**Tutorial 3**

Simulation of Dynamic Systems

## Aim

- To simulate zero-order systems and first-order systems
- To get familiar with Simulink
- To improve programming skills with Simulink

## Learning Objectives

Upon completion of this tutorial we will be able to

- Explain dynamics of zero-order systems
- Explain dynamics of first-order systems
- Solve first-order ODEs with Simulink
- Create a subsystem

## Hands-on Exercises

### Exercise 1 Zero-order systems (ideal systems)

A zero-order system is expressed by the following equation:

$$y(t) = Ku(t) \tag{1}$$

where K is gain, y(t) is output and u(t) the input. Make a Simulink model to simulate (1).

## SOLUTION

- Open a Simulink model and save as "ZeroOrderSysSimTute03_01.mdl"

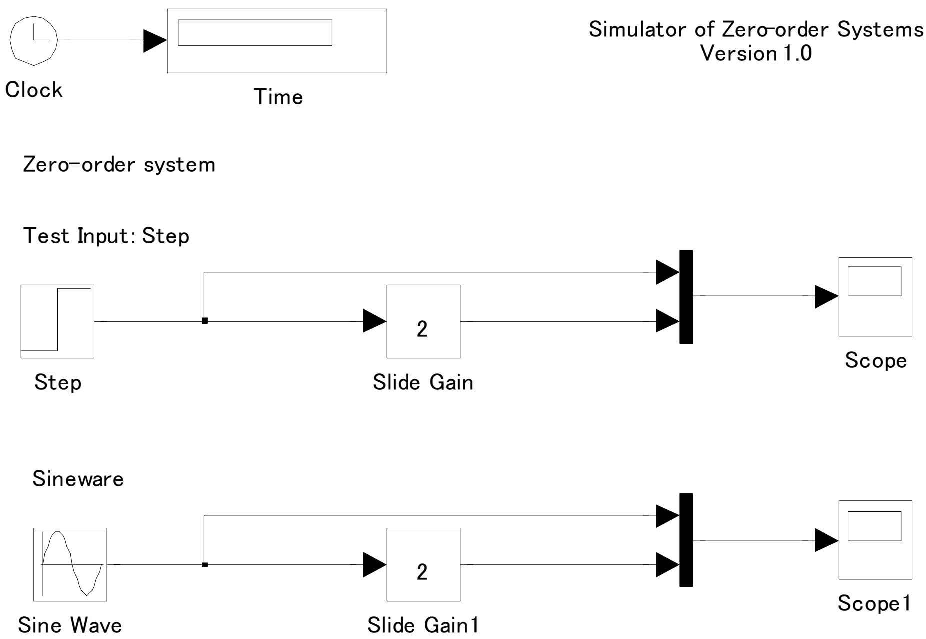
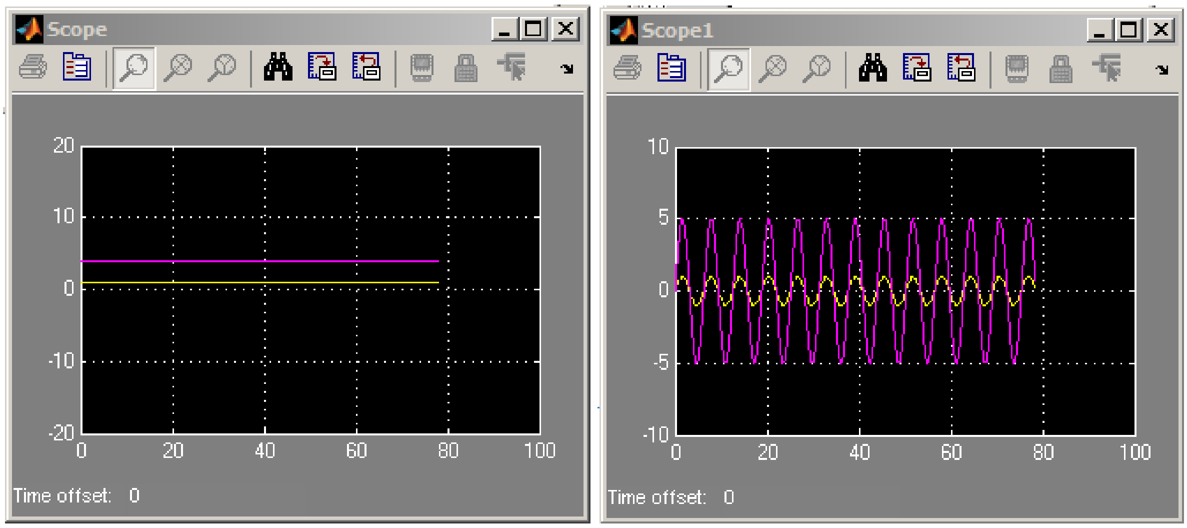



**Figure 1** Simulink models for zero-order systems

The resulting Scopes for K = 4 (step function) and K = 5 (sine wave input) are shown in **Fig. 2**.




(a) Step function response (b) Frequency response

**Figure 2** Zero-order system responses

**Exercise 2 First-order Systems**

**State of Problem:** A storage tank system is shown in **Fig. 3**. By applying the mass balance principle in Fluid Mechanics the relationship between the level (h) and the inlet flow rate ($m^3$/min) is derived as follows:

$$A\dot{h} + K_v\sqrt{h} = q_{in} \qquad (2)$$

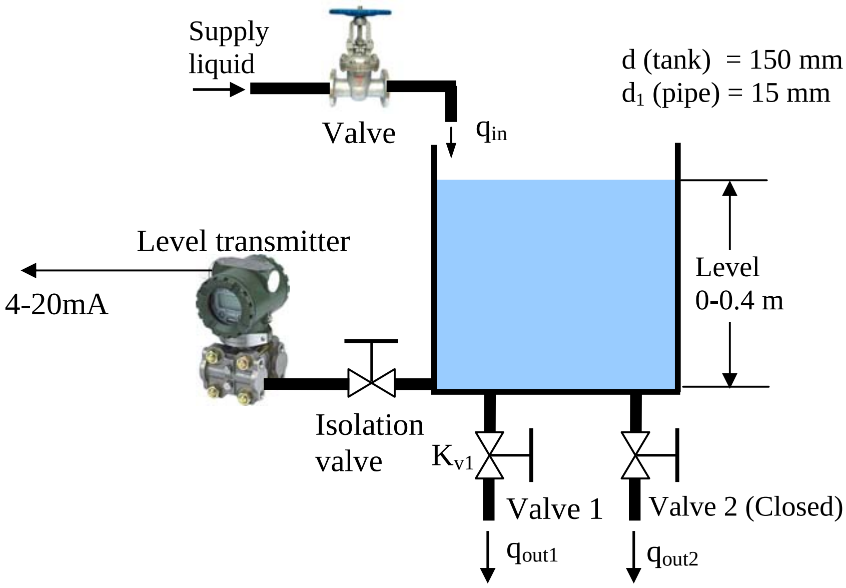



**Figure 3** Tank level system

Equation (2) is a non-linear differential equation. It is hard to solve ODEs with analytical methods. With the aid of computer and software a non-linear differential equation can be solved using numerical integration methods. Make a simulation program with Simulink to solve Equation (2). Use the following numerical values: $K_{v1}$ = 0.000187, d = 150 mm, level in range of 0 to 400 mm, $q_{in}$ in range of 0 to 0.0071 $m^3$/min. In the Simulink model, do the following tasks:

- Convert the flow rate $q_{in}$ from $m^3$/min to $m^3$/s
- Convert the level h from m to mm
- Set low alarm limit (20 mm) and high alarm limit (380 mm) for the level
- Simulate a level transmitter providing a level signal in range of 4 to 20 mm.

**SOLUTION**

**Block Diagram Algorithm**

Using the SI units the tank level system is represented by the block in **Fig. 4**.

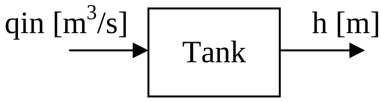



**Figure 4** Block diagram of the tank level system

Equation (2) for the tank level system can be rewritten as follows for a Simulink program:

$$\dot{h} = \frac{1}{A}\left(-K_{v1}\sqrt{h} + q_{in}\right) \qquad (3)$$

In order to solve equation (3) we develop a block diagram algorithm below:

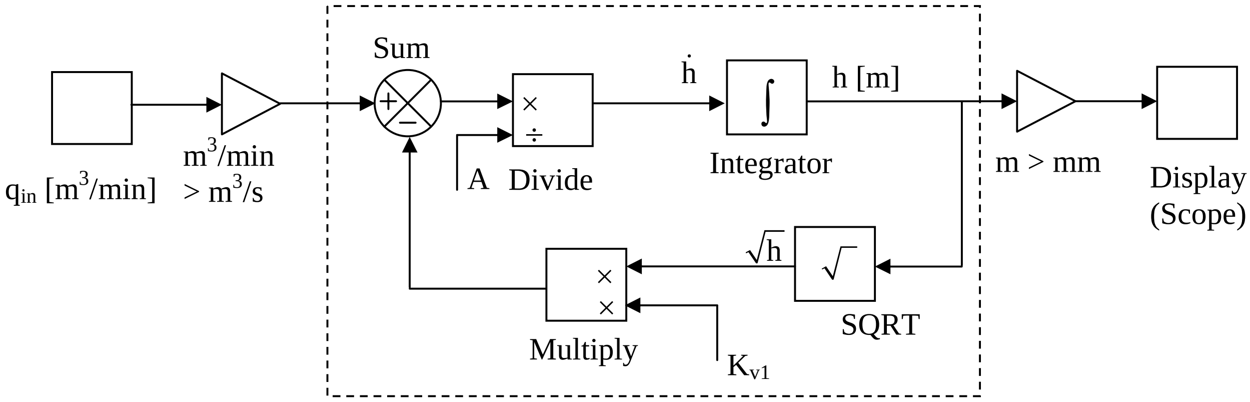



**Figure 5** Block diagram algorithm for solving a first-order differential equation with Simulink

The cross-sectional of the tank is calculated by the following equation:

$$A = \pi\frac{d^2}{4} \ \text{[m]} \qquad (4)$$

## Programming

- Open a Simulink model and save as “FirstOrderSysSim_Tute03_02_01.mdl”.
- Using the above block diagram and make a Simulink model for solving equation (3). The Simulink model may look like that in **Fig 6**.

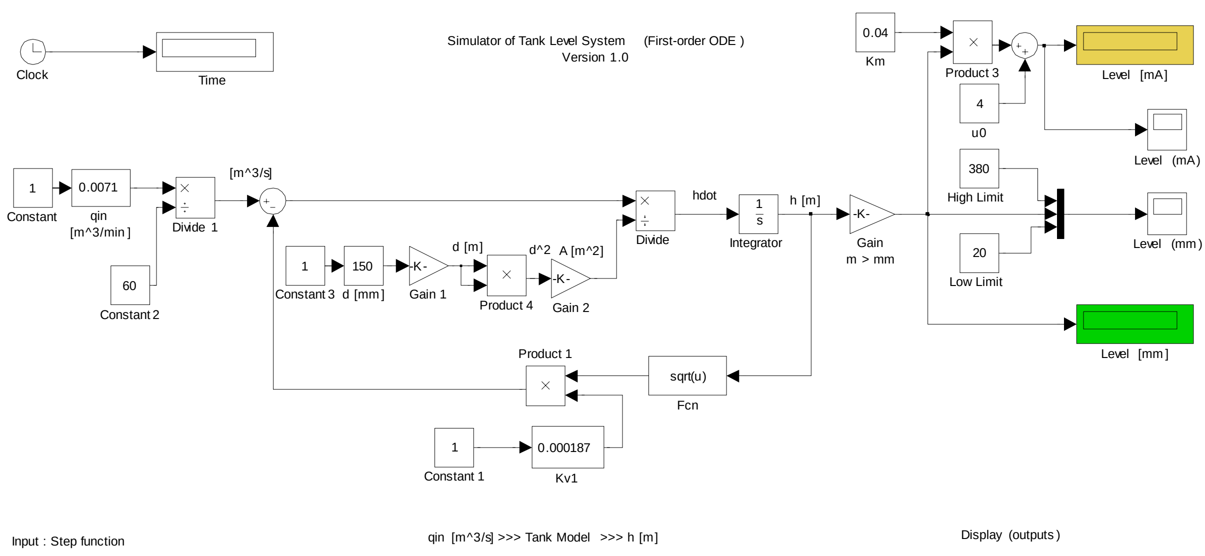



**Figure 6** Simulink model to solve a first-order ODE

- Save the model.

## System Parameters

- Set all system parameters for the model.
- Set initial conditions for the Integrator block by double-clicking it and set:
  - h(0) = 0, check Limit output (upper: 0.4 m, lower: 0.0)

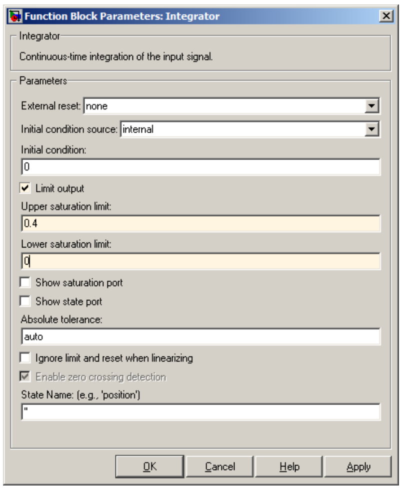



**Figure 7** Set initial conditions for the Integrator block

- Save the model. The resulting program looks like...

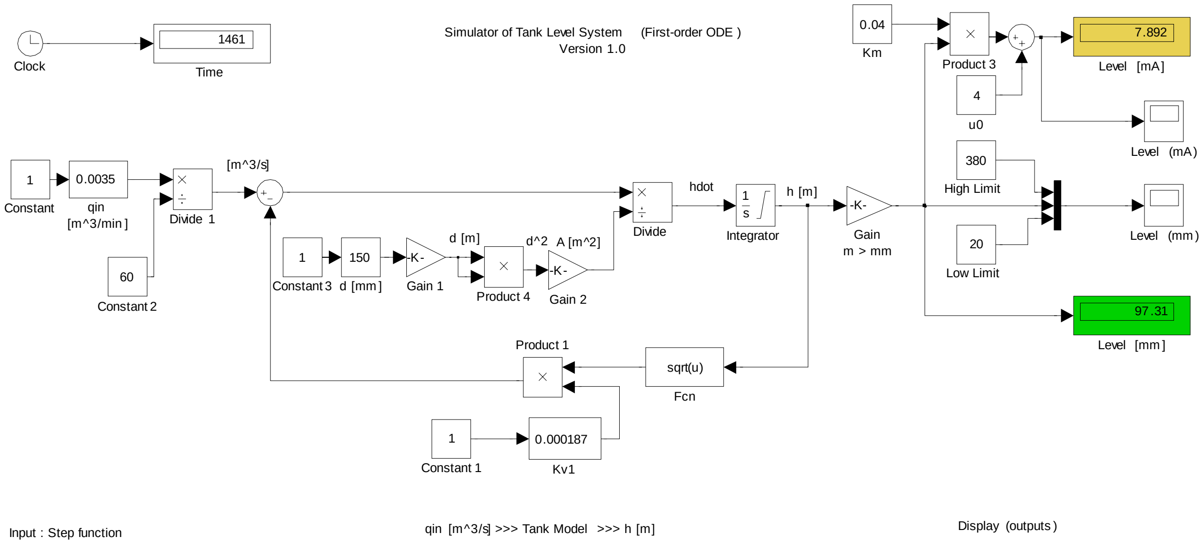



**Figure 8** Simulink model after setting the Integrator block

## Simulation Parameters

- Set the simulation parameters as in the following figure
- Click the OK button.

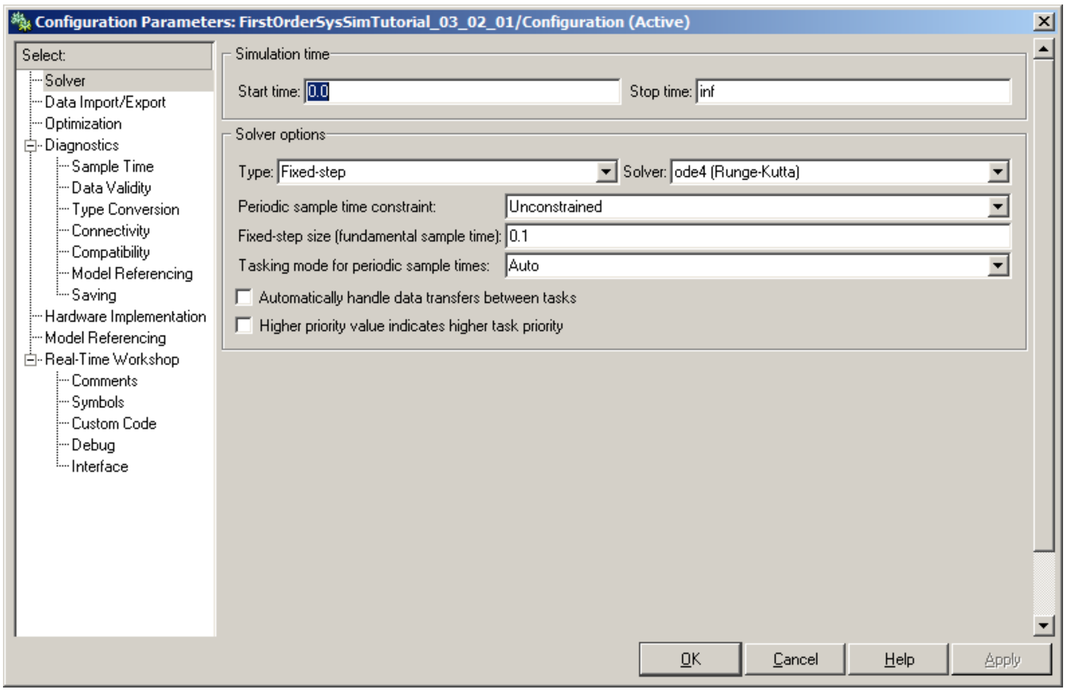



**Figure 9** Configuration parameters

- Save the model.
- Run the model and test its functionality. The results (for $q_{in}$ = 0.0035 $m^3$/min) are shown in **Fig. 10** and **Fig. 11**.

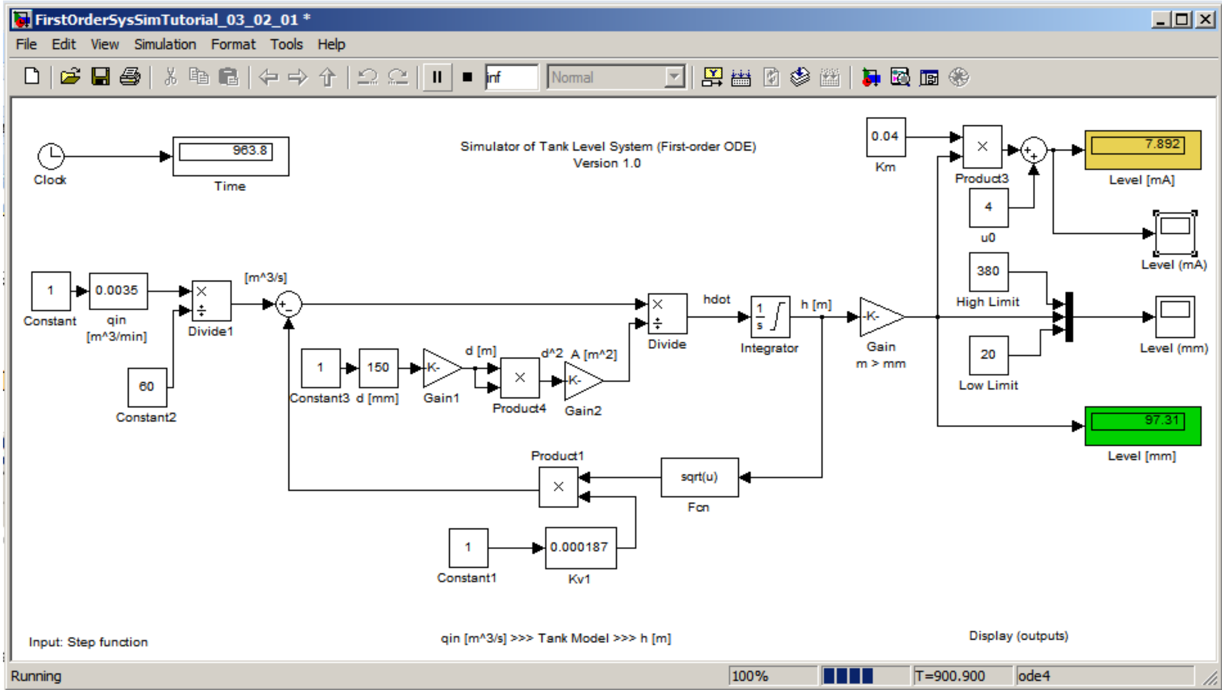



**Figure 10** The Simulink model is running for $q_{in}$ = 0.0035 $m^3$/min

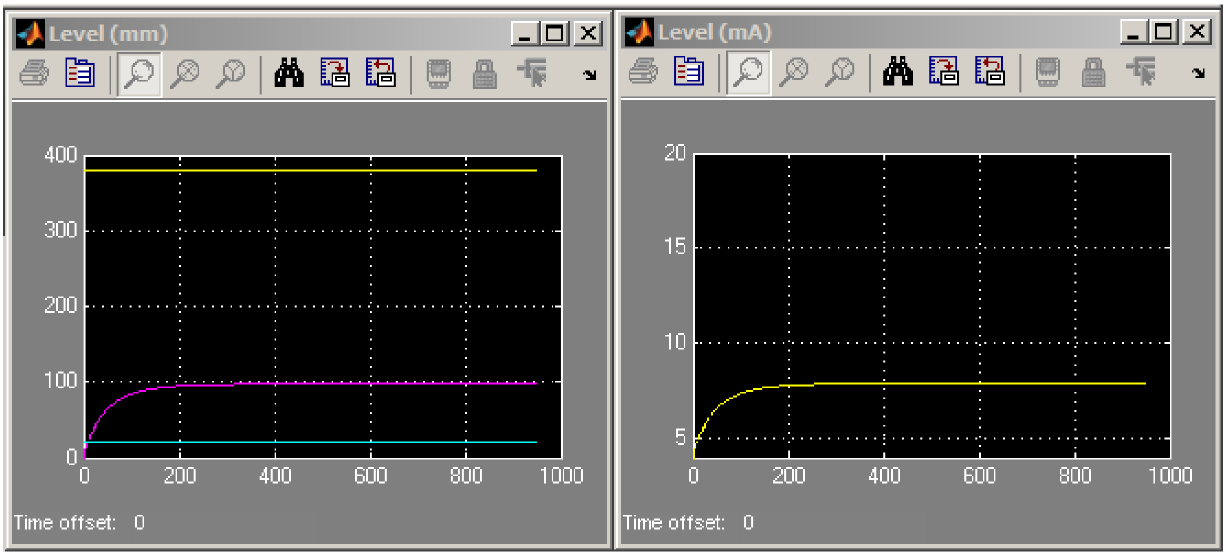



**Figure 11** Resulting Scopes (Level in mm and mA)

**Create a Subsystem** (for cross-sectional area calculating algorithm)

- Select blocks as in **Fig. 12**.

- Edit menu > Create Subsystem (or Right-click the selected blocks > Create Subsystem).

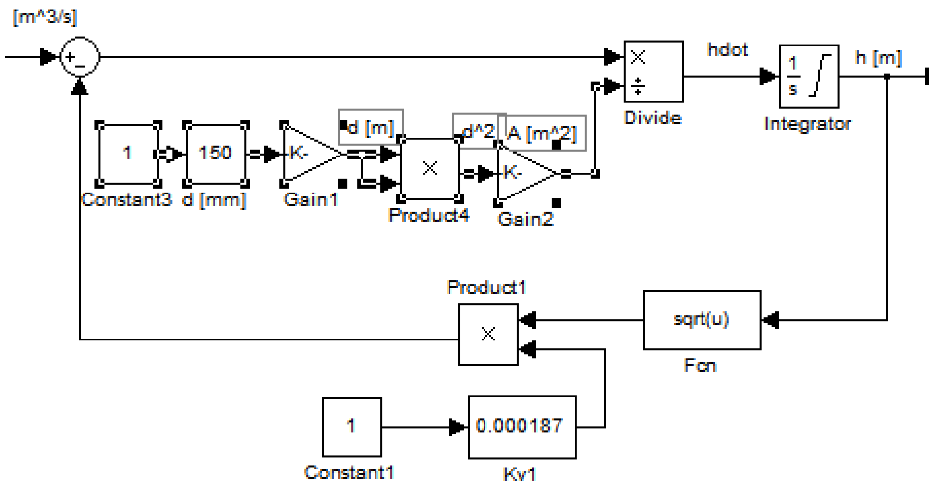



**Figure 12** Selecting blocks

The resulting program looks like...

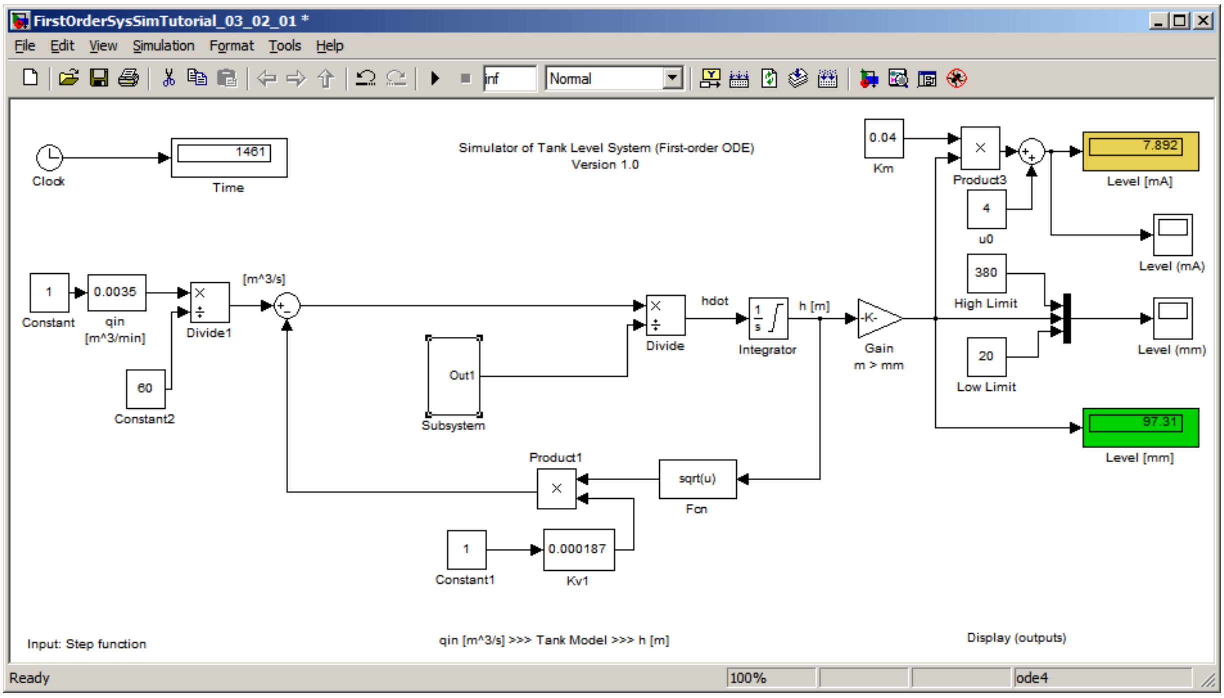



**Figure 13** Created a Subsystem

- Change Subsystem to A, double click the A block, change the output port (Out1) to A as the figure below.

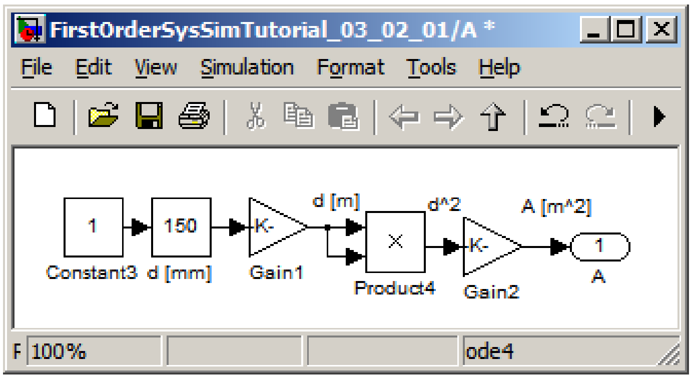



**Figure 9** Subsystem A for cross-sectional area

- Save the model as “FirstOrderSysSimTutorial_03_02_02.mdl”.
- Run the Simulink model and test its functionality. The same results should be obtained.

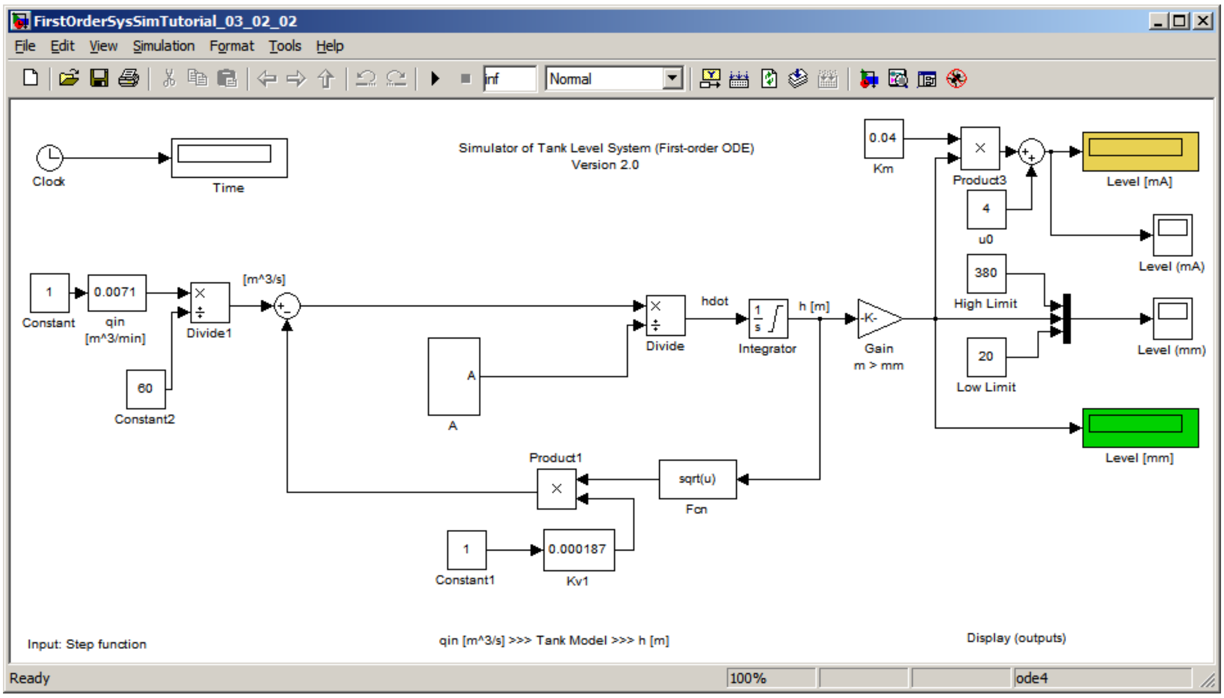



**Figure 10** Simulink model with a Subsystem (A)

## Conclusions

At this point, the following LOs have been met:

- Explain dynamics of zero-order systems
- Explain dynamics of first-order systems
- Solve first-order ODEs with Simulink
- Create a subsystem

**Follow-up Exercises**

**1**. Make a Simulink model to simulate the following RC Circuit expressed by the first-order differential equation:

$$RC\dot{e}_o + e_o = e_i$$

where $e_o$ and $e_i$ are output and input voltages, R and C are resistance and capacitance, respectively. This equation is rewritten in a standard first-order ODE below:

$$T\dot{e}_o + e_o = Ke_i$$

where T (= RC) is *time constant* (seconds), and K (=1) is *sensitivity* (*gain*, V/V).

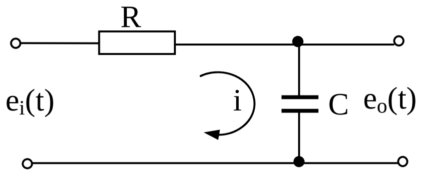



**Figure 11** An RC network

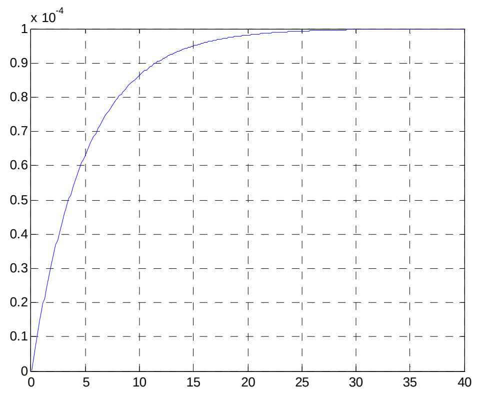
Use these numerical values: R = 10 k$\Omega$; C = $10^{-4}$ $\mu$F ; $e_i(t) = A_a\sin(2\pi f_a) + A_b\sin(2\pi f_b) + V_i$

Note: Use may compare results with those of the following RC circuit simulator:
http://techteach.no/simview/rc_circuit/index.php

**2.** A temperature transmitter is represented by a first-order differential equation below:

$$\tau\dot{y} + y = Ku$$

where $\tau$ is the *time constant* (5 sec), K the *sensitivity*/gain ($10^{-7}$ V/$^o$C), y and u are the output voltage (V) and input temperature ($^o$C), respectively. The input temperature is a step function having initial value of 0 $^o$C and final value of 100 $^o$C. Make a Simulink model to simulate the temperature transmitter displaying temperature in mA and $^o$C. Save the simulated results (input temp and output voltage). The resulting data should look like:




You can change the value of $\tau$ (e.g. 10 sec, 15 sec), run the simulation program. What is the value of output (percentage of span) when t = $\tau$? Suggest a method to estimate $\tau$ and K from the simulated data.