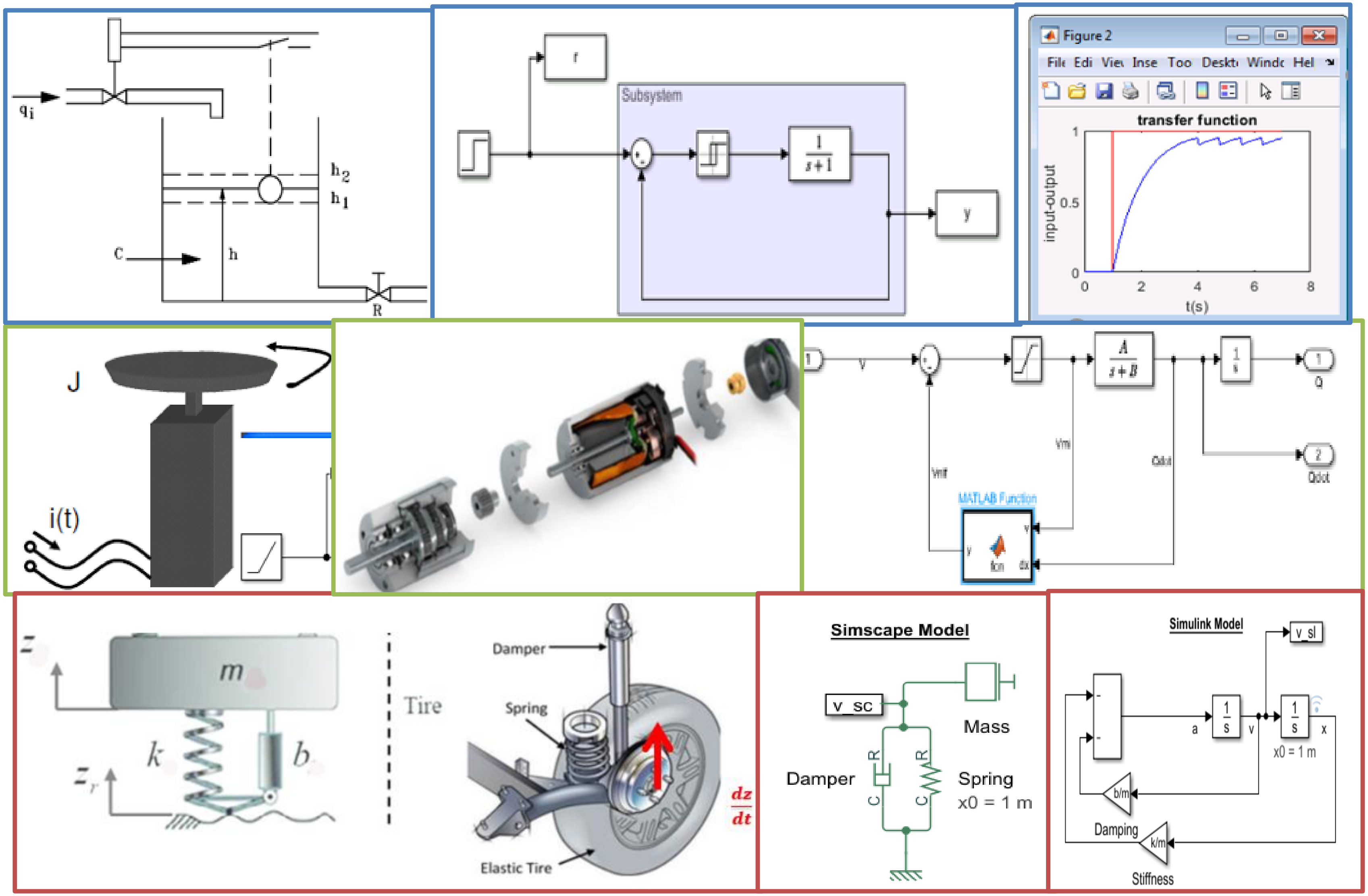
System Dynamics, Modeling and Simulation	**KOM5107**

Modeling & Simulation of Dynamic Systems (MSDS)

Simulation and Model Based Design

Introduction to System Modeling with MATLAB and Simulink and the Use of the Live Editor



Asst. Prof. Claudia F. YAŞAR YTU Control and
Automation Engineering Department-2024

In this course on system modeling with MATLAB and Simulink, you'll learn the basics of Simulink and Simscape, two powerful tools for simulating dynamic systems. We'll cover everything from connecting blocks to modeling complex mechanical systems, giving you practical experience in system analysis. Let's dive in and explore together!

MATLAB SIMULINK ® - Simulation and Model Based Design

http://www.mathworks.com

Asst. Prof. Claudia F. YAŞAR YTU Control and Automation Engineering Department-2024

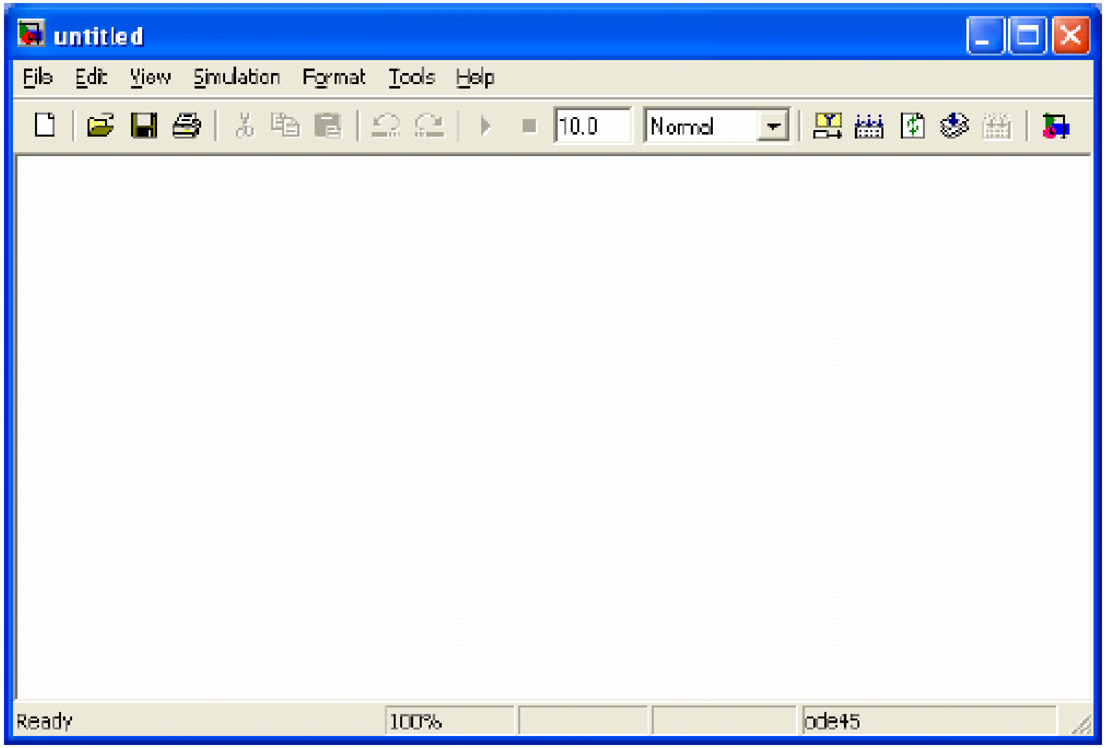
What is Simulink good for?

-Modeling/designing dynamic systems (including nonlinear dynamics)

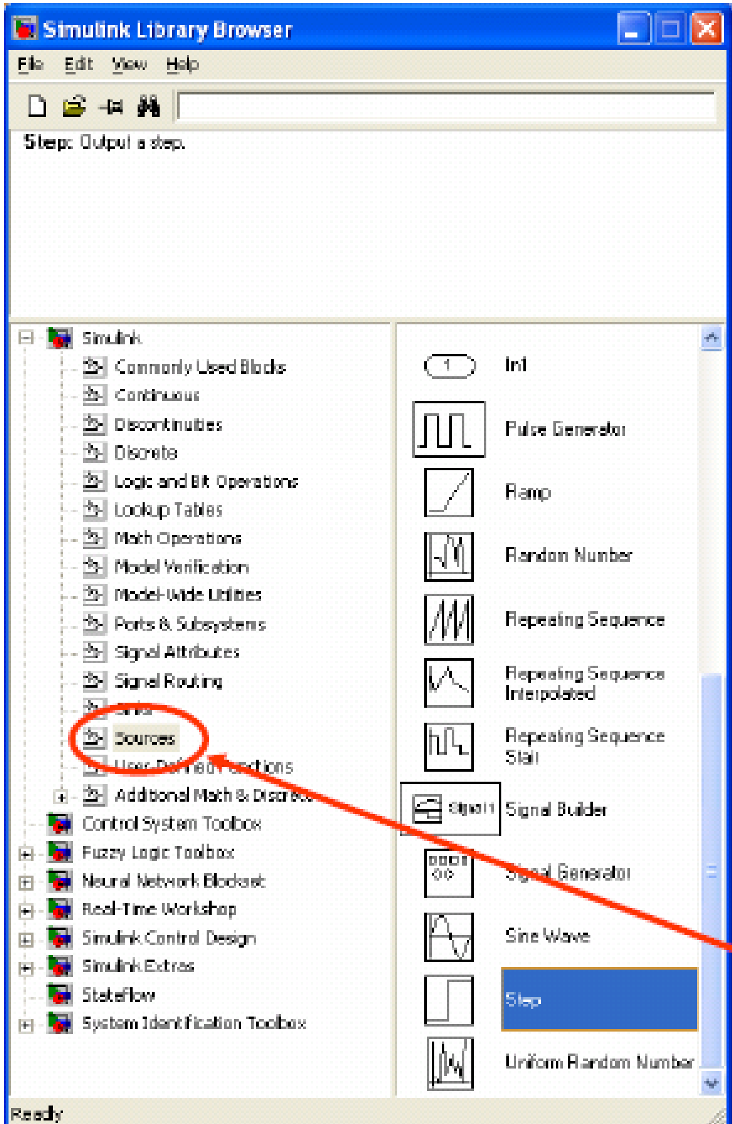
-Modeling/designing control systems (including nonlinear controllers and plants)

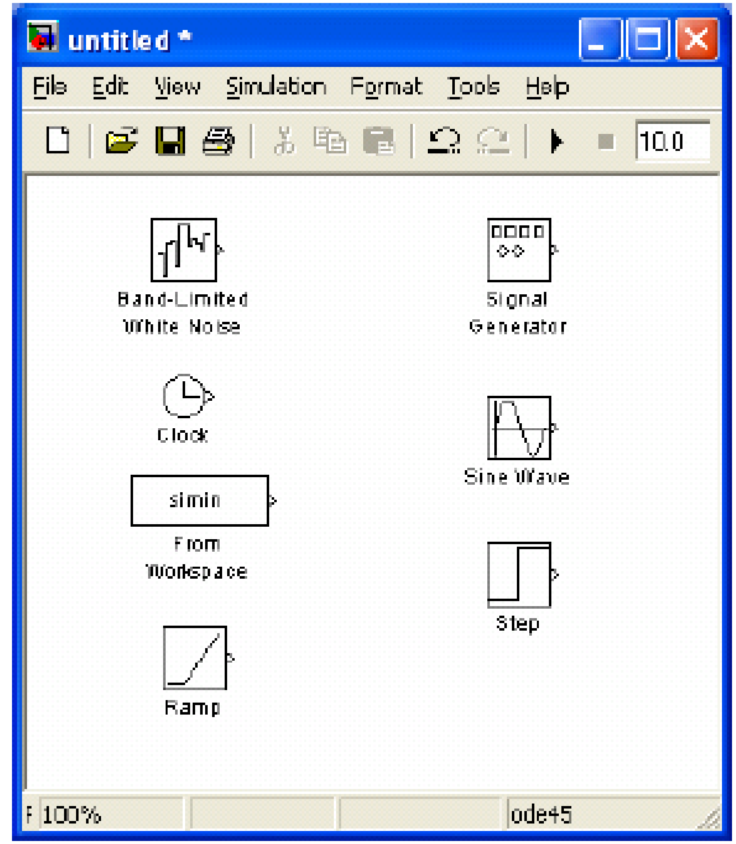
-Signal processing design/simulation

A Simulink model is a block diagram. Click "File|New|Model" in the Library Browser. An empty block diagram will pop up. You can drag blocks into the diagram from the library.



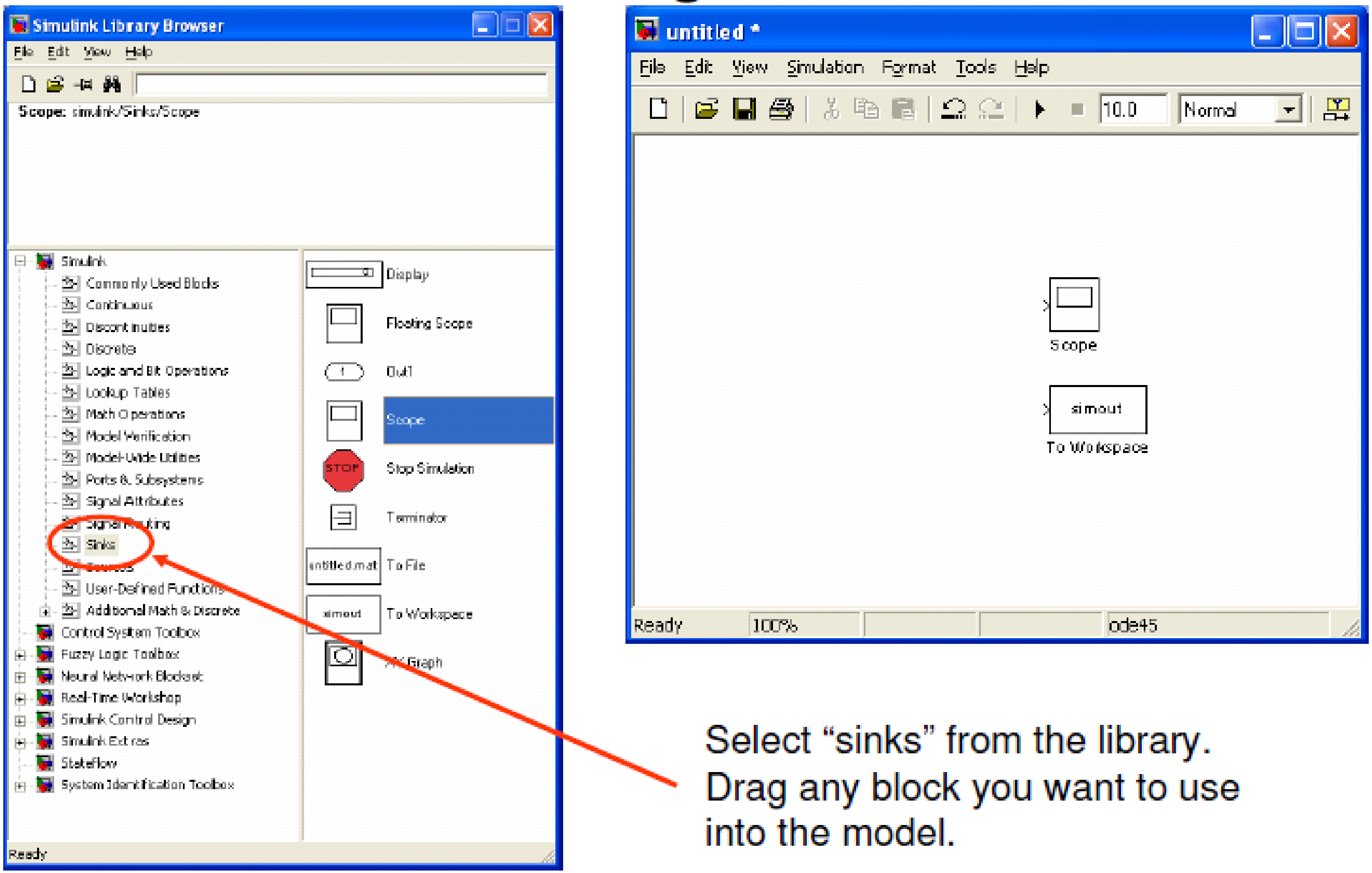
Sources: Produce Signals





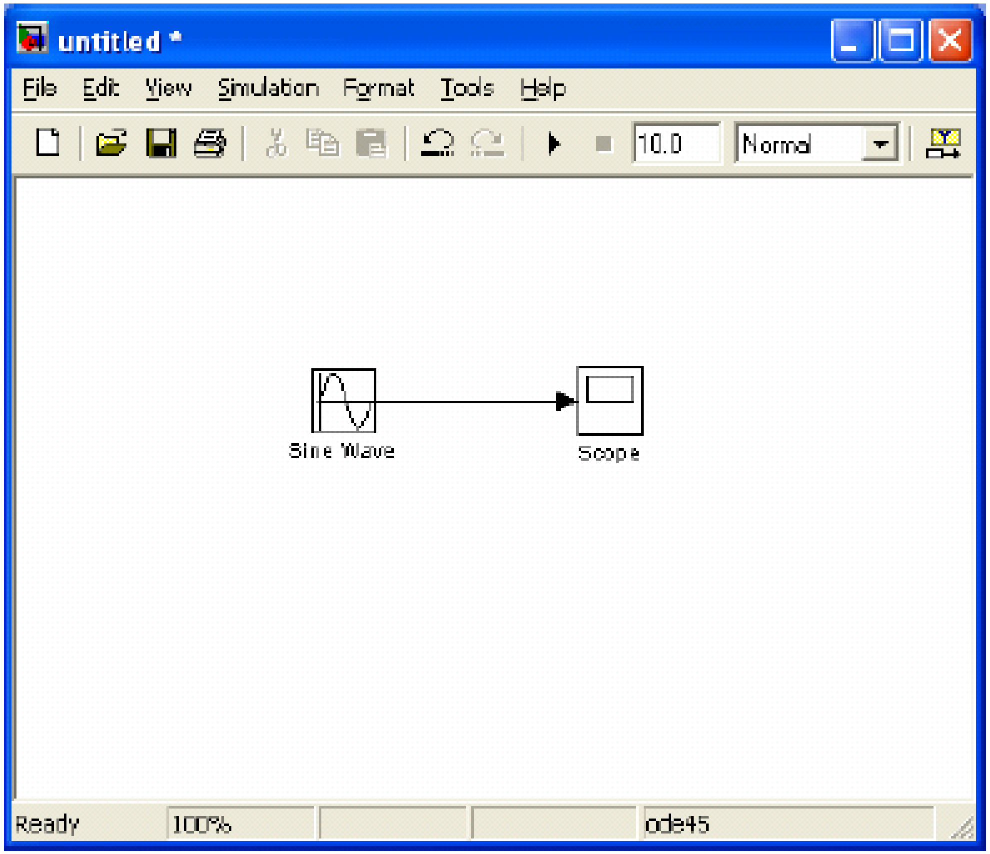
Select “sources” from the library. Drag any block you want to use into the model.

Sinks: Terminate Signals



Select “sinks” from the library. Drag any block you want to use into the model.

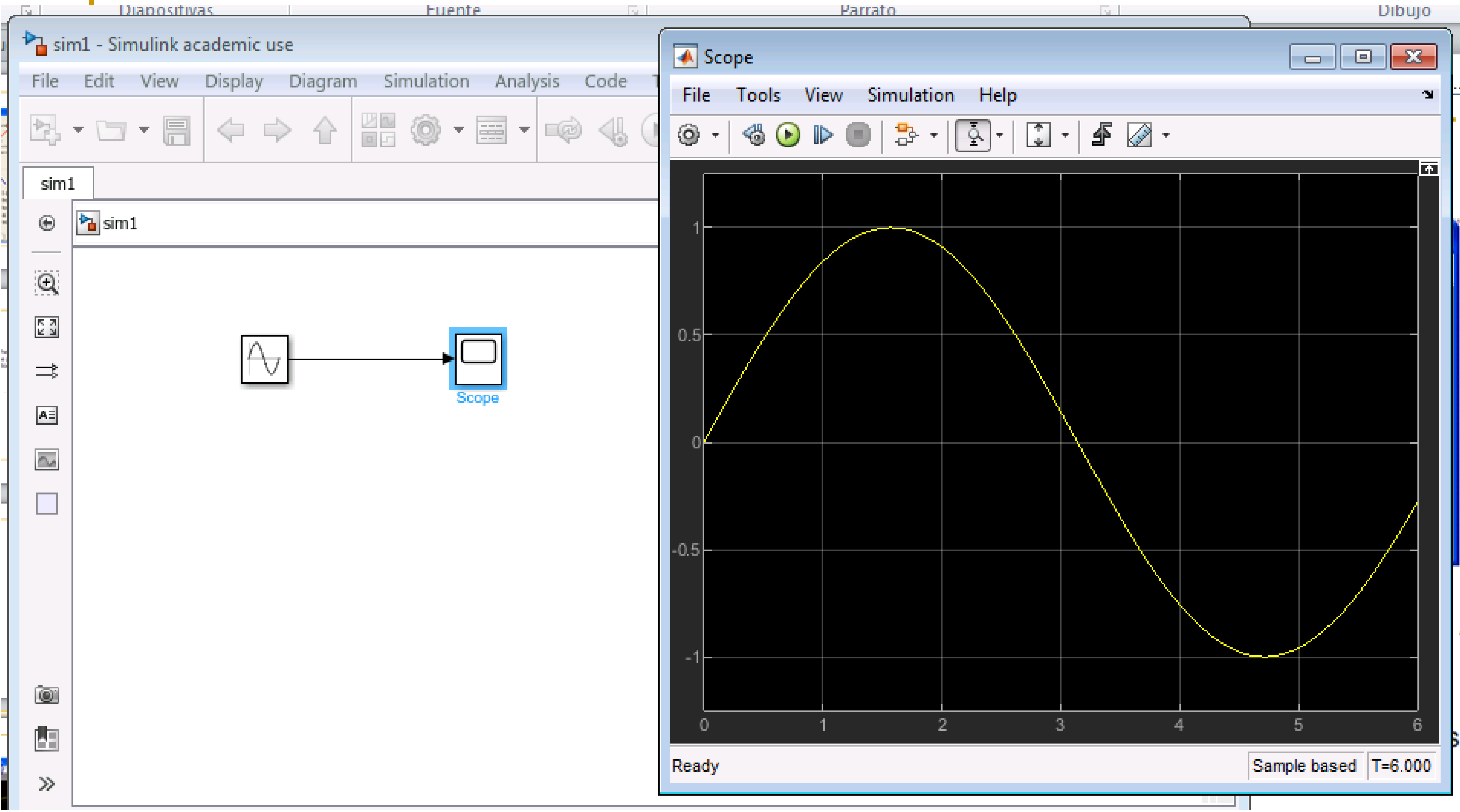
Connecting Blocks



Drag a signal line from the output of a block to the input of another block.

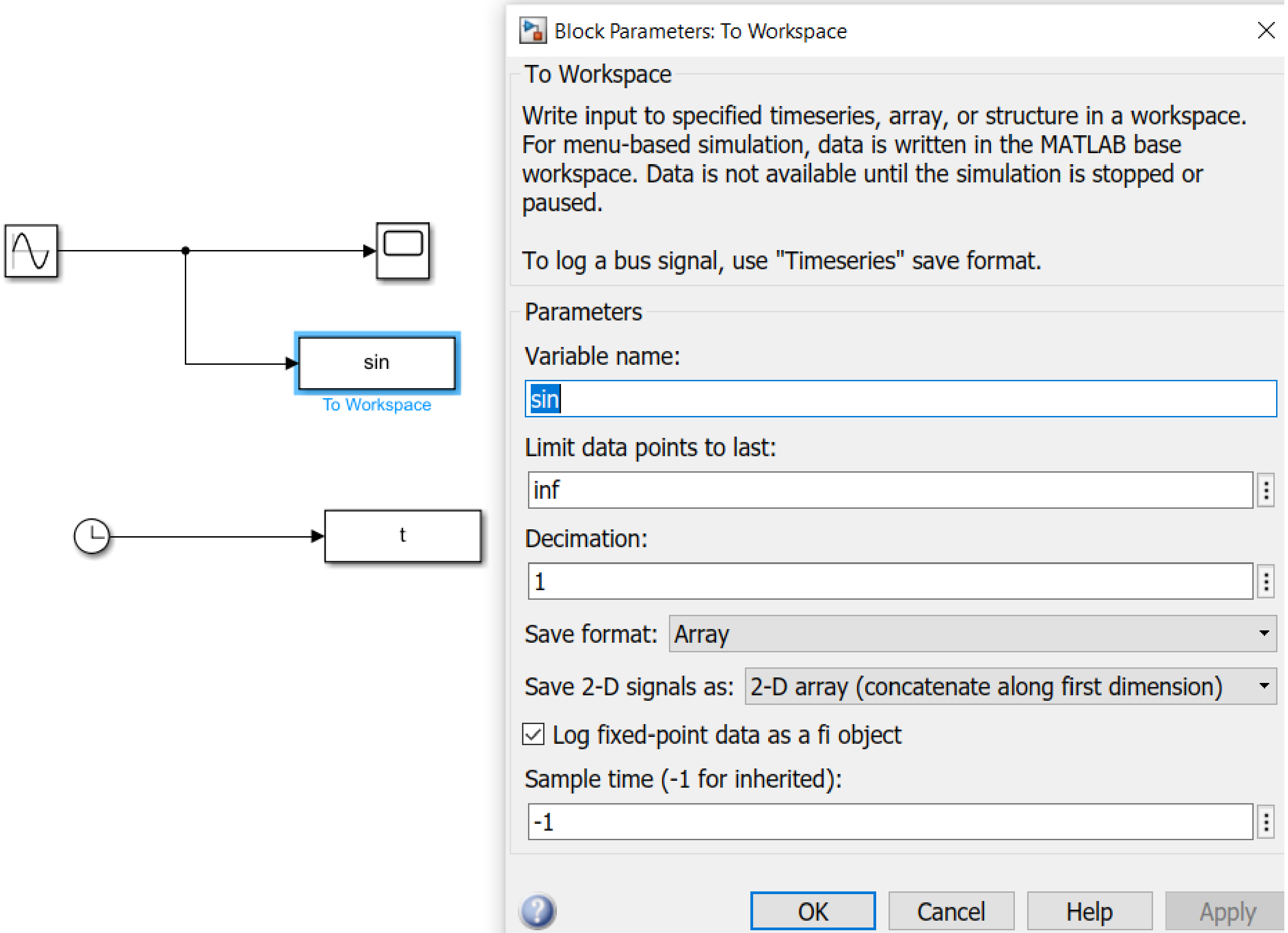
Ctrl-Click will automatically connect.

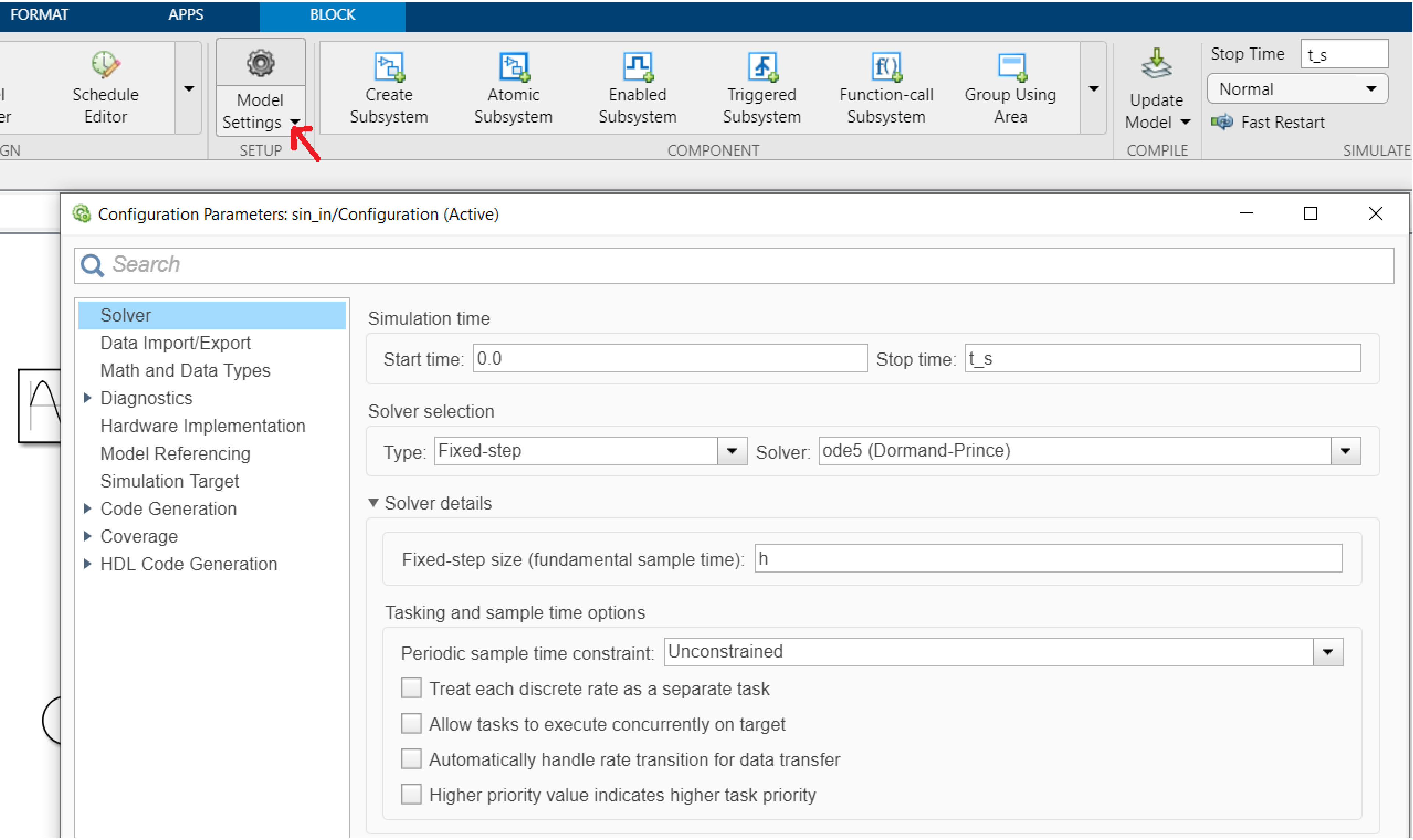
Viewing Results: Scope



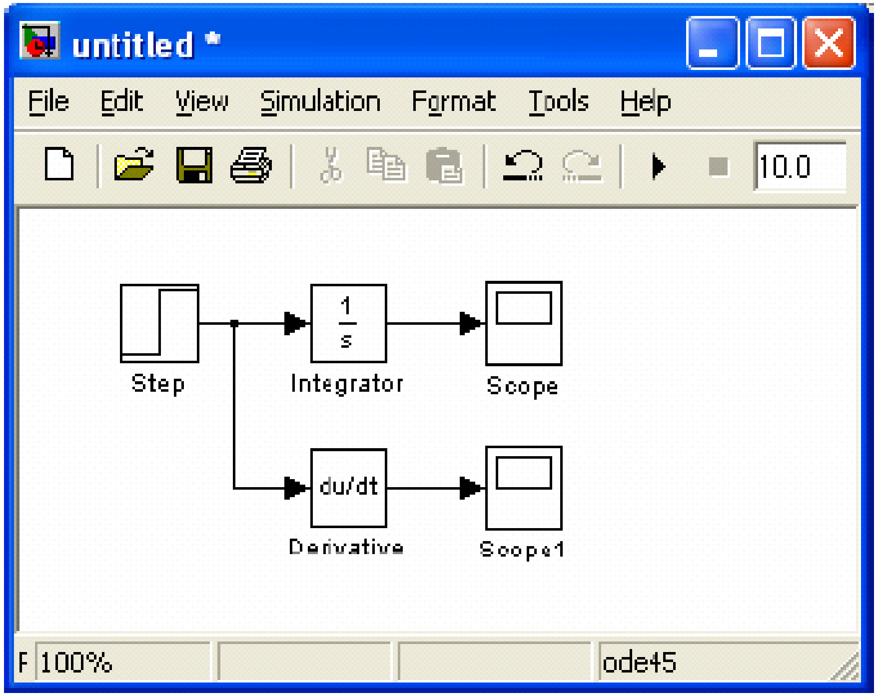
Running the Simulation

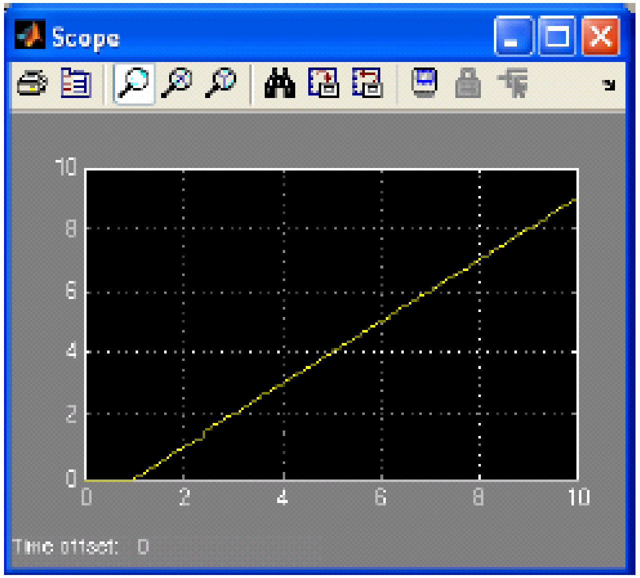
```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%% FIRST PROGRAM
%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% 1. PARAMETERS
t_s = 10;
h  = 0.001;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%
%%% 2. VALUES
% Sin Input
A=1;
f=3; %(rad)
% 3. SIMULATION AND PLOTS
sim sin_w
%%% plots
figure;
plot(t,sin_w);
xlabel('Time(s)');
ylabel('sin (rad.)');
title('Simula sin');
```

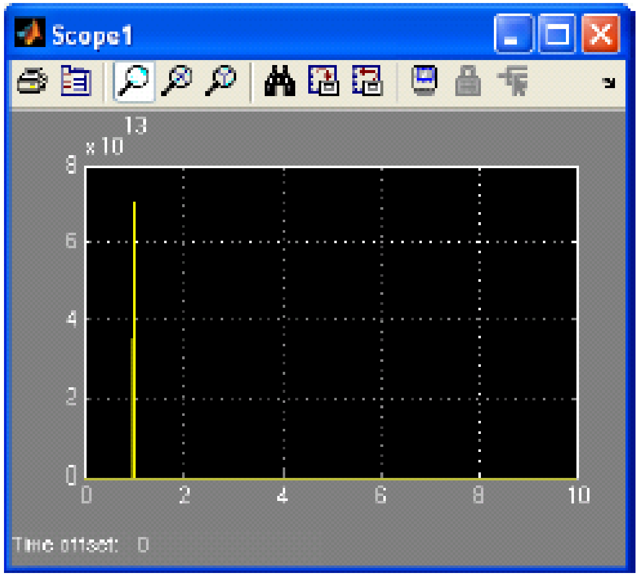

sin
To Workspace
t
Block Parameters: To Workspace
To Workspace
Write input to specified timeseries, array, or structure in a workspace. For menu-based simulation, data is written in the MATLAB base workspace. Data is not available until the simulation is stopped or paused.
To log a bus signal, use "Timeseries" save format.
Parameters
Variable name:
sin
Limit data points to last:
inf
Decimation:
1
Save format: Array
Save 2-D signals as: 2-D array (concatenate along first dimension)
Log fixed-point data as a fi object
Sample time (-1 for inherited):
-1
OK
Cancel
Help
Apply


FORMAT
APPS
BLOCK
Schedule Editor
Model Settings
SETUP
Create Subsystem
Atomic Subsystem
Enabled Subsystem
Triggered Subsystem
Function-call Subsystem
Group Using Area
COMPONENT
Update Model
COMPILE
Stop Time
t_s
Normal
Fast Restart
SIMULATE
Configuration Parameters: sin_in/Configuration (Active)
Search
Solver
Data Import/Export
Math and Data Types
Diagnostics
Hardware Implementation
Model Referencing
Simulation Target
Code Generation
Coverage
HDL Code Generation
Simulation time
Start time: 0.0
Stop time: t_s
Solver selection
Type: Fixed-step
Solver: ode5 (Dormand-Prince)
Solver details
Fixed-step size (fundamental sample time): h
Tasking and sample time options
Periodic sample time constraint: Unconstrained
Treat each discrete rate as a separate task
Allow tasks to execute concurrently on target
Automatically handle rate transition for data transfer
Higher priority value indicates higher task priority

Integrators and Derivatives

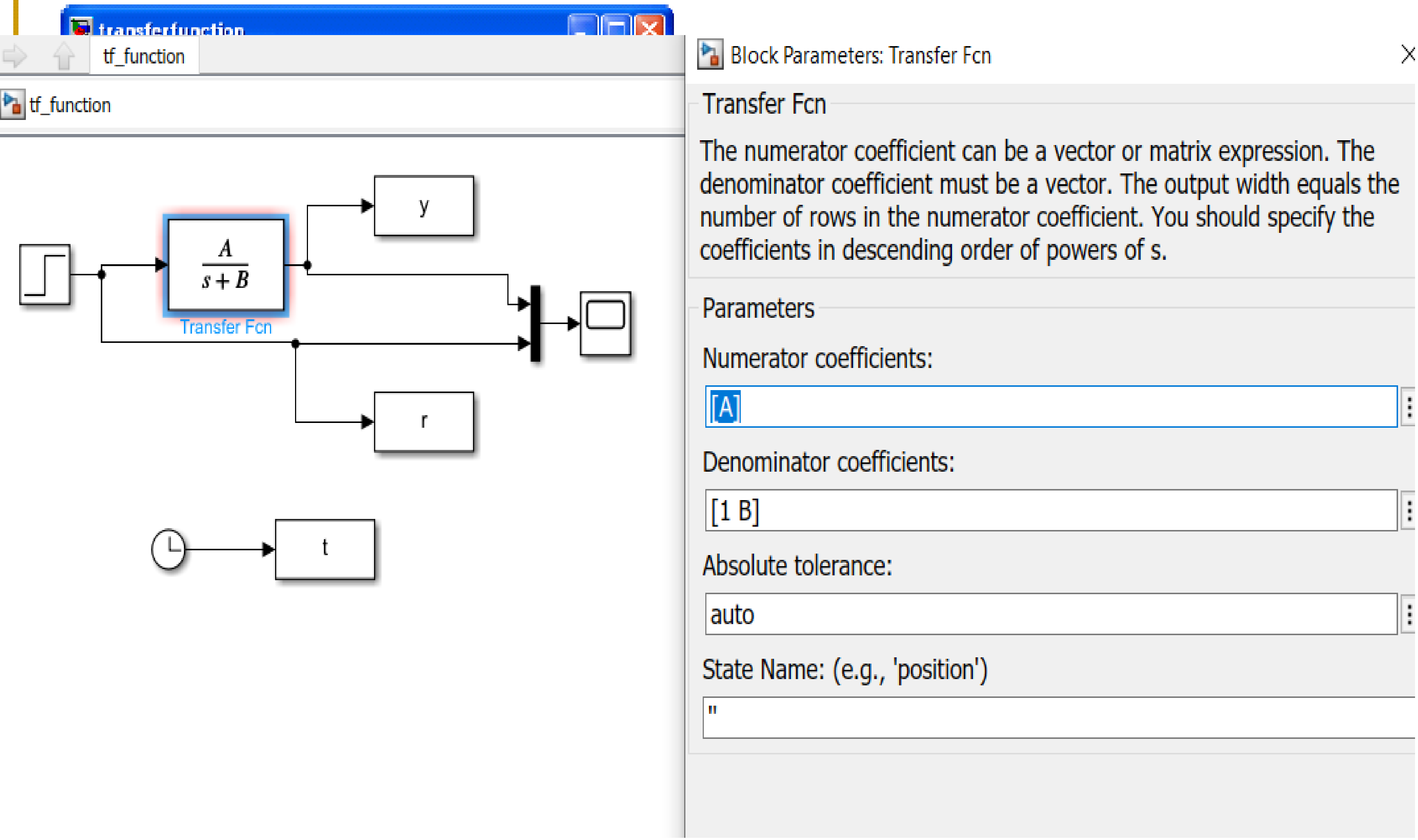




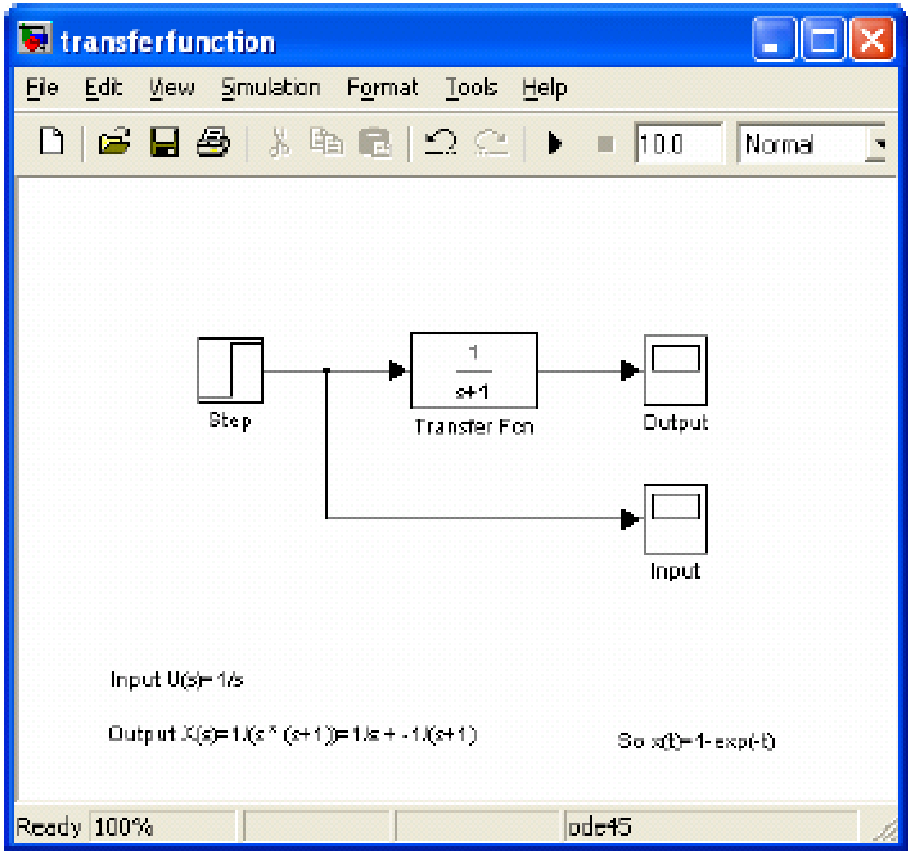


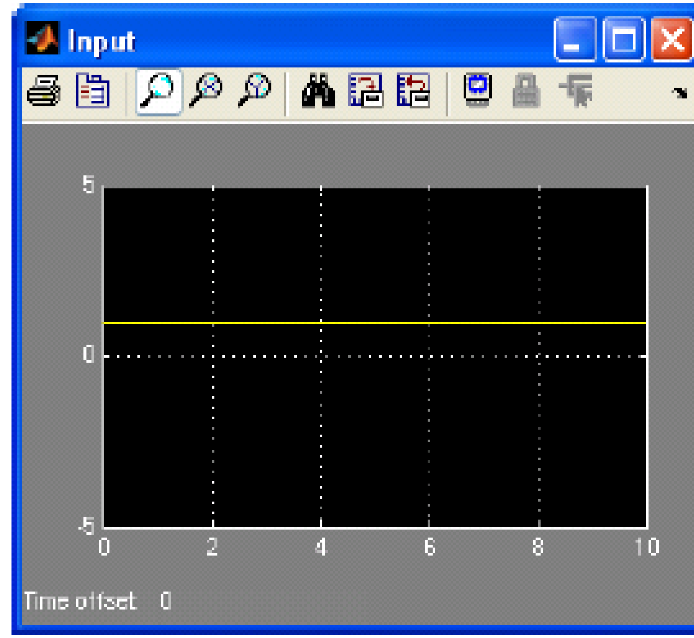
Integrators and derivatives are available in the “continuous” library.

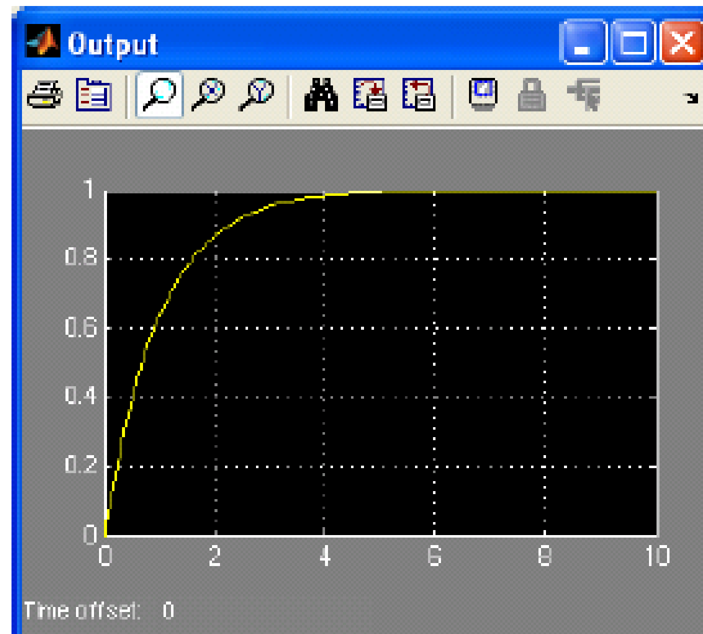
Transfer Functions



Transfer Functions







$U(s) = 1/s$

$X(s)=1/(s(s+1))=1/s-1/(s+1)$

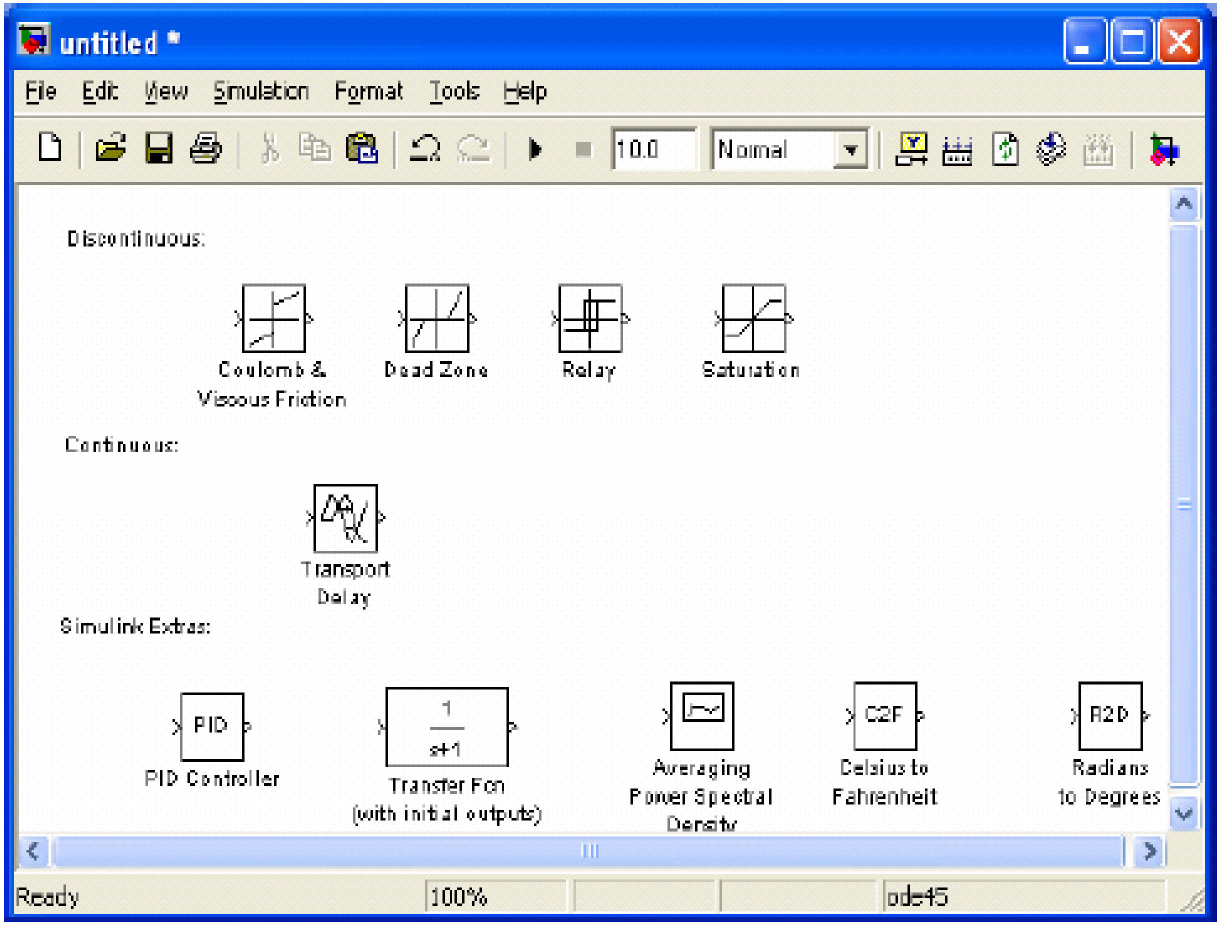
So $x(t)=1-\exp(-t)$

Other Useful Blocks

Under “Discontinuous” you will find coulomb friction, dead zone, saturation, and relay.

Under “Continuous” there is a delay block.

Under “Simulink Extras” there is a PID controller, transfer function with non-zero initial conditions, some useful sinks (such as power spectral density), and radians-to-degrees and Fahrenheit-to-Celsius converters.

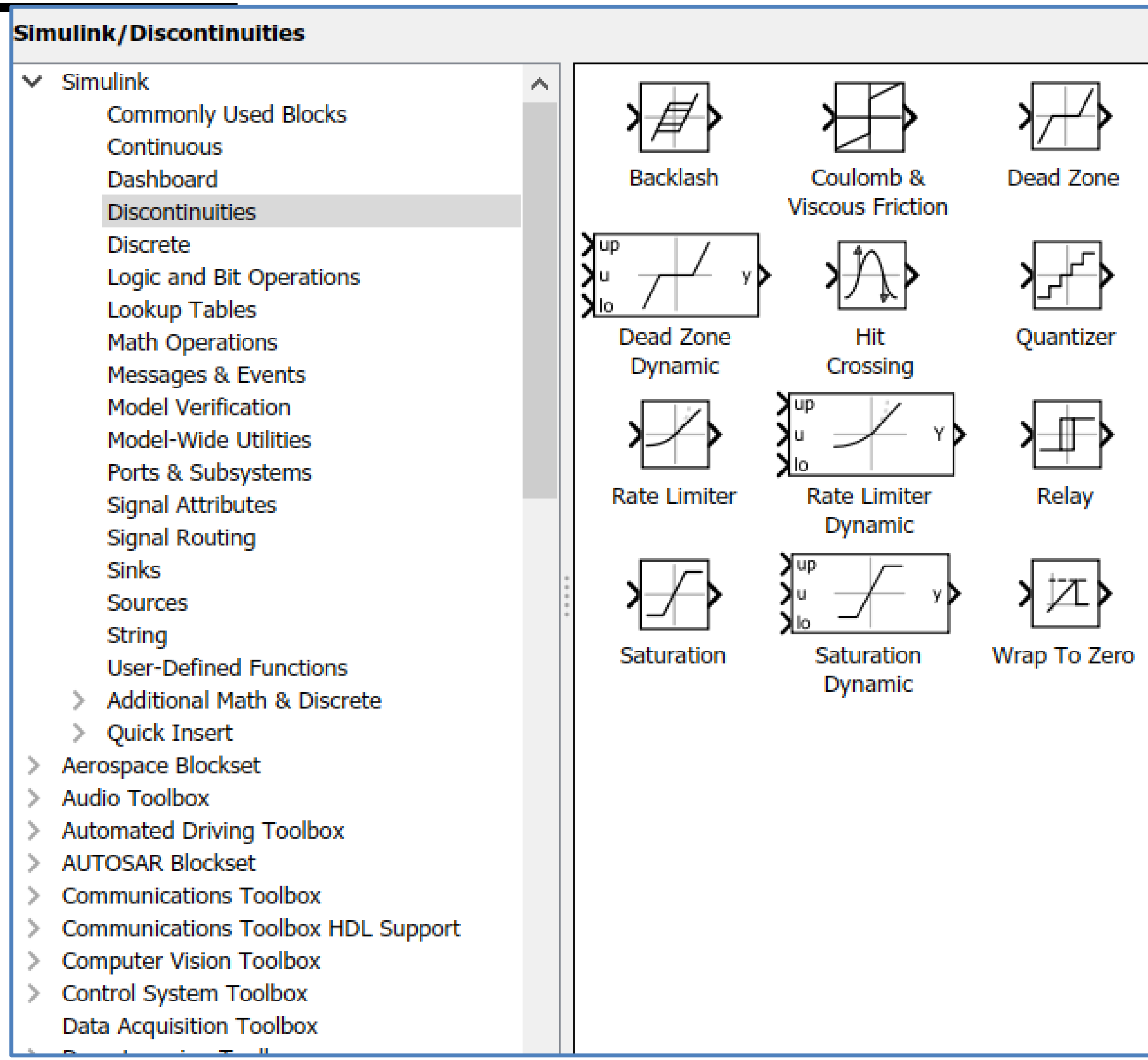


Other Useful Blocks

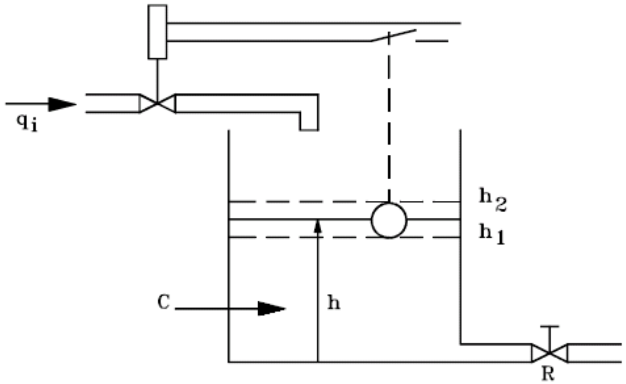
Under "Discontinuous" you will find coulomb friction, dead zone, saturation, and relay.

Under "Continuous" there is a delay block.

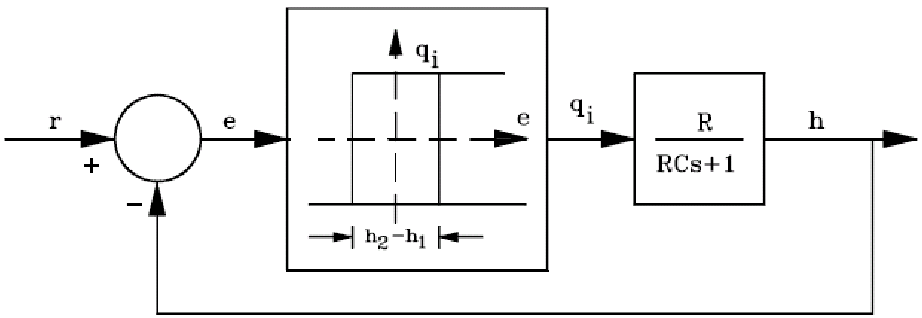
Under "Simulink Extras" there is a PID controller, transfer function with non-zero initial conditions, some useful sinks (such as power spectral density), and radians-to-degrees and Fahrenheit-to-Celsius converters.

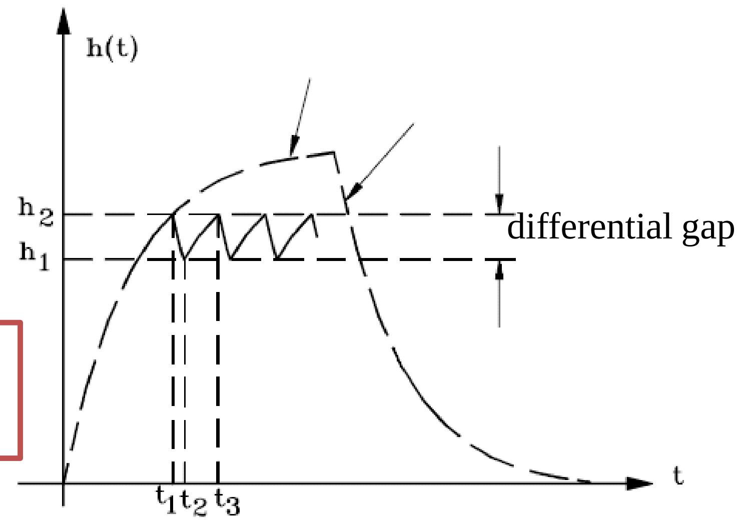


Modelling the tank



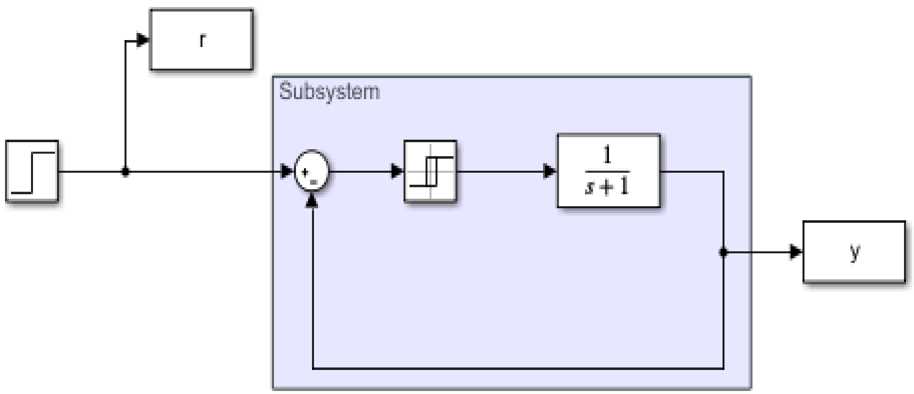
$$C\frac{dh}{dt}+\frac{h}{R}=q_i(t)$$

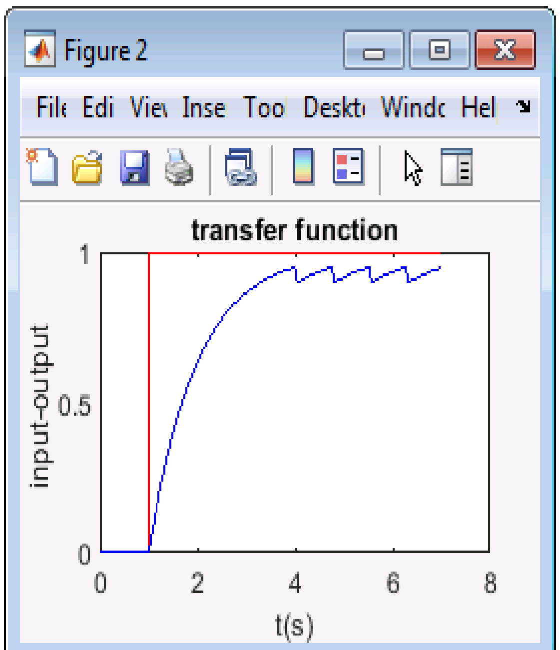


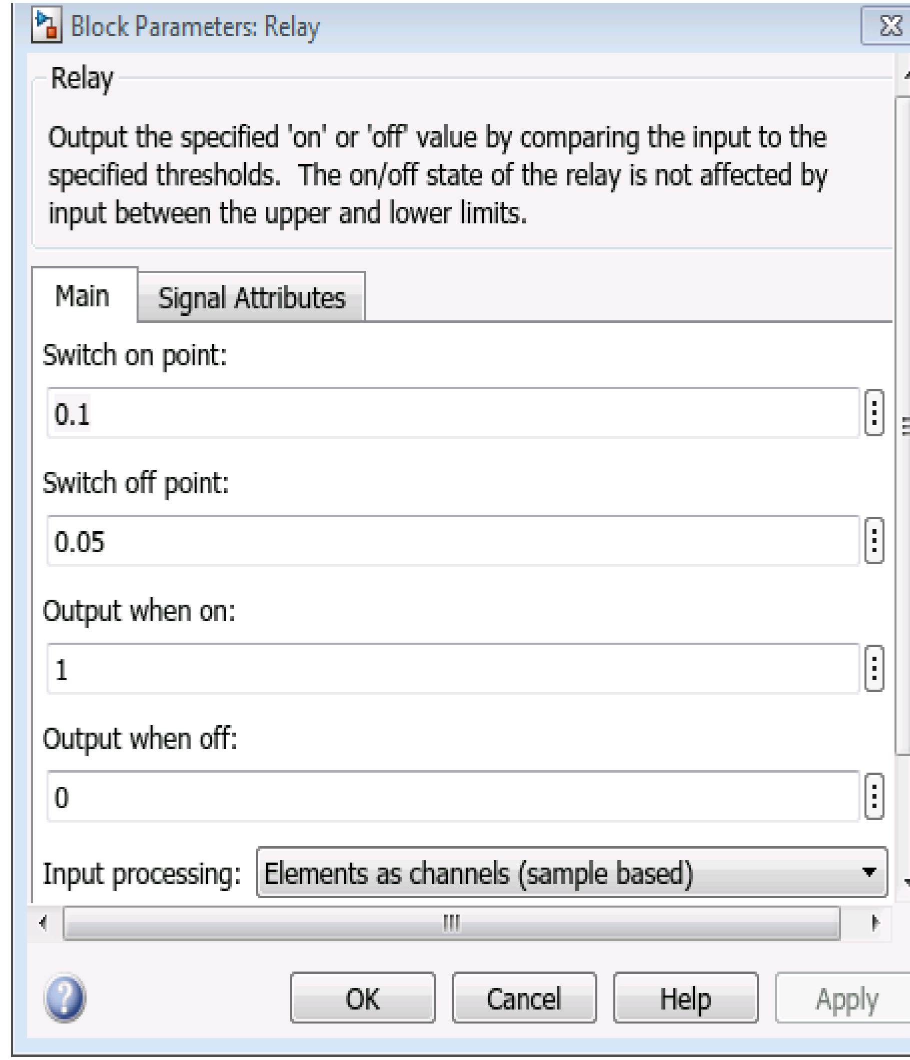


Response of the level of a tank with ON-OFF control and a differential gap

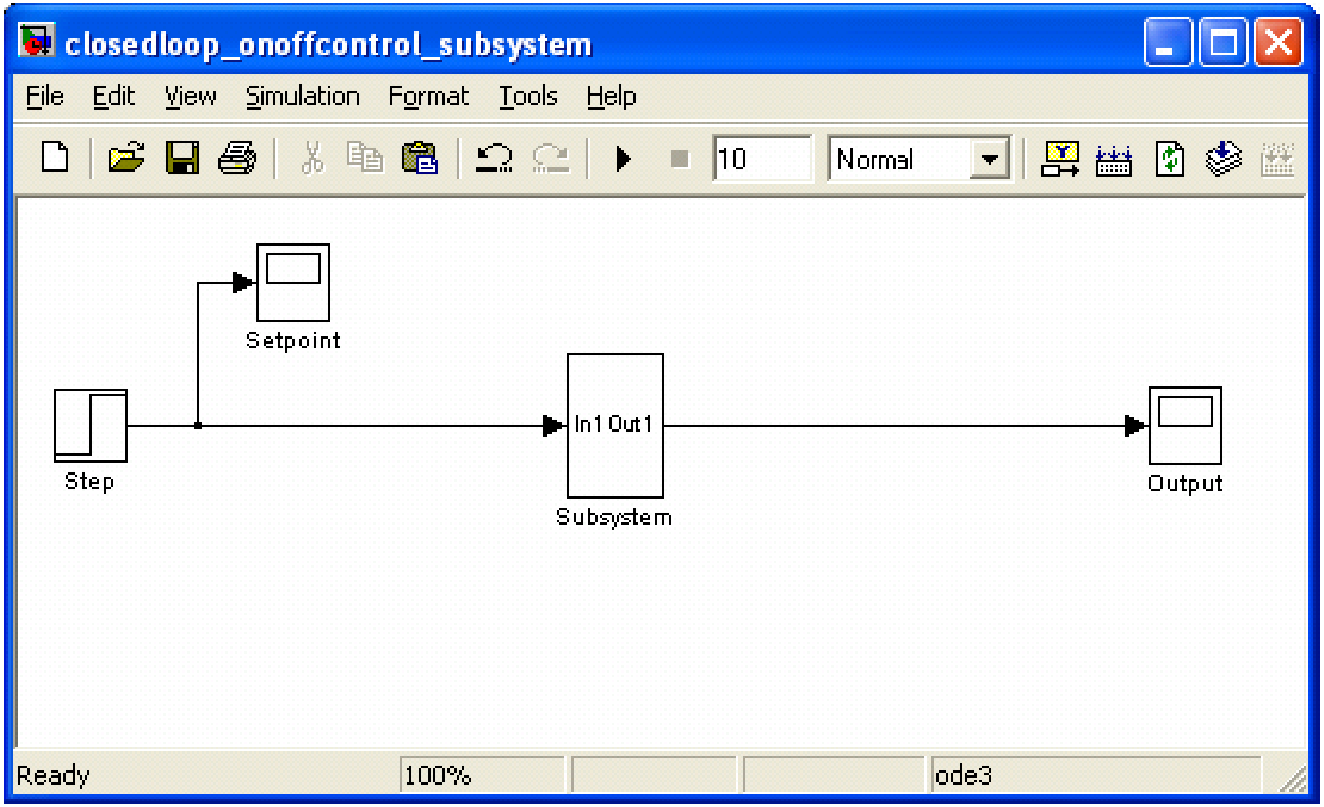
Closed Loop Control System:







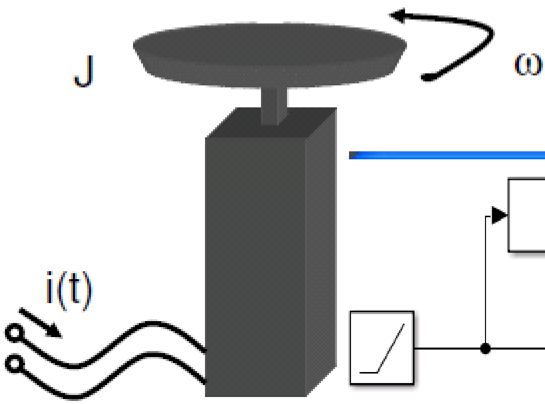
Subsystems



You can group a set of blocks together into a subsystem, by selecting them and right clicking and saying “Create Subsystem”. They will all go under a single block. If you double click the subsystem, you can see what is under the “mask”.

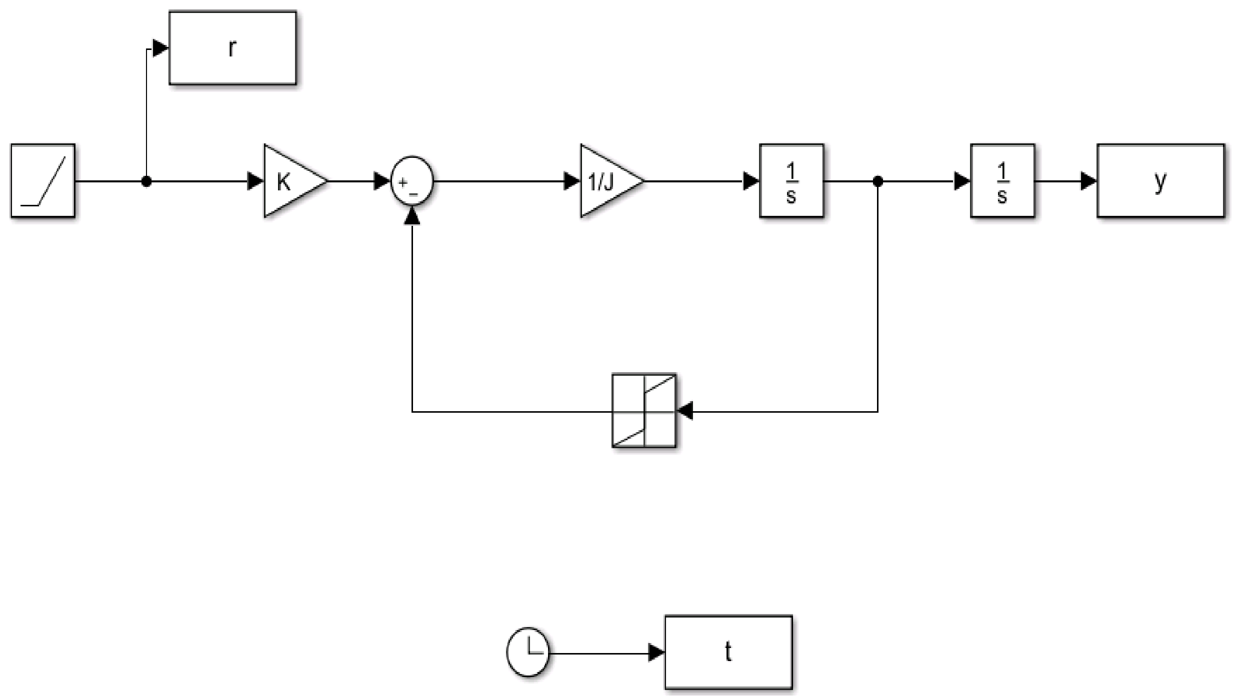
Setting Up Systems with Integrators

If you have a nonlinear equation system, you **can't** describe it with a transfer function. One option is to put all of the operations in as individual blocks:

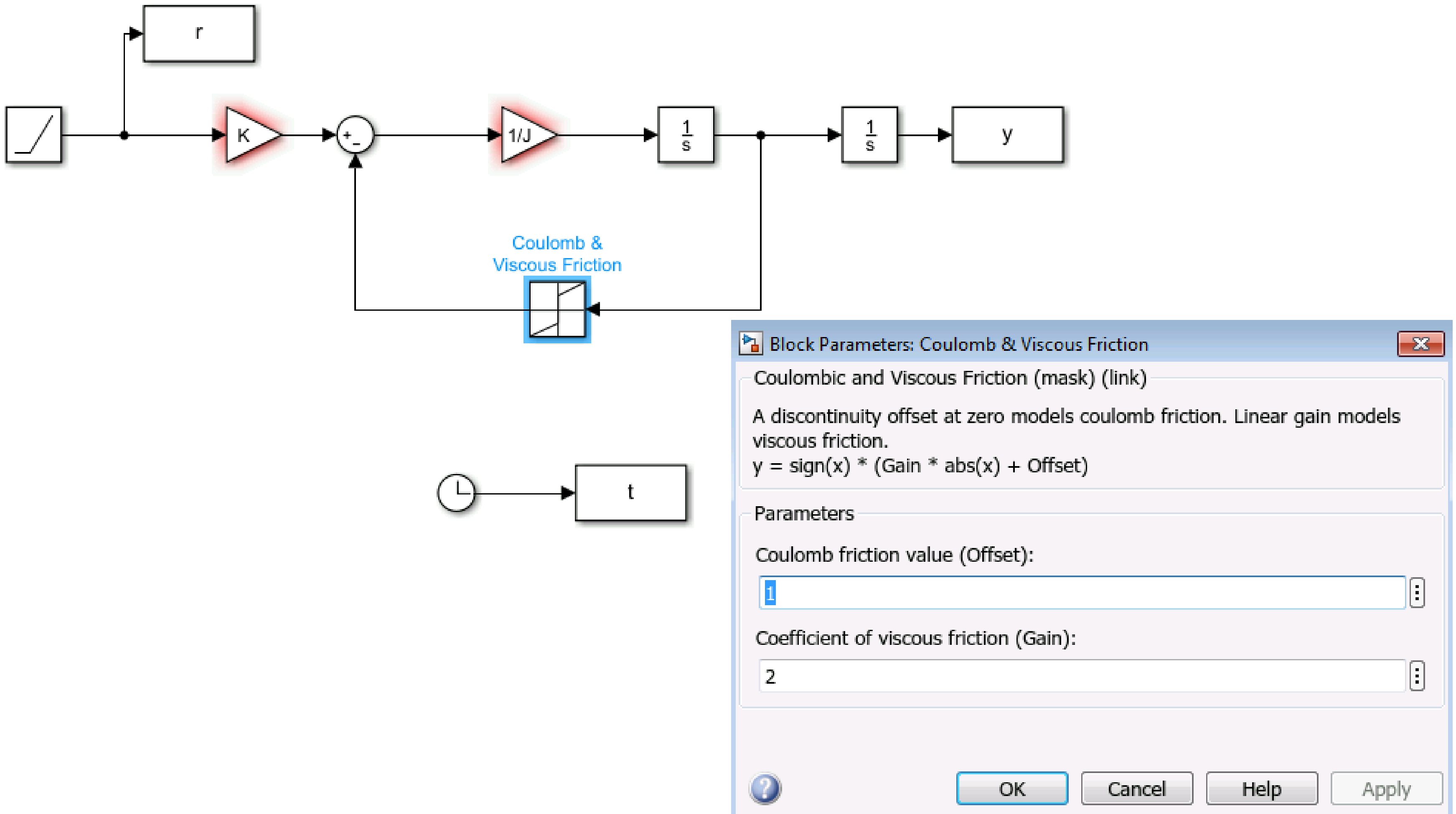


$$\tau(t) = ki(t) - f(\omega) = J\ddot{\theta}$$

where f(w) is coulomb and viscous friction.

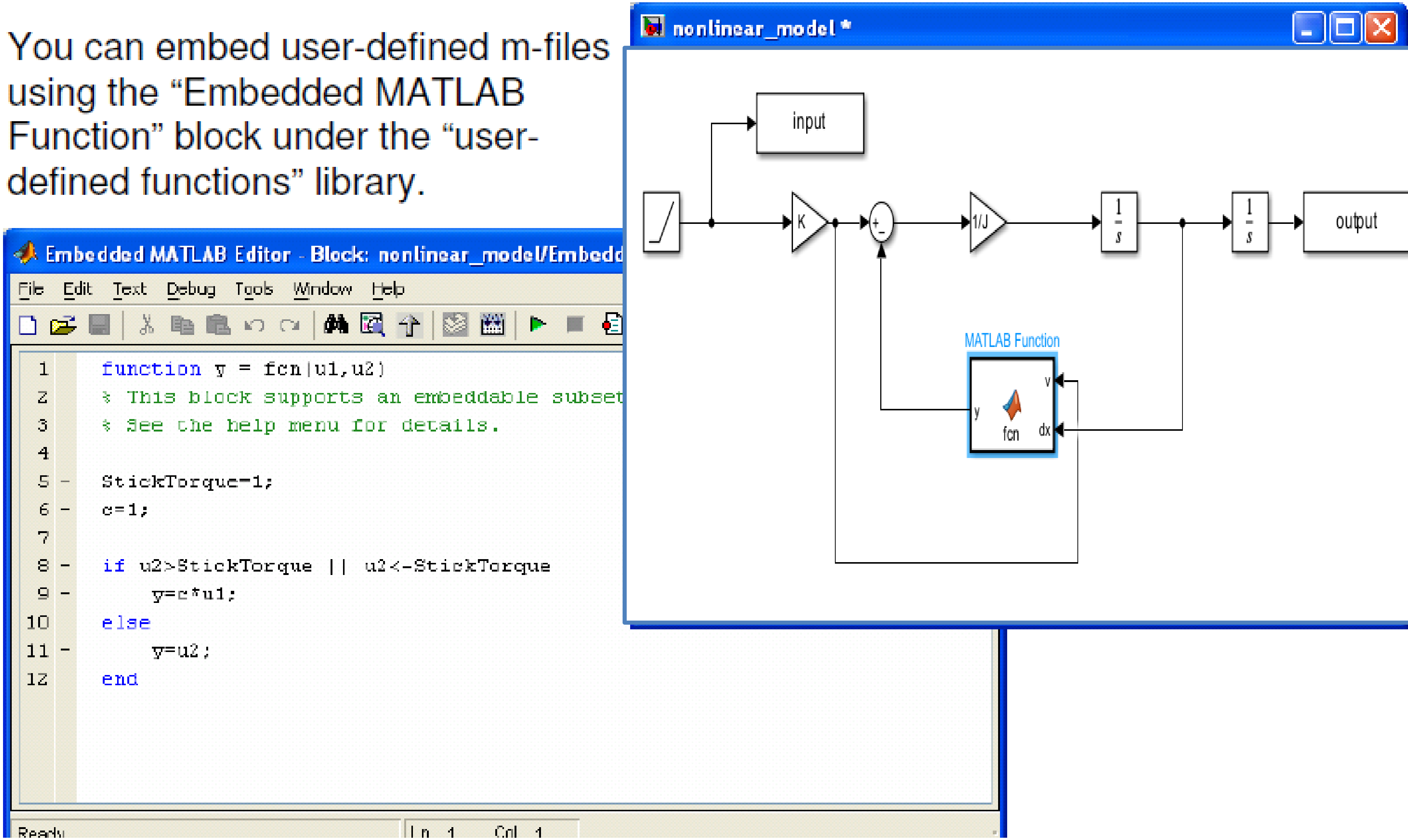


Use integrators rather than derivatives, and put in as many integrators as there are states. Then "bootstrap"!

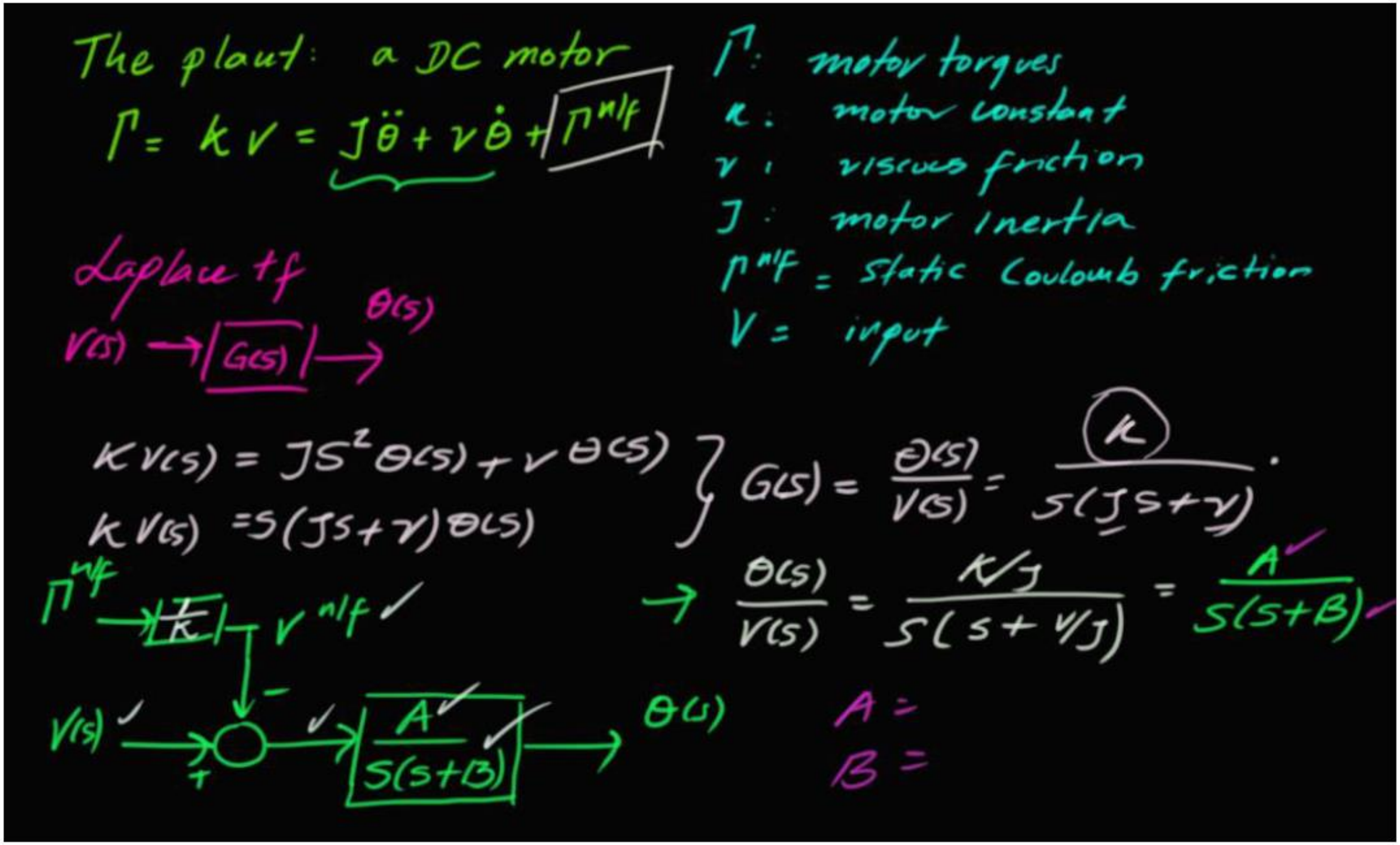

r
K
1/J
1
s
1
s
y
Coulomb &
Viscous Friction
t
Block Parameters: Coulomb & Viscous Friction
Coulombic and Viscous Friction (mask) (link)
A discontinuity offset at zero models coulomb friction. Linear gain models viscous friction.
y = sign(x) * (Gain * abs(x) + Offset)
Parameters
Coulomb friction value (Offset):
1
Coefficient of viscous friction (Gain):
2
OK
Cancel
Help
Apply

User Defined Functions

You can embed user-defined m-files using the “Embedded MATLAB Function” block under the “user-defined functions” library.



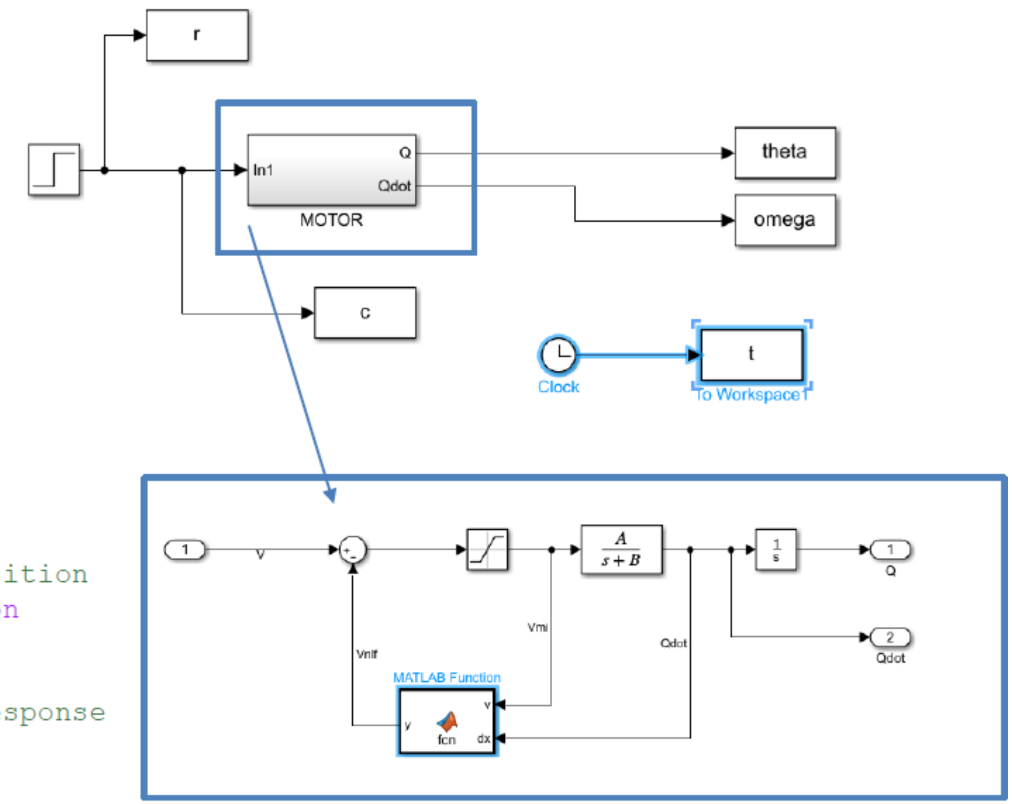
DC motor dynamics



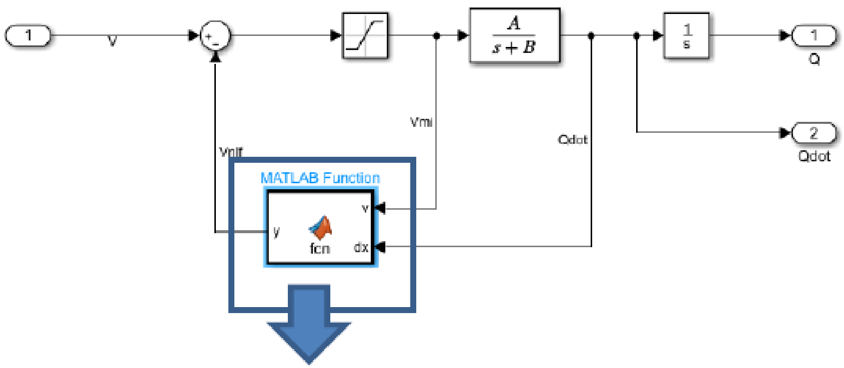
Asst. Prof. Claudia F. YAŞAR YTU Control and
Automation Engineering Department-2024

DC motor dynamics

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%% MOTOR SIMULATION
%%% System configuration
t_s=10;
h=0.001;
%Motor and simulation parameters
ref=1;
K1=0.00302;
J1=2.97E-07;
v1=8.23E-06;
A=K1/J1;
B=v1/J1;
sat1=1.0;
%%% Simulation and plots
sim motor_sim;
figure;
plot(t,theta,'b',t,r,'r'); %motor position
xlabel('t(s)'); ylabel('Motor position
(rad)');title('DC Motor');
figure;
plot(t,omega,'b',t,r,'r'); %system response
xlabel('t(s)'); ylabel('Motor speed
(rad/s)');title('DC Motor');
```



DC motor dynamics

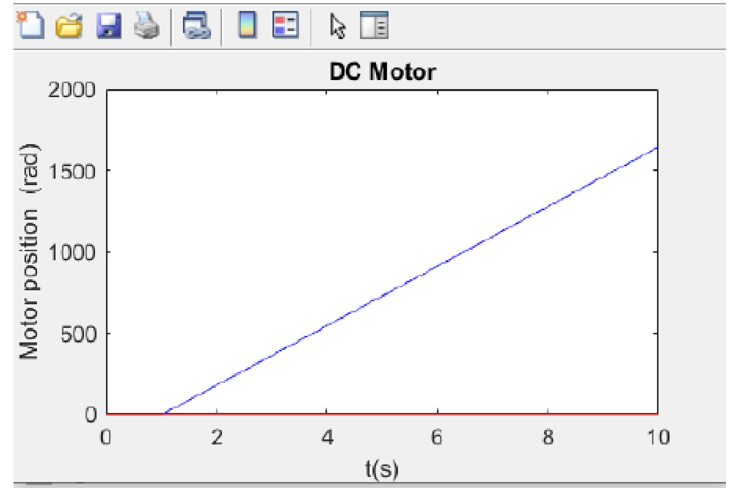


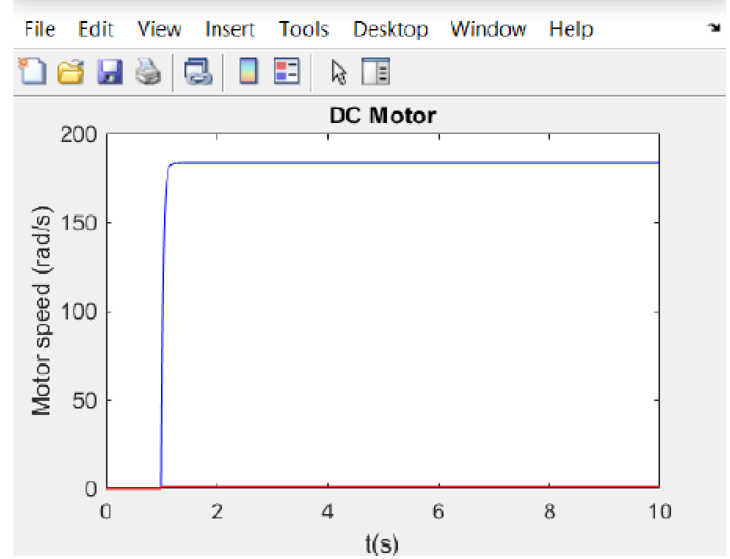

```
function y = fcn(v, dx)
%#codegen
Vie=0.15;
if dx==0
    if abs(v)<=Vie
        y=v;
    else
        y=sign(v)*Vie;
    end
else
    y=Vie*sign(dx);
end
```

$$f\dot{\theta}=0;$$

$$v^{nlf}=\begin{cases} |v_{mi}|\le v_{ie}^{nlf}; V_{mi} \\ |v_{mi}|>v_{ie}^{nlf}; \mathrm{sgn}(V_{mi}).v_{ie}^{nlf} \end{cases}$$

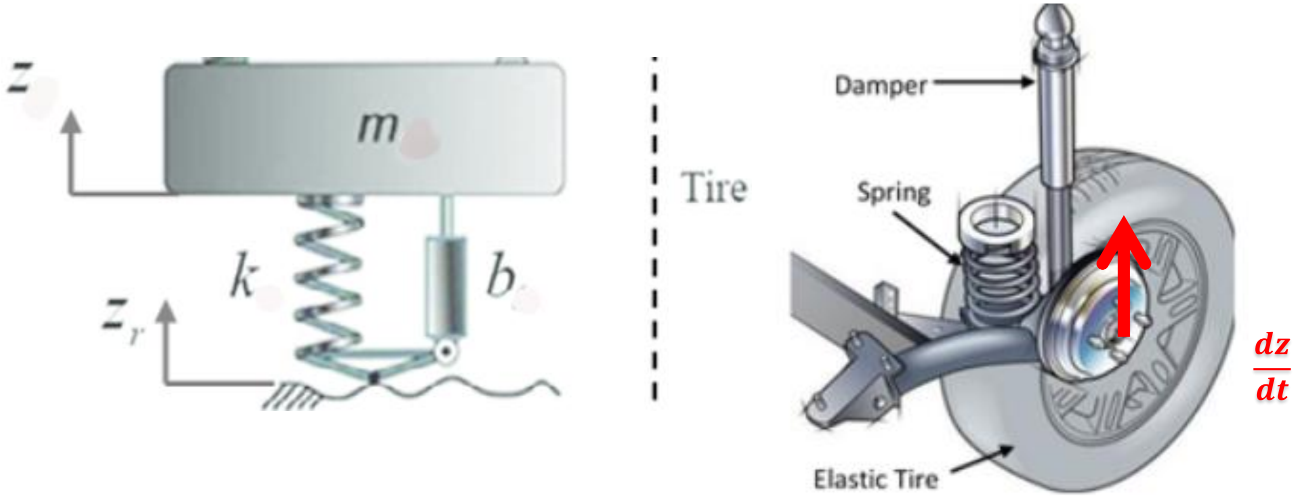
$$if\,\dot{\theta}\neq 0; v^{nlf}=v_{ie}^{nlf}.\mathrm{sgn}(\dot{\theta})$$

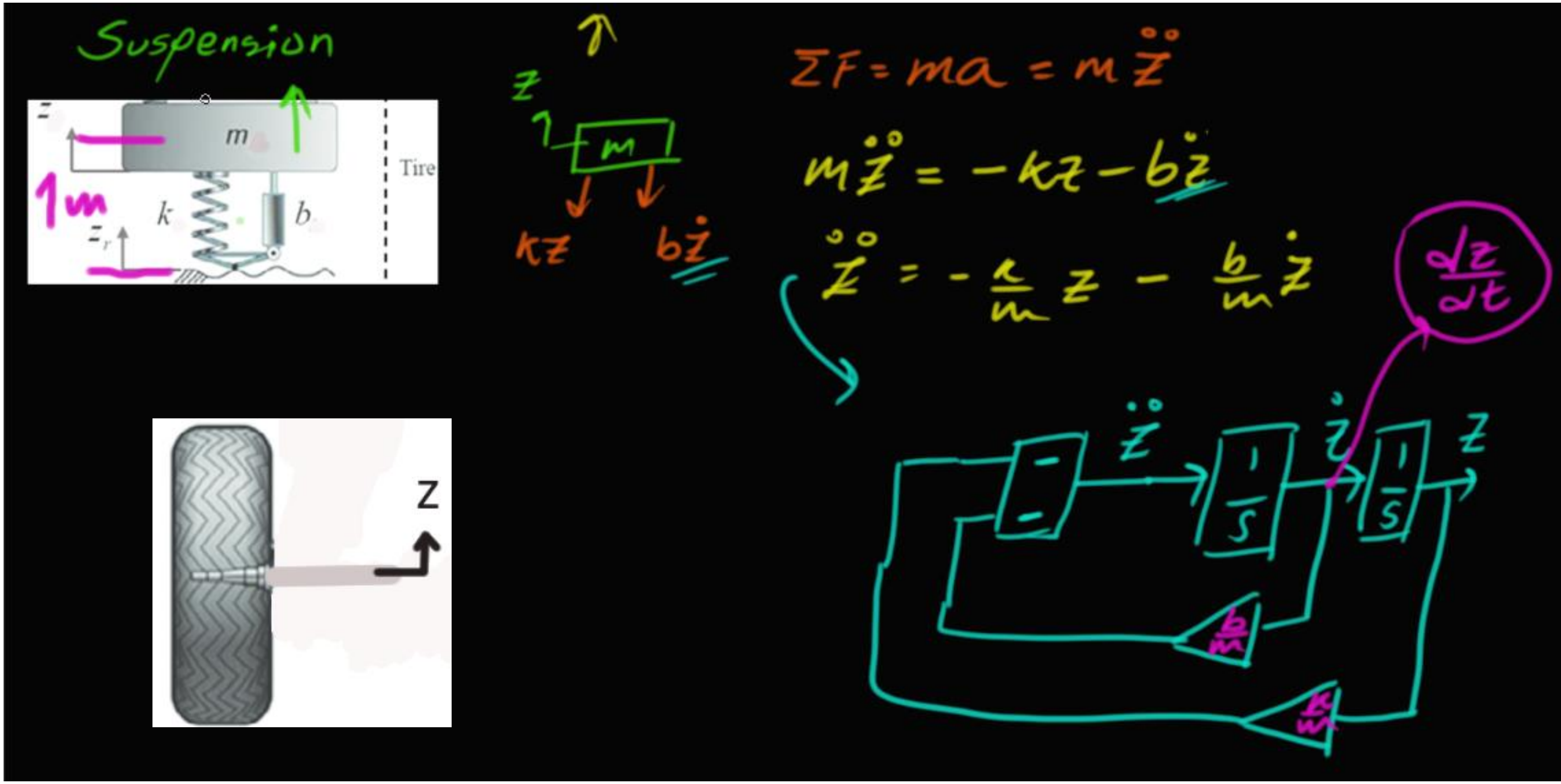


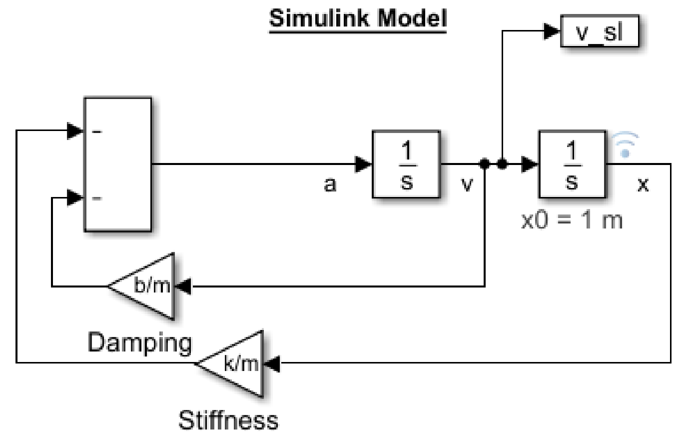


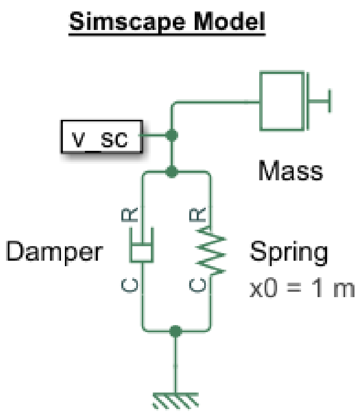
Suspension system

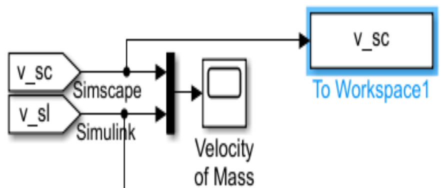
A suspension system has to be modelled and simulated. Here, z is the vertical displacement of the wheel center. We are interested in obtaining how the speed of mass m changes by time $\frac{dz}{dt}$.




Suspension
z
m
Tire
1m
k
b
z_r
Z
kz
$b\dot{z}$
$\Sigma F = ma = m\ddot{z}$
$m\ddot{z} = -kz - b\dot{z}$
$\ddot{z} = -\frac{k}{m} z - \frac{b}{m}\dot{z}$
$\frac{dz}{dt}$
$\ddot{z}$
$\dot{z}$
$\frac{1}{s}$
$\frac{1}{s}$
$\frac{b}{m}$
$\frac{k}{m}$

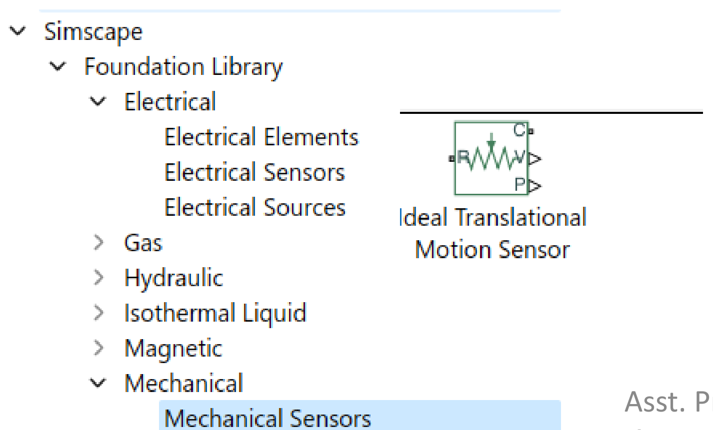


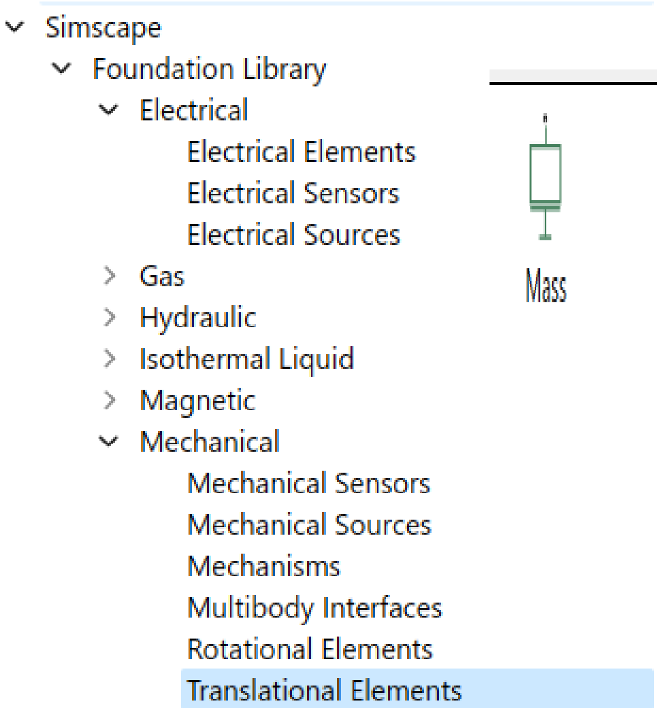


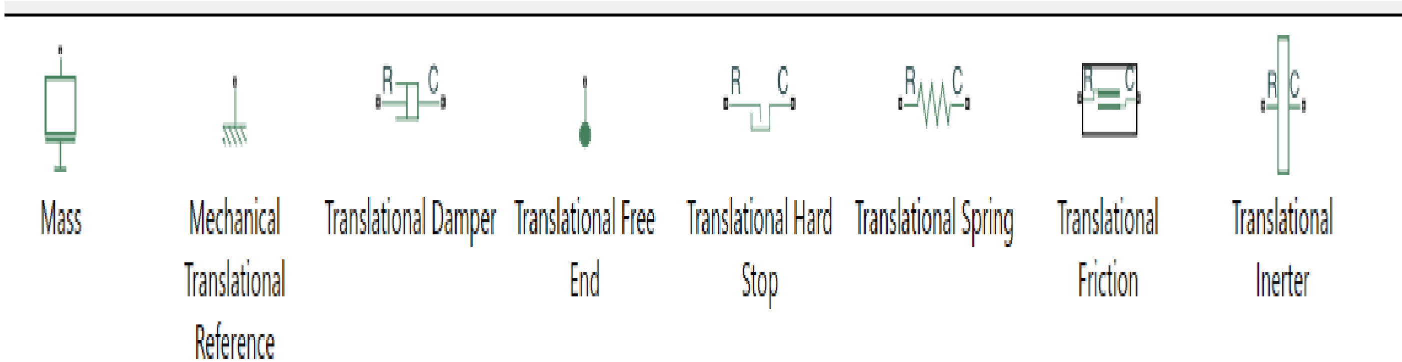


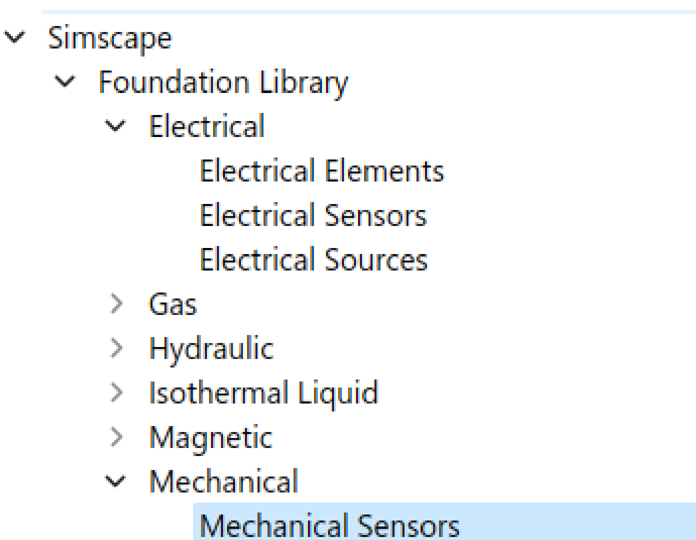
ssc_mass_spring_damper_sl

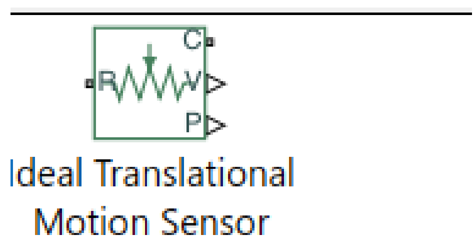
```
%%%%%  MASS SPRING DAMPER
%%%%%  SIMULINK AND SIMSCAPE
%%%%%
% 1. SIMULATION PARAMETERS
b=10;%N/(m/s)
k=400;% N/m
m=3.6;%kg
x0=1;%m
% 2. SIMULATION AND PLOTS
%
Sim ssc_mass_spring_damper_sl_2022;
figure;
plot(v_sc,'g','LineWidth',2);
hold on
plot(v_sl,'--r','LineWidth',2);
xlabel('Time(s)');
ylabel('Velocity (m/s)');
title('Velocity of Mass');
```

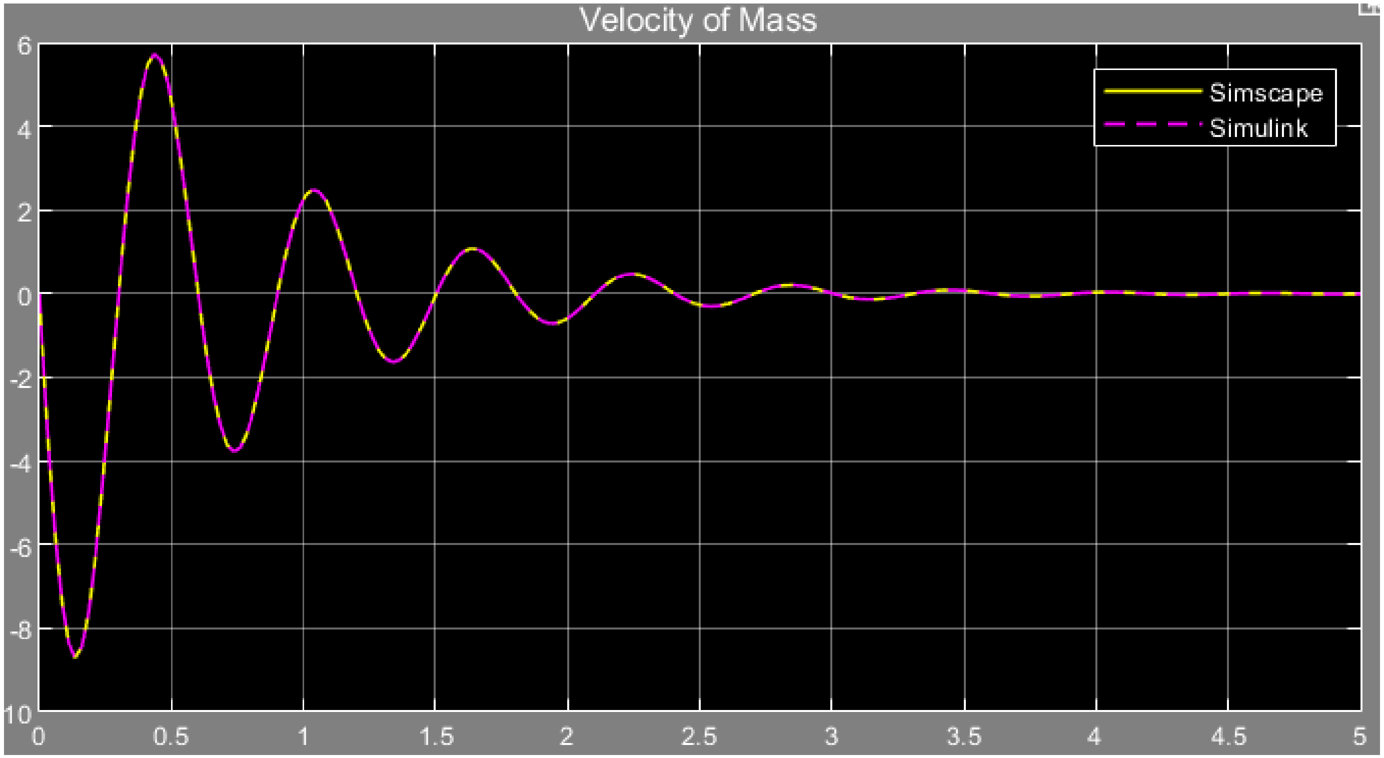



Simscape
Foundation Library
Electrical
Electrical Elements
Electrical Sensors
Electrical Sources
Gas
Hydraulic
Isothermal Liquid
Magnetic
Mechanical
Mechanical Sensors
Mechanical Sources
Mechanisms
Multibody Interfaces
Rotational Elements
Translational Elements


Mass
Mechanical Translational Reference
Translational Damper
Translational Free End
Translational Hard Stop
Translational Spring
Translational Friction
Translational Inerter

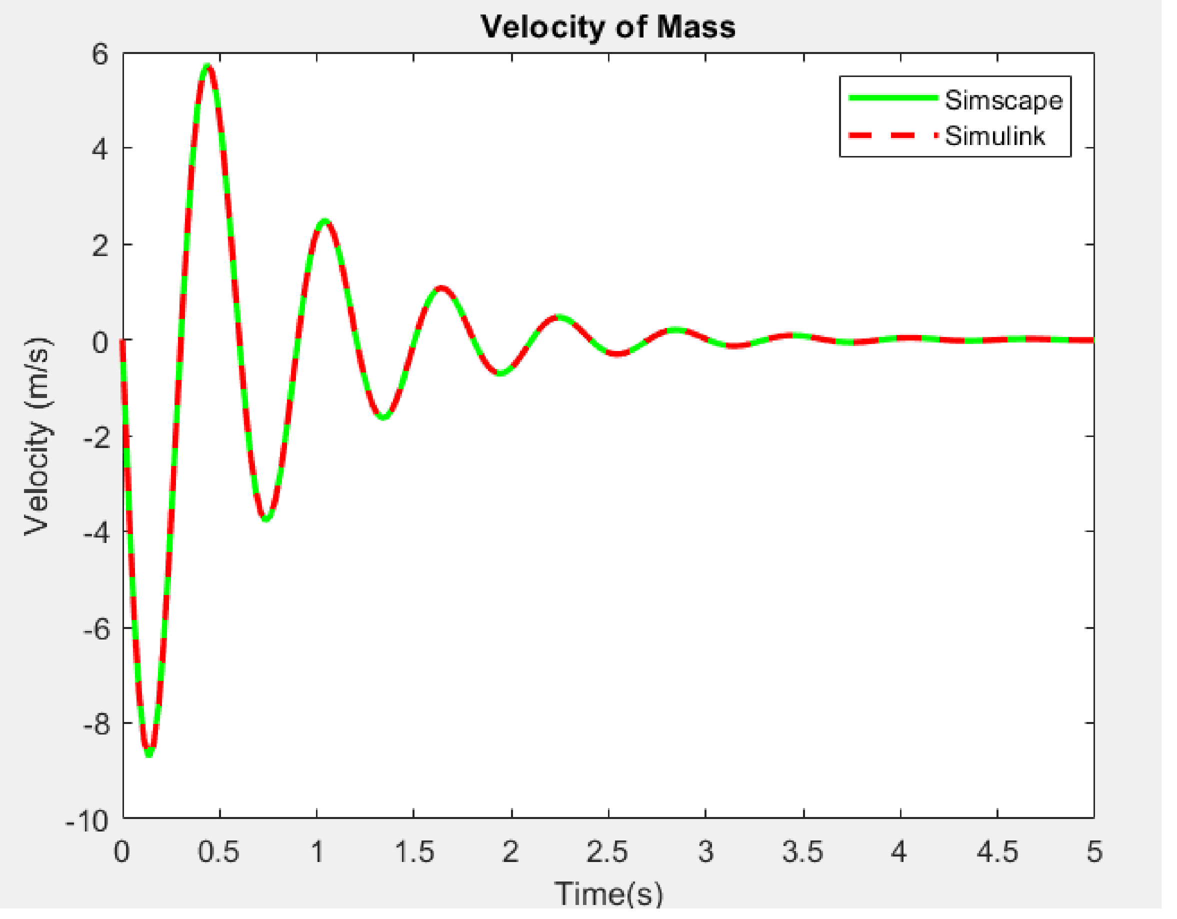

Simscape
Foundation Library
Electrical
Electrical Elements
Electrical Sensors
Electrical Sources
Gas
Hydraulic
Isothermal Liquid
Magnetic
Mechanical
Mechanical Sensors


C
R
V
P
Ideal Translational Motion Sensor



Velocity of mass (m/s)

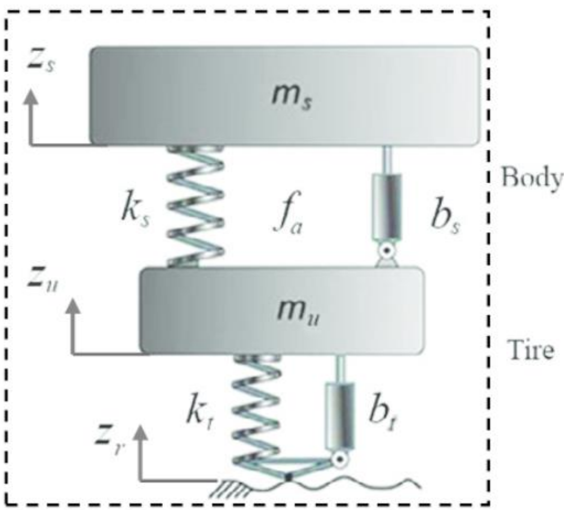
Time (s)



Velocity of Mass
Simscape
Simulink
Velocity (m/s)
Time(s)
6
4
2
0
-2
-4
-6
-8
-10
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5

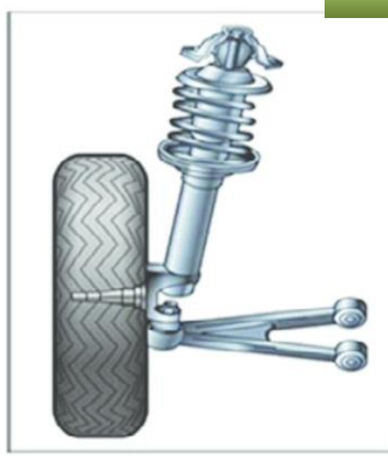
Suspension system

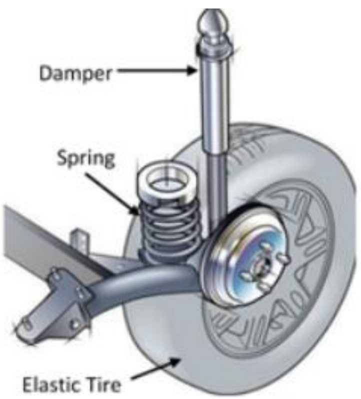
A quarter-car and its suspension system has to be modelled and simulated. Here, z_u is the vertical displacement of the wheel center and z_s is the vertical displacement of vehicle chassis. We are interested in obtaining how the speed of masses m_s and m_u change by time.



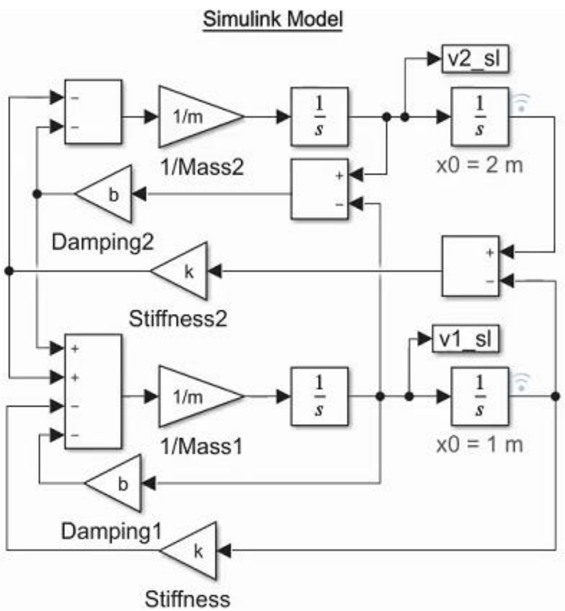
A quarter-car model of suspension system

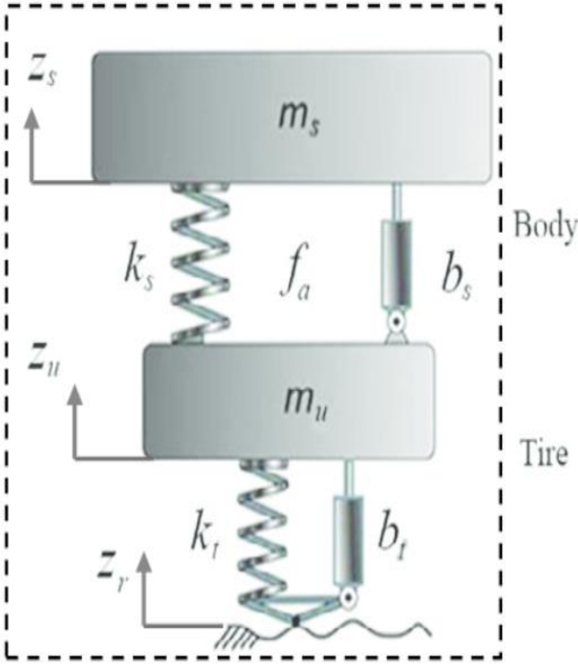
Quarter-car





Mass-Spring-Damper Simulink



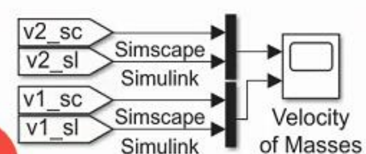


https://www.mathworks.com/help/physmod/simscape/ug/double-mass-spring-damper-in-simulink-and-simscape.html

https://ctms.engin.umich.edu/CTMS/index.php?example=Suspension&section=SimulinkControl

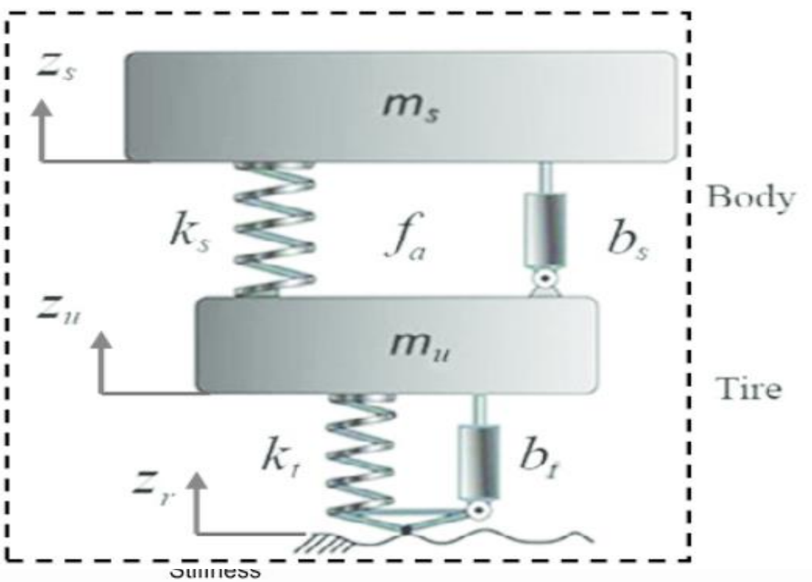
Double Mass-Spring-Damper in Simulink and Simscape

1. Plot position of masses (see code)
2. Explore simulation results using sscexplore
3. Learn more about this example
4. Open model of single mass-spring-damper
5. Learn more about modeling physical networks

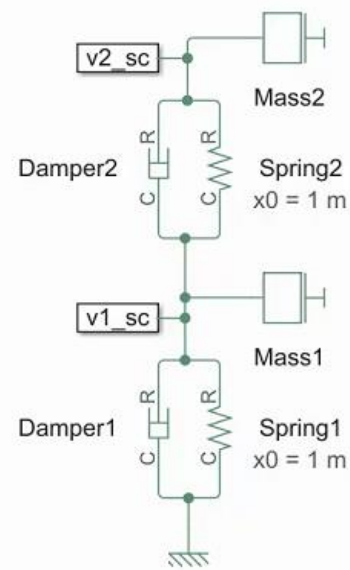


Mass-Spring-Damper Simscape

Simulink Model



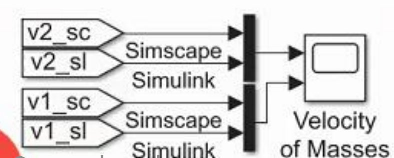
Simscape Model



https://www.mathworks.com/services/training.html?s_tid=gn_trg_ov

Double Mass-Spring-Damper in Simulink and Simscape

1. Plot position of masses (see code)
2. Explore simulation results using sscexplore
3. Learn more about this example
4. Open model of single mass-spring-damper
5. Learn more about modeling physical networks



Mass-Spring-Damper