

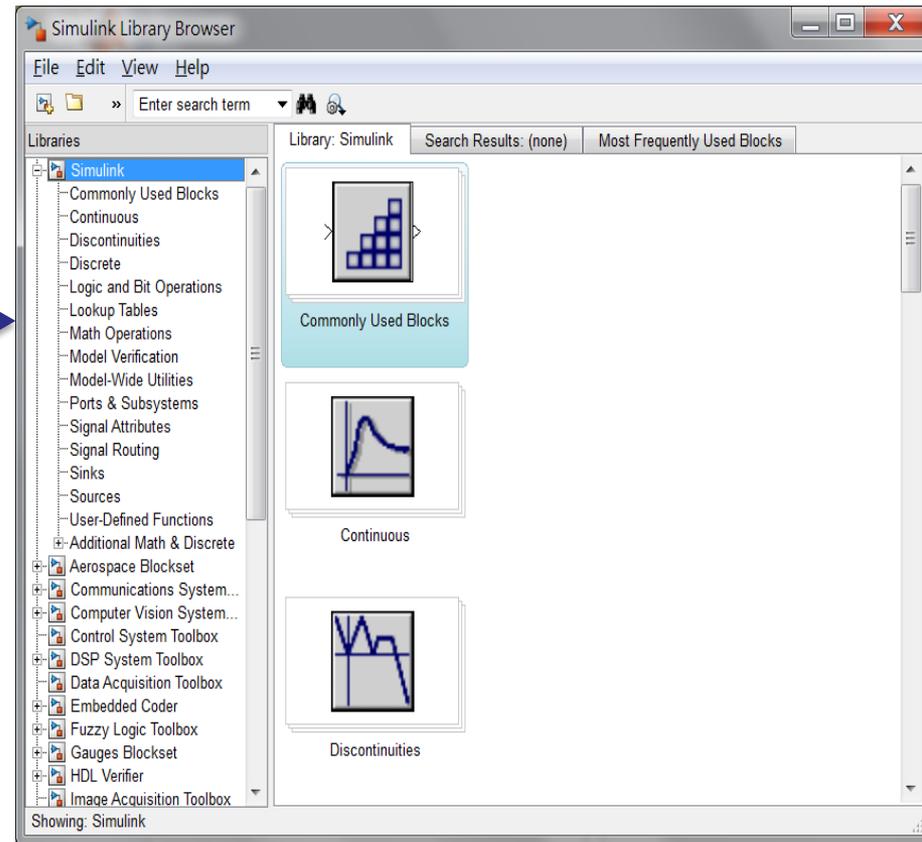
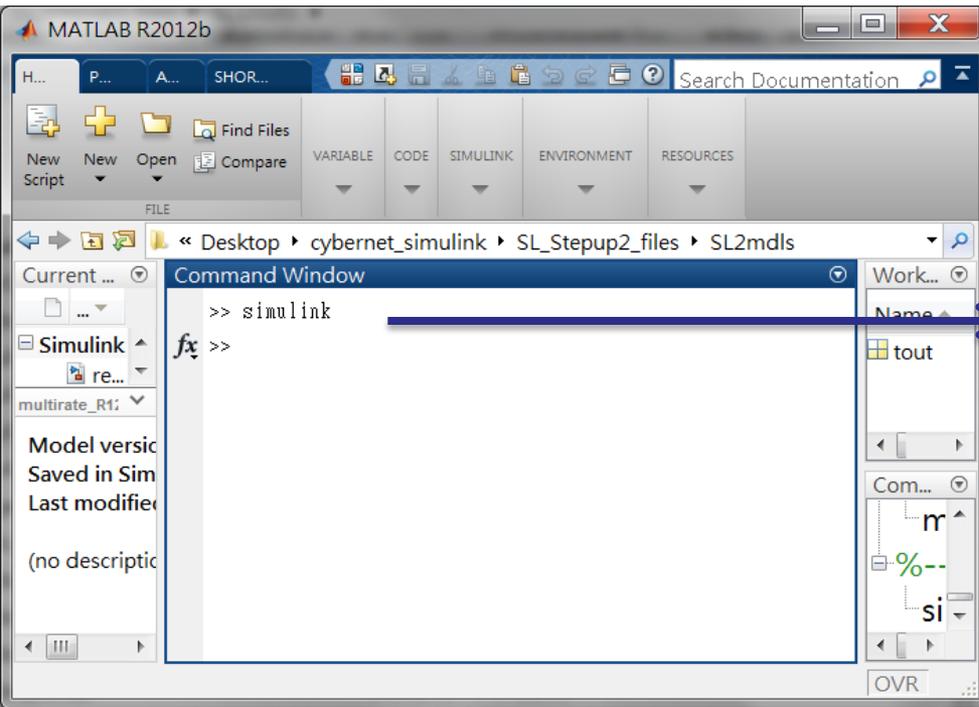
使用Simulink建構微分方程

support@terasoft.com.tw
Presenter: Michael Chen

Agenda

- Simulink 基本操作介紹
- 一個自由度的動態系統微分方程建構
- 多個自由度的動態系統微分方程建構
- 非線性系統
- 離散時間系統建構

Simulink 基本操作介紹

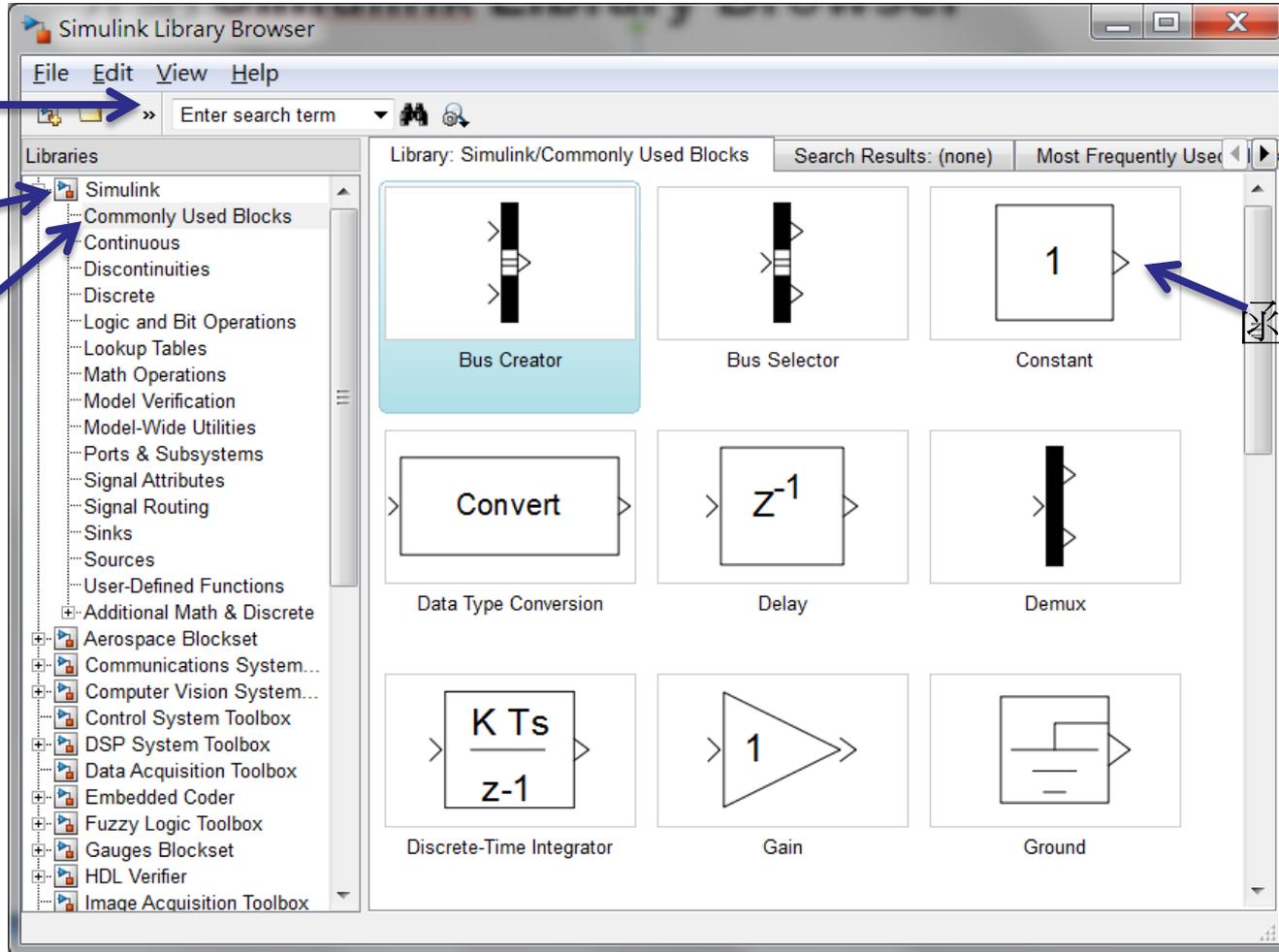


介紹 Simulink Library Browser

方塊搜尋

函式庫

子函式庫



函式庫方塊

Simulink 函式庫

- Commonly used blocks – 一般常用的方塊
- Continuous – 連續時間方塊, 如 Integrator, Derivative
- Discontinuities – 非連續的方塊, 如 Saturation
- Discrete – 離散時間方塊, 如 Unit Delay
- Logic and Bit Operations – 邏輯或位元運算方塊 如邏輯運算子 (Logic operator), 關係運算子 (Relational operator)
- Look-Up Tables – 查表法方塊 如 sine, cosine
- Math Operations – 數學運算; 如總和 (sum), 乘積 (product)
- Model Verification – 錯誤偵測
- Model-Wide Utilities – Model 方塊的資訊, 如 Model Info

Simulink 函式庫 (續)

- Ports & Subsystems – 埠與子系統方塊如 In1, Out1, Subsystem
- Signals Attributes – 信號的屬性, 如資料型態轉換 (Data Type Conversion)
- Signals Routing – 信號的路由選擇, 如 Switch, From, Goto
- Sinks – 顯示或輸出資料的方塊, 如 Display, Scope
- Sources – 訊號源方塊, 如 Clock, Sine Wave, Constant, Pulse Generator
- User-Defined Functions – 使用者定義的方塊, 如 MATLAB Function, S-Function
- Additional Math & Discrete – 其他數學與離散方塊

建立一個新的Model

The image shows two windows from the Simulink environment. The left window is the 'Simulink Library Browser', displaying a grid of blocks such as 'Bus Creator', 'Bus Selector', 'Constant', 'Data Type Conversion', 'Delay', 'Demux', 'Discrete-Time Integrator', 'Gain', and 'Ground'. The right window is the 'Model Explorer' for an 'untitled' model, showing a menu with options like 'Library Browser', 'Model Explorer', 'Simulink Project', 'Model Dependency Viewer', 'Requirements at This Level', 'Model Browser', 'Configure Toolbars', 'Toolbars', 'Status Bar', 'Explorer Bar', 'Navigate', 'Zoom', 'Smart Guides', and 'MATLAB Desktop'. Annotations with blue arrows point to the 'New Model' icon in the Model Explorer menu, the 'Simulink 工具欄' (Simulink toolbar) in the Model Explorer window, and the 'Status bar' at the bottom of the Model Explorer window.

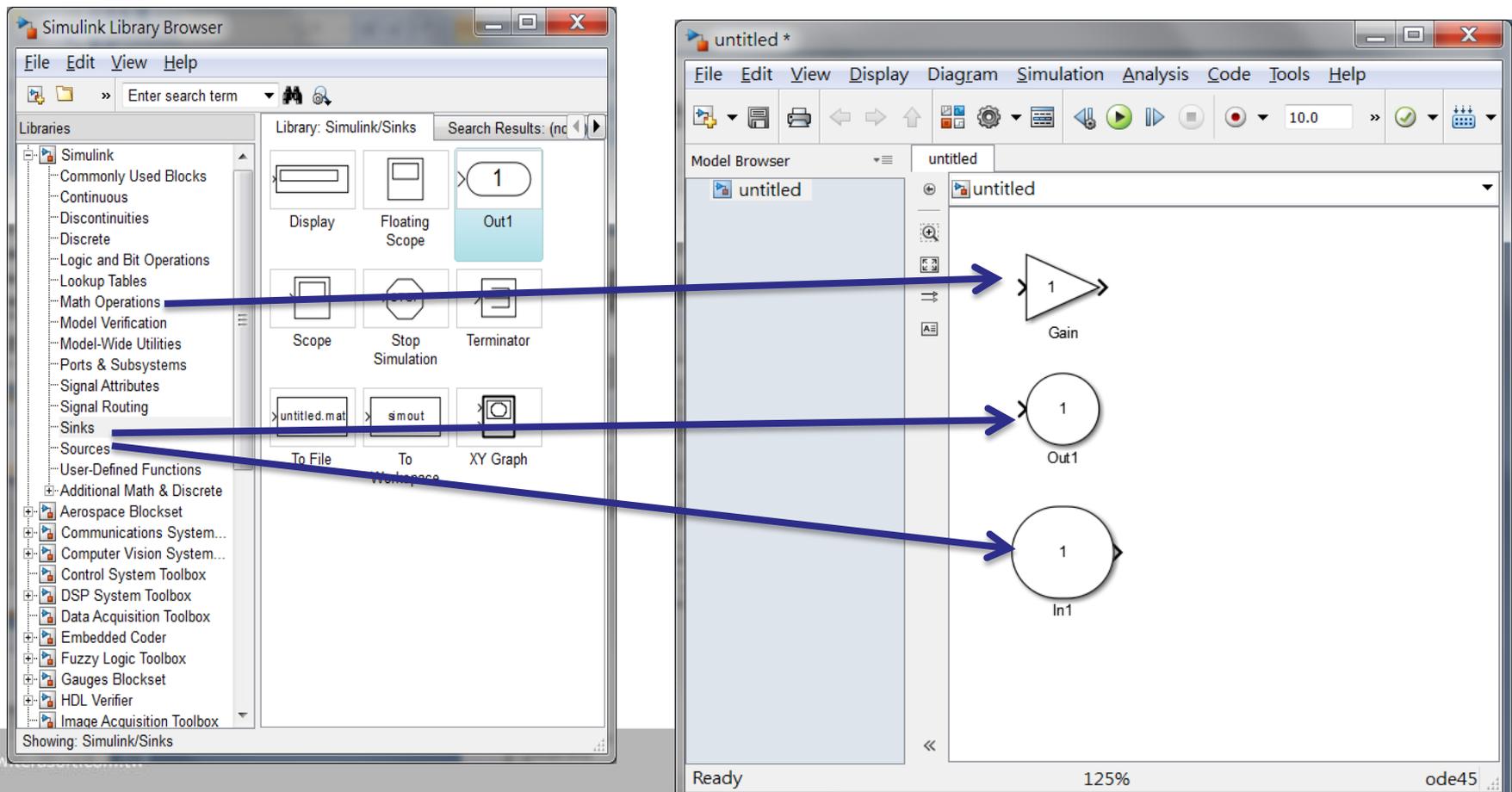
新model icon

Simulink 工具欄

Status bar

於Model加入方塊

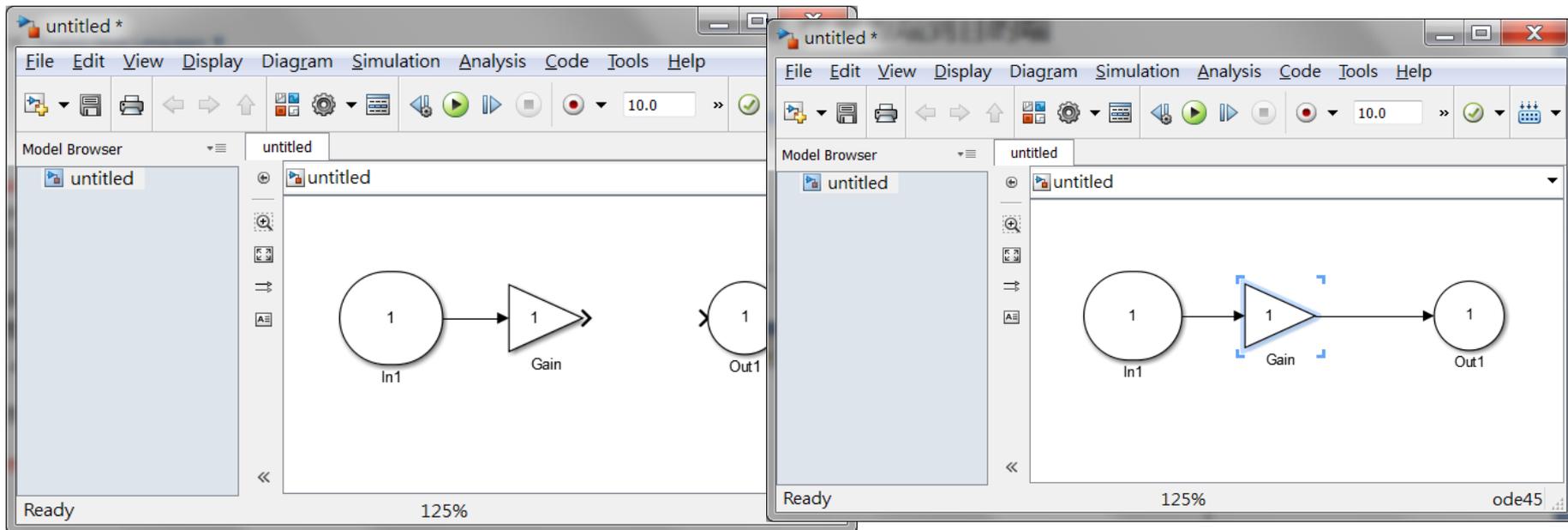
可以用從資料庫瀏覽器拖拉方塊到model裡面,方塊依字母的順序排列在函式庫裡



The image shows two windows from the Simulink environment. The left window is the 'Simulink Library Browser' with the 'Sinks' category selected. Three blue arrows point from blocks in the library to the right window. The right window is the 'Model Browser' for an 'untitled' model, showing three blocks added to the workspace: a Gain block (triangle), an Out1 block (circle), and an In1 block (circle). The status bar at the bottom indicates 'Ready', '125%', and 'ode45'.

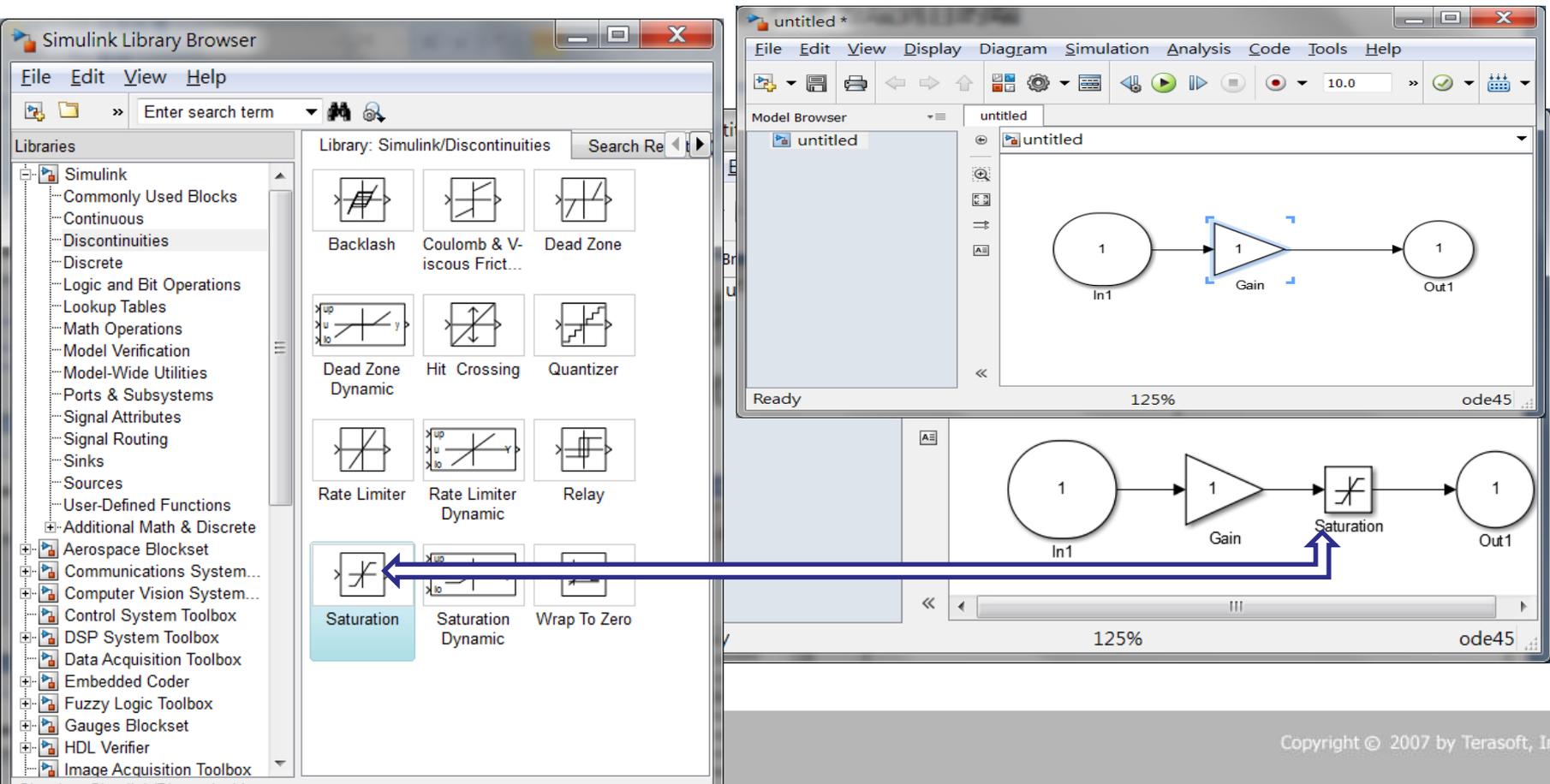
連結訊號

- **Ctrl+滑鼠左鍵** 從來源端到目的端



插入方塊

- 在訊號中間插入方塊,把方塊放在訊號線上



The image displays two Simulink windows. The left window is the 'Simulink Library Browser', showing a list of libraries on the left and a grid of blocks in the 'Simulink/Discontinuities' library. The 'Saturation' block is highlighted with a blue selection box. A blue arrow points from this block to the right window. The right window is the 'Model Browser' for an 'untitled' model. It shows two diagrams. The top diagram shows a signal path from 'In1' through a 'Gain' block to 'Out1'. The bottom diagram shows the same signal path but with a 'Saturation' block inserted between the 'Gain' and 'Out1' blocks. The 'Saturation' block is highlighted with a blue selection box, and a blue arrow points from the Library Browser to it.

設定方塊參數

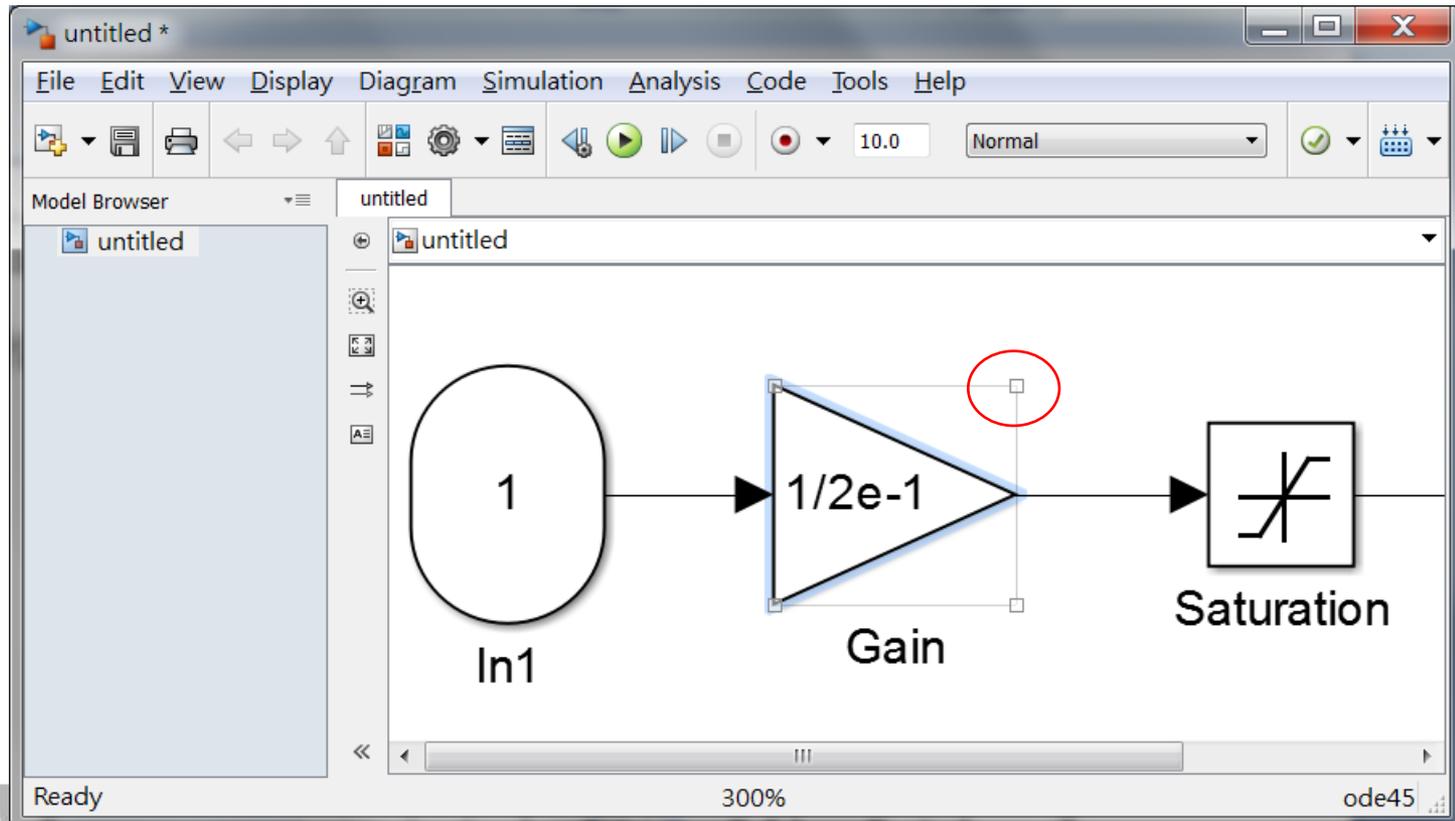
- 連點滑鼠兩下開啟一個方塊參數,在欄位輸入是當的值

The screenshot illustrates the process of setting parameters for a Gain block and a Saturation block in a Simulink model. The main window shows a Simulink diagram with an input port 'in1', a Gain block, a Saturation block, and an output port 'Out1'. Two dialog boxes are open:

- Function Block Parameters: Gain**: Shows the Gain value set to $1/2e-1$. The Multiplication is set to Element-wise($K.*u$). The Sample time is set to -1.
- Function Block Parameters: Saturation**: Shows the Upper limit set to 0.5 and the Lower limit set to -0.5. The checkboxes for 'Treat as gain when linearizing' and 'Enable zero-crossing detection' are checked. The Sample time is set to -1.

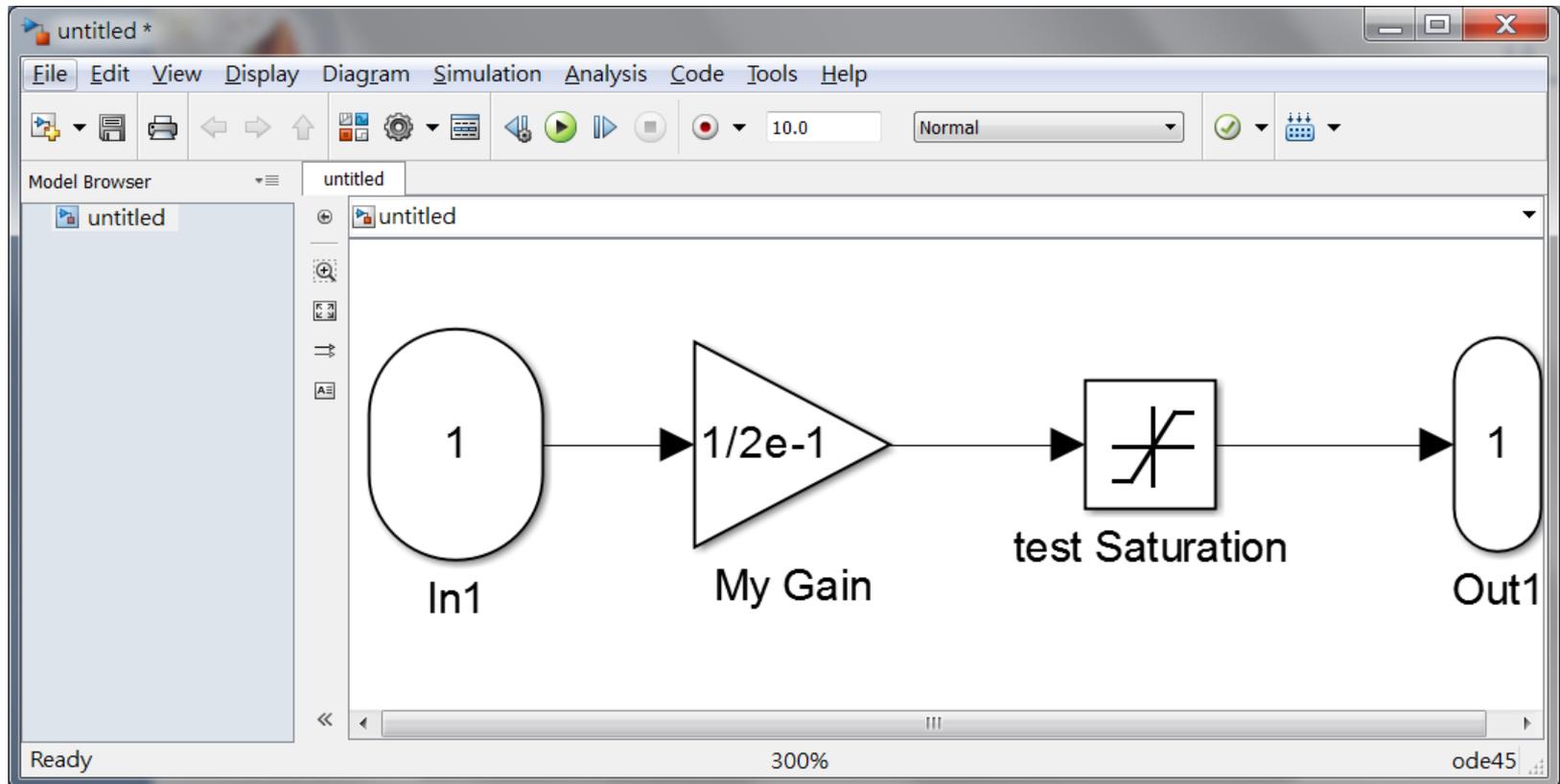
調整方塊的大小

滑鼠點到指定的方塊;方塊有四個角,透過四個角可以調整其方塊大小



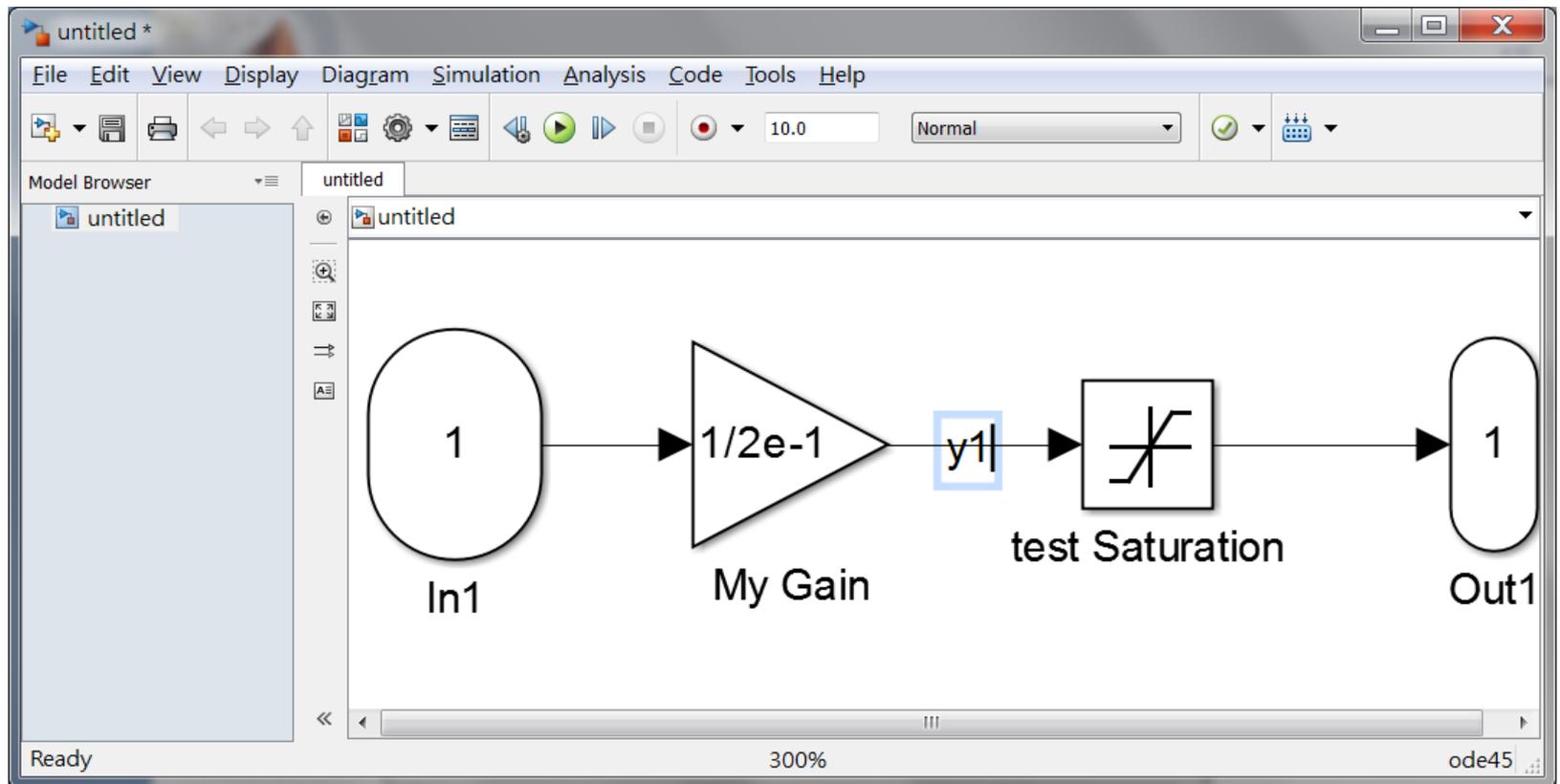
標記方塊

點選方塊標籤然後標註方塊名，每個方塊的名字都要是唯一的。



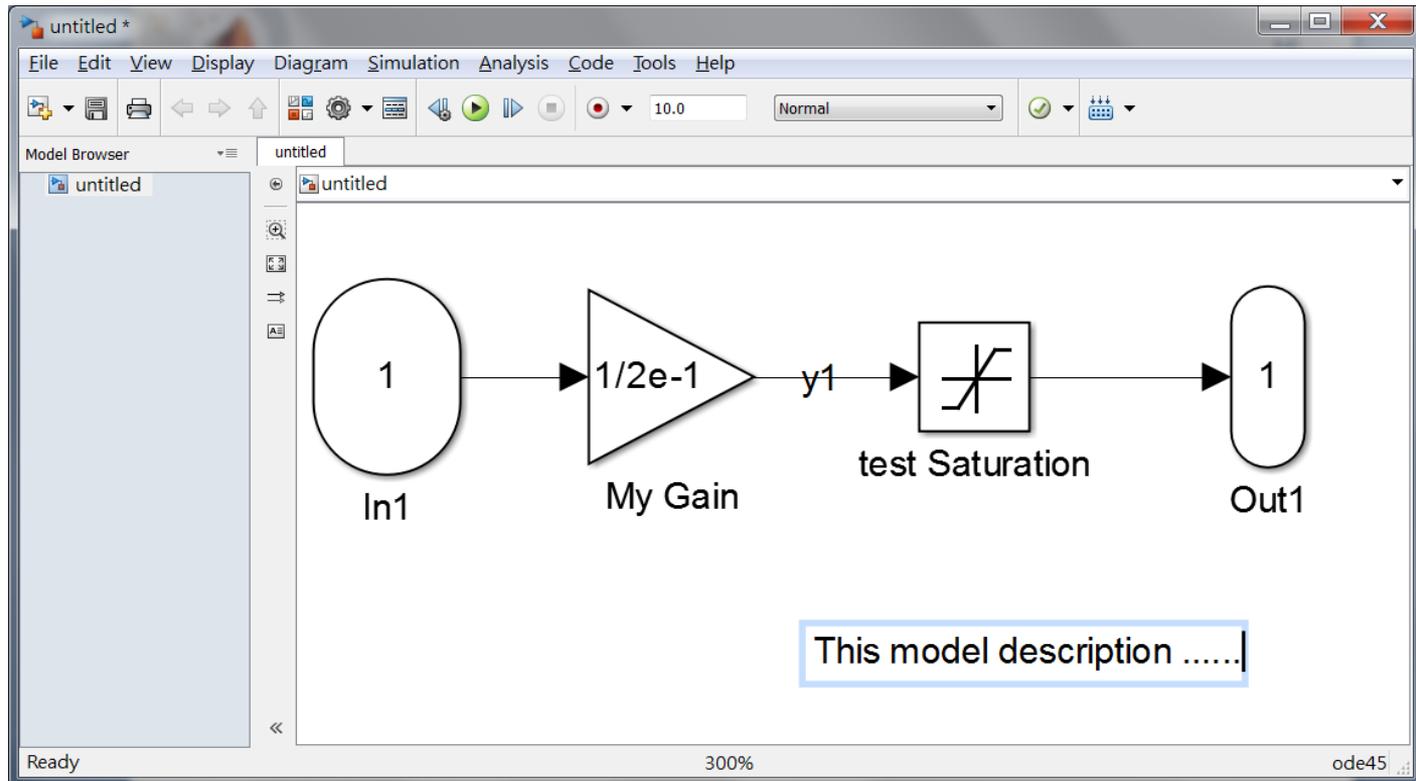
標示訊號

點訊號線兩下 輸入你要給的訊號名。



增加註解

- 可以在model 空白處 連點兩下輸入文字註解。



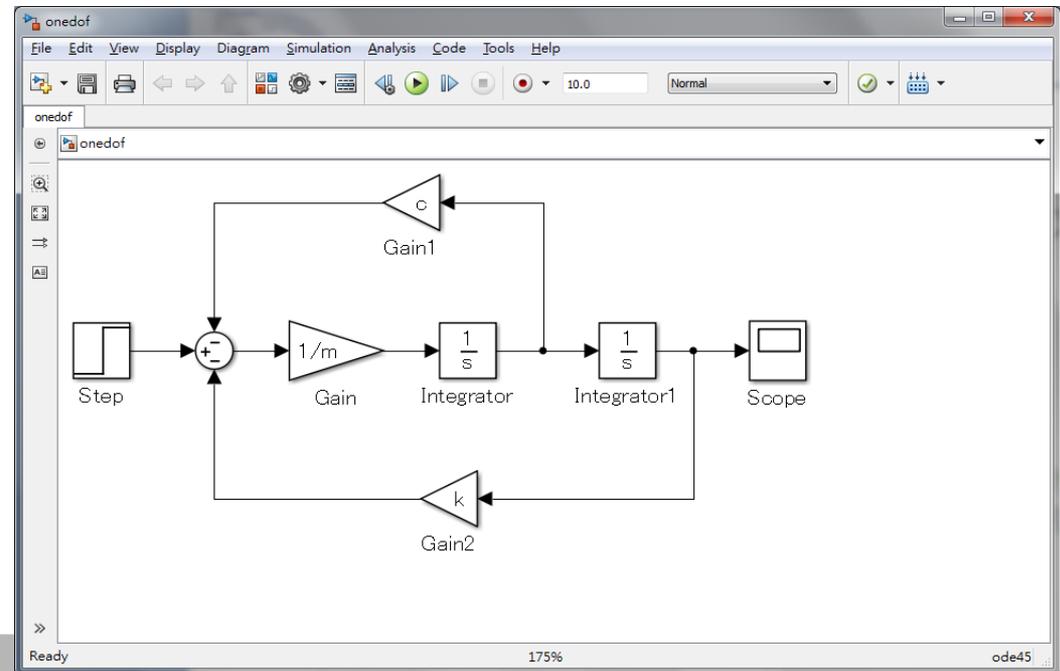
一個簡單的模型

The image shows a Simulink model window titled 'cn_dc'. The model contains two 'Sine Wave' blocks: 'Continuous' and 'Discrete'. Both are connected to a summing junction, which is then connected to a 'Scope' block. The 'Continuous' block has a smooth sine wave icon, while 'Discrete' has a stepped sine wave icon. The 'Scope' block displays a plot of the signal over time, showing a smooth yellow sine wave and a purple stepped sine wave. The plot axes range from -1 to 1 on the y-axis and 0 to 10 on the x-axis. The 'Scope' window title is 'Scope' and it shows 'Time offset: 0'.

Two 'Source Block Parameters: Sine Wave' dialog boxes are overlaid on the model. The top dialog box has 'Sample time: 0' and is annotated with 'Sample time 設 0 表示continuous'. The bottom dialog box has 'Sample time: 0.5' and is annotated with '0.5秒update 一次'. Both dialog boxes show parameters: Time (t): Use simulation time, Amplitude: 1, Bias: 0, Frequency: 0, Phase (rad): 0, and a checked box for 'Interpret vector parameters as 1-D'.

一個自由度微分方程

- 方程式如右 $m\ddot{x} + c\dot{x} + kx = f$ ，二階方程式使用兩個積分器。
- 使用積分器實現微分。



微分方程式的實現

- 使用積分器建立連續系統

$$\dot{y} \xrightarrow{\int} y$$

- 二次微分經過積分器變成一次微分項,一次微分的函數經過積分得原函數

$$\ddot{y} \xrightarrow{\int} \dot{y} \xrightarrow{\int} y$$

- 建立不同狀態的代數關係
- 設定積分器的初始條件

連續時間系統:使用積分器(in continuous block library)

- 系統的輸出是連續的變化
- 用積分的方式表示輸入與輸出的關係

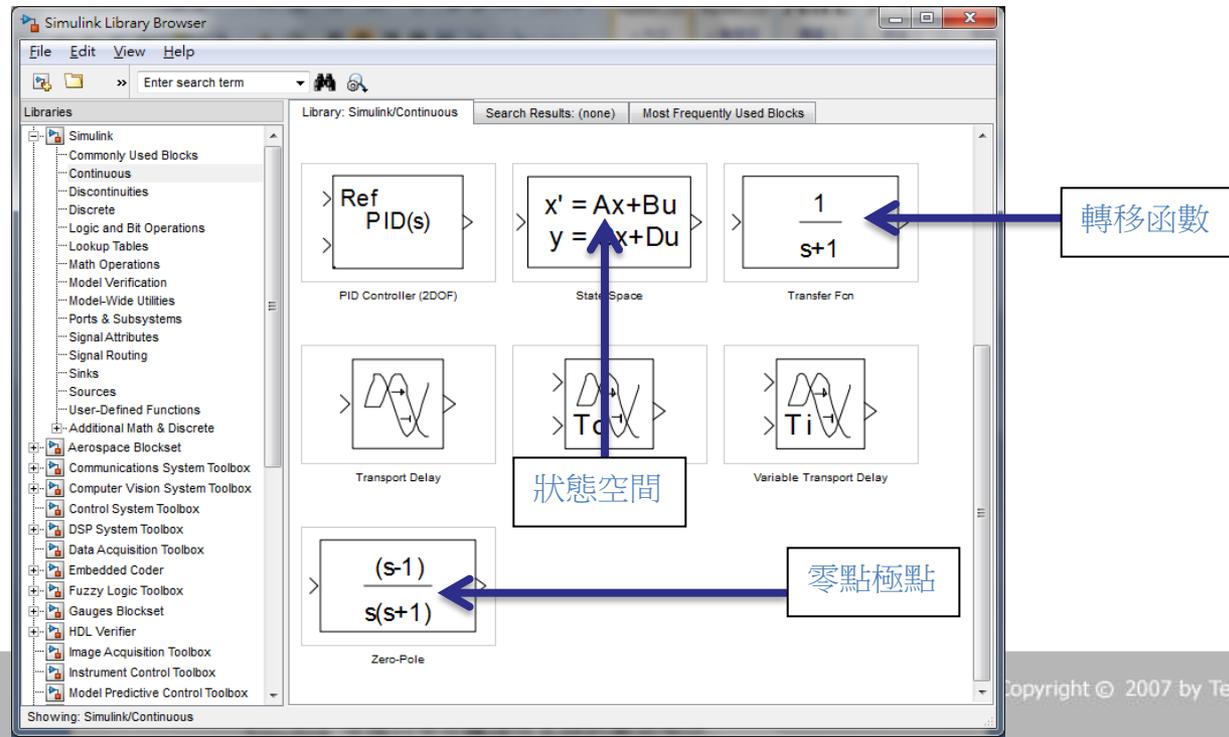
Example: 積分器(Integrator)

Output is the integral of the input

$$\dot{y} = u \quad \Rightarrow \quad \begin{aligned} \dot{x} &= u \\ y &= x \end{aligned}$$

三種線性系統的表示方法

- 轉移函數(transfer function)
- 狀態空間(state space)
- 零點極點(zero-pole)



線性系統的表示方法: transfer function

- 假設 initial value 等於零

$$m\ddot{x} + c\dot{x} + kx = f \longrightarrow mX(s)s^2 + cX(s)s + kX(s) = F(s)$$

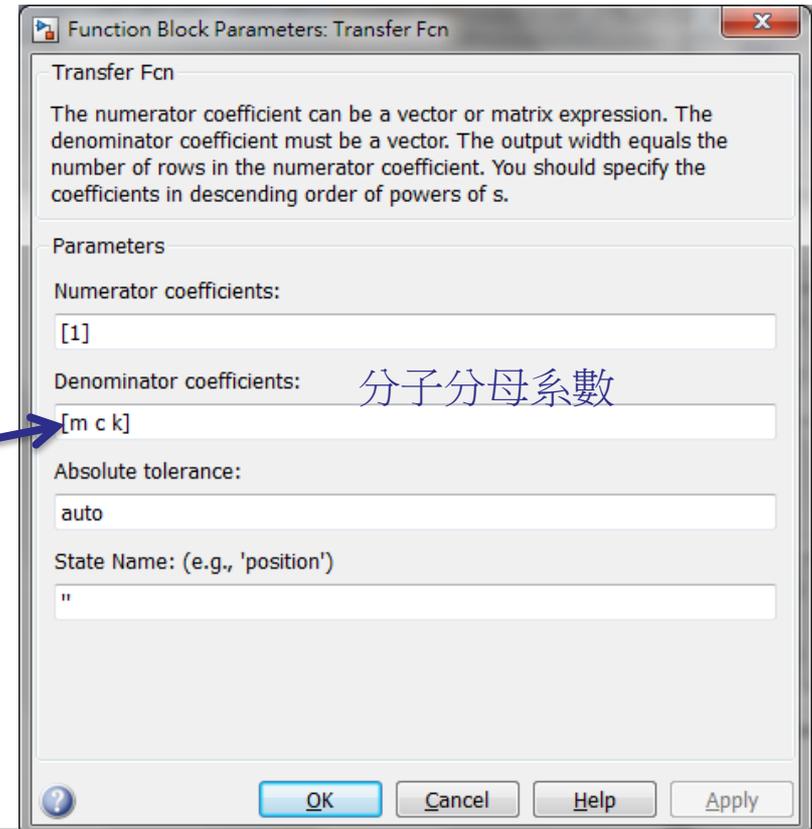
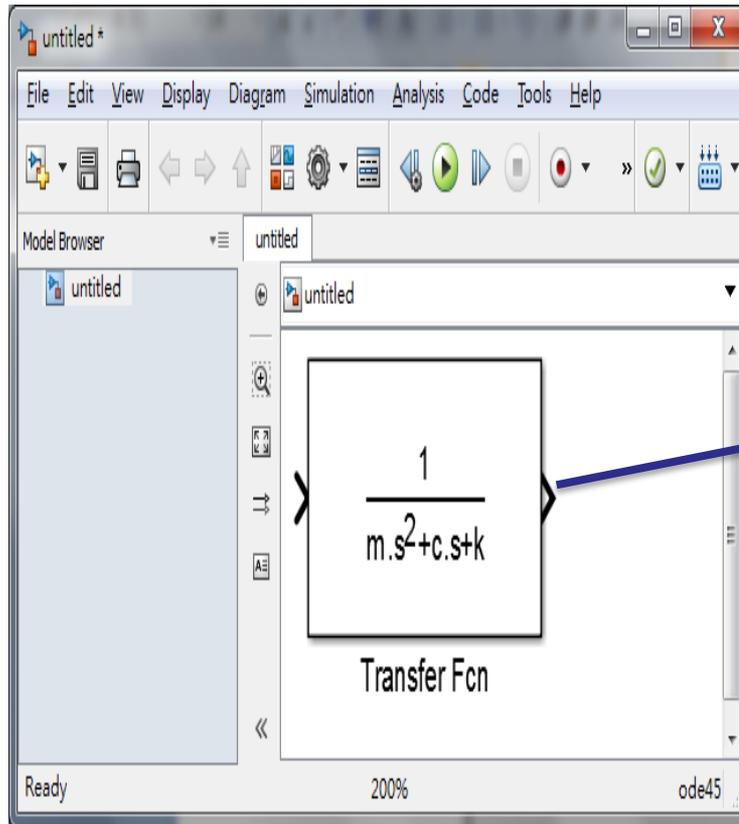
Laplace transform

整理得如下轉移函數

$$G(S) = \frac{X(S)}{F(S)} = \frac{1}{ms^2 + cs + k}$$

Transfer function block 設定

$$G(S) = \frac{X(S)}{F(S)} = \frac{1}{ms^2 + cs + k}$$



狀態空間表示法

- 常用表示法如下

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

$$y = Cx + Du$$

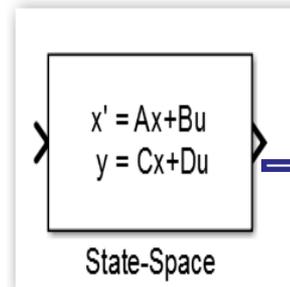
- Ex: $m\ddot{x} + c\dot{x} + kx = f$

$$\text{Assume } X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} x \\ \dot{x} \end{bmatrix}$$

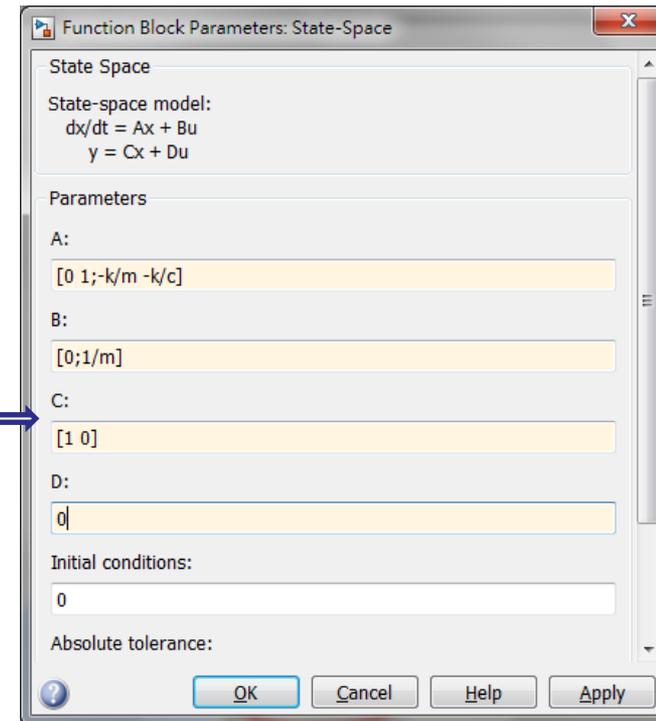
$$\Rightarrow \begin{cases} \dot{x}_1 = x_2 \\ m\dot{x}_2 + cx_2 + kx_1 = f \end{cases}$$

$$\dot{X} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} X + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} f$$

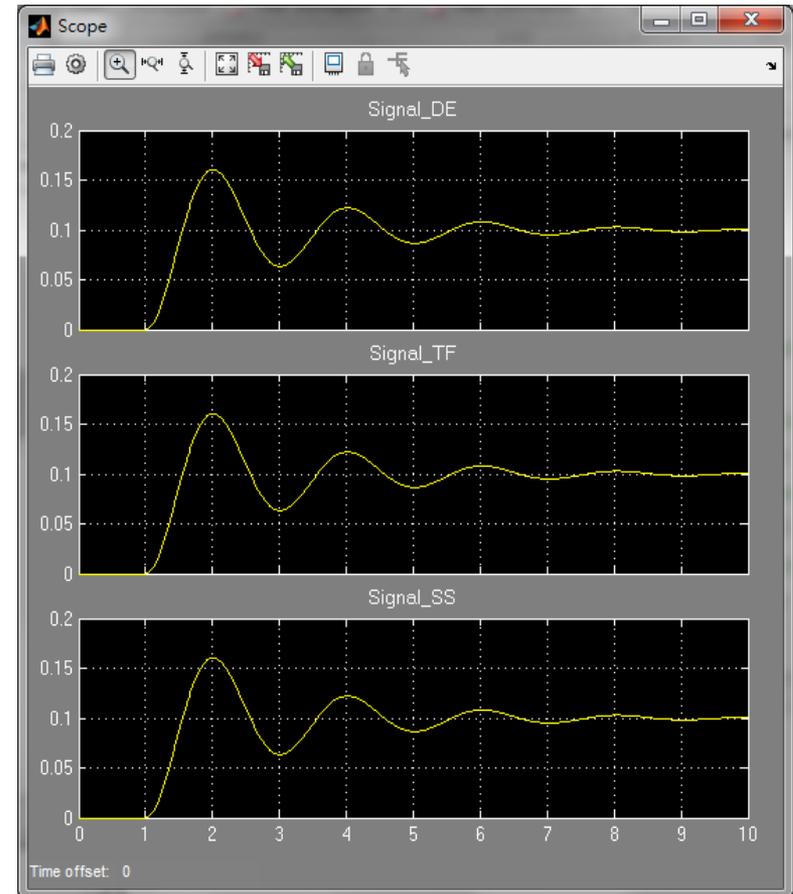
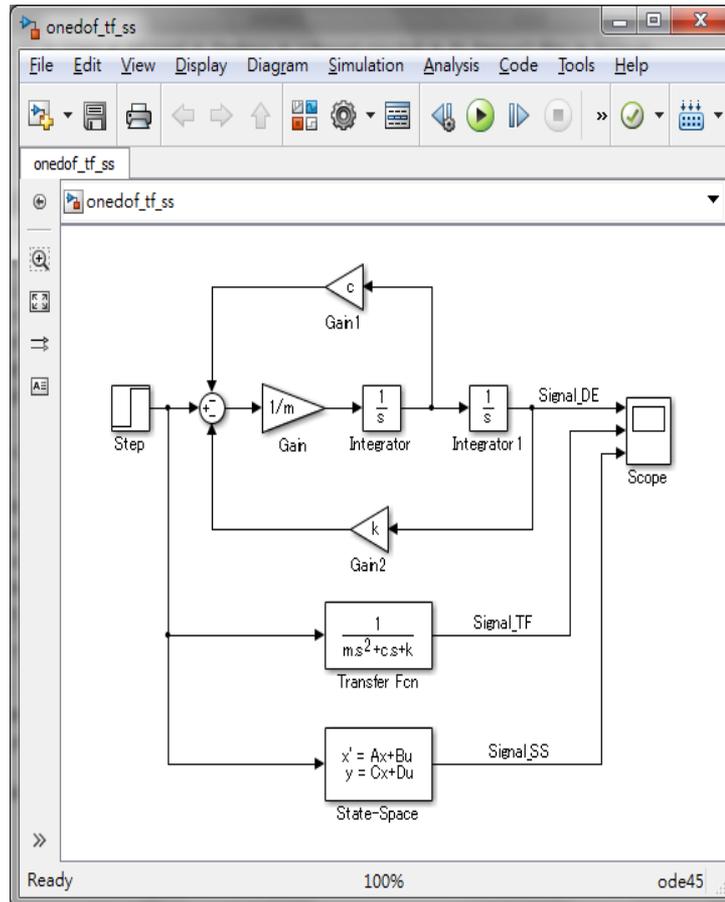
$$y = [1 \ 0]X + [0]f$$



A, B, C, D 係數

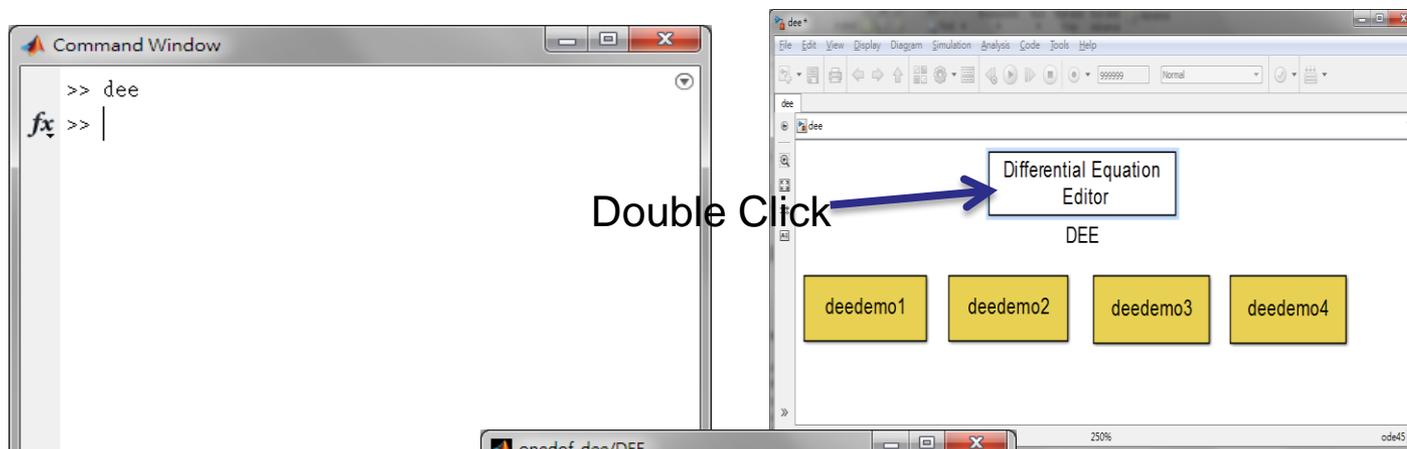


Simulation結果比較(DE, TF, SS)



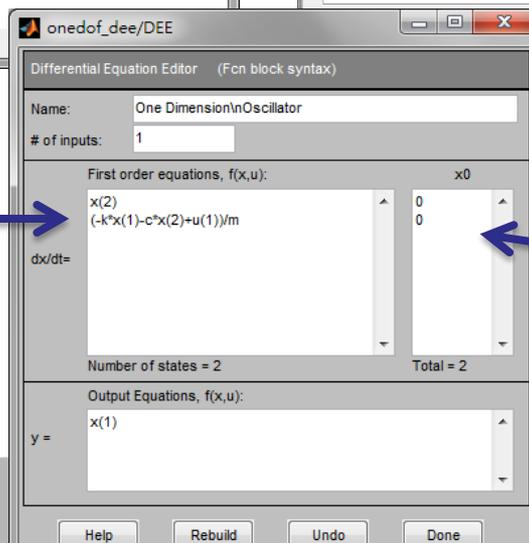
微分方程式編輯器(differential equation editor)

Dee 的使用方法:Command Window 輸入dee



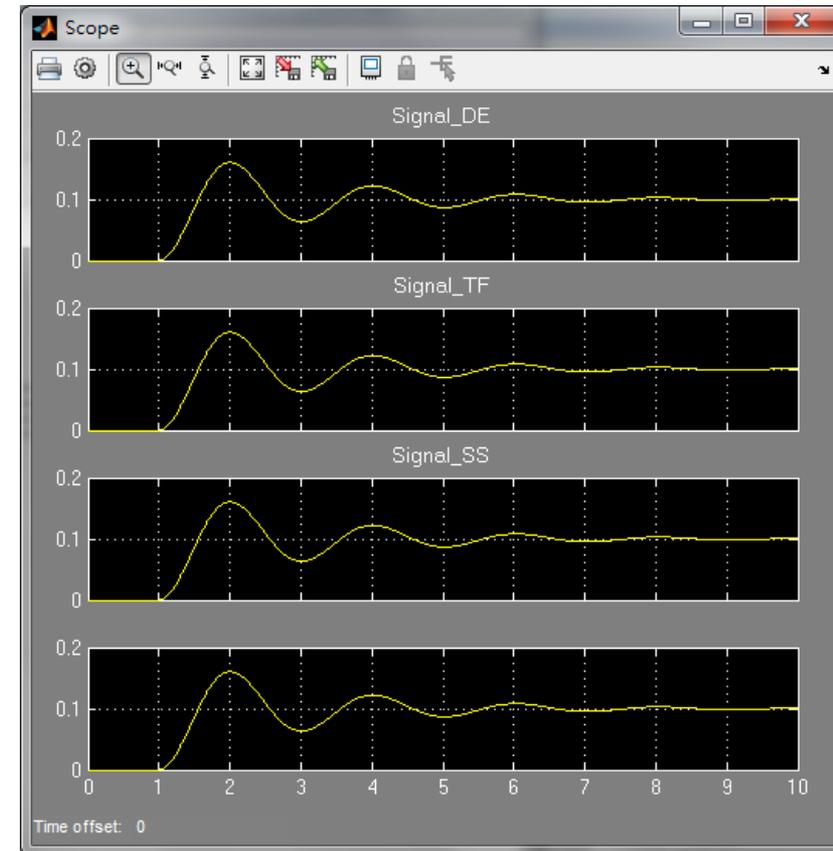
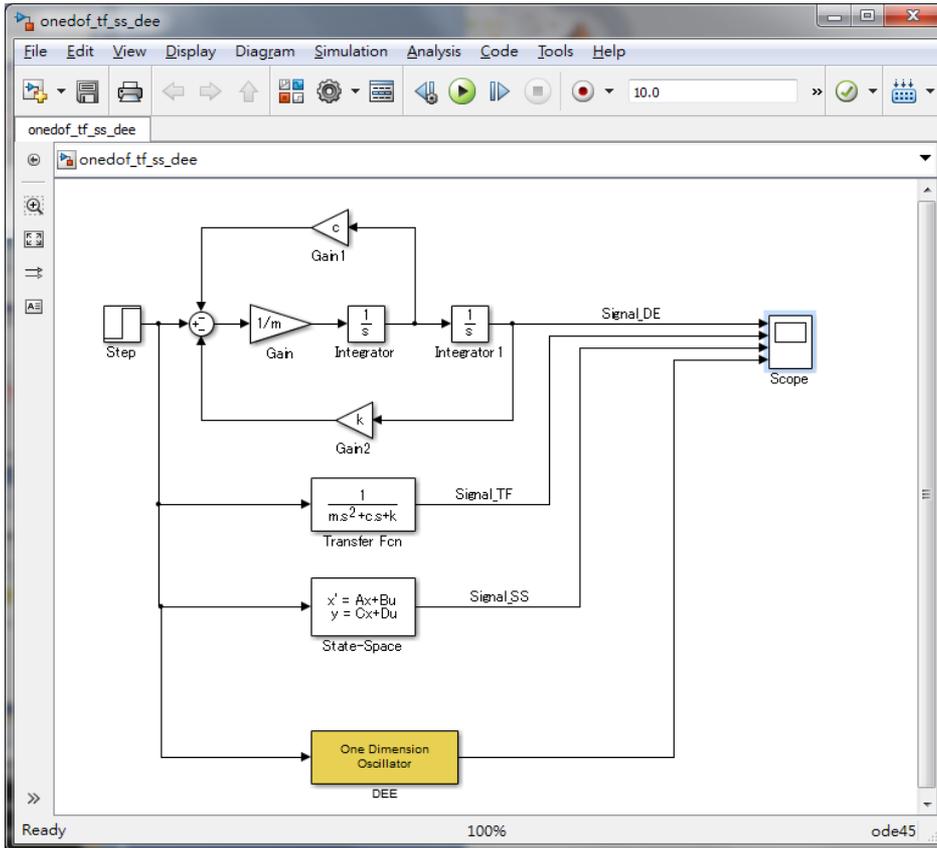
設定參數

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -\frac{k}{m}x_1 - \frac{c}{m}x_2 + \frac{f}{m} \end{cases}$$



$$\begin{cases} \dot{x}_1(0) = 0 \\ \dot{x}_2(0) = 0 \end{cases}$$

模擬結果比較(DE, TF, SS, DEE)

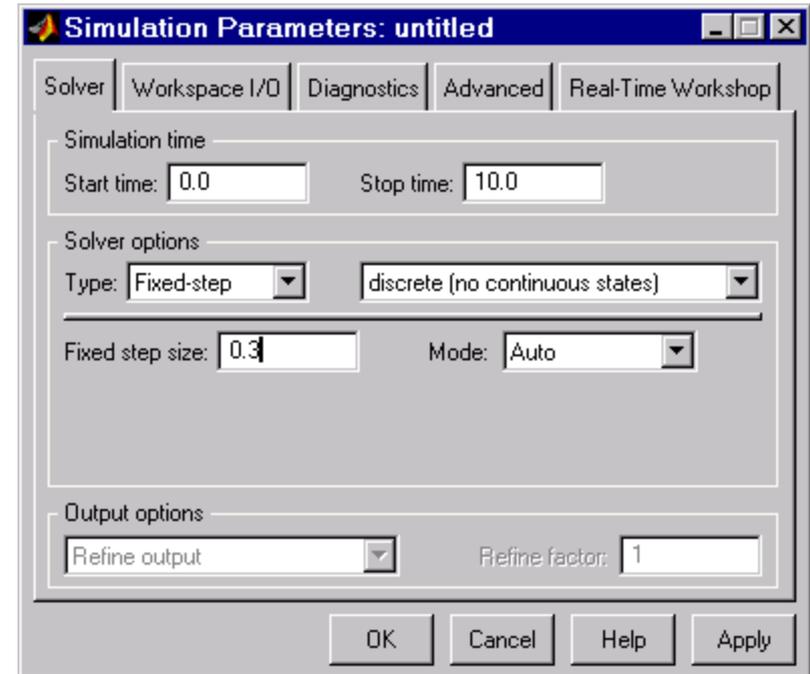


Exercise: 非線性差分方程

- 每年的人口跟以下參數有關
 1. 前年的人口
 2. 人口增加率, r ($r=1.05$)
 3. 其它變數, K ($K=1e6$)
- $p(n) = r * p(n-1) * (1 - p(n-1)/K)$ (非線性, $p(n-1)$ 的平方造成非線性)
- Initial value for population is needed! ($p(0) = 100000$)

Solver參數設定

- 固定步階解題器:
ode45,ode23...etc.
- 可變步階幾題器:
ode8,ode5....etc.
- 離散解題器

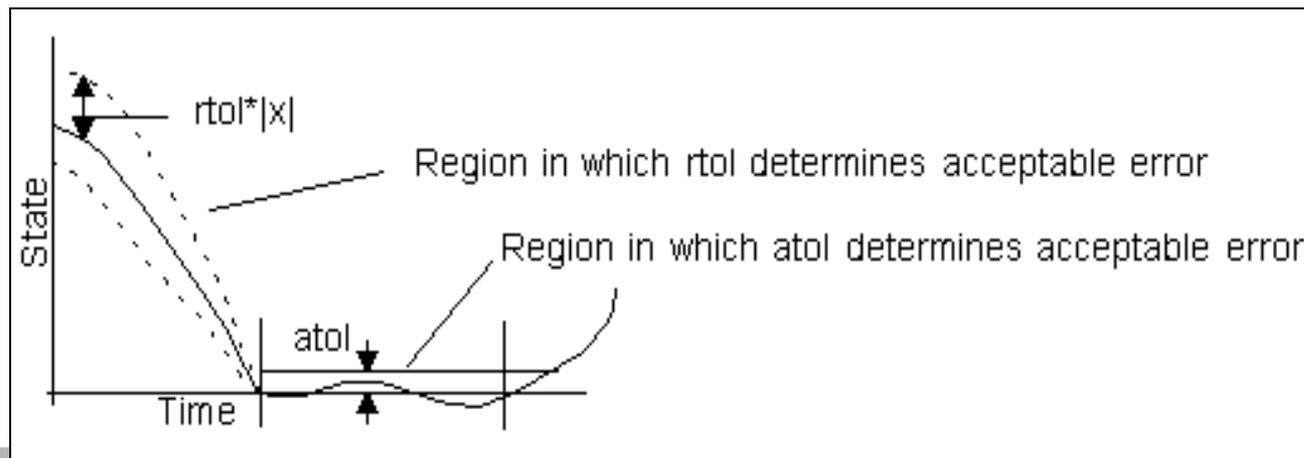


連續時間解題器(solver)

- 解題器的主要功能是計算方塊的輸出
- 在連續系統, 用積分來表示微分的狀態
- 不同的連續解題器使用不同的方法(數值分析)來逼近積分的結果
 - 固定步階(Fixed step): ode5, ode4, ode3, ode2, ode1
 - 可變步階(Variable step): ode45, ode23, ode113, ode15s, ode23s, ode23t ode23tb

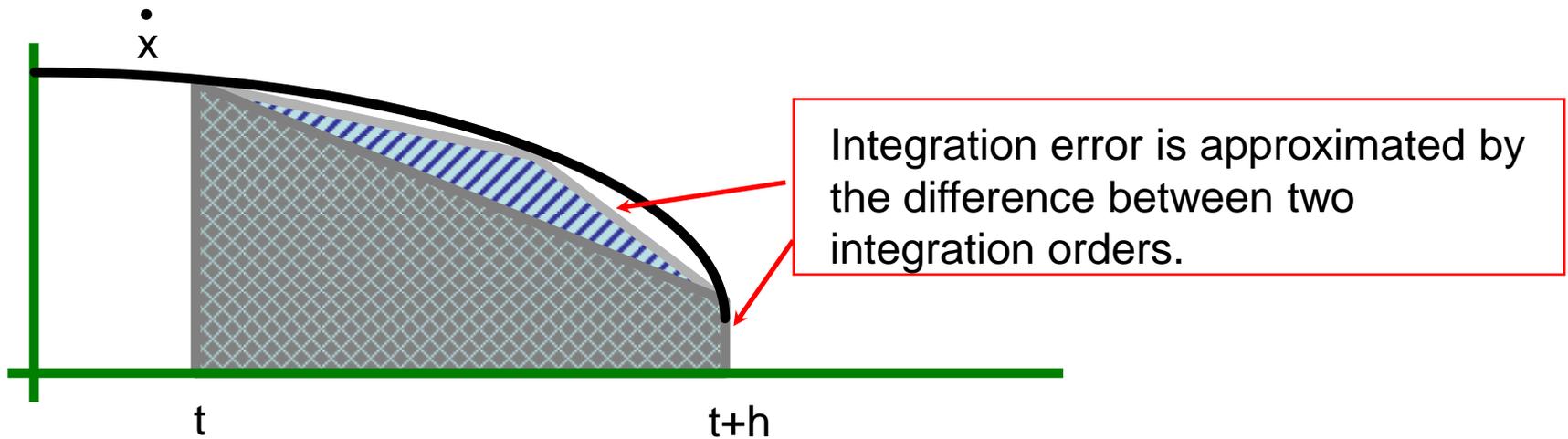
錯誤控制(Error Control)

- 可變步階解題器可以用來做錯誤的控制
- 絕對誤差: 積分錯誤的上限
- 相對誤差: 積分錯誤的上限除以變數的值
- 解題器可以滿足其中之一!



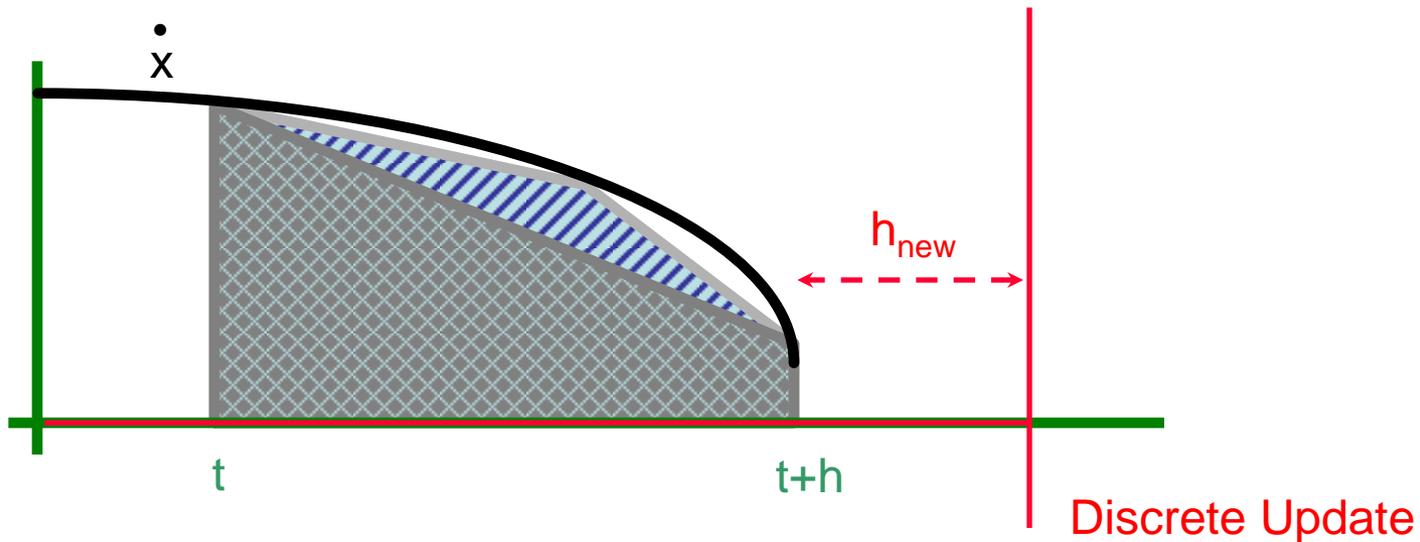
步階的計算 (Variable Step)

SIMULINK 用積分積derivatives的方式計算連續狀態



如誤差在可以接受範圍，模擬將繼續。否則將減小步階再做積分。

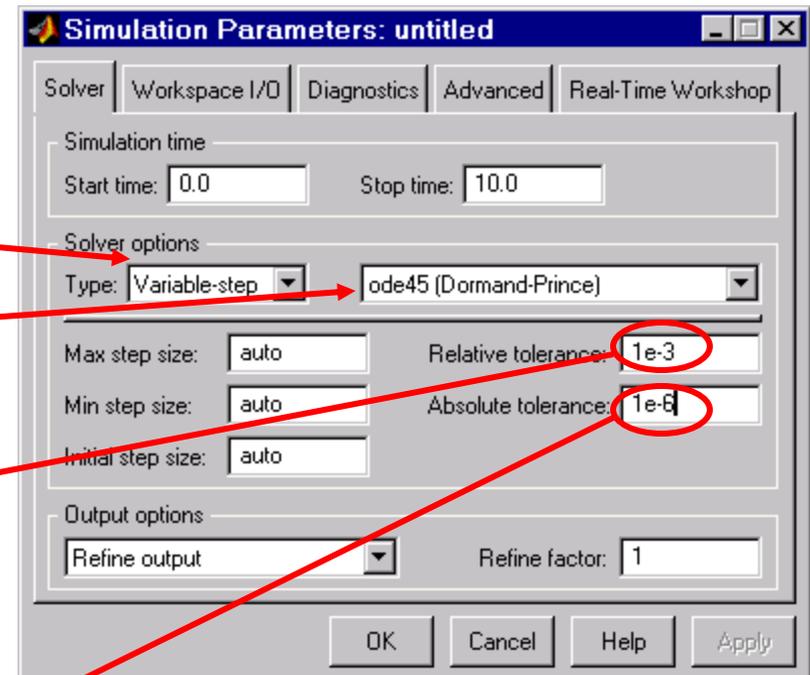
Step Size Calculation (Variable Step)



- 假如錯誤可以接受，模擬將繼續。否則將減小步階再積分。
- 步階的大小調整到適合的大小。

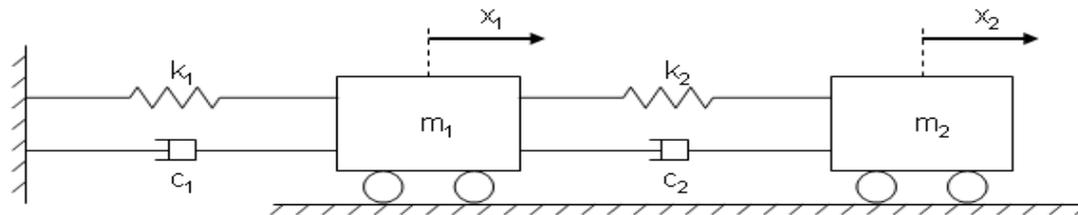
Setting Parameters

- 設定解題器
 - 選擇可變步階做誤差控制
 - 選擇適當的解題器
- 設定相對誤差
$$\frac{\text{error in } x}{x} < 1e-3$$
- 設定絕對誤差
$$\text{error in } x < 1e-6$$



多自由度的動態系統

- EX: 自由度為2的運動系統如下圖

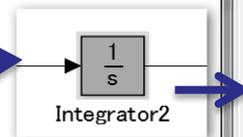
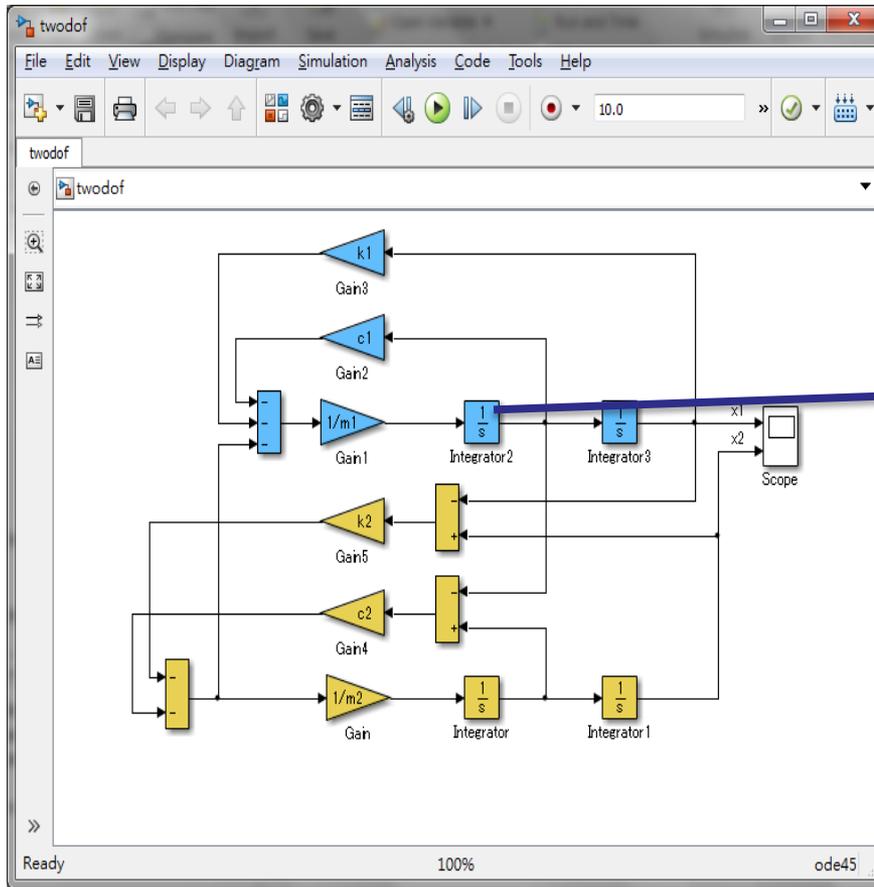


- 其聯立方程式如下

$$\begin{cases} m_1 \ddot{x}_1 + c_1 \dot{x}_1 + k_1 x_1 + m_2 \ddot{x}_2 = 0 \\ m_2 \ddot{x}_2 + c_2 (\dot{x}_2 - \dot{x}_1) + k_2 (x_2 - x_1) = 0 \end{cases}$$

- initial value 為 $x_2(0) = 1, x_1(0) = \dot{x}_1(0) = \dot{x}_2(0) = 0$

自由度為2 的動態系統 Simulink model 如下



Function Block Parameters: Integrator2

Integrator
Continuous-time integration of the input signal.

Parameters

External reset: none

Initial condition source: internal

Initial condition:
0 Initial value

Limit output

Upper saturation limit:
inf

Lower saturation limit:
-inf

Show saturation port

Show state port

OK Cancel Help Apply

非線性系統

- 線性與非線性

假設 x_1, x_2 ，其output 為 y_1, y_2

滿足 input 的線性組合 $\alpha x_1 + x_2$ ，其
output $\alpha y_1 + x y_2$ ， $\forall \alpha \in \mathbb{R}$

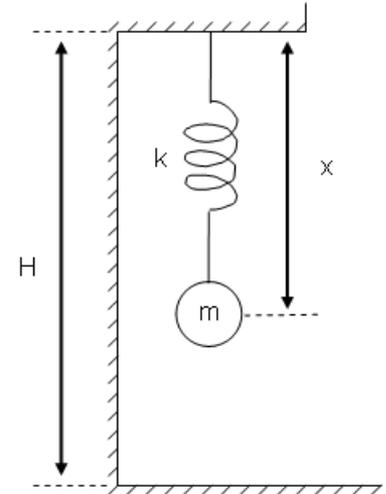
則為線性，反之 則為非線性

Example: 高空彈跳

- 運動方程式如下

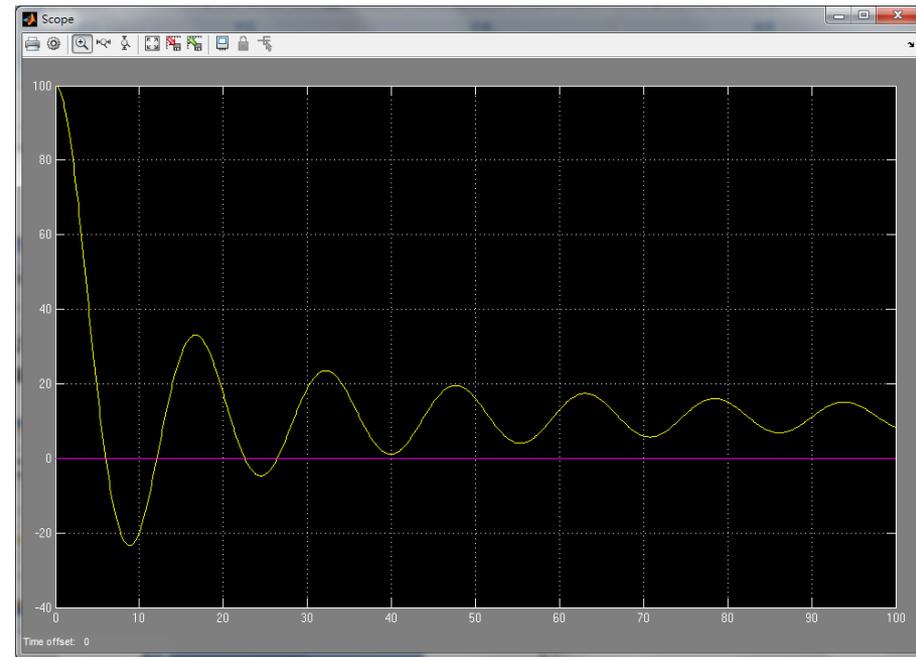
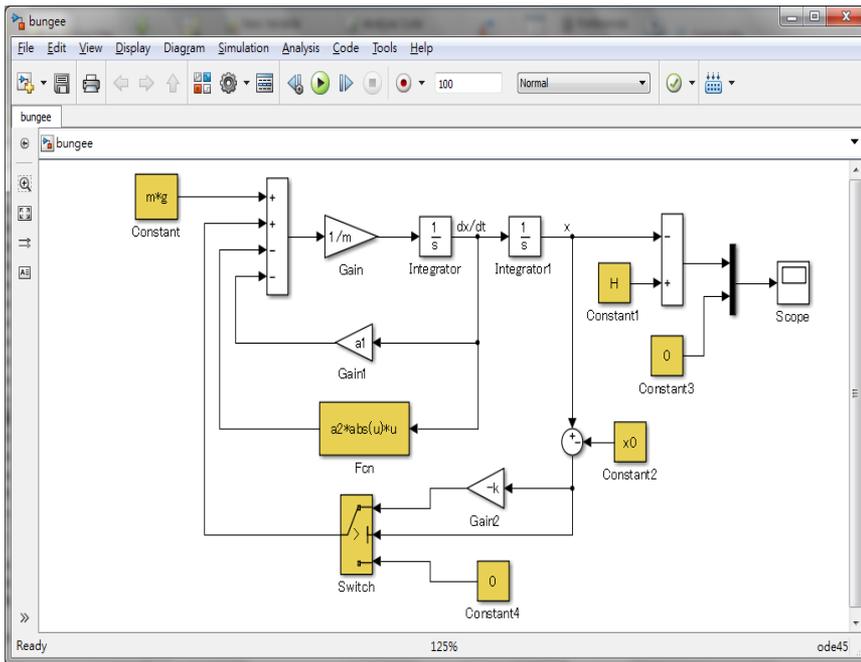
$$m\ddot{x} = mg + b(x) - a_1 \dot{x} - a_2 |\dot{x}|\dot{x}$$

$$b(x) = \begin{cases} -k(x - x_0) & (x > x_0) \\ 0 & (x \leq x_0) \end{cases}$$



- x 是落下距離， x_0 彈簧的自然長， g 重力加速度， m 質量， k 彈性係數， a_1, a_2 空氣阻力係數。
- $b(x)$ 及 $a_2 |\dot{x}|\dot{x}$ 非線性項，所以此運動方程式非線性微分方程式。直接建模如下。

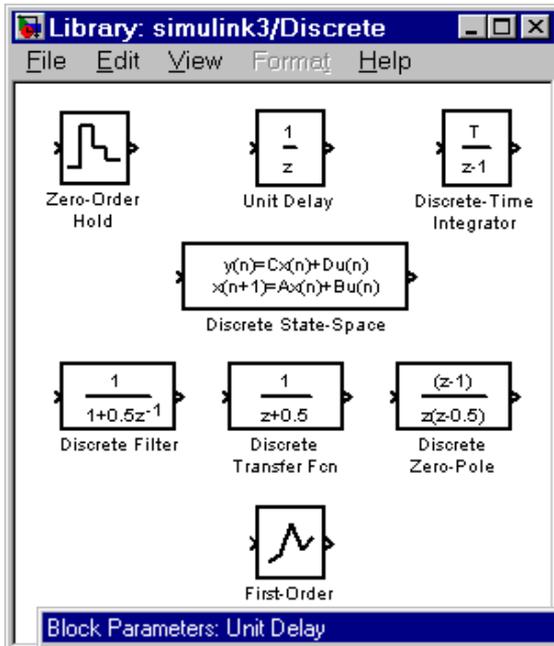
Nonlinear Simulink model 與模擬結果



離散時間系統建構

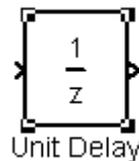
- 固定時間間隔更新系統的狀態
- 輸出跟現在的輸入或前一個輸入或輸出有關
$$y(n)=u(n)+u(n-1)+3y(n-1)$$
$$\text{time}=n \times \text{sampling time}$$
- 離散動態系統: 離散狀態(discrete state)
 - 前一個output 表示一個 狀態(state)
 - 離散狀態相當於存前一個訊號值的記憶體！

離散系統函式庫



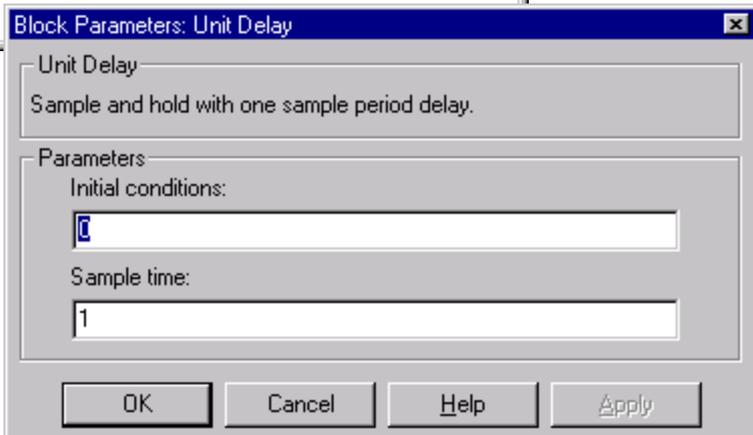
• Zero-Order Hold

- 把連續的訊號轉成離散的訊號
- 設定取樣速率



• Unit Delay

- Input延遲一個單位時間 (sampling time)
- 設定initial value



離散系統:差分方程 (Difference equation) 使用Unit Delay

設計步驟

- 決定block需要幾個delay:
 Ex: $y(n-1)$ 表示一個delay; $y(n-2)$ 表示兩個delay
- 根據方程式 連接各個input 與output blocks
- 設定初始值(initial value).
- 設定取樣時間(sampling time).

線性離散系統

- Z-transform
 - $x(n) \rightarrow X(z)$
 - $x(n-1) \rightarrow z^{-1}X(z)$ (delay one unit)
- $y(n)=u(n)+u(n-1)+3y(n-1)$
 $\rightarrow Y(z)=U(z)+z^{-1}U(z)+3z^{-1}Y(z)$

$$\rightarrow \frac{Y(z)}{U(z)} = \frac{1+z^{-1}}{1-3z^{-1}}$$

Filter representation: num=[1 1], den=[1 -3]

Z-Domain表示式

- 濾波器(filter): num=[n0 n1 n2], den=[d0 d1]

$$\frac{n_0 + n_1 z^{-1} + n_2 z^{-2}}{d_0 + d_1 z^{-1}}$$

- 轉移函數(transfer function): num=[n0 n1 n2], den=[d0 d1 0]

$$\frac{n_0 z^2 + n_1 z^1 + n_2}{d_0 z^2 + d_1 z}$$

- 零點極點Zero-pole: gain=K, zeros=[z1 z2], poles=[0 p1]

$$K \frac{(z - z_1)(z - z_2)}{z(z - p_1)}$$

Moving average filter 5階

- Moving average filter 方程式如下

$$y_n = \frac{1}{5} (u_n + u_{n-1} + u_{n-2} + u_{n-3} + u_{n-4})$$

計算目前的 output 與前四個 output 的平均

- $$Y(z) = \frac{1}{5} (1 + z^{-1} + z^{-2} + z^{-3} + z^{-4}) U(z)$$

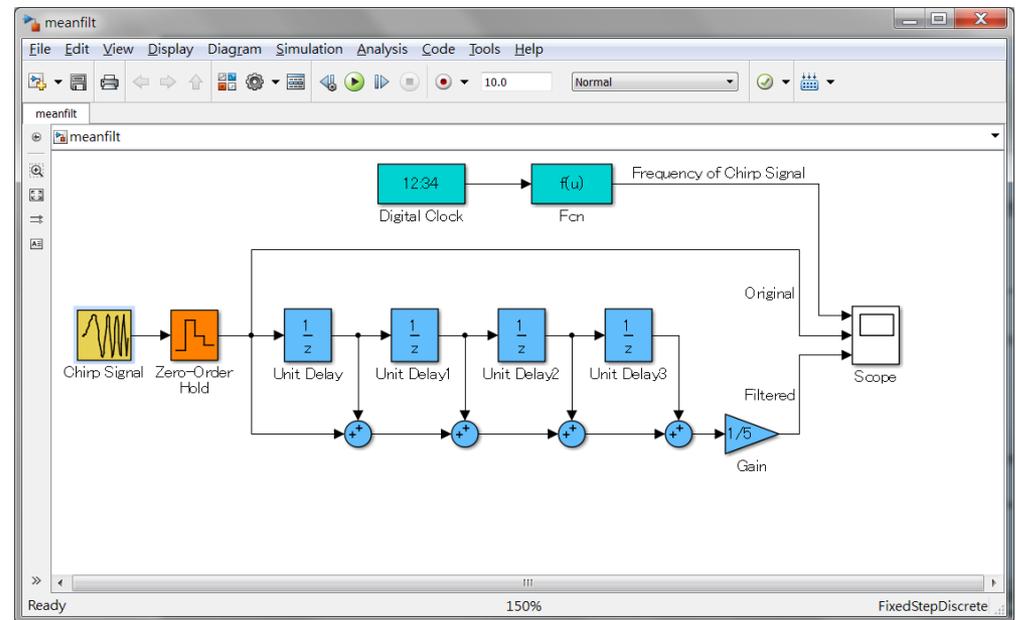
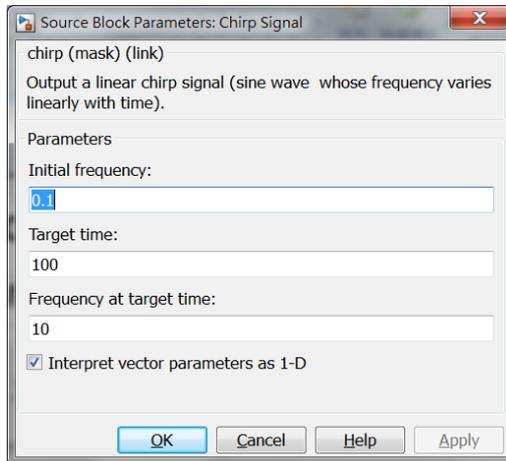
- $$H(z) = \frac{1 + z^{-1} + z^{-2} + z^{-3} + z^{-4}}{5}$$

Moving average filter 5階(續)

- 組成的 Simulink model 如下

Chirp f 設定如下

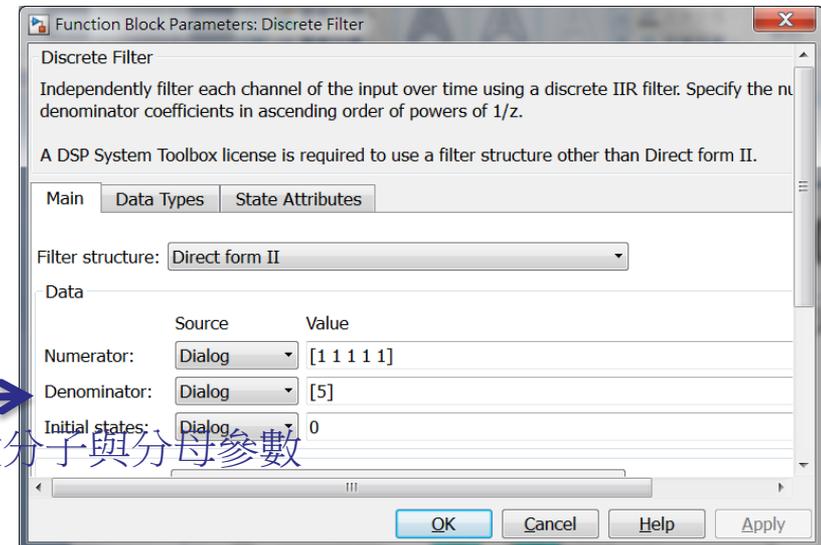
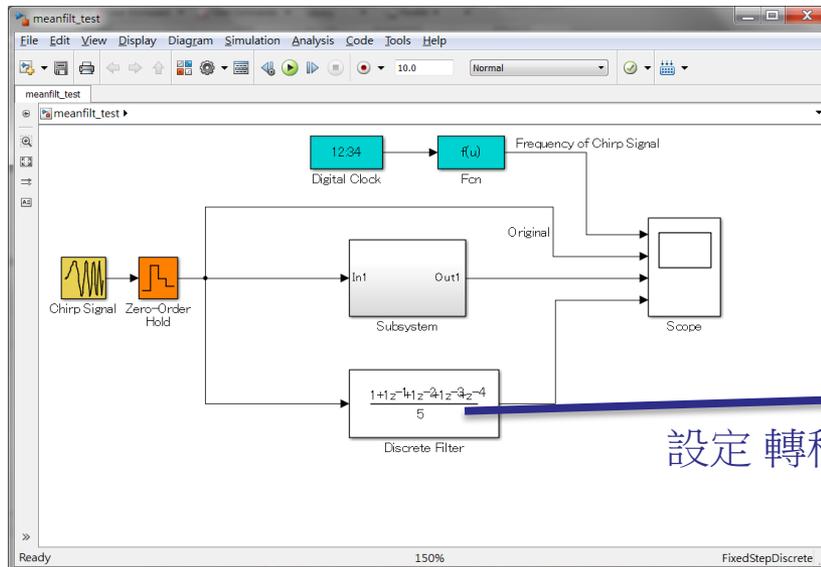
$$f(t) = f_0 + \frac{f_t \arg et - f_0}{T_t \arg et} \times t$$



計算目前的 output 與前四個 output 的平均

Moving average filter 5階(續)

- 可以直接透過discrete filter block 設定轉移函數分子與分母參數



設定轉移函數分子與分母參數

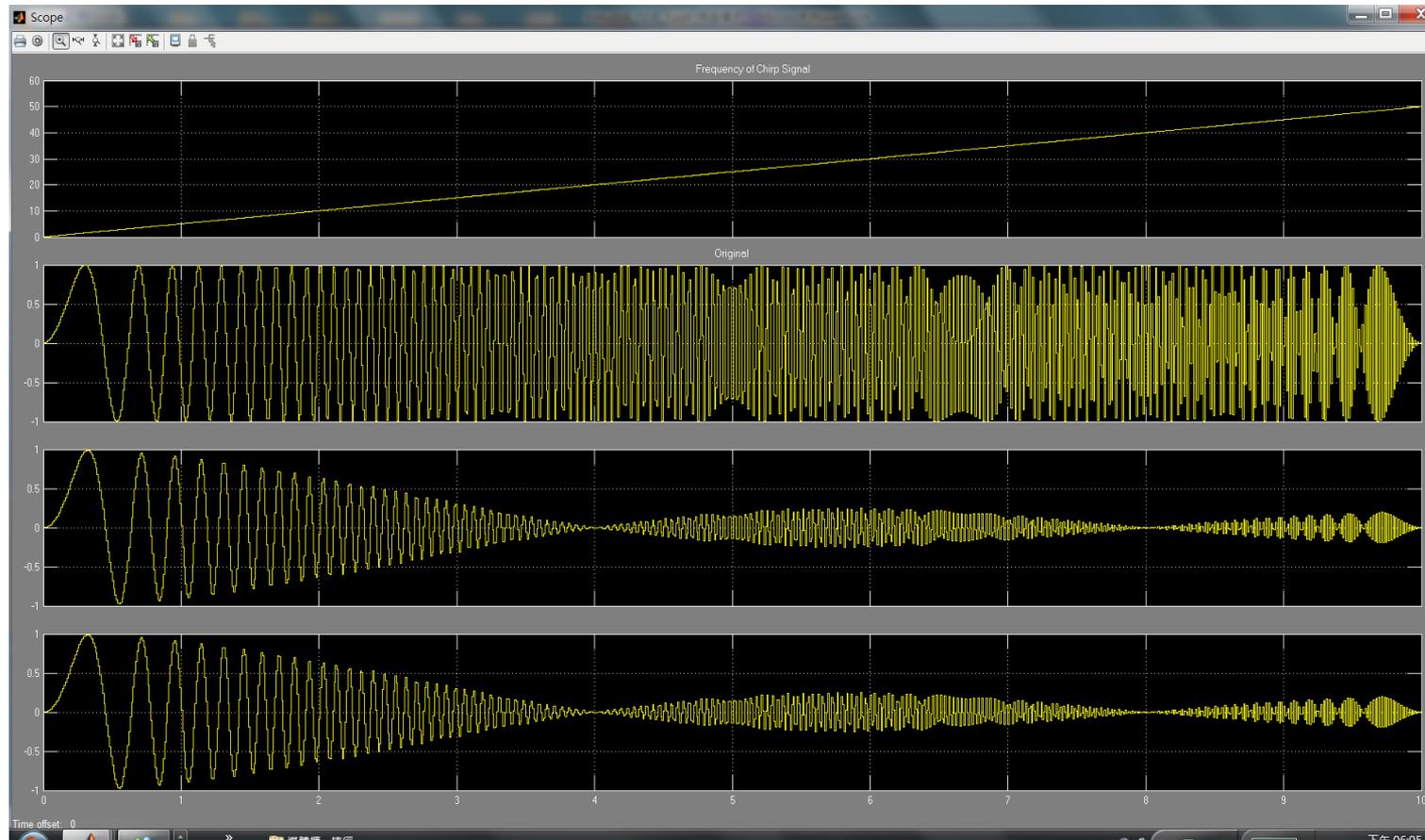
比較模擬結果

Chirp signal
frequency 0.1-
50Hz

Chirp signal
原始訊號

Chirp signal
經過moving
average filter

Chirp signal
經過moving
average filter
使用discrete
filter block



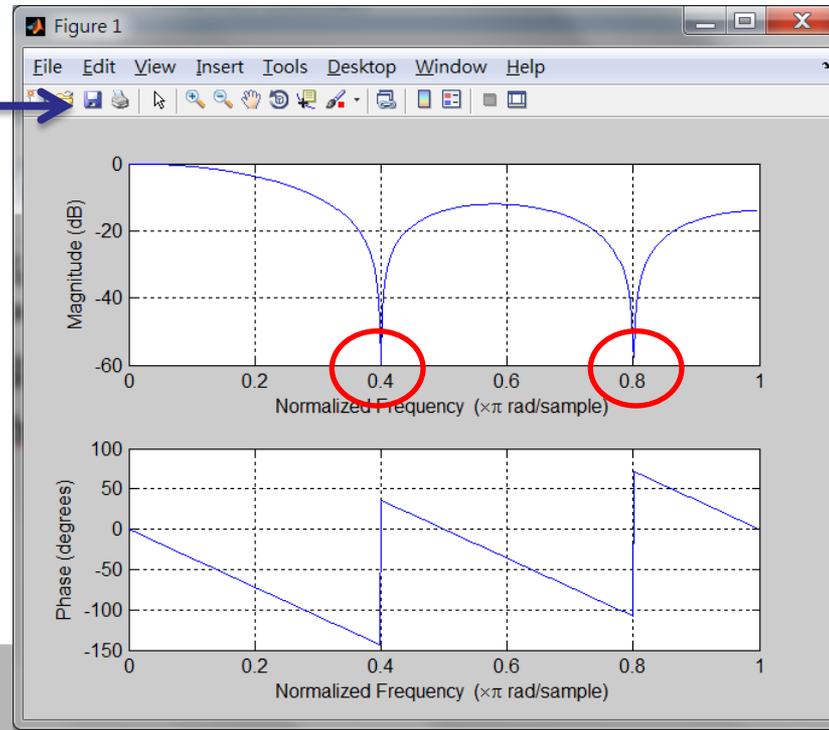
比較模擬結果(續)

- 由結果知:直接建模實現濾波器與用discrete filter block 輸入濾波器分子分母參數 結果是一致的。
- 由圖知 訊號在頻率20 Hz 與 40 Hz 被壓抑。
- 由freqz 指令做頻率響應圖驗證。

```
Command Window  
>> freqz([1 1 1 1 1],5)  
fx >>
```

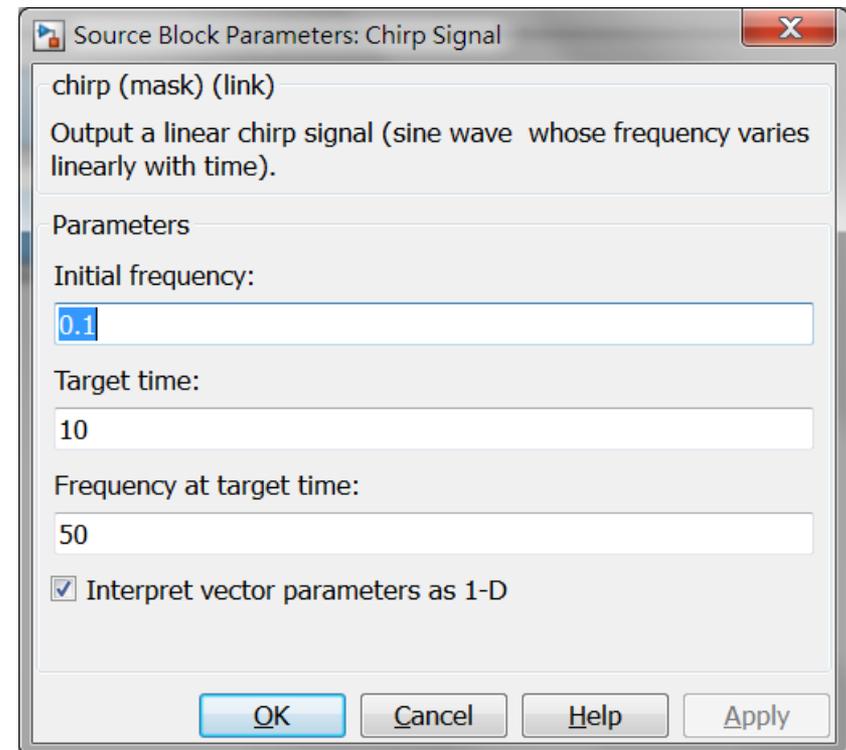
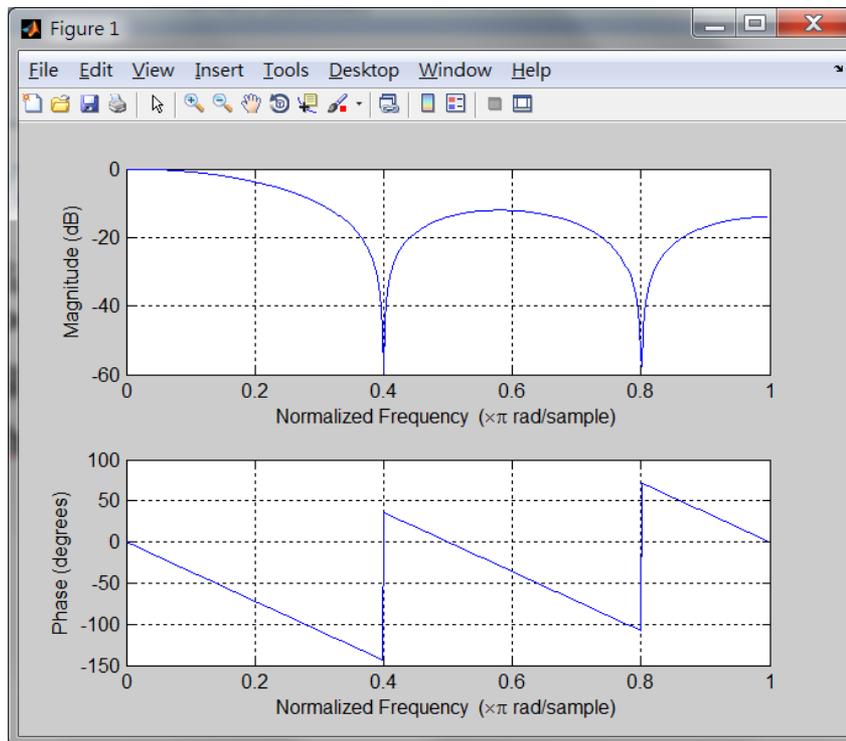
振幅響應

相位響應



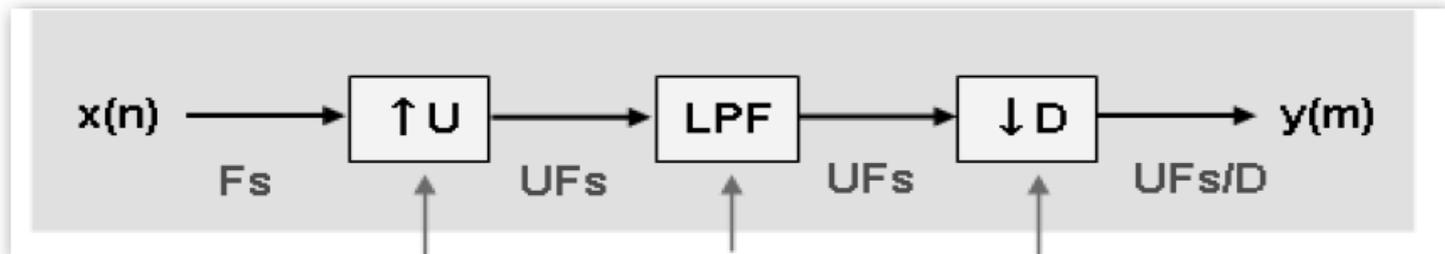
比較模擬結果(續)

- 因為頻率軸經過正規劃，Normalized Frequency 0~1 對應到0~50Hz，所以0.4與0.8對應到20 與40 Hz。



Multi-rate system 模擬

- 不同取樣速率的離散系統：upsampling, downsampling, decimation, interpolation
- LPF(anti-aliasing filter) 加在 Upsampling 之後 downsampling 之前



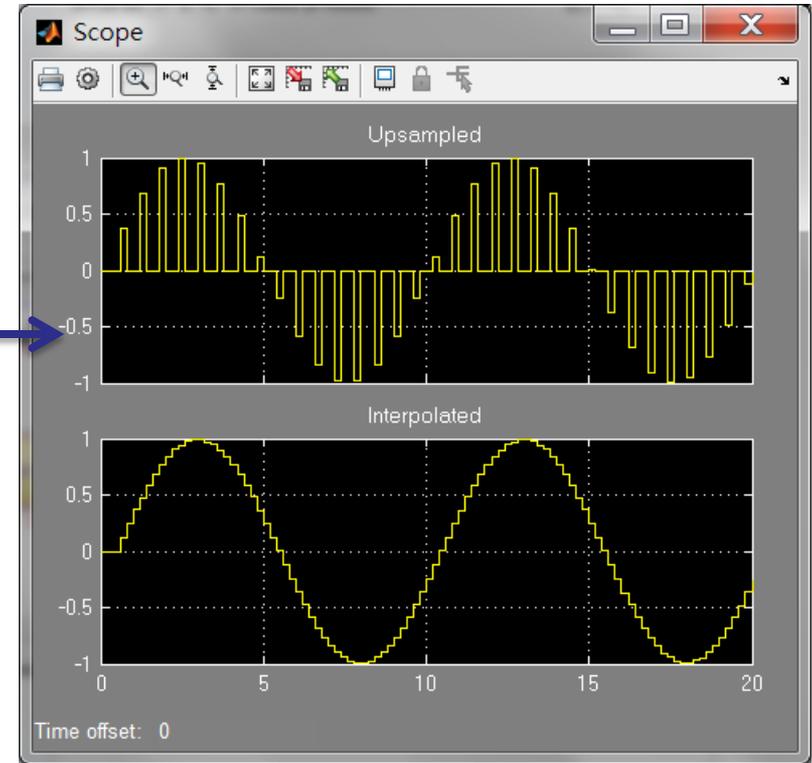
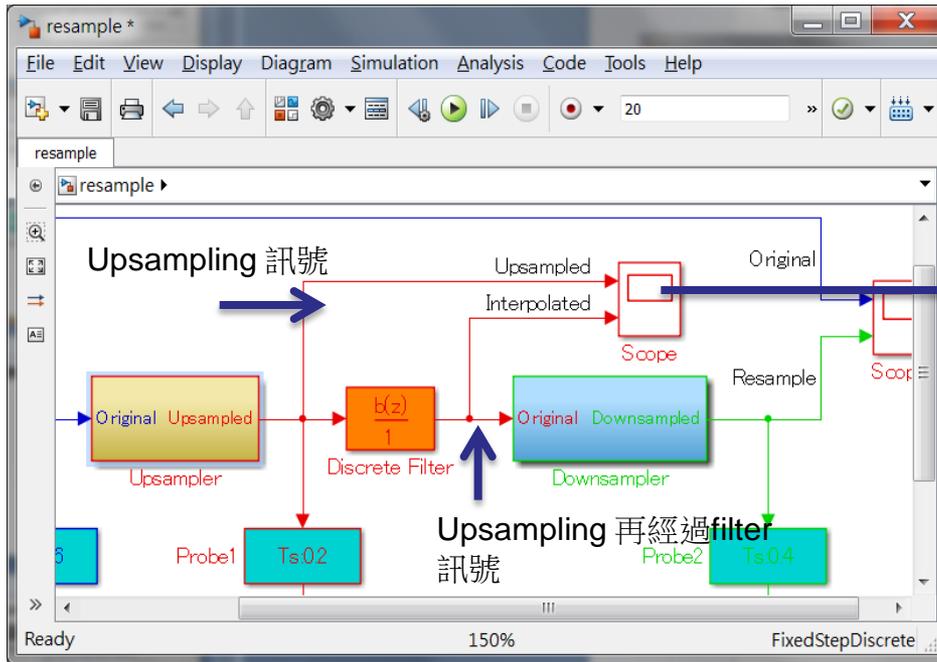
Example : Multi-rate model (up sampling)

The figure illustrates a multi-rate model for up-sampling in MATLAB/Simulink. It consists of several windows:

- resample**: The main Simulink model window showing a block diagram. A Sine Wave block is connected to an Upsampler block (Ts=0.6), which then feeds into a Discrete Filter block (Ts=0.2), and finally a Downsampler block (Ts=0.4). The output is labeled 'Resample' and is connected to a Scope block. Probes and displays are used to monitor the signals at different sampling rates.
- resample/Upsampler ***: A zoomed-in view of the Upsampler block. It shows a Pulse Generator block connected to a Product block. The Product block also receives an input from the 'Original' signal. The output of the Product block is connected to a Scope block and an Upsampled signal output.
- Sample Time Legend**: A window showing the sample times for the 'resample' model. It lists three discrete sample times: Discrete 1 (0.2), Discrete 2 (0.4), and Discrete 3 (0.6), along with a Constant value (Inf).
- Scope**: A window showing three plots of the signals over time (0 to 20 seconds). The top plot is 'Pulse', the middle is 'Original', and the bottom is 'Upsampled'. The 'Upsampled' plot shows a higher frequency signal compared to the 'Original' plot.

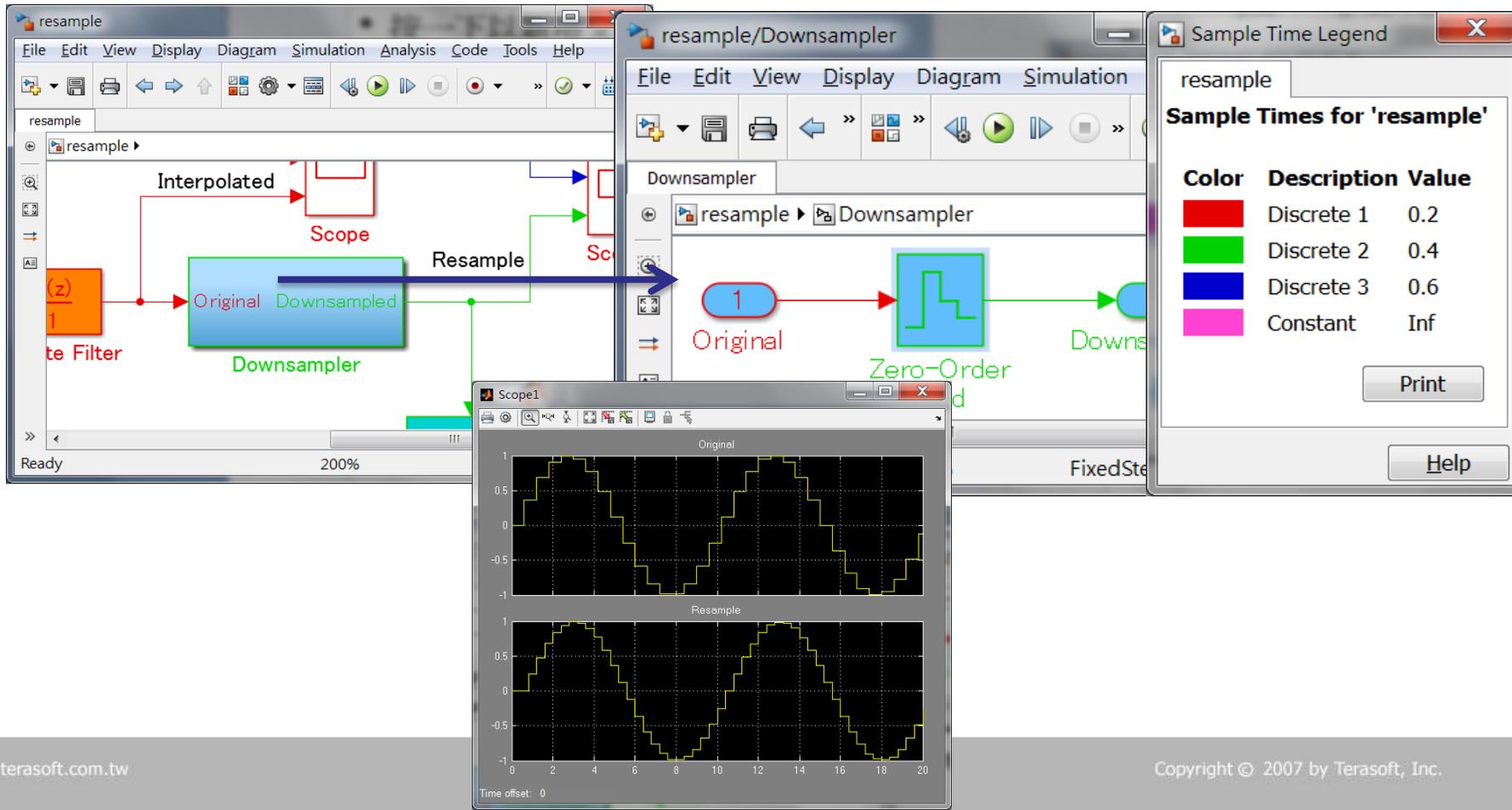
Sampling time從0.6sec 到 0.2sec

觀察經過filter 後的訊號



Example : Multi-rate model (down sampling)

- 取樣時間由0.2秒取樣一次降到0.4秒取樣一次。原始訊號取樣速率0.6秒。Overall 來講最後取樣速率是原本的1.5倍。



Hybrid system的建模

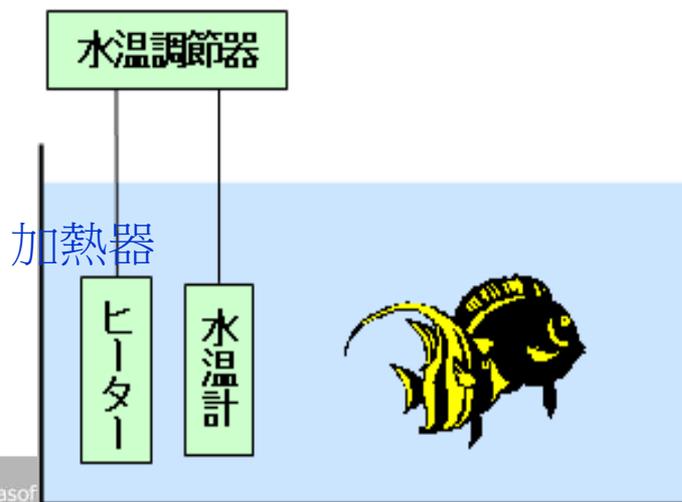
- 連續系統和離散系統混合

Example: 恆溫器的水槽的自動溫度調節

把水溫作為 T [$^{\circ}\text{C}$] 的話，那個時間變化用下面的方程式表示。

$$C \frac{dT}{dt} = Q_H + Q_R$$

一階；須一個積分器



水溫控制系統

- **C** 水槽系全部的熱容 [**kcal/°C**]，**Q_H**、**Q_R** 各表示來自加熱器及外部的單位時間的發熱量 [**kcal/sec**]。其中

$$Q_R = \frac{T_R - T}{R}$$

- **R** 表示單位時間的熱阻的比例係數 [**sec·°C/kcal**]，**T_R** 室溫。關於室溫，考慮日常的溫度變化 **T_R(0) ±4°C** 的正弦波表示。

$$T_R(t) = T_R(0) + 4 \sin\left(\frac{2\pi}{24 \times 3600} t\right)$$

頻率設一天

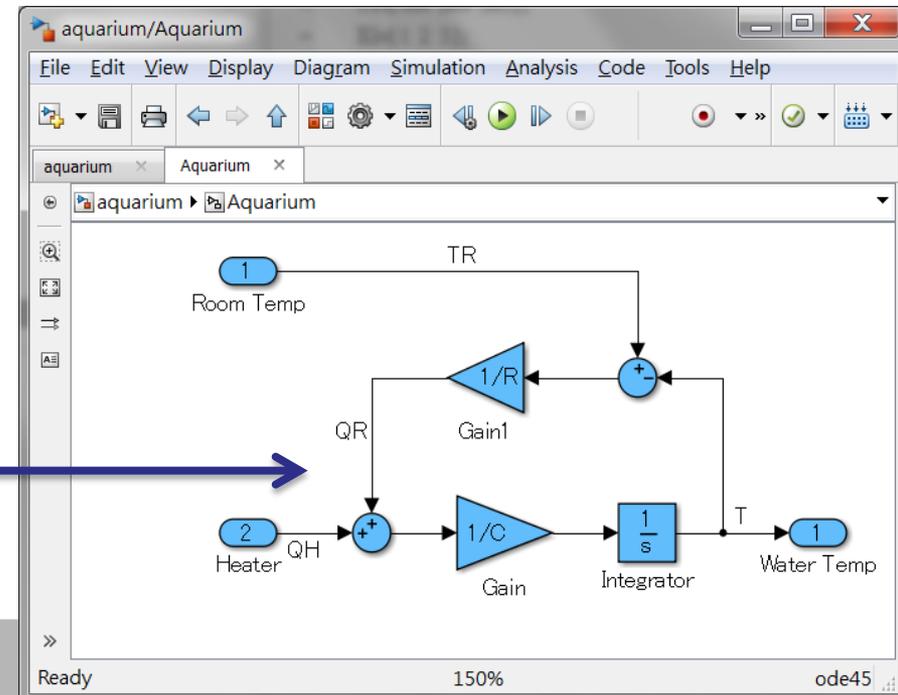
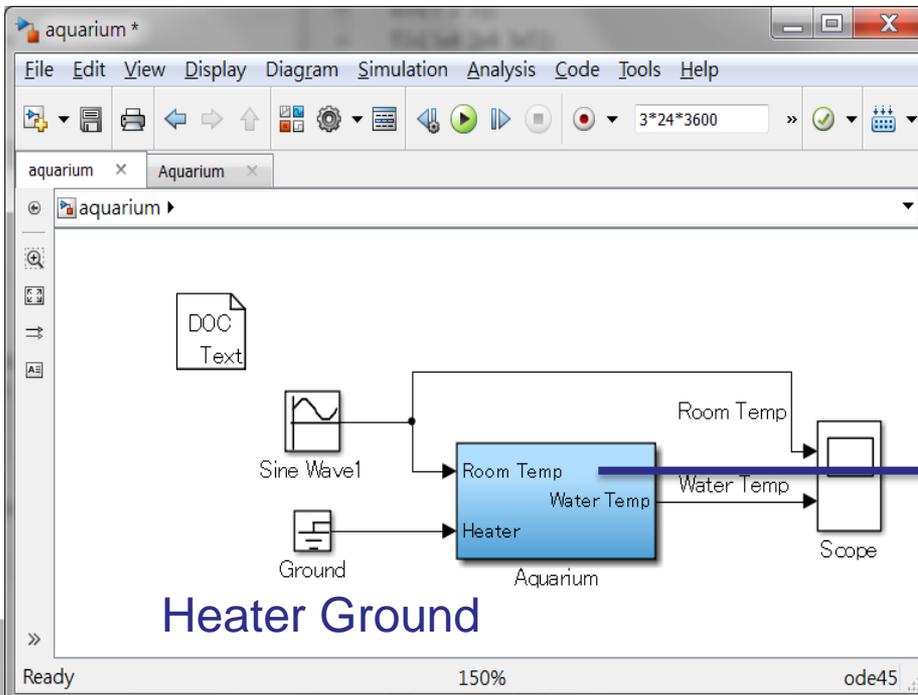
沒有加熱器的model

- 由三個等式導出底下的model

$$C \frac{dT}{dt} = Q_H + Q_R$$

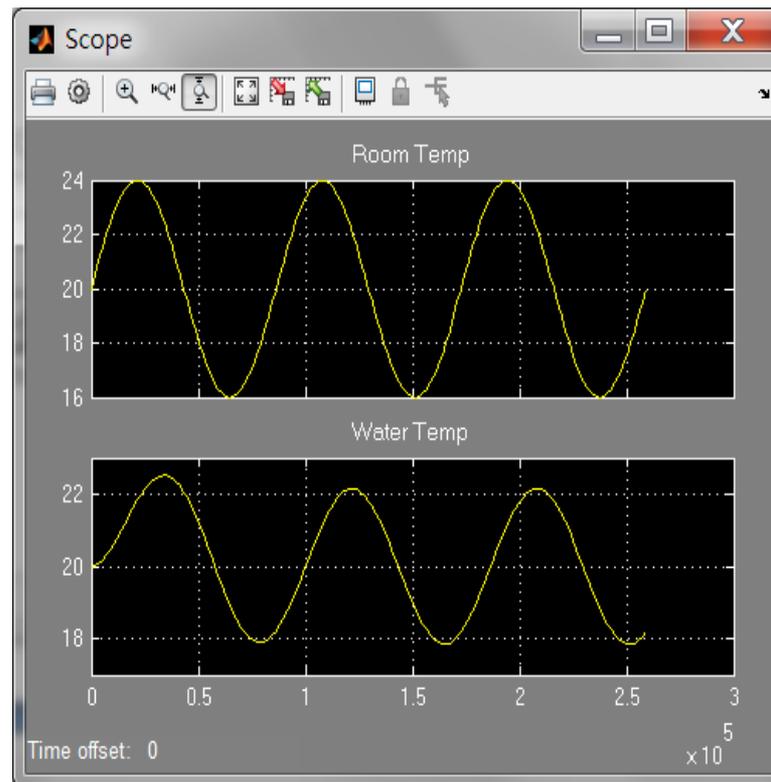
$$Q_R = \frac{T_R - T}{R}$$

$$T_R(t) = T_R(0) + 4 \sin\left(\frac{2\pi}{24 \times 3600} t\right)$$



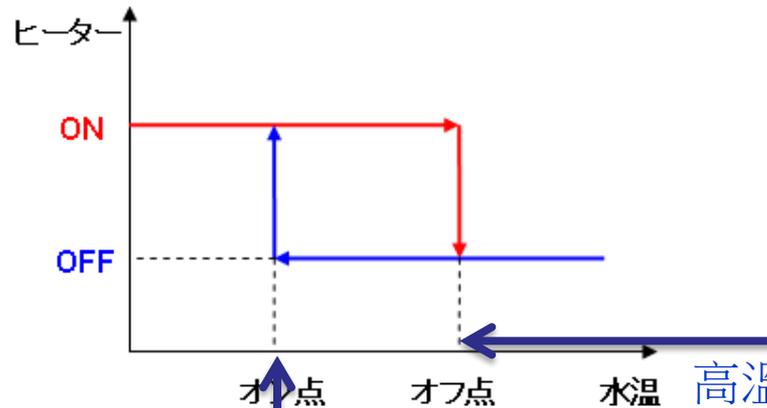
沒有加熱器模擬結果

- Initial 溫度設20度，室溫在20正負4度變化 觀察三天結果如下。



有加熱器的model(band-band)

- 水溫變得低的話打開加熱器的開關，反過來變得高的話切斷控制開關。



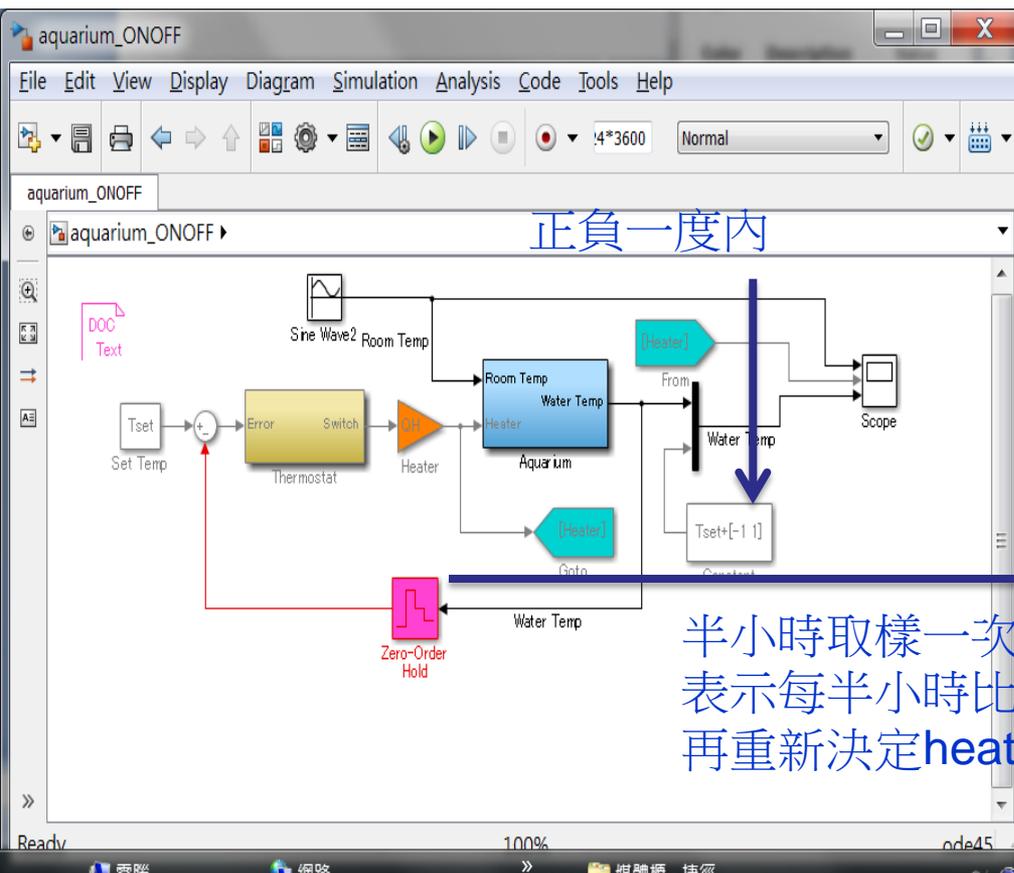
高溫，高於這個溫度須降溫,加熱器 turn off 停止加熱

低溫，低於這個溫度開始加熱,加熱器 turn on

有加熱器的model (續)

- 同樣模擬三天

Hybrid system:
連續+離散



正負一度內

半小時取樣一次
表示每半小時比較一次溫度
再重新決定heater狀態

Sample Time Legend

Color	Description	Value
Black	Continuous	0
Grey	Fixed in Minor Step [0,1]	
Red	Discrete 1	1800
Pink	Constant	Inf

Function Block Parameters

Zero-Order Hold

Zero-order hold.

Parameters

Sample time (-1 for default)

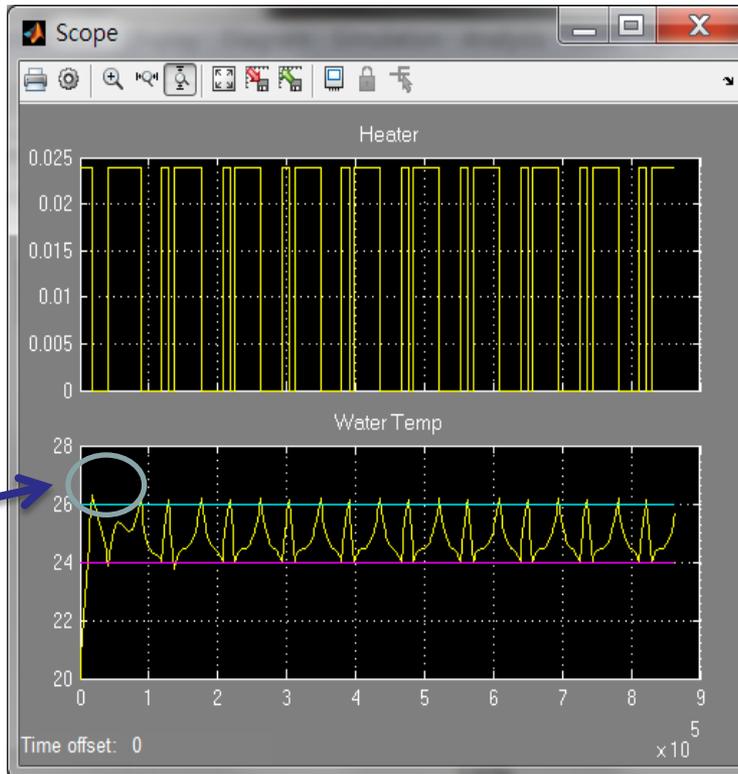
0.5*3600

OK Cancel Help Apply

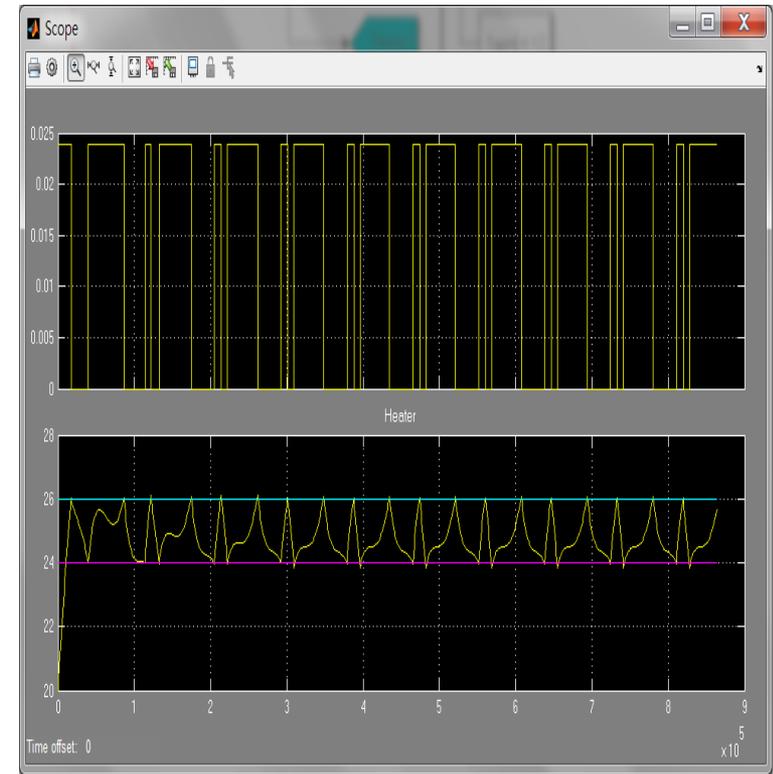
有加熱器模擬結果

High low表示
heater off

溫度超過
spec
可以調整
sampling
time 降低誤
差



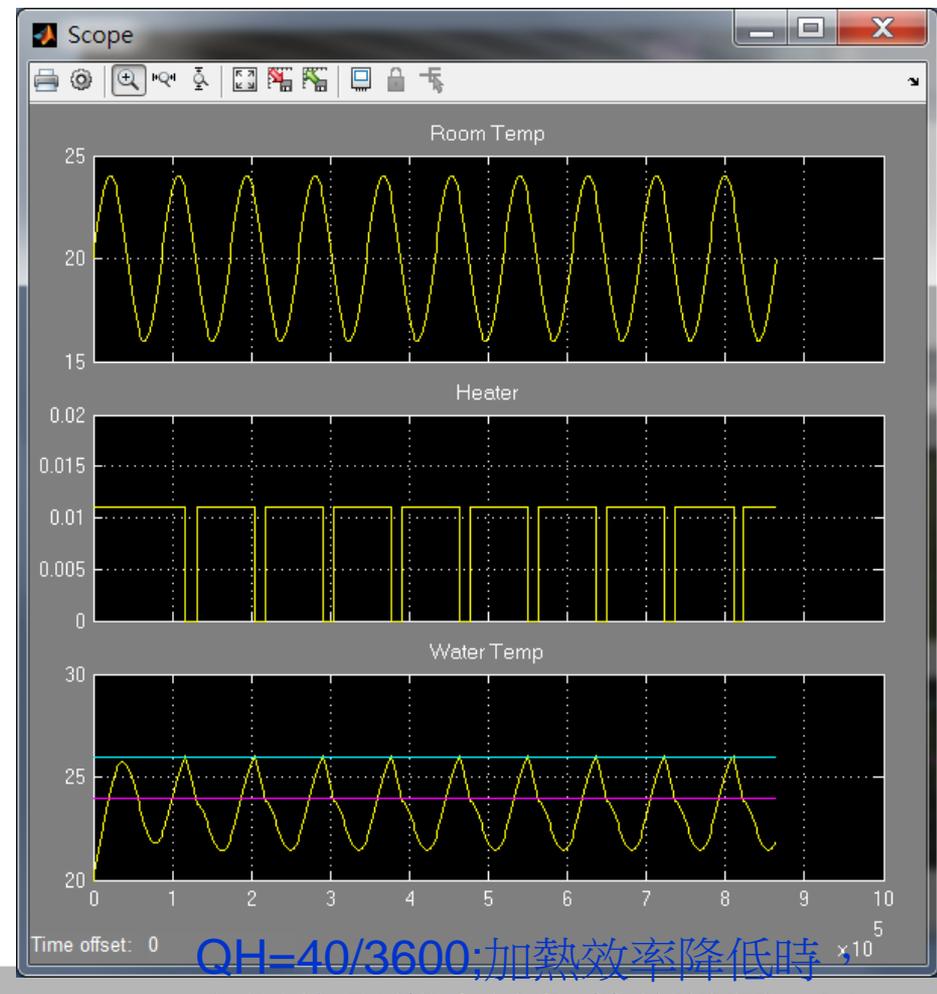
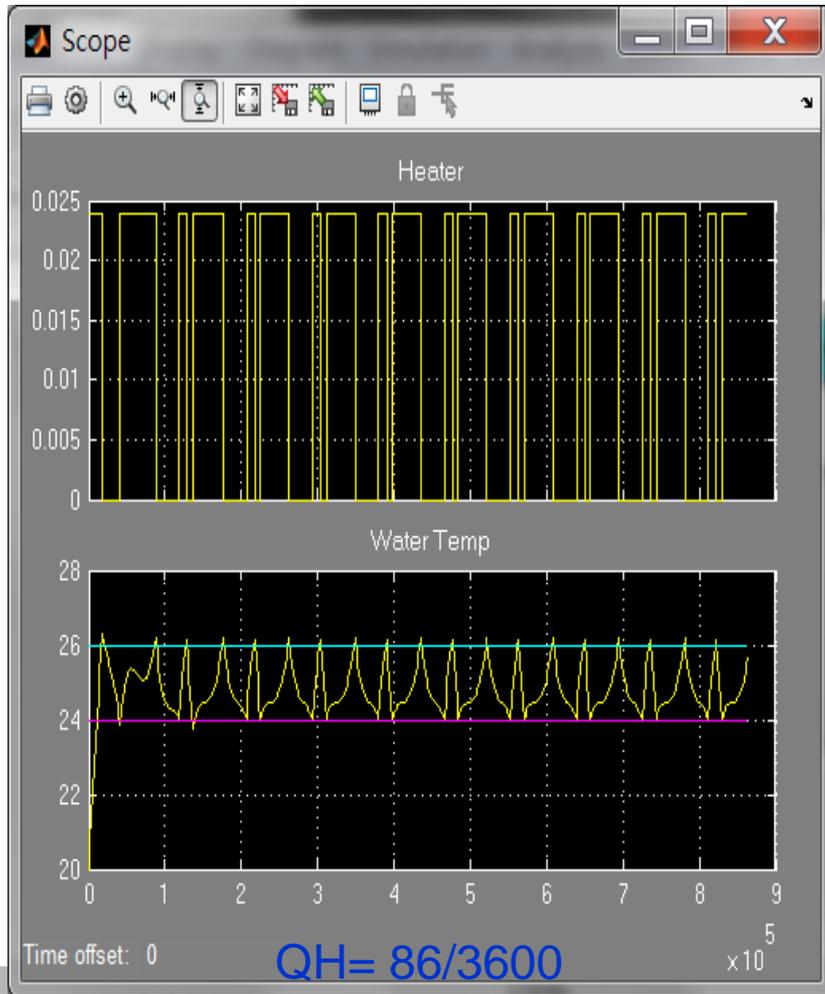
Sampling time
=1800sec



Sampling time
=900sec

其它條件模擬結果

- Q_H 從原本 86/3600 降到 40/3600 。



Q & A

Thank you very much

Have a good time