





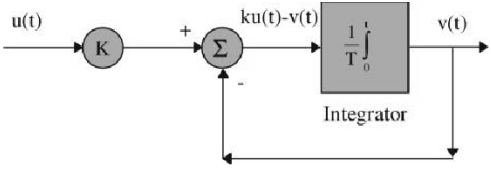
Giới thiệu về Simulink®

- Giới thiệu chung
- Nguyên lý hoạt động và quản lý
- Giải phương trình vi phân
- Đơn giản hóa hệ thống
- Tương tác với MATLAB


85



Giới thiệu chung

- **Simulink®:**
 - Các quá trình phụ thuộc thời gian tuyến tính hay phi tuyến có thể được mô tả bởi các phương trình vi phân.
 - Một cách khác mô tả hệ thống động: thông qua sơ đồ khối



Mô tả một hệ thống động bằng sơ đồ khối

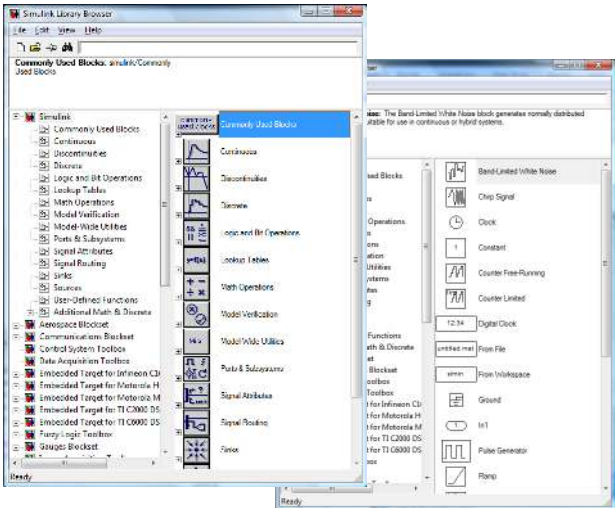
- *Simulink*: bộ giải phương trình vi phân số.


86

Giới thiệu chung

• **Simulink®:**

Thư viện các khối được tổ chức thành các nhóm chức năng



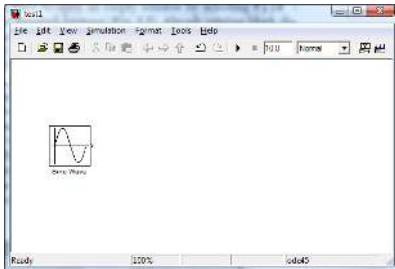
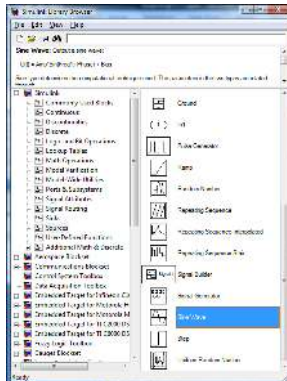
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Posts & Telecommunications Institute of Technology

87

Nguyên lý hoạt động và quản lý

• **Xây dựng một mô hình Simulink®:**

- Mở cửa sổ mô hình mới: **File – New Model**
- Lưu model dưới một tên phù hợp: **File – Save As (mdl=model)**
- Mở model đã tồn tại: **File – Open**
- Ví dụ: model test1.mdl

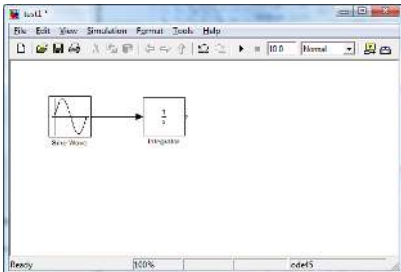
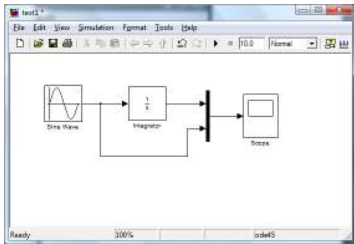



HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Posts & Telecommunications Institute of Technology

88

Nguyên lý hoạt động và quản lý

- Xây dựng một mô hình *Simulink*[®]:
 - Dùng khối *Sine Wave* trong thư viện *Sources* làm nguồn tín hiệu
 - Dùng khối *Integrator* trong thư viện *Continuous* để lấy tích phân
 - Sử dụng khối *Mux* từ thư viện *Signal Routing* để ghép 2 tín hiệu trước và sau khi lấy tích phân.
 - Dùng khối *Scope* từ thư viện *Sinks* để hiển thị kết quả.

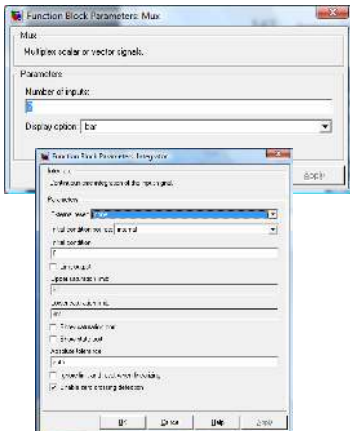
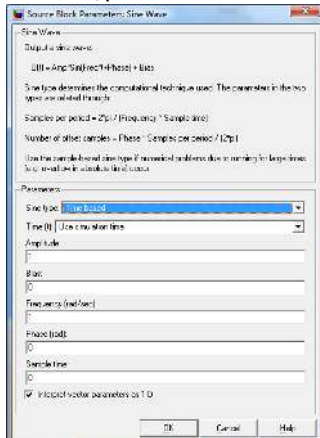
Hoàn thành mô hình ví dụ test1.mdl

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Posts & Telecommunications Institute of Technology

89

Nguyên lý hoạt động và quản lý

- Thiết lập tham số cho các khối *Simulink*[®]:
 - Kích đúp vào mỗi khối chức năng để thiết lập tham số

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Posts & Telecommunications Institute of Technology

90

Nguyên lý hoạt động và quản lý

- Thiết lập tham số cho các khối *Simulink*[®]:
 - Thiết lập tham số tại cửa sổ hiển thị của khối Scope

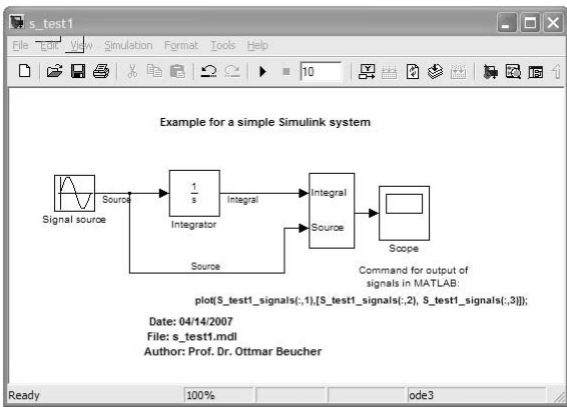


HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Posts & Telecommunications Institute of Technology

91

Nguyên lý hoạt động và quản lý

- Thiết lập tham số cho các khối *Simulink*[®]:



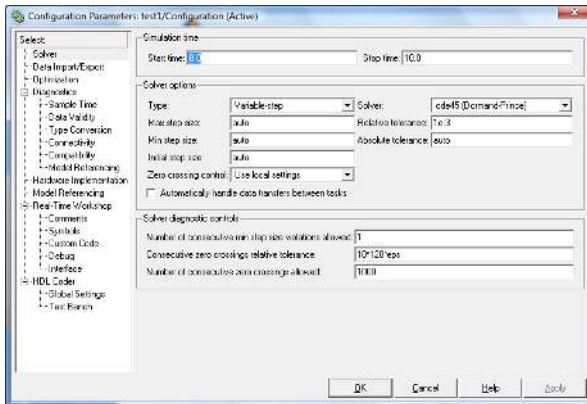
Hệ thống Simulink test1 hoàn thiện

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Posts & Telecommunications Institute of Technology

92

Nguyên lý hoạt động và quản lý

- Chạy mô phỏng *Simulink*[®]:



Thiết lập tham số cấu hình mô phỏng

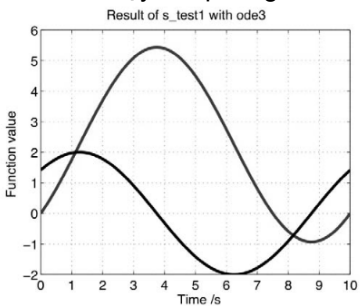
– Sau khi thiết lập xong các tham số → Chạy mô phỏng:
Simulation - start

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Posts & Telecommunications Institute of Technology

93

Nguyên lý hoạt động và quản lý

- Chạy mô phỏng *Simulink*[®]:
 - Các kết quả sau khi chạy mô phỏng có thể xử lý bằng MATLAB

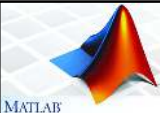


```

>> plot(S_test1_signals(:,1), ...
        [S_test1_signals(:,2),S_test1_signals(:,3)])
>> title('Result of s_test1 with ode3')
>> xlabel('Time /s')
>> ylabel('Function value')
>> grid
  
```

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Posts & Telecommunications Institute of Technology

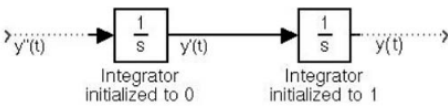
94




Giải phương trình vi phân

- Ví dụ đơn giản:
 - Giải các phương trình vi phân phi tuyến phức tạp đơn giản hơn
 - Chuyển đổi ptr. vi phân thành một hệ thống động → mô tả bằng sơ đồ khối trong *Simulink*®.
 - Bài toán ptr. vi phân bậc 2: $\ddot{y}(t) = -y(t)$, $y(0) = 1, \dot{y}(0) = 0$.

Nghiệm của ptr. : $y(t) = \cos(t)$.



Kỹ thuật lấy tích phân để tìm $y(t)$



HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Posts & Telecommunications Institute of Technology

95



Giải phương trình vi phân

- Ví dụ đơn giản:



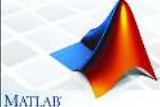
Mô hình giải ptr. vi phân bằng *Simulink*®

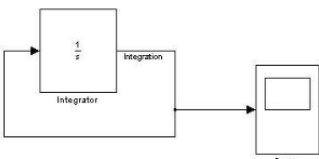
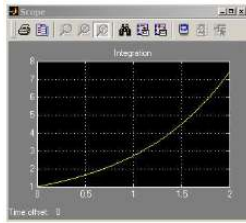
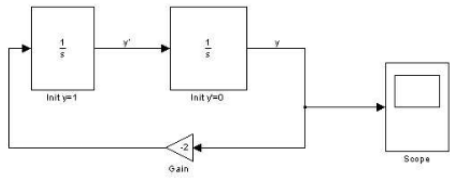
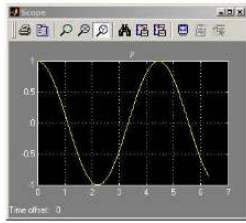




HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Posts & Telecommunications Institute of Technology

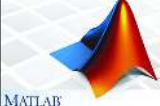
96

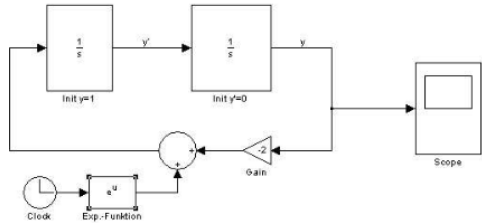
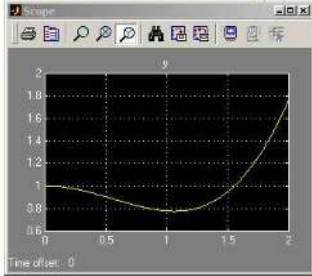
 **Giải phương trình vi phân**

- Một số ví dụ khác:
 - Ptr. vi phân: $y' = y$ $y(0) = 1$


 - Ptr. vi phân: $y'' = -2y$ $y(0) = 1, \quad y'(0) = 0$



HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Posts & Telecommunications Institute of Technology

97

 **Giải phương trình vi phân**

- Một số ví dụ khác:
 - Ptr. vi phân: $y'' = e^x - 2y$ $y(0) = 1, \quad y'(0) = 0$



HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Posts & Telecommunications Institute of Technology

98

Đơn giản hóa hệ thống
Simulink

- Khối *Fcn*:
 - Hệ thống *Simulink* có thể được đơn giản hóa đáng kể bằng việc sử dụng khối *Fcn* trong thư viện *User-Defined Functions* → các khối phân tử mức thấp nhất (VD: khối *Sum* hoặc *Gain*) được loại bỏ.

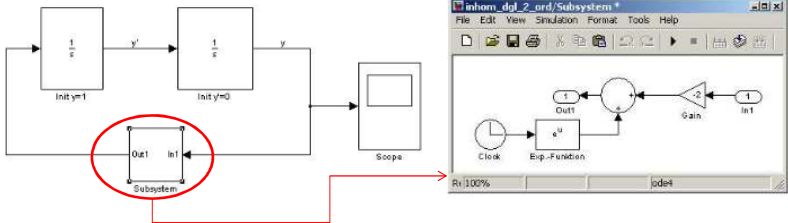


HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Posts & Telecommunications Institute of Technology

99

Đơn giản hóa hệ thống
Simulink

- Xây dựng các phân hệ (*Subsystem*):
 - Trong trường hợp bài toán cỡ lớn → module hóa hệ thống
 - Một số các khối có thể được kết hợp lại thành một phân hệ
 - Cho phép tổ chức hệ thống *Simulink*® dạng phân cấp
 - Tạo một subsystem: sử dụng menu *Edit/Create Subsystem*



HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Posts & Telecommunications Institute of Technology

100



Tương tác với MATLAB

- Truyền các biến giữa *Simulink*[®] và *MATLAB*:
 - Có nhiều cách thực hiện trong *Simulink*[®]:
 - Sử dụng khối *To Workspace* trong *Sinks*
 - Trong thiết lập tham số cấu hình *Simulink*[®]: vào các biến phù hợp trong khối tham số ở mục *Data Import/Export – Save to Workspace*.
 - Sử dụng phần thiết lập tham số trong khối *Scope/ Data History*.
 - Giá trị các biến sử dụng trong chương trình *Simulink*[®] có thể được định nghĩa trong *MATLAB* bằng việc viết một chương trình m-file → vào tên biến thay cho giá trị cụ thể trong phần thiết lập tham số cho các khối.

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Posts & Telecommunications Institute of Technology

101

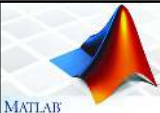


Tương tác với MATLAB

- Lặp lại các mô phỏng *Simulink*[®] trong *MATLAB*:
 - Khi mô phỏng *Simulink*[®] phụ thuộc vào số lượng lớn các tham số → thiết lập sự phụ thuộc hệ thống mô phỏng vào các tham số.
 - Gọi chương trình *Simulink*[®] thông qua *MATLAB*:
 - Sử dụng hàm `sim` `[T,X,Y] = SIM('model',TIMESPAN,OPTIONS,UT)`
 - Ví dụ: `[t,x,y] = sim('system', [starttime, endtime]);`
 - Thiết lập các tham số *Simulink*[®]:
 - Sử dụng hàm `set_param` `SET_PARAM('OBJ','PARAMETER1',VALUE1,'PARAMETER2',VALUE2,...)`
 - Ví dụ: `set_param('vdp','Solver','ode15s','StopTime','3000')`
 - Xem các tham số *Simulink*[®]:
 - Sử dụng hàm `get_param` `GET_PARAM('OBJ','PARAMETER')`
 - Ví dụ: `currentGain = get_param('s_denon3/Factorc','Gain')`

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
Posts & Telecommunications Institute of Technology

102




Tương tác với MATLAB

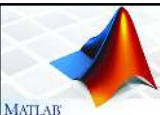
- Lặp lại các mô phỏng *Simulink*[®] trong *MATLAB*
 - Gọi chương trình *Simulink*[®] nhiều lần:


```
for i=1:iterations
    % Set the block parameters with set_param
    % for each new iteration.
    reciprocalmass = num2str(1/M(i));
    set_param('s_denon3/Invm','Gain',reciprocalmass);
    b = num2str(B(i));
    set_param('s_denon3/Factorb','Gain',b);
    c = num2str(Fc);
    set_param('s_denon3/Factorc','Gain',c);
    [t,x,y] = sim('s_denon3', [0,tm]);
    Y = [Y,y];
end;
```
- Truyền các biến thông qua biến toàn cục:
 - Khai báo tham số như biến toàn cục:


```
global x y z p1 a2 ...
```



103




Giới thiệu về *Simulink*[®]

- Bài tập:
 - Thiết kế một hệ thống *Simulink* để giải bài toán giá trị ban đầu:

$$\ddot{y}(t) + y(t) = 0, \quad y(0) = 1, \dot{y}(0) = 0.$$
 Chạy và so sánh kết quả với nghiệm chính xác.
 Biến đổi hệ thống để mô phỏng một hàm nhiễu loạn (vế bên phải $\neq 0$) được xác định là e^{-t} .
 - Giải bài toán giá trị ban đầu: $t\ddot{y}(t) + 2\dot{y}(t) + 4y(t) = 4, \quad y(1) = 1, \dot{y}(1) = 1$ bằng một hệ thống *Simulink* phù hợp.
 - Giải hệ phương trình vi phân sau:

$$\begin{aligned} \dot{y}_1(t) &= -3y_1(t) - 2y_2(t), & y_1(0) &= 1, \\ \dot{y}_2(t) &= 4y_1(t) + 2y_2(t), & y_2(0) &= 1 \end{aligned}$$
 bằng một hệ thống *Simulink* phù hợp.



104