



# 控工實驗 LAB3 結報

MATLAB 教學\_part1

Group 2

林珮玉 E24084096

林玠志 E24083074

林哲緯 E24086129

薛博文 F14071148

蔡孟宗 F44071055

# Control Engineering Laboratory – Lab3 Report

Group 2 - 林珮玉 林哲緯 林玠志 薛博文 蔡孟宗

## I. Objectives

1. MATLAB 的基本操作
2. Simulink Tutorial
3. Sisotool Tutorial

## II. Exploration

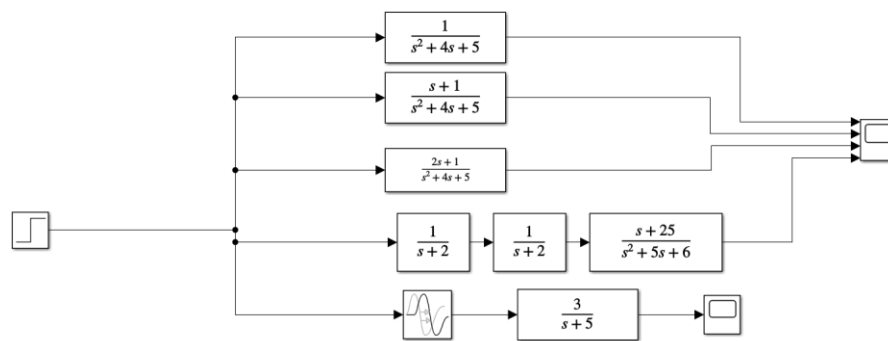
### ● 實驗練習 1-1

(1)  $T(s) = \frac{as + 1}{s^2 + 4s + 5}$   $a = 0, 1, 2$ , 試以 simulink 探討步階響應變化(畫在同一個圖上)

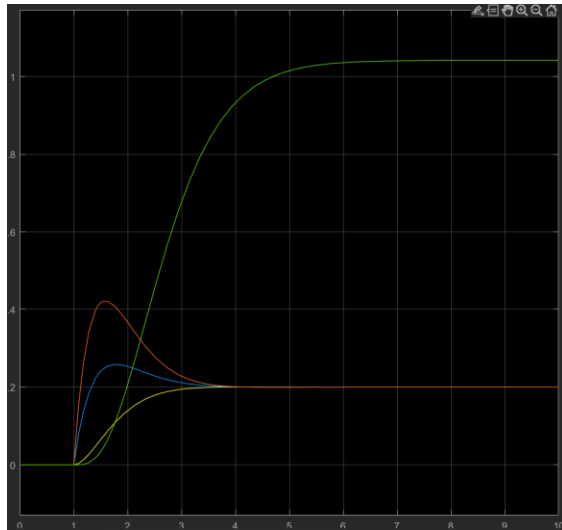
(2)  $T(s) = \frac{s + 25}{(s + 2)^2(s^2 + 5s + 6)}$  求其步階響應(畫在同一圖上)

(3)  $T(s) = \frac{3e^{-0.5s}}{(s + 5)}$  輸入系統左方延遲的轉移函數，並以simulink畫出步階響應圖

### Block diagram



### Simulation result

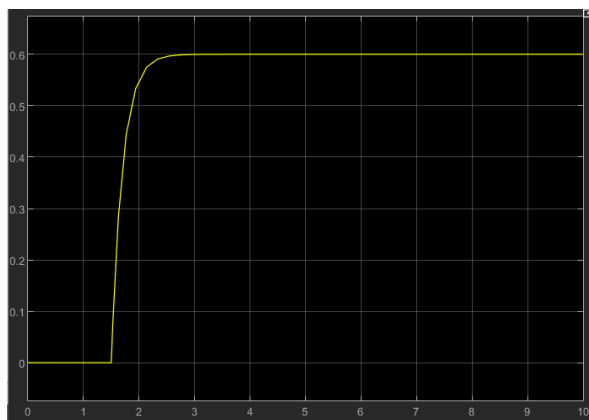


(第一小題  $a=0$ ) Yellow line

(第一小題  $a=1$ ) Blue line

(第一小題  $a=2$ ) Red line

(第二小題) Green line



(第三小題)

● 實驗練習 1-2

- 建立  $\text{sys1} = \frac{1}{s+1}$  與  $\text{sys2} = \frac{(s-20)}{(s-2)^2(s-5)}$
- 將  $\text{sys1}$  與  $\text{sys2}$  並聯
- 將  $\text{sys1}$  與  $\text{sys2}$  串聯
- 將  $\text{sys1}$  做單位回授
- 利用 `tfdata`、`zpkdata` 與 `get` 取得  $\text{sys1}$  單位回授後的系統資訊

程式碼

```

1 % $ domain declaration
2 - s = tf('s');
3
4 % create system1 and system2
5 - sys1 = 1/(s+1);
6 - sys2 = (s-20)/((s-2)^2*(s-5));
7
8 % parallel and series system1 and system2
9 - sysp = parallel(sys1,sys2);
10 - sysss = series(sys1,sys2);
11 |
12 % Feedback system1 and unity
13 - sysf = feedback(sys1,1,-1);
14
15 %Get system information
16 - [num,den,Ts_t] = tfdata(sysf,'v');
17 - [z, p, k, Ts_z] = zpkdata(sysf,'v');
18 - get(sysf);

```

系統資訊

```

Command Window

Numerator: {[0 1]}
Denominator: {[1 2]}
Variable: 's'
IODelay: 0
InputDelay: 0
OutputDelay: 0
Ts: 0
TimeUnit: 'seconds'
InputName: ''
InputUnit: ''
InputGroup: [1x1 struct]
OutputName: ''
OutputUnit: ''
OutputGroup: [1x1 struct]
Notes: [0x1 string]
UserData: []
Name: ''
SamplingGrid: [1x1 struct]

```

工作區參數

Name ▼	Value
z	[]
Ts_z	0
Ts_t	0
sysss	1x1 tf
sysp	1x1 tf
sysf	1x1 tf
sys2	1x1 tf
sys1	1x1 tf
s	1x1 tf
p	-2
num	[0,1]
k	1
den	[1,2]

### ● 實驗練習 1-3

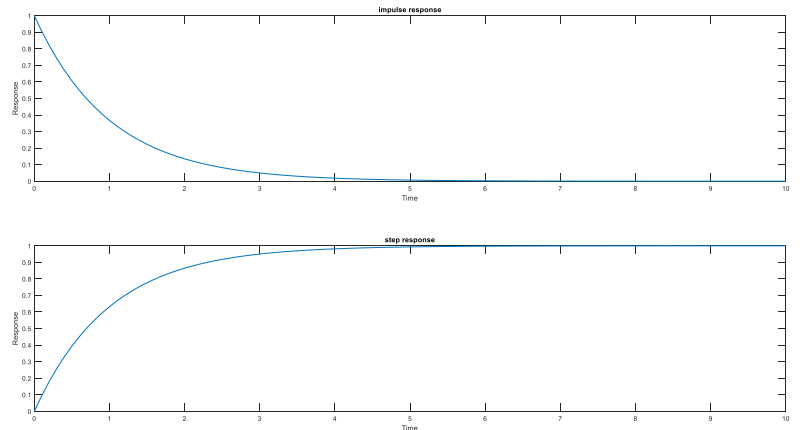
#### □ 畫出sys1的單位脈衝響應與單位步階響應

- ▢ 設定色彩、線條樣式和點的標記樣式
- ▢ 時間範圍、時間間隔
- ▢ 利用figure、subplot、hold on
- ▢ 標記、文字內容、附註、標題

程式碼

模擬結果

```
Editor - D:\google drive 21469\NCKUdata\控制
lab1_1.m x lab2.m x lab3.m x la
1 % S domain declaration
2 s = tf('s');
3
4 % create system1
5 sys1 = 1/(s+1);
6
7 t=0:0.1:10;
8 [y_impulse,t]=impz(sys1,t);
9 [y_step,t]=step(sys1,t);
10
11 subplot(211);
12 plot(t,y_impulse);
13 title('impulse response');
14 xlabel('Time');
15 ylabel('Response');
16
17 subplot(212);
18 plot(t,y_step);
19 title('step response');
20 xlabel('Time');
21 ylabel('Response');
```

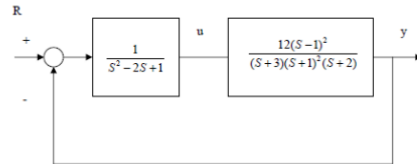


## ● 實驗練習 1-4

- 考慮下圖的系統求取30秒的步階響應及u的輸出響應結果

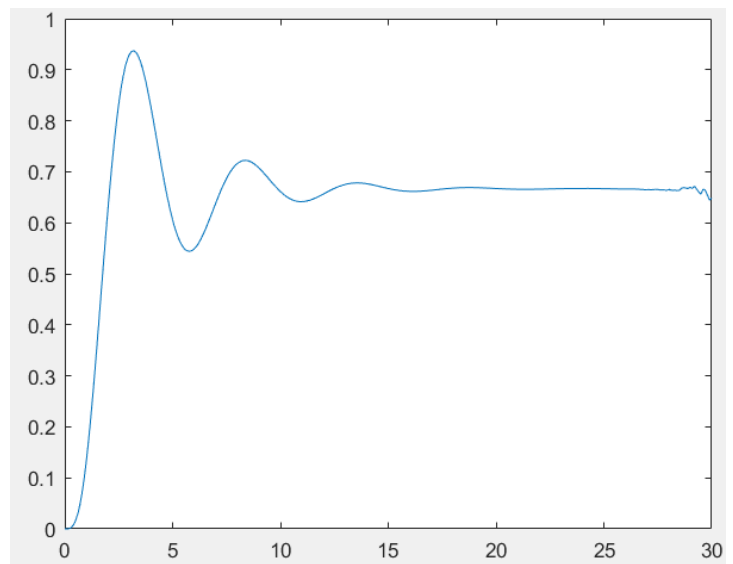
(1)M-file

(2)simulink

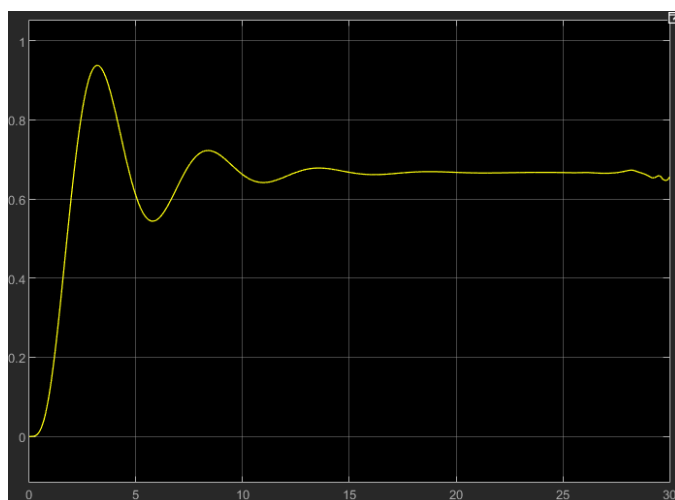
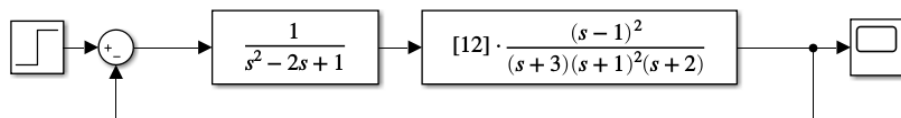


M-file:

```
Num = conv([12],[1 -1]);
Num = conv(Num,[1 -1]);
Den = conv([1 3],[1 1]);
Den = conv(Den,[1 1]);
Den = conv(Den,[1 2]);
Den = conv(Den,[1 -2 1]);
sys1 = tf(Num,Den);
sys1 = feedback(sys1,1,-1);
t=0:0.1:30;
[y_step,t]=step(sys1,t);
plot(t,y_step);
```



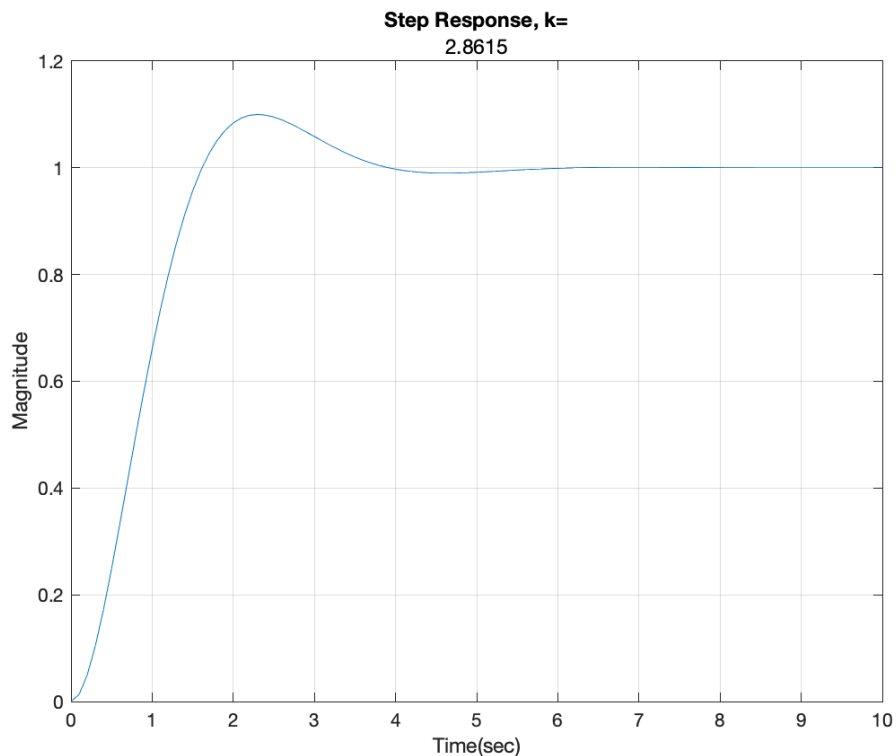
Simulink:



● 實驗練習 2-1

假設有一個二階系統的轉移函數  $G(s) = \frac{K}{s^2 + 2s + K}$   
如果我們需要%os=10%;且Ts=3.8秒,請問K  
為多少可符合需求?(使用Matlab並求步階響應將結果畫出)

```
Ts = 3.8;  
pos = 10;  
zeta = -(log(pos/100))/(sqrt(pi^2+(log(pos/100))^2))  
omegan=2/(2*zeta)  
k = omegan^2  
  
num = k;  
den = [1 2 k];  
sys = tf(num,den);  
  
t = 0:0.1:10;  
[y,t]=step(sys,t);  
plot(t,y);%繪出曲線  
title('Step Response, k=',k)  
xlabel('Time(sec));  
ylabel('Magnitude');  
grid on
```

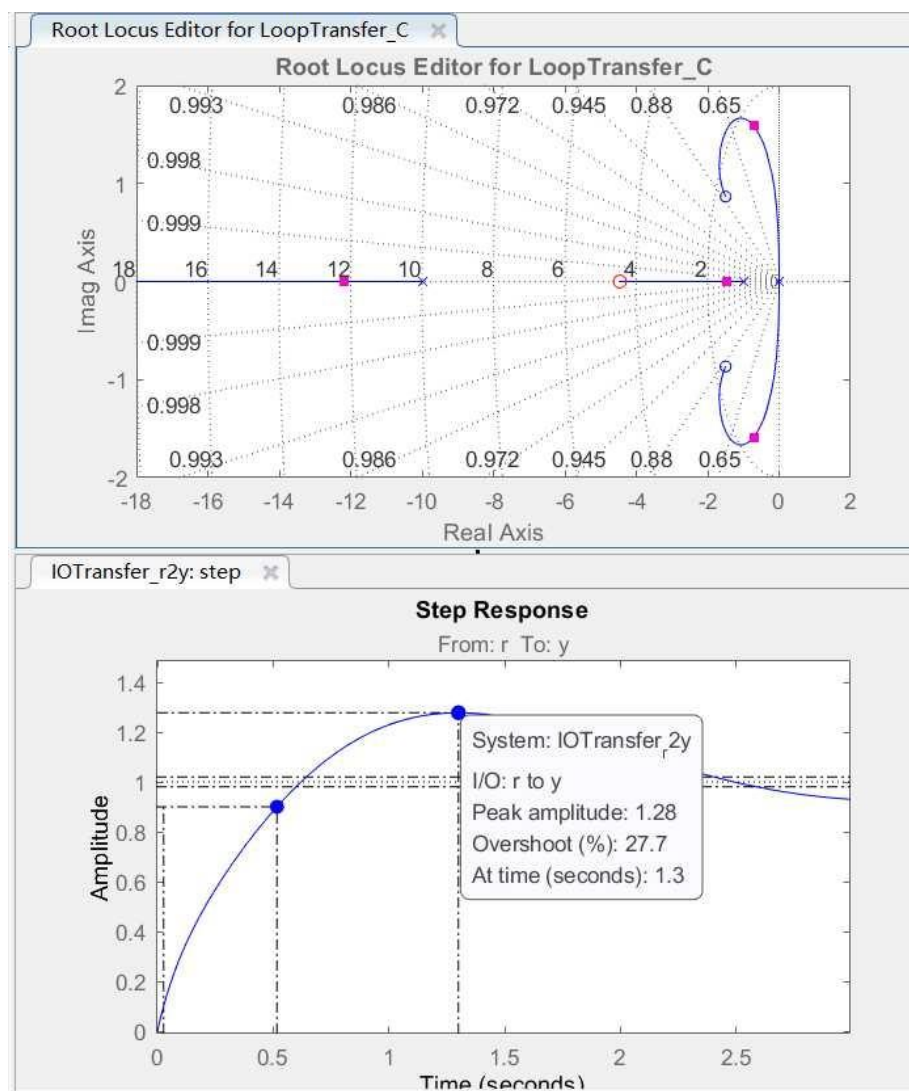


- 實驗練習 2-2

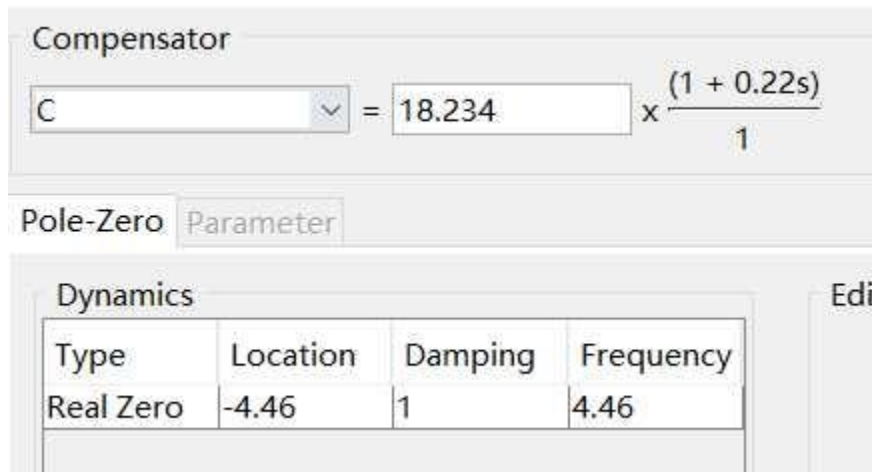
- Code

```
num_scalor=1;
num=[1 3 3];
num_re=conv(num_scalor,num);
den1=conv([1 0 0],[1 11 10]);
den2=[1 20];
den_re=conv(den1,den2);
sys=tf(num,den);
```

- In sisotool 調整 k 值







K 值為  $18.234 \cdot (0.22s + 1)$

### III. Conclusion

#### 1. 林珮玉

從第一堂課就非常期待實作的到來，這次課程算是相當簡單，主要是讓我們熟悉 ROBOTC 編譯環境，以及用 Lego 提供的 C-like 語言讓馬達轉動。其中我覺得最酷的是 EV3 這顆小主機，只要透過 USB 接上電腦，就可以將程式燒錄到 EV3 上使其驅動。語法也有點像是之前單晶片在玩的東西，期待下週上課！！

#### 2. 林哲緯

本次實驗算是相當簡單，使用 Lego 提供的 C-like 語言以及 API，主要算是熟悉了一下整體的使用環境，覺得還蠻有趣的！

#### 3. 林玠志

這週的實驗主要是熟悉 EV3 控制器的操作方式，用內建 function 來控制馬達或 sensor，希望下次的機器人會有趣一點，學到更多東西！

#### 4. 薛博文

這次 Lab 用 C-like 語言來控制 Lego 馬達，做起來的感覺很像是之前玩過的 Arduino，感覺也還有不少好玩的東西有待探索！

#### 5. 蔡孟宗

這周做的實驗是學習如何用 ROBOTC 提供的 function 控制馬達和感測器的行為，雖然在做 Bonus 時一開始沒搞懂題目的意思因此繞了彎路做了許多嘗試，不過藉著這些嘗試我們也因此能更了解 EV3 的能耐和運作機制，總體來說還是很有收穫。後來同組的組員發現 `setMotorTarget()` 可以用來指定馬達到特定的角度，利用這個 function 我們便順利的完成了 Bonus 題