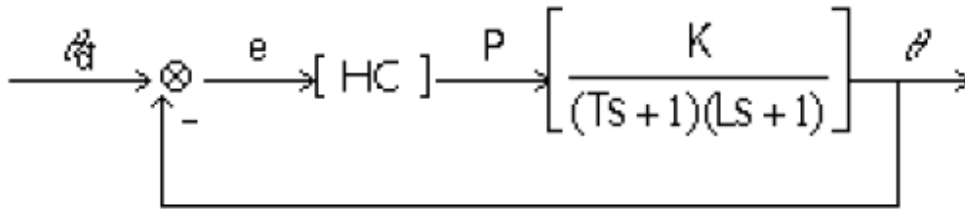
Khác với nguyên lý ON-OFF chỉ có hai giá trị công suất ở ngõ ra, sơ đồ điều khiển tuyến tính cung cấp công suất cho tải thay đổi một cách liên tục theo sai lệch giữa nhiệt độ đặt và nhiệt độ thực. Bộ hiệu chỉnh xử lý tín hiệu sai lệch đặt - phản hồi, cho ra tín hiệu điều khiển làm sai lệch tiến về zero với đặc tính quá độ mong muốn. Như vậy ở chế độ xác lập, nhiệt độ lò và công suất cung cấp cho lò sẽ có một giá trị xác lập, phụ thuộc vào đặc tính hệ thống.

Chất lượng hệ thống như vậy sẽ phụ thuộc vào thông số của sơ đồ hiệu chỉnh. Một trong những nguyên lý hiệu chỉnh thường dùng là PID (vi phân tích phân tỉ lệ), và trong bài

thí nghiệm ta dùng P (điều khiển tỉ lệ) vì không thể thực hiện các thời hằng của mạch vi tích phân (ở đây có trị số khá lớn) bằng mạch analog.

Như vậy, khâu hiệu chỉnh chỉ là mạch khuếch đại có hệ số thích hợp, dung hoà giữa độ chính xác và chất lượng quá độ vì nếu tăng hệ số khuếch đại, sai số sẽ giảm nhưng sẽ xuất hiện dao động và vọt lố ở ngõ ra.

Sơ đồ khối hệ thống điều khiển tuyến tính như sau:



Hình 4.3

HC: hàm truyền bộ hiệu chỉnh, trong thí nghiệm này chỉ là bộ khuếch đại.

Phương pháp điều khiển công suất được dùng là điều rộng xung. Tải sẽ nhận công suất trong khoảng  $T_{on}$  của chu kì  $T$  không đổi. Công suất trên tải có thể điều khiển được bằng cách thay đổi độ rộng xung tương đối  $a$ :

$$A = T_{on}/T.$$

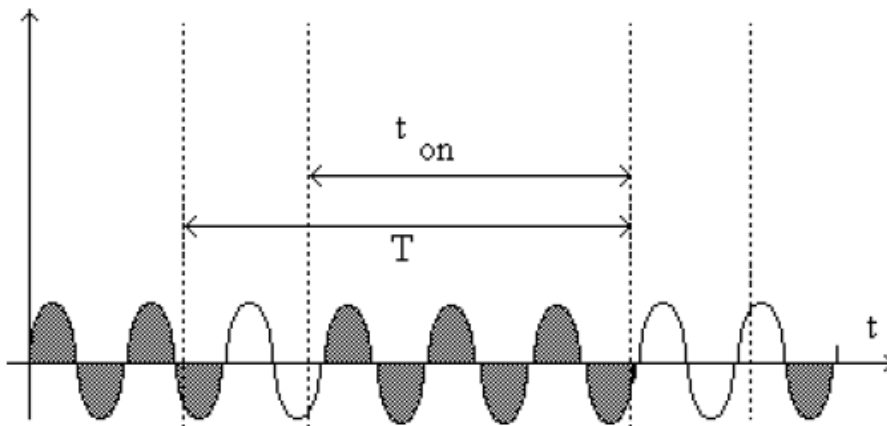
Và công suất cung cấp cho tải :

$$P = a * P_{max}.$$

$P_{max}$  : công suất cực đại ứng với trường hợp  $a=1$ , khi phần tử điều khiển công suất là TRIAC đóng mạch liên tục.

Vì TRIAC chỉ ngắt mạch khi dòng qua nó về zero, chu kì  $T$  phải đủ lớn để cho TRIAC có thể dẫn điện trong nhiều chu kì điện áp lưới (tần số lưới điện là 50 Hz).

Trong bài thí nghiệm,  $T$  khoảng 3 giây, bộ thí nghiệm có chỉ thị hệ số  $a$  theo đơn vị % gọi là % công suất.



$t_{on}$  : thời gian TRIAC đóng

$T$  : chu kỳ điều khiển

Hình 4.4

**4.3. Thí nghiệm: Sinh viên đọc tham khảo [2]**

**4.3.1. Đo quá trình quá độ hệ hở, đầu vào hàm nấc**

**. Mục đích:**

Vẽ đặc tính quá độ hệ hở, suy ra hàm truyền lò điện theo mô hình Ziegler-Nichol.

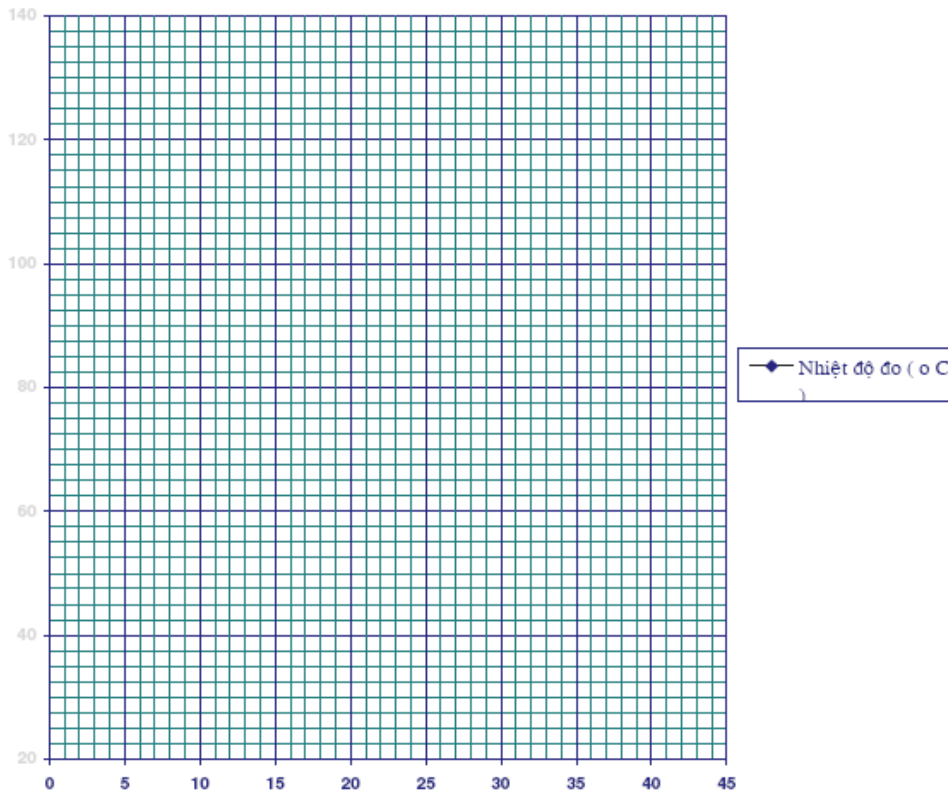
**. Thí nghiệm:**

Đặt chế độ vòng hở (công tắc chọn chế độ HỞ-KÍN bật qua hở), dùng biến trở đặt chỉnh công suất ra bằng 20%, ghi nhiệt độ lò từng 1 phút trong khoảng 30 phút vào bảng 1, vẽ đặc tính quá độ nhiệt độ ngõ ra trên hình 1 của cáo cáo và trả lời các câu hỏi trong báo cáo. (xem lại phần hướng dẫn).

Bảng 1: Quá trình quá độ hệ hở khi cung cấp 20% công suất.

T(min)	0	1	2	3	4	5	6	7	8	9	10	11	12
T°C													
$\Delta T^{\circ}\text{C}$													
T(min)	13	14	15	16	17	18	19	20	21	22	23	24	25
T°C													
$\Delta T^{\circ}\text{C}$													
T(min)	26	27	28	29	30	31	32	33	34	35	36	37	38
T°C													
$\Delta T^{\circ}\text{C}$													

Hình vẽ 4.5: Quá trình quá độ hệ hở, đầu vào hàm nấc.



Tính toán các hệ số của hàm truyền lò theo kiểu mẫu của Ziegler-Nichol:

**4.3.2. Điều khiển ON-OFF**

**. Mục đích:**

Khảo sát hệ thống điều khiển nhiệt độ theo nguyên tắc điều khiển ON-OFF.

**. Thí nghiệm:**

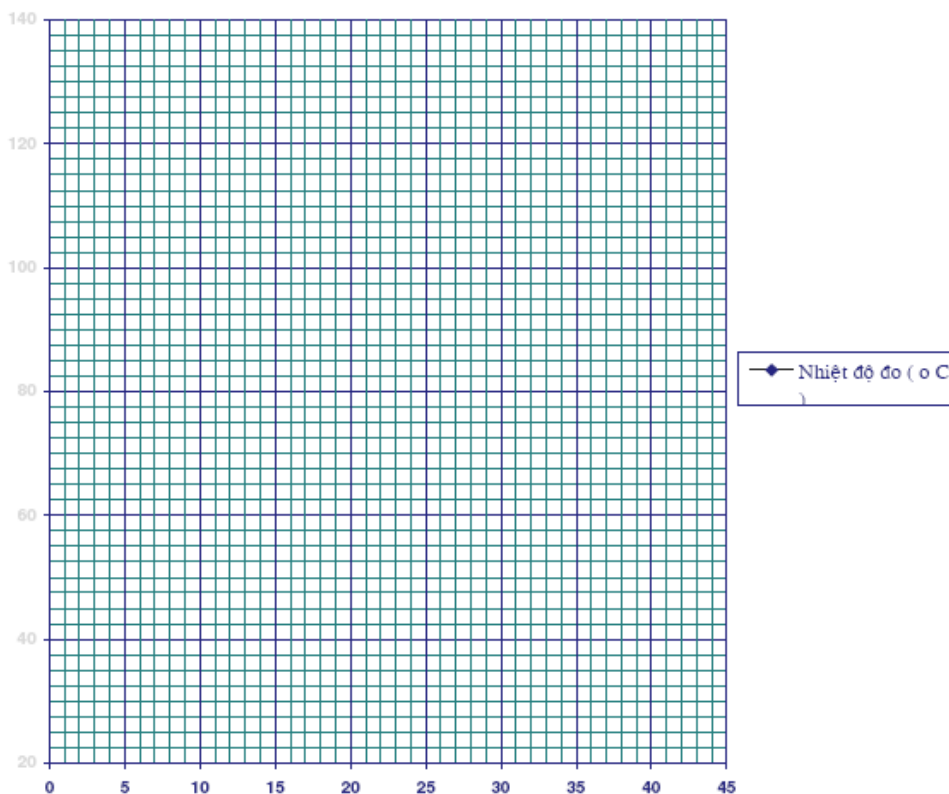
Mở cửa lò, dùng quạt để hạ nhiệt độ lò xuống dưới 50°C, đặt chế độ vòng kín (công tắc chọn chế độ), chuyển công tắc phương pháp điều khiển sang ON-OFF, bật công tắc ĐO-ĐẶT

sang ĐẶT, chỉnh biến trở để có nhiệt độ đặt là  $T(^{\circ}\text{C})=100^{\circ}\text{C}$ . Bật công tắc Đo-Đặt sang Đo để thí nghiệm. Đóng điện lò, ghi quá trình tăng nhiệt trong 30 phút.

Kết quả ghi theo bảng 2 và vẽ QTQĐ nhiệt độ lò, % công suất cung cấp trên cùng đồ thị. Sai số nhiệt độ của hệ thống được điều khiển ON-OFF tính bằng sai lệch dương +e1 và âm -e2 (viết ở dạng +e1/-e2) của biên độ đường cong dao động nhiệt độ ngõ ra so với giá trị đặt khi nhiệt độ ngõ ra có dao động ổn định, còn gọi là chế độ tựa xác lập.

Bảng 2: Bảng kết quả điều khiển ON-OFF, nhiệt độ đặt  $100^{\circ}\text{C}$ .

T(min)	0	1	2	3	4	5	6	7	8	9	10	11	12
T $^{\circ}\text{C}$													
%CS													
T(min)	13	14	15	16	17	18	19	20	21	22	23	24	25
T $^{\circ}\text{C}$													
%CS													
T(min)	26	27	28	29	30	31	32	33	34	35	36	37	38
T $^{\circ}\text{C}$													
%CS													



Từ hình vẽ QTQĐ, xác định vùng trễ của khâu role và sai số thực tế. Giải thích lý do tại sao sai số điều khiển lại khá lớn so với vùng trễ.

### 4.3.3. Điều khiển tuyến tính.

#### . Mục đích:

Khảo sát hệ thống điều khiển tuyến tính nhiệt độ với hiệu chỉnh P (bộ hiệu chỉnh là bộ khuếch đại tín hiệu sai số nhiệt độ).

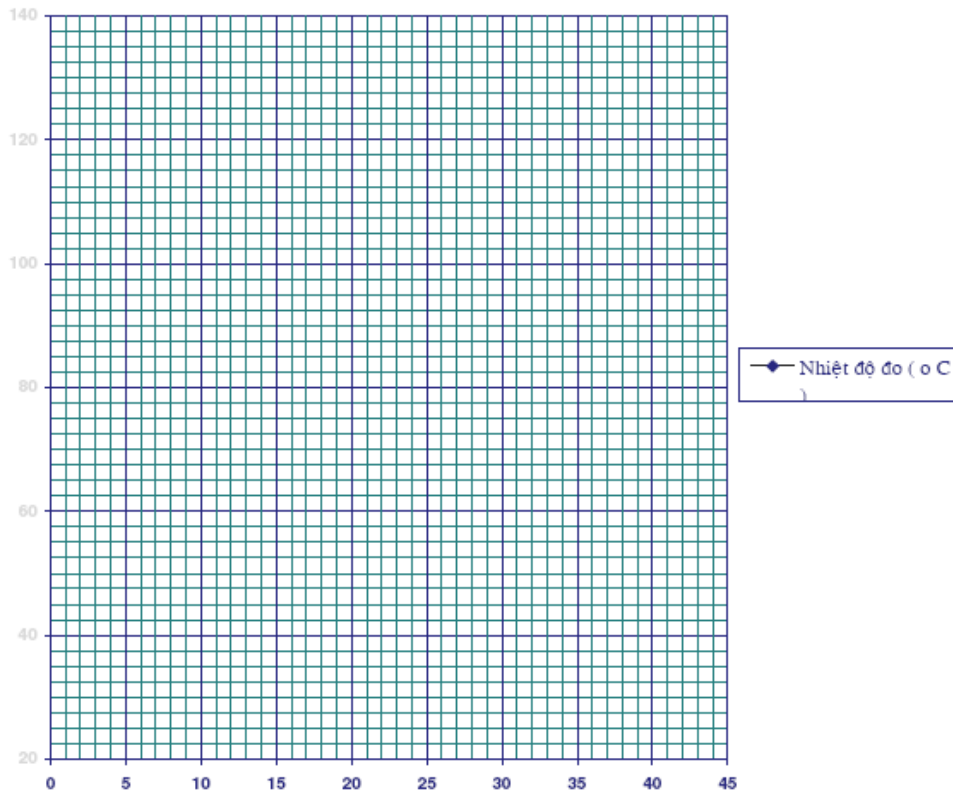
#### . Thí nghiệm:

Mở cửa lò để hạ nhiệt độ xuống dưới  $50^{\circ}\text{C}$ , vẫn giữ nhiệt độ đặt là  $100^{\circ}\text{C}$ , chuyển công tắc phương pháp điều khiển sang tuyến tính (PI). Đo như phần 4.3.2 trong 30 phút. Báo cáo kết quả ghi theo bảng 3 và vẽ QTQĐ nhiệt độ lò, % công suất trên cùng đồ thị.



Bảng 3:

T(min)	0	1	2	3	4	5	6	7	8	9	10	11	12
T°C													
%CS													
T(min)	13	14	15	16	17	18	19	20	21	22	23	24	25
T°C													
%CS													
T(min)	26	27	28	29	30	31	32	33	34	35	36	37	38
T°C													
%CS													



Đánh giá thời gian quá độ, độ vọt lố và sai số xác lập.

Nhận xét về chất lượng điều khiển (sai số xác lập và đặc tính quá độ).

Kết luận : So sánh kết quả điều khiển tuyến tính (hiệu chỉnh P) với điều khiển ON-OFF, giải thích.

#### 4.4. Kiểm tra đánh giá

-Sinh viên nộp báo thí nghiệm cuối mỗi buổi học.

-Giảng viên đánh giá và cho điểm.

Trích dẫn : bài thí nghiệm Hệ thống điều khiển nhiệt độ , của Bộ môn Điều khiển tự động, Khoa Điện - Điện tử, Đại học Bách Khoa Tp. HCM,

## 4.5. ĐIỀU KHIỂN NHIỆT ĐỘ BẰNG CÁC PHƯƠNG PHÁP

### ĐIỀU KHIỂN CỖ ĐIỆN

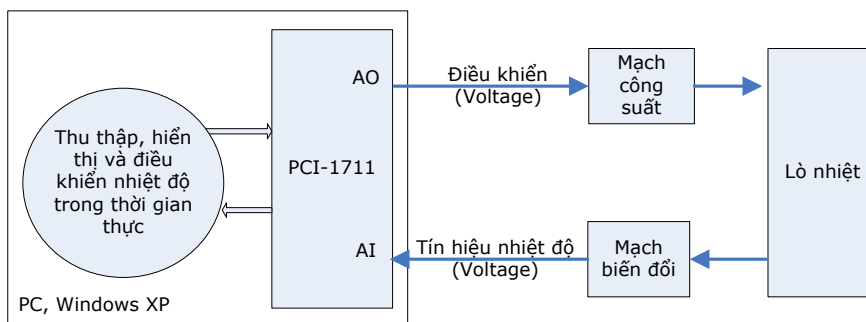
#### Mục tiêu:

Bài thí nghiệm này giúp sinh viên làm quen với đối tượng lò nhiệt, một đối tượng hay gặp trong công nghiệp và phương thức điều khiển chúng bằng các phương pháp điều khiển cở điện đồng thời biết cách viết chương trình thực hiện thuật toán điều khiển và giao diện dùng ngôn ngữ lập trình Borland C++ Builder 5.0/Visual C++ 6.0 trên cơ sở hệ thống máy tính, card PCI 1718H/DU/PCI 1711 và đối tượng lò nhiệt.

#### Nội dung :

- Điều khiển ON-OFF.
- Điều khiển PID.

#### Sơ đồ khối và kết nối của hệ thống điều khiển nhiệt độ dùng PC:



Hình 4.6

#### Thí nghiệm:

1. Viết chương trình điều khiển nhiệt độ dùng ngôn ngữ lập trình Borland C++ Builder 5.0.
2. Viết chương trình điều khiển nhiệt độ dùng ngôn ngữ lập trình Visual C++6.0.
3. Viết chương trình điều khiển nhiệt độ dùng ngôn ngữ lập trình Borland Delphi 5.0.
4. Viết chương trình điều khiển nhiệt độ dùng ngôn ngữ lập trình Visual Basic 6.0.

Trích dẫn : bài thí nghiệm Điều khiển nhiệt độ bằng các phương pháp điều khiển cở điện, của Bộ môn Điều khiển tự động, Khoa Điện - Điện tử, Đại học Bách Khoa Tp. HCM, 2008.

## Bài 5

# HỆ THỐNG ĐIỀU KHIỂN VỊ TRÍ VÀ TỐC ĐỘ ĐỘNG CƠ MỘT CHIỀU

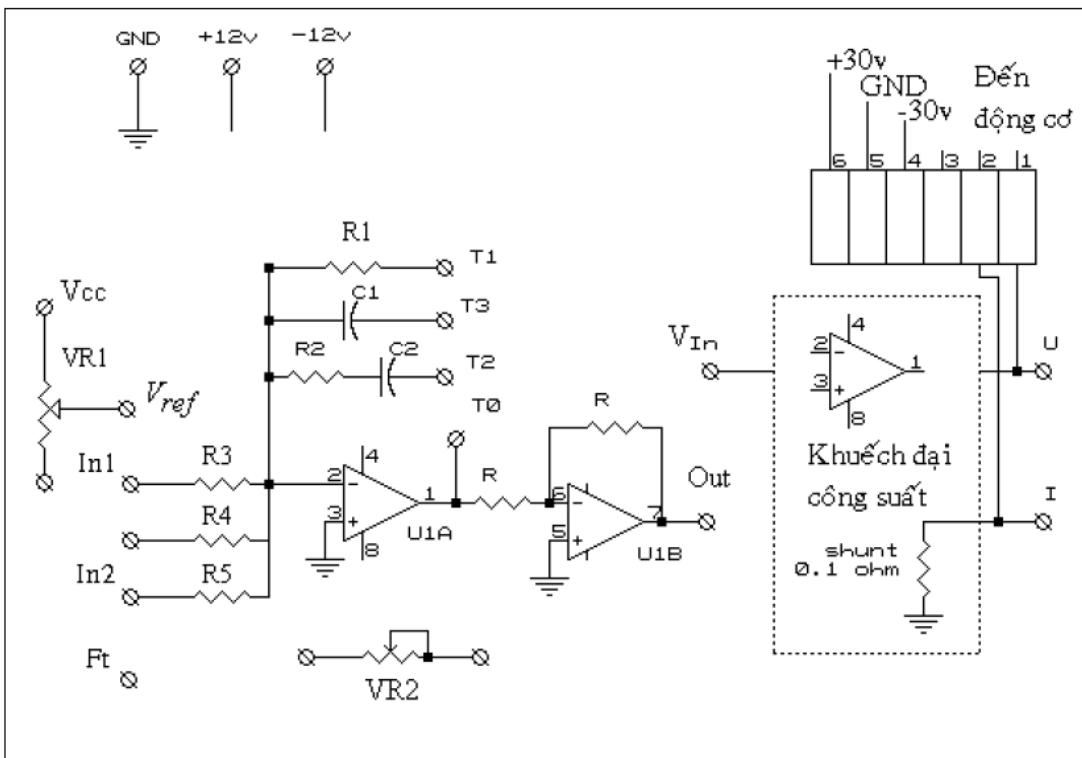
### 5.1. Mục tiêu

Khảo sát trong chế độ xác lập hệ thống truyền động điện động cơ một chiều kích từ bằng nam châm vĩnh cửu (động cơ chấp hành một chiều) khi:

- . Điều khiển tốc độ: đặc tính cơ hệ hở và vòng kín.
- . Điều khiển vị trí: có và không có phản hồi tốc độ với các bộ hiệu chỉnh khác nhau.

### 5.2. Nội dung

#### 5.2.1. Phần điều khiển tốc độ:



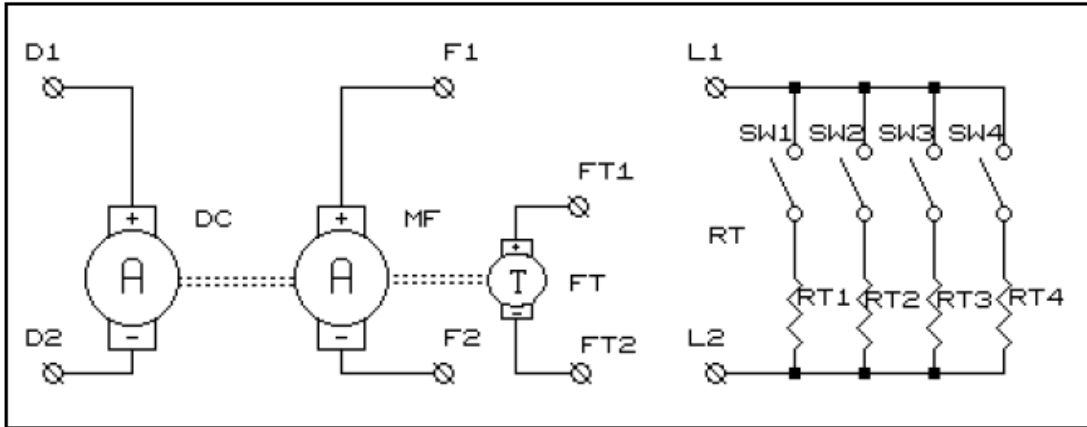
Hình 5.1: Sơ đồ nguyên lý bộ điều khiển động cơ chấp hành

#### 5.2.1.1. Bộ điều khiển tốc độ

Có sơ đồ nguyên lý như hình 1, bao gồm:

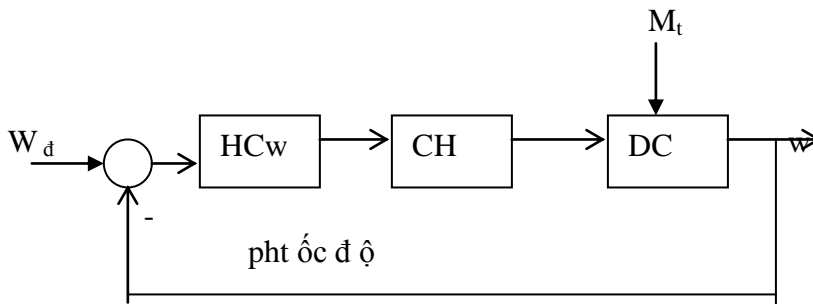
- . Mạch động lực là một bộ khuếch đại một chiều công suất, cấp điện  $\pm 30V$  từ bên ngoài, ngõ ra nối với động cơ một chiều qua trạm nối. U và I là các trạm dùng để đo.
- . VR1 là biến trở đặt tín hiệu tốc độ.
- . U1 làm chức năng điều khiển.
- . Ngoài ra còn có đầu ra  $\pm 12V$  làm tín hiệu đặt.

5.2.1.2. Nhóm động cơ máy phát : (hình 5.2)



Hình 5.2 : Sơ đồ nguyên lý nhóm động cơ máy phát

Hình 5.3 cho ta sơ đồ khối bộ điều khiển tốc độ, tín hiệu phản hồi tốc độ lấy từ máy phát tốc FT :



Hình 5.3 : Sơ đồ khối bộ điều khiển tốc độ

Trong đó :

$w_d$  : tốc độ đặt ,  $w$ : tốc độ động cơ.

HCw : khối hiệu chỉnh tốc độ.

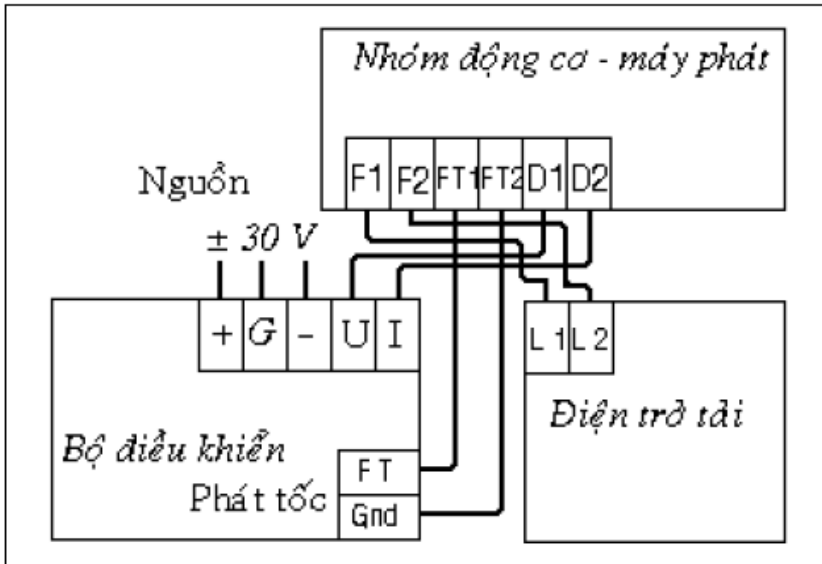
CH : khối chấp hành, là mạch khuếch đại công suất.

DC : động cơ có tải- tương ứng moment cản  $M_t$ .

5.2.1.3. Thiết bị đo để khảo sát HT điều khiển tốc độ :

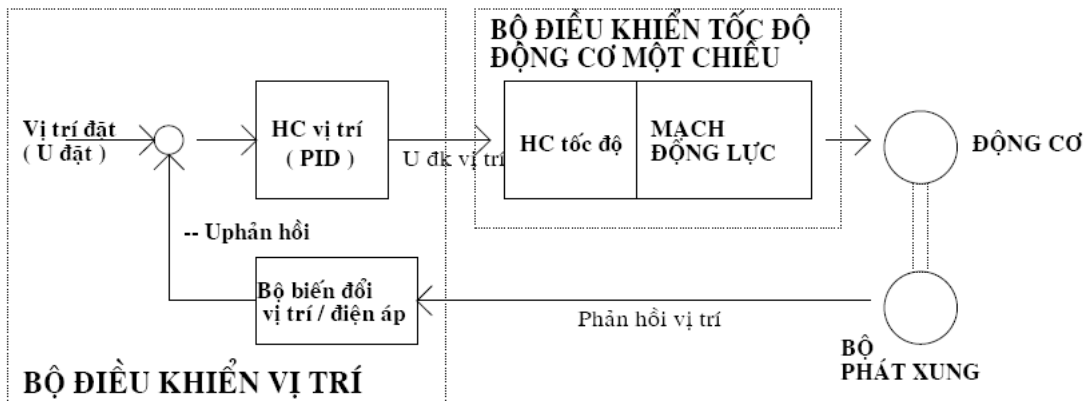
Gồm bộ hiển thị số dòng điện, điện áp động cơ trên mặt bộ thí nghiệm và một volt kế số để đo các điện áp cần thiết khác (tín hiệu đặt và phản hồi tốc độ).

Bộ điều khiển tốc độ, động cơ , và tải đã nối sẵn theo hình 5.4 cho cả bài thí nghiệm :



Hình 5.4 : Sơ đồ nối dây cho thí nghiệm điều khiển tốc độ động cơ :

**5.2.2. Phần điều khiển vị trí :**



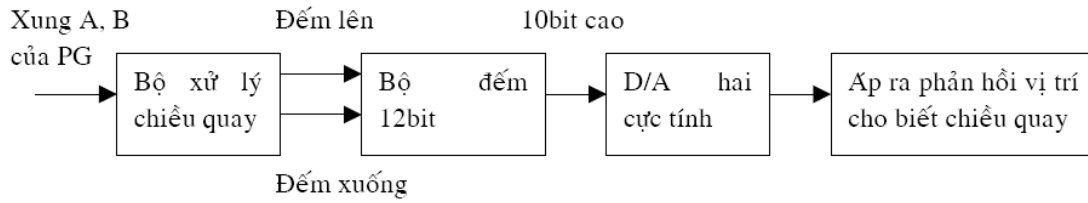
Hình 5.5 : Sơ đồ khối bộ điều khiển vị trí.

Để điều khiển vị trí, ta thêm vào phía trước bộ điều khiển tốc độ của phần 5.2.1 bộ điều khiển vị trí như trên hình 5.5. Bộ điều khiển vị trí xử lý sai lệch vị trí đặt và phản hồi, cho ra tín hiệu điều khiển động cơ sao cho sai lệch này tiến về zero với quá trình quá độ mong muốn.

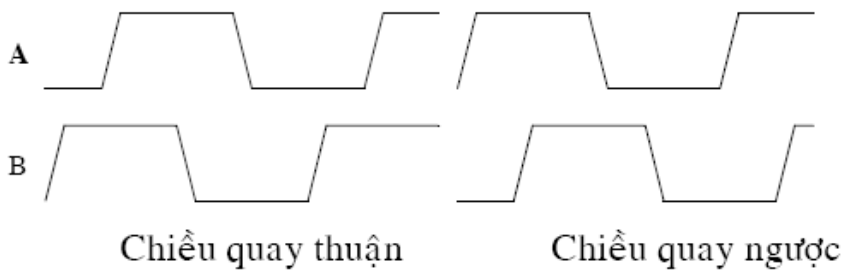
**5.2.2.1. Bộ điều khiển vị trí:**

- . Khối hiệu chỉnh vị trí (PID vị trí).
- . Khối biến đổi vị trí/ điện áp.( hình 5.6)

**Bộ phản hồi vị trí**



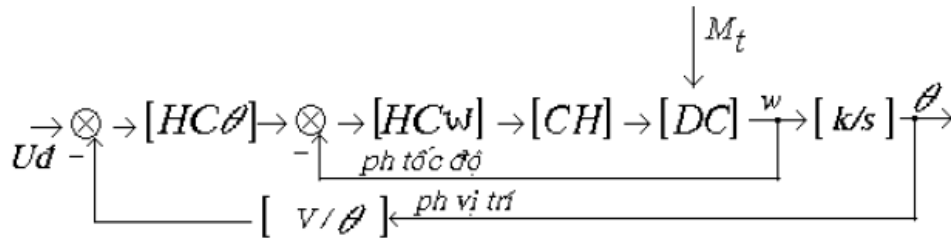
Hình 5.6 : Sơ đồ khối bộ biến đổi xung PG ra áp.



Hình 5.7 : Dạng xung ra của bộ phát xung PG.

5.2.2.2. Thiết bị đo cho thí nghiệm điều khiển vị trí:

- . Trục động cơ mang kim chuy ền động trước bảng chia 200 vạch/vòng cho phép đánh giá chuyển động thực của hệ thống.
- . Trong bài thí nghiệm ta dùng một vôn kế số để đo điện áp đặt và điện áp phản hồi để kiểm tra sai số.



Hình 5.8: Sơ đồ khối hệ thống điều khiển vị trí.

Sơ đồ khối bộ điều khiển vị trí được cho ở hình 5.8, trong đó:

HCθ: bộ hiệu chỉnh vị trí.

HCω: hiệu chỉnh tốc độ.

U đ: tín hiệu đặt.

θ: vị trí, là tích phân của tín hiệu tốc độ ω.

V/θ: khối biến đổi vị trí điện áp.

### 5.2.3. Truyền động điện động cơ DC điều chỉnh áp phản ứng:

#### 5.2.3.1. Hàm truyền động cơ:

Khi bỏ qua phản ứng phân ứng và giả sử các phần tử trong hệ thống là tuyến tính, có các phương trình sau:

$$U = E + Ri + L \frac{di}{dt}$$

$$M_{dc} - M_t = J \frac{d\varpi}{dt}$$

$$M_{dc} = C_e i$$

$$E = C_e \varpi$$

trong đó:

U : điện áp hai đầu phần ứng.

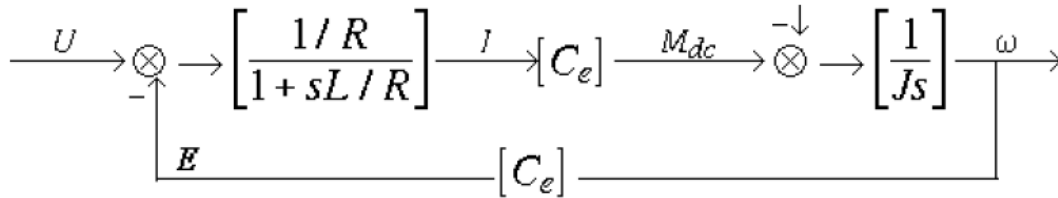
I : dòng điện qua động cơ.

R,L: điện trở, t ự cảm mạch điện phần ứng.

E, M<sub>dc</sub>: sức điện động, moment quay của động cơ.

J: moment quán tính của các phần quay và M<sub>t</sub> là moment cản.

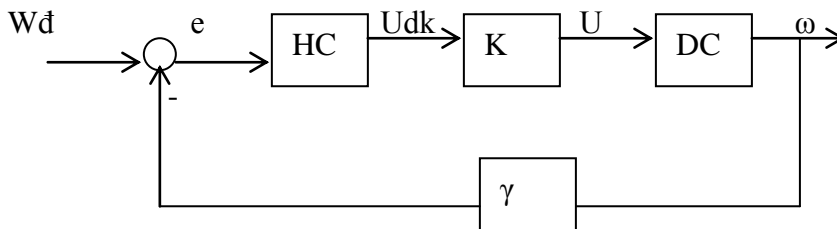
Từ phương trình trên, có thể suy ra hàm truyền động cơ như hình sau :



Hình 5.9 : Sơ đồ khối động cơ một chiều khi từ thông không đổi.

### 5.2.3.2. Đặc tính cơ của động cơ :

Trong chế độ xác lập, moment động cơ bằng moment cản, sụt áp qua tự cảm L bằng zero.



Hình 5.10 : Hàm truyền hệ thống điều khiển tốc độ.

Trong đó :

- K : hàm truyền bộ chấp hành CH, là hằng số.
- $\omega_d$  : là tín hiệu tốc độ đặt, có dạng điện áp.
- $\gamma$  : hệ số phát tốc.

### 5.2.3.3. Hàm truyền của khâu hiệu chỉnh PI trong bài thí nghiệm :

Từ hình vẽ có thể tính tín hiệu ra Out của khâu điều khiển PI theo hai tín hiệu vào In1 và In2 (khi T0 nối với T2).

$$HC(s) = \frac{sR_2C_2 + 1}{sRC_2} \left( In_1 + \frac{R_5}{R_3} In_2 \right)$$

Để có hiệu số giữa tín hiệu tốc độ đặt  $\omega_d$  và phản hồi  $\gamma \cdot \omega$  khi dùng mạch điện trên, ta phải đảo cực tính áp ra của máy phát tốc.

### 5.3. Thí nghiệm : Sinh viên đọc tham khảo [2]

Trong bài thí nghiệm, chỉ cần nối dây giữa các bộ điều khiển theo sơ đồ kèm theo từng phần, các dây nối bên ngoài thiết bị đã được nối sẵn.



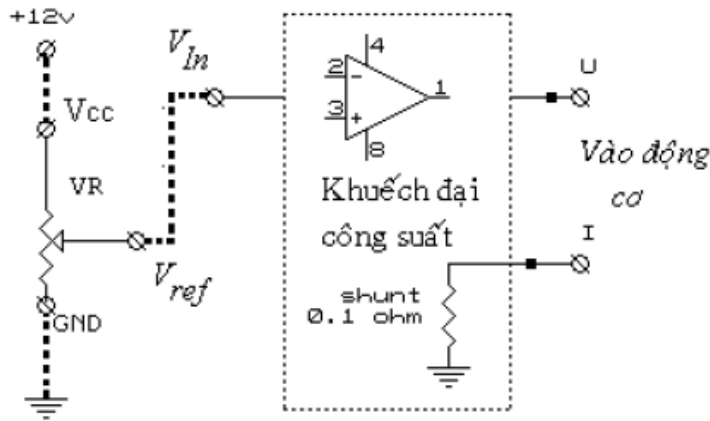
**5.3.1. Thí nghiệm điều khiển tốc độ:**

**5.3.1.a. Thí nghiệm hệ hở:**

. **Mục đích:** Vẽ đặc tính tốc độ  $\omega=f(I)$  khi áp đặt vào động cơ không đổi, suy ra các thông số tĩnh của hệ thống.

. **Thí nghiệm:**

-Nối nguồn 12 V, GND cho biến trở đặt tốc độ như H5.11.



Hình 5.11: Nối dây thí nghiệm hệ hở.

-Nối ngõ ra biến trở đặt tốc độ Vref đến ngõ vào bộ khuếch đại công suất Vin. Chính Vref =1,5 V, đo U,I động cơ, điện áp ra máy phát tốc Uf trên trạm FT khi thay đổi điện trở tải RT của máy phát. (Xem phần mô tả thiết bị thí nghiệm).

-Ghi kết quả vào theo như bảng 1 để viết báo cáo. Trong đó tốc độ động cơ  $\omega$  tính bằng rad/giây, suy ra từ tỉ lệ :  $\omega=Uf/\gamma$ .

$\gamma$ : hệ số phát tốc tính bằng V/(rad/giây), tính từ quan hệ : áp phát tốc 8,2 V tương ứng tốc độ quay 1000 vòng /phút.

- Vẽ đặc tính tốc độ  $\omega=f(I)$  trong đồ thị theo như ở bảng 2, với chú ý  $\omega$  tính bằng rad/sec; suy

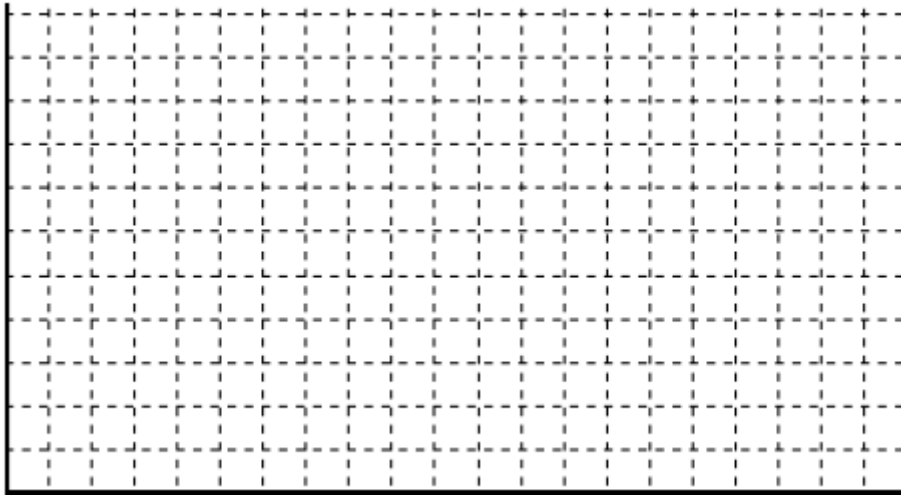
ra hệ số  $C_e = \frac{U}{\omega_0}$ ,  $\omega_0$ : là tốc độ động cơ khi không tải lí tưởng (lúc dòng qua động cơ bằng

không-tương ứng tung độ giao điểm đặc tính tốc độ với trục tung).

I(A)						
U(V)						
Uf(V)						
$\omega$ (rad/sec)						

Bảng 1: Kết quả thí nghiệm hệ hở Vref=1,5 V.

Bảng 2: Đặc tính tốc độ  $\omega=f(I)$  (tốc độ theo dòng điện) của hệ hở, trục hoành là dòng điện.



Điện áp:  $U=?$

Suy ra ở tốc độ không tải lý tưởng (khi  $I=0$ ):  $\omega_0=?$

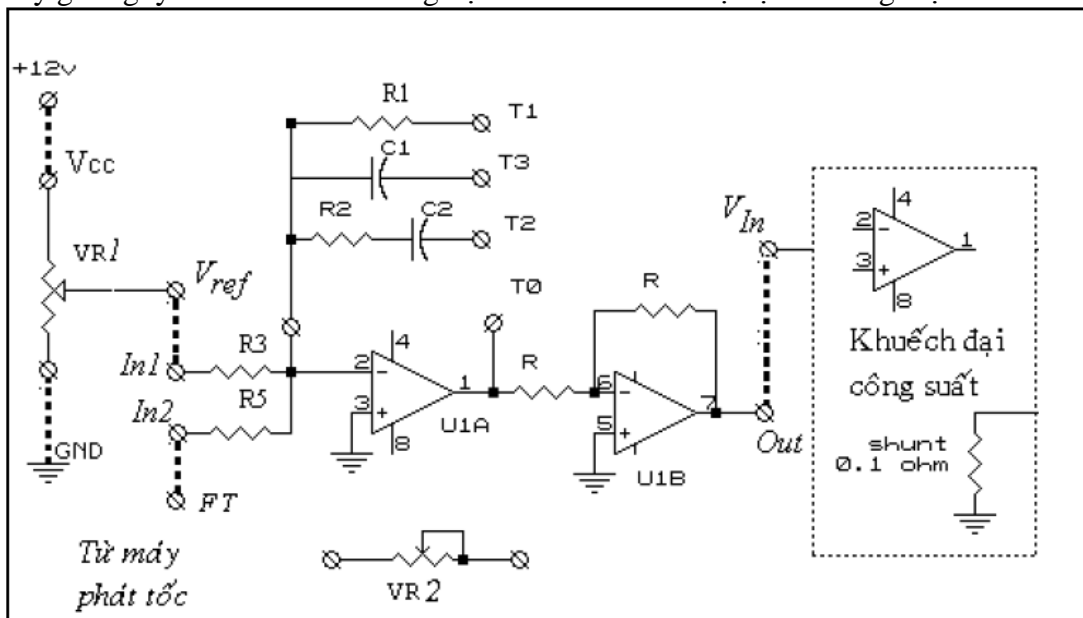
Hằng số điện từ của động cơ:  $C_e=?$

**5.3.1.b. Thí nghiệm hệ vòng kín:**

. **Mục đích** : khảo sát đặc tính tốc độ hệ thống kín với các khâu hiệu chỉnh khác nhau.

. **Thí nghiệm:**

Nối sơ đồ điều khiển tốc độ như hình 5.12, đặt  $V_{ref}=3$  V, tín hiệu đặt tốc độ này giữ nguyên cho đến hết thí nghiệm điều khiển tốc độ hệ kín trong mục 5.3.1.b



Hình 5.12: Nối dây bộ điều khiển hệ kín.

**i) Khảo sát đặc tính của hệ thống khi điều khiển tỉ lệ (P):** đặt  $V_{ref}=3$  volt.

Nối T1 và T0 để U1A là mạch khuếch đại DC. Thay đổi tải  $R_T$  của máy phát, ghi giá trị điện áp, dòng điện của động cơ, điện áp máy phát tốc (đo tại trạm FT) theo như bảng 3 để viết báo cáo.

I(A)					
U(V)					
U <sub>f</sub> (V)					
$\omega$ (rad/sec)					

Bảng 3 : Đặc tính tốc độ khi điều khiển tỉ lệ (P).

Nhận xét về khả năng cải thiện độ sụt dọc theo tải khi có phản hồi tốc độ.

**ii) Khảo sát đặc tính của hệ thống khi thay đổi Kp:** đặt  $V_{ref} = 3V$ .

Nối hai đầu biến trở VR2 vào T1 và T0 để có thể thay đổi hệ số khuếch đại bộ hiệu chỉnh (hệ số Kp). Chỉnh biến trở theo chiều kim đồng hồ đến cuối để có hệ số khuếch đại lớn nhất. Thay đổi tải RT của máy phát để lấy số liệu, kết quả ghi theo bảng 4 để viết báo cáo.

I(A)						
U(V)						
Uf(V)						
$\omega$ (rad/sec)						

Bảng 4 : Đặc tính tốc độ khi điều khiển tỉ lệ (P) với Kp lớn.

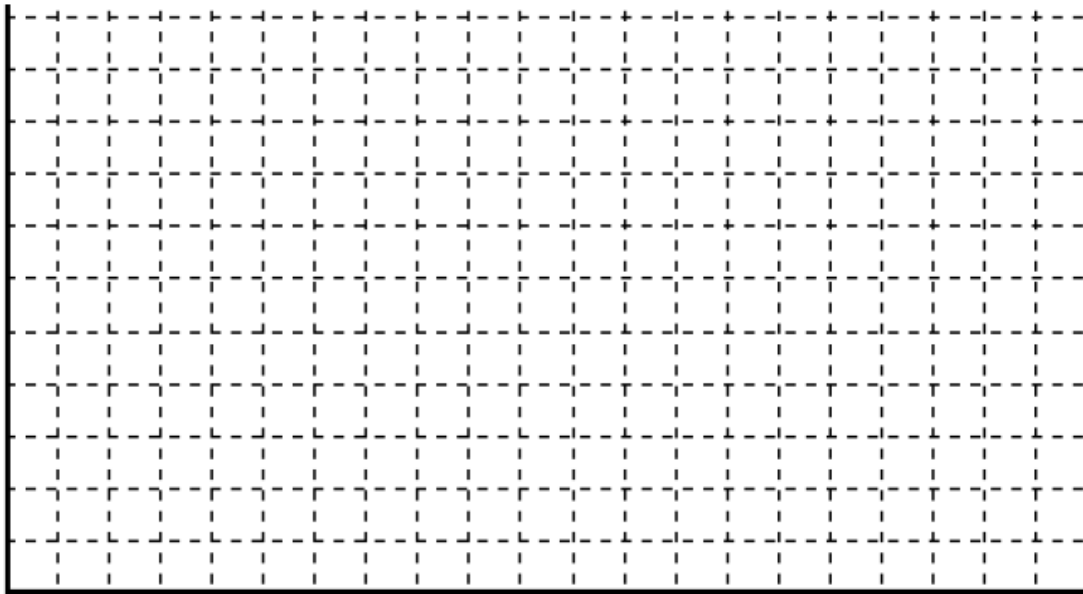
**iii) Khảo sát đặc tính của hệ thống khi điều khiển tích phân tỉ lệ (PI):** đặt  $V_{ref} = 3V$ .

Nối T2 vào T0 để U1A là một khâu PI. Thay đổi tải RT của máy phát để lấy số liệu, kết quả ghi theo bảng 5 để viết báo cáo.

I(A)						
U(V)						
Uf(V)						
$\omega$ (rad/sec)						

Bảng 5 : Đặc tính tốc độ khi điều khiển tích phân tỉ lệ (PI).

Vẽ đặc tính tốc độ các khảo sát trong phần i), ii), iii) trên cùng một hình vẽ như ở bảng 6. Nên sử dụng cùng tỉ lệ để dễ so sánh các trường hợp. Cho nhận xét và giải thích kết quả về sai số xác lập của hệ thống điều khiển tốc độ với các bộ hiệu chỉnh khác nhau khi tải thay đổi.



Bảng 6: Đồ thị đặc tính tốc độ (tốc độ theo dòng điện) khi hiệu chỉnh P, PI.

(trục hoành là dòng điện, trục tung là các đặc tính tốc độ).

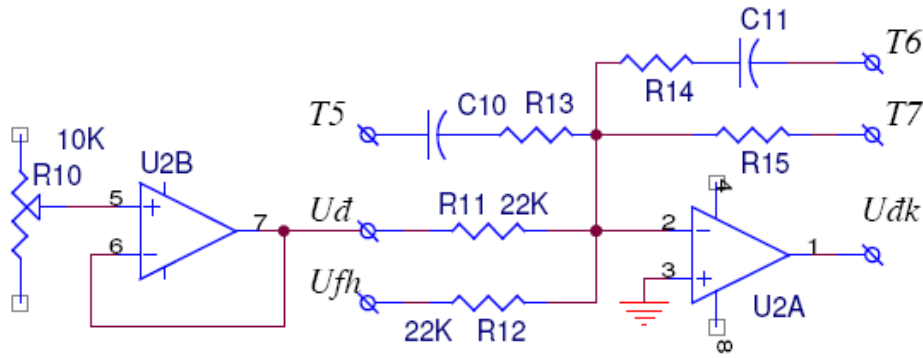
**5.3.2. Thí nghiệm điều khiển vị trí:**

Nối dây sử dụng chung cho tất cả các phần thí nghiệm ở mục 5.3.2.

. Cấp điện 220 VAC cho bộ điều khiển vị trí.

. Hệ thống được khảo sát ở chế độ không tải (các công tắc điện trở tải của máy phát ở vị trí hở mạch).

. Công tắc RESET/RUN ở vị trí RESET, để reset bộ phản hồi vị trí và không cho hệ thống hoạt động. Bộ phát xung PG đã được nối sẵn, tại chân  $U_{TH}$  có điện áp phản hồi vị trí (phản ánh sai lệch vị trí hiện tại so với vị trí ban đầu-khi RESET hệ thống-xem lại 5.2.2.1 giới thiệu đặc tính cảm biến vị trí).



Hình 5.13: Sơ đồ nguyên lý mạch điều khiển vị trí.

5.3.2.1. Khảo sát hệ thống điều khiển vị trí không có phản hồi tốc độ:

. Mục đích: Khảo sát đặc tính tĩnh và động của hệ thống điều khiển vị trí với các bộ hiệu chỉnh P và PI với ngõ vào là hàm nấc.

. Thí nghiệm:

Nội dung chung cho mục này: sử dụng cả hai bộ điều khiển tốc độ và vị trí.

i) Khảo sát khâu hiệu chỉnh P (tỉ lệ) cho bộ điều khiển vị trí:

ii) Khảo sát khâu hiệu chỉnh PI (tích phân tỉ lệ) cho bộ điều khiển vị trí:

**5.3.2.2. Khảo sát hệ thống điều khiển vị trí có vòng phản hồi tốc độ:**

. Mục đích: khảo sát ảnh hưởng của vòng tốc độ đến chất lượng của hệ điều khiển vị trí.

. Thí nghiệm:

Thực hiện nội dung cho 2 vòng hồi tiếp với vòng vị trí điều khiển PI và vòng tốc độ điều khiển P như sau:

Nhận xét về độ chính xác của hệ thống điều khiển vị trí và vọt lố của HT với đầu vào hàm nấc khi có và không có vòng phản hồi tốc độ. Từ đó nêu tác dụng của vòng phản hồi tốc độ.

Chú ý: Do tầm phản hồi vị trí bị hạn chế ở xấp xỉ 2V (xem lý thuyết phần 5.2), vọt lố lớn có thể làm sai tín hiệu phản hồi dẫn đến hệ thống không hoạt động được ở giá trị đặt lớn, phân biệt với sự mất ổn định của hệ thống.

### 5.3. Nhận xét đánh giá

Trích dẫn: bài thí nghiệm Hệ thống điều khiển tốc độ và vị trí động cơ một chiều, của Bộ môn Điều khiển tự động, Khoa Điện - Điện tử, Đại học Bách Khoa Tp. HCM.

## Bài 6

# HỆ THỐNG ĐIỀU KHIỂN SỐ ĐỘNG CƠ MỘT CHIỀU

### 6.1. Mục tiêu

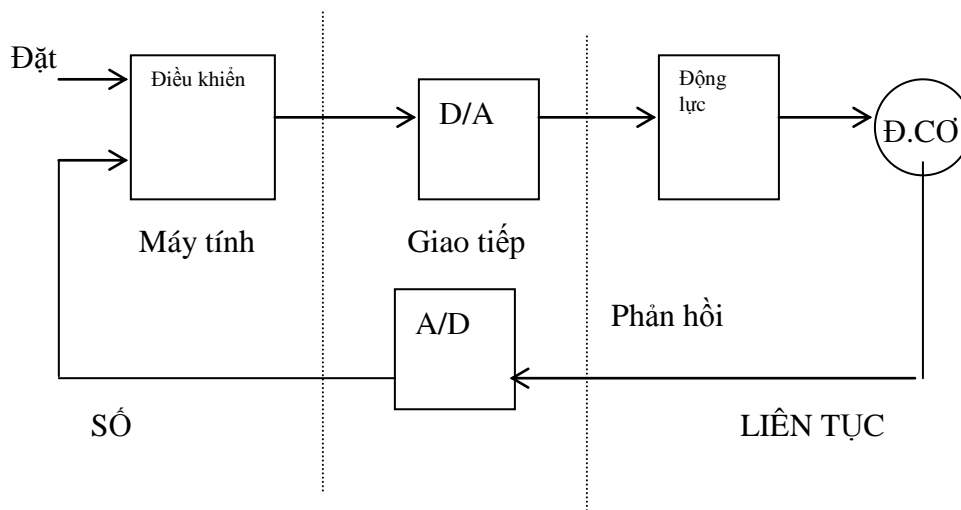
-Làm quen với hệ thống sử dụng máy tính số(PC) để điều khiển vòng kín một đối tượng cụ thể trong bài là động cơ một chiều.

-Khảo sát ảnh hưởng của các thành phần điều khiển PID trên đặc tính tĩnh và động của hệ thống điều khiển tốc độ động cơ một chiều bằng cách quan sát ảnh hưởng của từng thông số trên quá trình quá độ của hệ thống với đầu vào hàm nấc.

### 6.2. Nội dung

Bộ thí nghiệm bao gồm :

- Bộ điều khiển bao gồm mạch giao tiếp , máy tính và mạch động lực.
- Động cơ một chiều có bộ phát tốc xung.
- Bộ cấp điện một chiều  $\pm 24V$ .
- Chương trình điều khiển.



Hình 6.1 : Sơ đồ khối bài thí nghiệm

#### 6.2.1. Bộ điều khiển :

Bộ điều khiển động cơ chứa trong hộp riêng, có sơ đồ khối như hình 6.2, bao gồm :

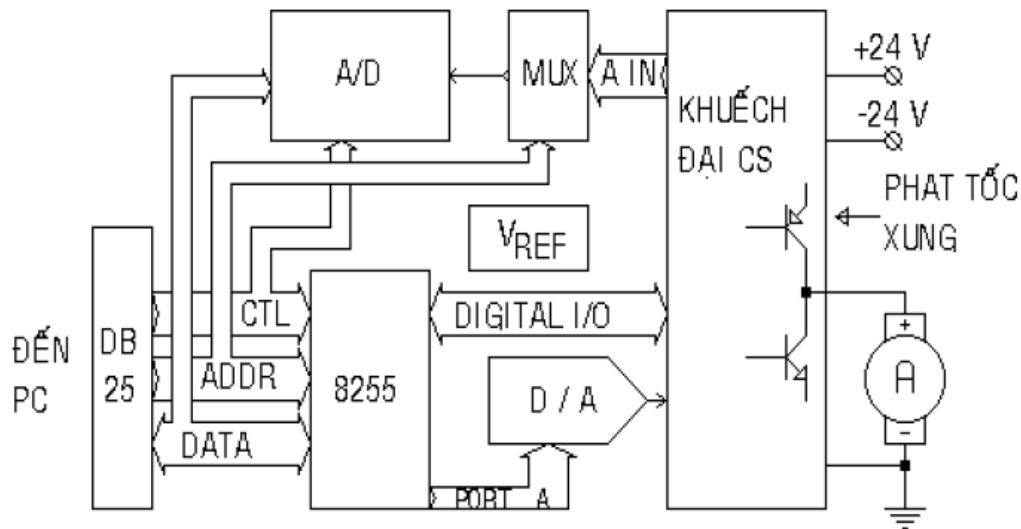
-Card giao tiếp tương tự và số gồm có :

. Bộ chuyển đổi tương tự-số ADC0809 8bit cho phép máy tính thu thập và tính toán số liệu điều khiển động cơ. Trong thí nghiệm này, tốc độ, dòng điện, và điện áp động cơ được đọc về máy tính số. Máy tính cũng ghi lại và trình bày đồ thị quá trình quá độ để ta phân tích đặc tính hệ thống.

Chuyển đổi số-tương tự (DAC) 8 bit AD7520 nối port A của PIA 8255, biến đổi tín hiệu điều khiển từ số ra điện áp, giữ nguyên trong chu kỳ lấy mẫu làm thành bộ giữ bậc 0 (ZOH).

-Máy tính AT có card giao tiếp PROTOTYPE ADAPTER(TK-1), địa chỉ \$300-\$31F(có thể dùng card PCI 1711/PCI 1718HDU).

-Bộ khuếch đại công suất có mạch bảo vệ dòng điện, sử dụng nguồn  $\pm 24V$  khuếch đại tín hiệu điều khiển (từ ngõ ra DAC) để cung cấp cho động cơ DC.



Hình 6.2 : Sơ đồ khối bộ điều khiển động cơ.

6.2.2. Động cơ một chiều.

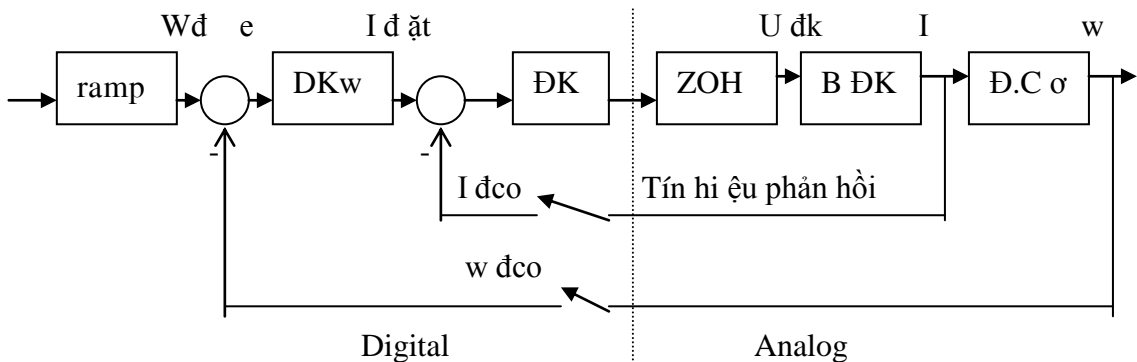
6.2.3. Chương trình điều khiển.

**6.3. Thí nghiệm : sinh viên đọc tham khảo [2]**

6.3.1. Thí nghiệm khảo sát hệ hở :

6.3.2. Thí nghiệm hệ vòng kín 1(Không có phản hồi dòng điện)

6.4. Sơ đồ khối hệ thống điều khiển tốc độ động cơ :



Hình 6.3 Sơ đồ khối HT điều khiển động cơ

6.5. Nhận xét đánh giá :

Trích dẫn : bài thí nghiệm Điều khiển số động cơ DC, của Bộ môn Điều khiển tự động, Khoa Điện - Điện tử, Đại học Bách Khoa Tp. HCM.

## Bài 7

# HỆ THỐNG ĐIỀU KHIỂN MỰC NƯỚC

### 7.1.Mục tiêu

Sinh viên xây dựng mô hình toán học bồn đơn và dùng Simulink mô phỏng hệ bồn đơn; và khảo sát hệ thống điều khiển mực nước cụ thể.

### 7.2.Nội dung

Dùng máy tính có sử dụng card giao tiếp PCL818L/PC I 1711 để điều khiển bồn nước. Card dùng đọc giá trị từ cảm biến và xuất ra điện áp điều khiển bơm nước vào bồn. Có sử dụng XPC-target trong Matlab và có viết chương trình kết nối bằng Visual C hay Borland C++ Builder.

Sơ đồ khối hệ thống điều khiển bồn nước:

#### xPC-Target:

xPC Target là giải pháp của Matlab nhằm đưa các kết quả mô hình hệ thống dùng Simulink vào hệ thống chạy thời gian thực. xPC target cần có một máy tính chủ và máy tính đích ghép nối với nhau qua cổng RS-232 hay mạng Ethernet, chương trình điều khiển dạng mã thực thi, tạo ra bởi Real Time Workshops và Visual C++, xPC target được chuyển từ máy chủ đến máy đích, máy tính đích có các card giao tiếp với đối tượng sẽ điều khiển đối tượng theo thời gian thực, các đáp ứng được gửi lên máy chủ và các tình chỉnh gửi xuống máy đích. Máy tính chủ cài hệ điều hành Windows 95, 98, 2000, Xp hay NT4. Máy tính đích không cần cấu hình mạnh (CPU 386, RAM 8MB), không cần hệ điều hành nhưng BIOS phải tương thích máy PC, mainboard có sẵn các slot gắn card giao tiếp và có cổng COM(dùng chipset tương thích UART) hay card mạng, chỉ cần ổ đĩa mềm còn ổ cứng và màn hình không cần thiết. Sự giao tiếp giữa hai máy tính thực hiện qua cổng COM hay mạng Ethernet.

xPC Target hỗ trợ giao tiếp qua cổng COM và một số card giao tiếp của các hãng Advantech, ADDI-DATA, National Instruments, Real time devices,... Sinh viên đọc thêm trong tài liệu[4].

### 7.3.Thí nghiệm

#### 7.3.1.Mô phỏng bồn chứa I

$u(t)$  : tín hiệu vào ,  $h(t)$ : tín hiệu ra là chiều cao cột nước, tối đa 40cm.

$$dh(t)/dt = (1/A).(K.u(t) - C.a.\sqrt{2gh(t)}) \quad (1)$$

Cho:

A: tiết diện ngang của bồn chứa=100cm<sup>2</sup>.

a: tiết diện van=0,5 cm<sup>2</sup>.

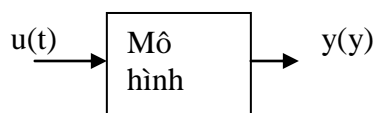
C: hệ số xả=0,6.

g: gia tốc trọng trường=981 cm/s<sup>2</sup>

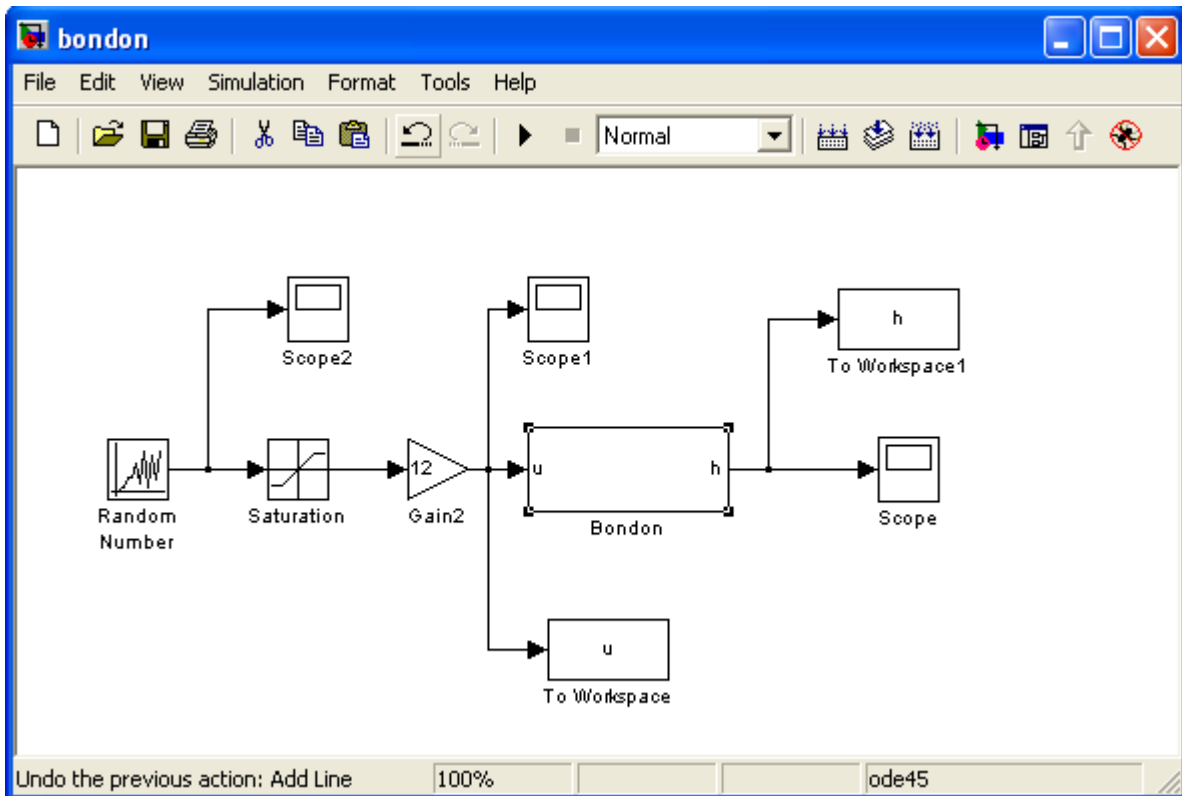
$u(t)$ : [0->1]; tối đa 12V, ngẫu nhiên.

K: hệ số đặc trưng cho công suất lớn=18 lit/phút=300 cm<sup>3</sup>/s.

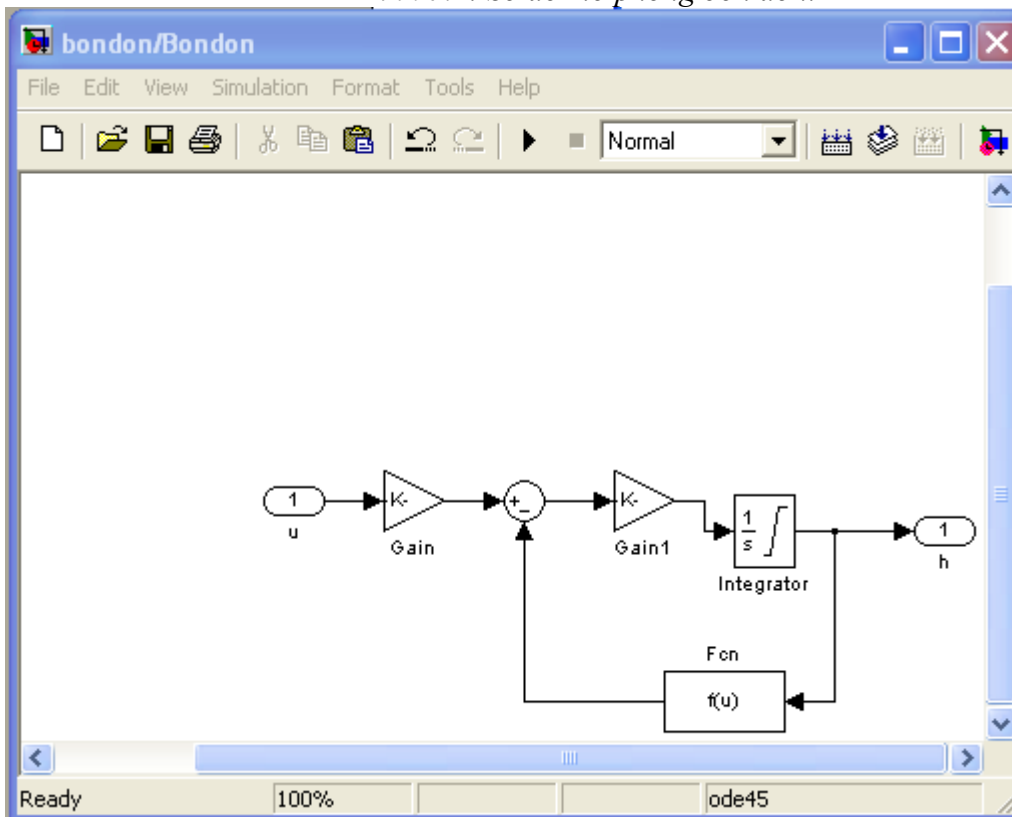
Xây dựng mô hình mô phỏng :



Sơ đồ mô phỏng lưu ở tập tin bondon.mdl  
 Mô phỏng hệ thống với  $u(t)$  ngẫu nhiên



Hình 7.1: Sơ đồ mô phỏng bồn đơn.



Hình 7.2 : Bên trong subsystem “Bondon”

Gain là  $K=300$ , Gain1 là  $1/A=1/100$ , Khối Fcn là  $C*a*\sqrt{2*981*u[1]}=0.6*0.5*\sqrt{2*981*u[1]}$ . Khối tích phân có giới hạn cao nhất là 40.

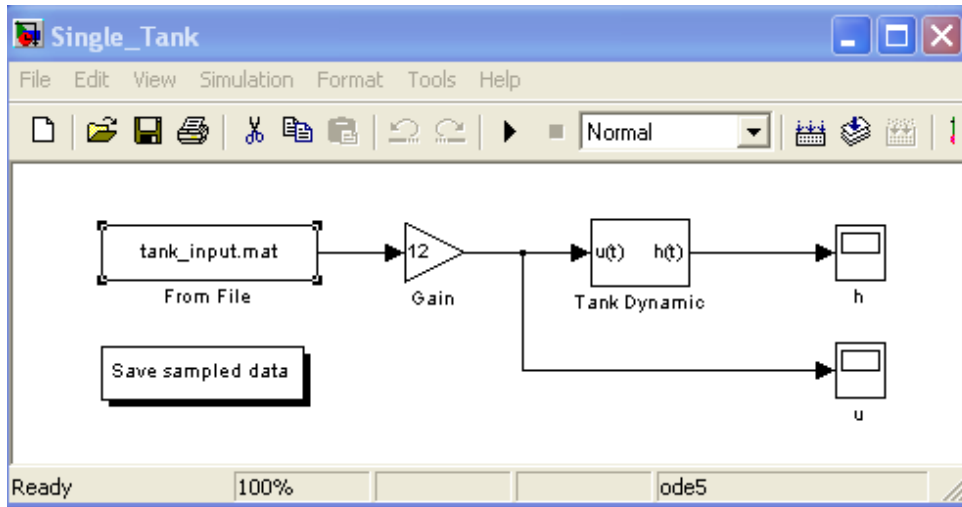


Kết quả mô phỏng

Ngõ vào  $u(t)$

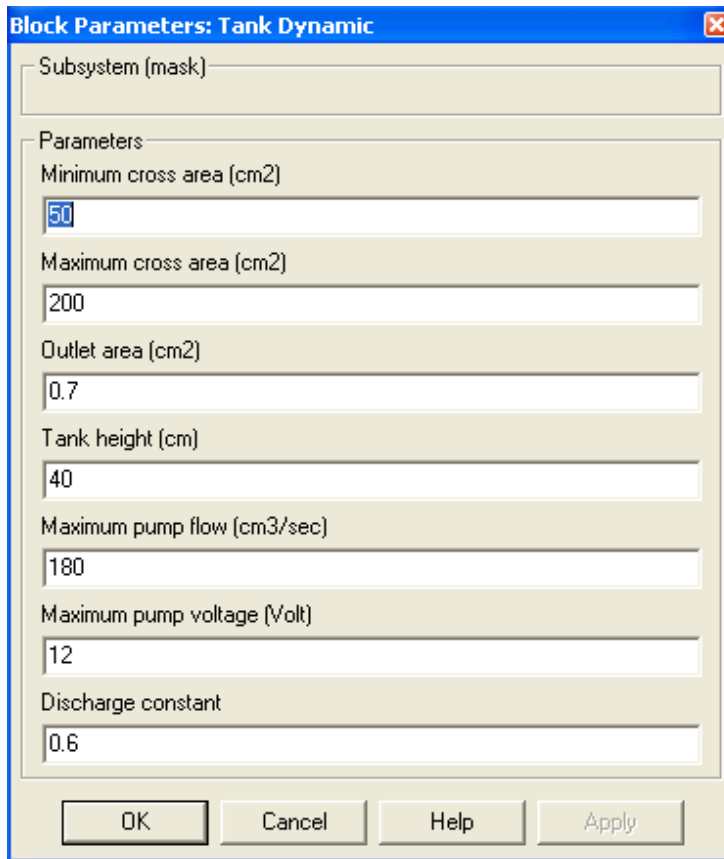
Ngõ ra  $h(t)$

### 7.3.2. Sơ đồ mô phỏng II:



Hình 7.3.

Thông số bồn nước :



Hình 7.4

Kết quả mô phỏng:

$U(t)$ :

$H(t)$ :

7.4. Nhận xét đánh giá.

7.5. Khảo sát mô hình điều khiển bồn nước của hãng Festo.

# PHỤ LỤC

**Bảng biến đổi Laplace**

<i>STT</i>	<i>Ảnh laplace F(s)</i>	<i>Hàm thời gian f(t)</i>
<b>1</b>	<b>1</b>	<b>Hàm Dirac <math>\delta(t)</math></b>
<b>2</b>	$\frac{1}{s}$	<b>Hàm nấc đơn <math>u_s(t)=1(t)</math></b>
<b>3</b>	$\frac{1}{s^2}$	<b>Hàm dốc (hàm RAMP)= <math>t.1(t)</math></b>
<b>4</b>	$\frac{n!}{s^{n+1}}$	<b><math>t^n</math> (<math>n=</math> số nguyên dương)</b>
<b>5</b>	$\frac{1}{s+\alpha}$	<b><math>e^{-\alpha t}</math></b>
<b>6</b>	$\frac{1}{(s+\alpha)^2}$	<b><math>t.e^{-\alpha t}</math></b>
<b>7</b>	$\frac{n!}{(s+\alpha)^{n+1}}$	<b><math>t^n.e^{-\alpha t}</math>, (<math>n=</math> số nguyên dương)</b>
<b>8</b>	$\frac{1}{(s+\alpha)(s+\beta)}$	<b><math>\frac{1}{(\beta-\alpha)}(e^{-\alpha t} - e^{-\beta t})</math> (<math>\alpha \neq \beta</math>)</b>
<b>9</b>	$\frac{1}{s(s+\alpha)}$	<b><math>\frac{1}{\alpha}(1 - e^{-\alpha t})</math></b>
<b>10</b>	$\frac{1}{s(s+\alpha)^2}$	<b><math>\frac{1}{\alpha^2}(1 - e^{-\alpha t} - \alpha t e^{-\alpha t})</math></b>
<b>11</b>	$\frac{1}{s^2(s+\alpha)}$	<b><math>\frac{1}{\alpha^2}(\alpha t - 1 + e^{-\alpha t})</math></b>
<b>12</b>	$\frac{1}{s^2(s+\alpha)^2}$	<b><math>\frac{1}{\alpha^2}[t - \frac{1}{\alpha} + (t + \frac{2}{\alpha})e^{-\alpha t}]</math></b>

13	$\frac{s}{(s + \alpha)^2}$	$(1 - \alpha t)e^{-\alpha t}$
14	$\frac{\omega_n^2}{s^2 + \omega_n^2}$	$\sin \omega_n t$
15	$\frac{s}{s^2 + \omega_n^2}$	$\cos \omega_n t$
16	$\frac{\omega_n^2}{s(s^2 + \omega_n^2)}$	$1 - \cos \omega_n t$
17	$\frac{\omega_n^2(s + \alpha)}{s^2 + \omega_n^2}$	$\omega \sqrt{\alpha^2 + \omega_n^2} \sin(\omega_n t + \theta)$ với $\theta = \tan^{-1}(\omega_n / \alpha)$
18	$\frac{\omega_n}{(s + \alpha)(s^2 + \omega_n^2)}$	$\frac{\omega_n}{\alpha^2 + \omega_n^2} e^{-\alpha t} + \frac{1}{\alpha^2 + \omega_n^2} \sin(\omega_n t + \theta)$ , với $\theta = \tan^{-1}(\omega_n / \alpha)$
19	$\frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$	$\frac{\omega_n}{\sqrt{1 - \xi^2}} e^{-\xi\omega_n t} \sin \omega_n \sqrt{1 - \xi^2} t$ ( $\xi < 1$ )
20	$\frac{\omega_n^2}{s(s^2 + 2\xi\omega_n s + \omega_n^2)}$	$1 - \frac{\omega_n}{\sqrt{1 - \xi^2}} e^{-\xi\omega_n t} \sin(\omega_n \sqrt{1 - \xi^2} t + \theta)$ , với $\theta = \cos^{-1} \xi$ ( $\xi < 1$ )
21	$\frac{s\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2}$	$\frac{-\omega_n^2}{\sqrt{1 - \xi^2}} e^{-\xi\omega_n t} \sin(\omega_n \sqrt{1 - \xi^2} t - \theta)$ với $\theta = \cos^{-1} \xi$ ( $\xi < 1$ )
22	$\frac{\omega_n^2(s + \alpha)}{s^2 + 2\xi\omega_n s + \omega_n^2}$	$\omega_n \sqrt{\frac{\alpha^2 - 2\alpha\xi\omega_n + \omega_n^2}{1 - \xi^2}} e^{-\xi\omega_n t} \sin(\omega_n \sqrt{1 - \xi^2} t + \theta)$ với $\theta = \tan^{-1} \frac{\omega_n \sqrt{1 - \xi^2}}{\alpha - \xi\omega_n}$ ( $\xi < 1$ )
23	$\frac{\omega_n^2}{s^2(s^2 + 2\xi\omega_n s + \omega_n^2)}$	$t - \frac{2\xi}{\omega_n} + \frac{1}{\omega_n^2 \sqrt{1 - \xi^2}} e^{-\xi\omega_n t} \sin(\omega_n \sqrt{1 - \xi^2} t + \theta)$ , với $\theta = \cos^{-1}(2\xi^2 - 1)$ ( $\xi < 1$ )

**Các hàm MATLAB dùng trong Control systems toolbox (một số):**

Tên hàm	Mô tả chức năng
abs	Tính giá trị tuyệt đối
Acos	Tính arcos
Ans	biên được tạo ra trong biểu thức
Asin	Tính giá trị arcsin
Atan	Tính giá trị arctan (2 quadrant)
Atan2	Tính giá trị arctan(4 quadrant)
Axis	chỉ ra tỉ lệ trục bằng tay trên đồ thị.
Bode	tạo ra đồ thị đáp ứng tần số Bode.
C2d	chuyển đổi thể hiện hệ thống biến trạng thái liên tục thời gian sang thể hiện hệ thống rời rạc thời gian.
Clear	Xoá workspace
Clf	Xoá xử số đồ thị
Conj	Tính liên hợp phức (complex conjugate)
Conv	Nhân hai đa thức (convolution)
Cos	Tính giá trị cosin
Ctrb	Tính ma trận điều khiển được
diary	Lưu phiên làm việc vào tập tin đĩa
D2c	chuyển đổi hệ biến trạng thái rời rạc thời gian sang hệ liên tục thời gian
Eig	Tính giá trị riêng và vector riêng.
End	kết thúc cấu trúc điều khiển
Exp	Tính lũy thừa với cơ số e.
Expm	Tính lũy thừa ma trận với cơ số e.
eye	tạo ra ma trận đơn vị.
Feedback	Tính kết nối vòng kín của hai hệ .
For	tạo ra vòng lặp
Format	thiết lập định dạng hiển thị ngõ ra.
Grid on	Thêm lưới vào đồ thị hiện hành.
Help	In danh sách của chủ đề HELP(trợ giúp).
Hold on	giữ đồ thị hiện hành trên màn hình
i	$\sqrt{-1}$
imag	Tính phần ảo của số phức.
impulse	Tính đáp ứng xung đơn vị của hệ thống.
Inf	thể hiện không xác định
j	$\sqrt{-1}$
legend	đặt chú thích trên đồ thị hiện hành
lsim	Tính đáp ứng thời gian của hệ.
minreal	Rút gọn hàm truyền
Nichols	Tính toán đồ thị đáp ứng tần số Nichols
Nyquist	Tính toán đáp ứng tần số Nyquist
Plot	vẽ đồ thị tuyến tính.
Pole	Tính cực của hệ
Rlocus	Tính quỹ đạo nghiệm số.
Roots	Tính nghiệm của đa thức.
Ss	tạo ra đối tượng mô hình không gian trạng thái.
Step	Tính đáp ứng nấc đơn vị của hệ thống.
Tf	tạo ra đối tượng mô hình hàm truyền.
Xlabel	Thêm nhãn vào trục x của đồ thị
Ylabel	Thêm nhãn vào trục y của đồ thị.

## TÀI LIỆU THAM KHẢO

- [1]. Nguyễn Thị Phương Hà(chủ biên), Huỳnh Thái Hoàng, Lý thuyết điều khiển tự động, NXB ĐHQG Tp. HCM, 2003.
- [2]. ĐHBK Tp. HCM, Bộ môn Điều khiển tự động, Thí nghiệm Điều khiển tự động 1, 2008.
- [3]. B.C. Kuo, Automatic Control Systems, 8<sup>th</sup> ed. , Wiley, 2003.
- [4]. Nguyễn Đức Thành, Matlab và ứng dụng trong điều khiển, NXB ĐHQG Tp. HCM, 2005.
- [5]. Phần mềm Matlab 6.5 và 7.0 và Simulink.
- [6]. Toolbox ACSYS2002 chạy trên Matlab.
- [7]. William J. Palm III, Introduction to Matlab 6 for Engineers, McGraw-Hill, 2001.