

# ME40064: System Modelling & Simulation

## ME50344: Engineering Systems Simulation

### Lecture 16

Dr Andrew Cookson  
University of Bath, 2019-20

# LECTURE 16

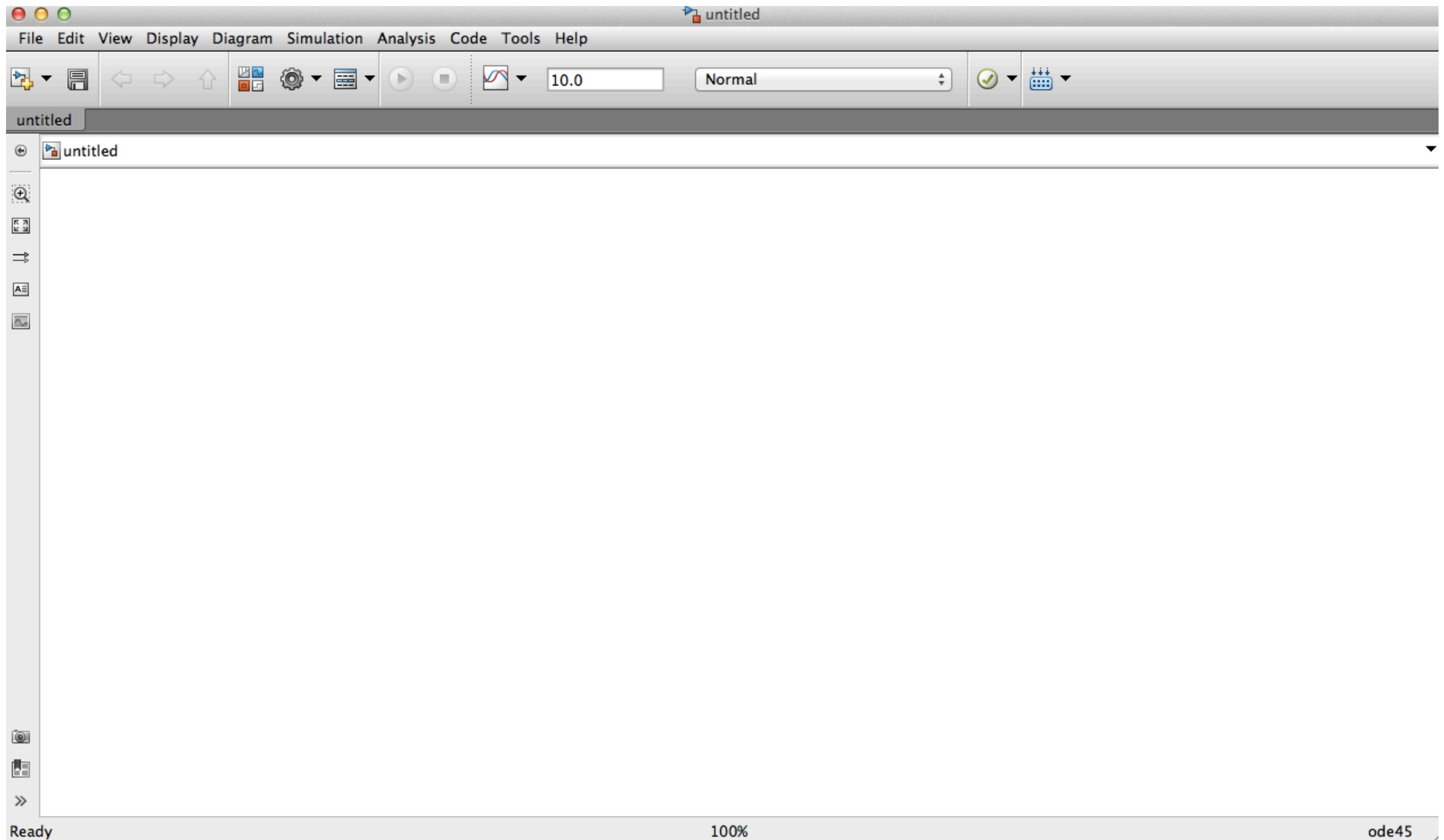
## Introduction To Using Simulink

- Introduction to Simulink
- Ability to construct and solve a block diagram model in Simulink

# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

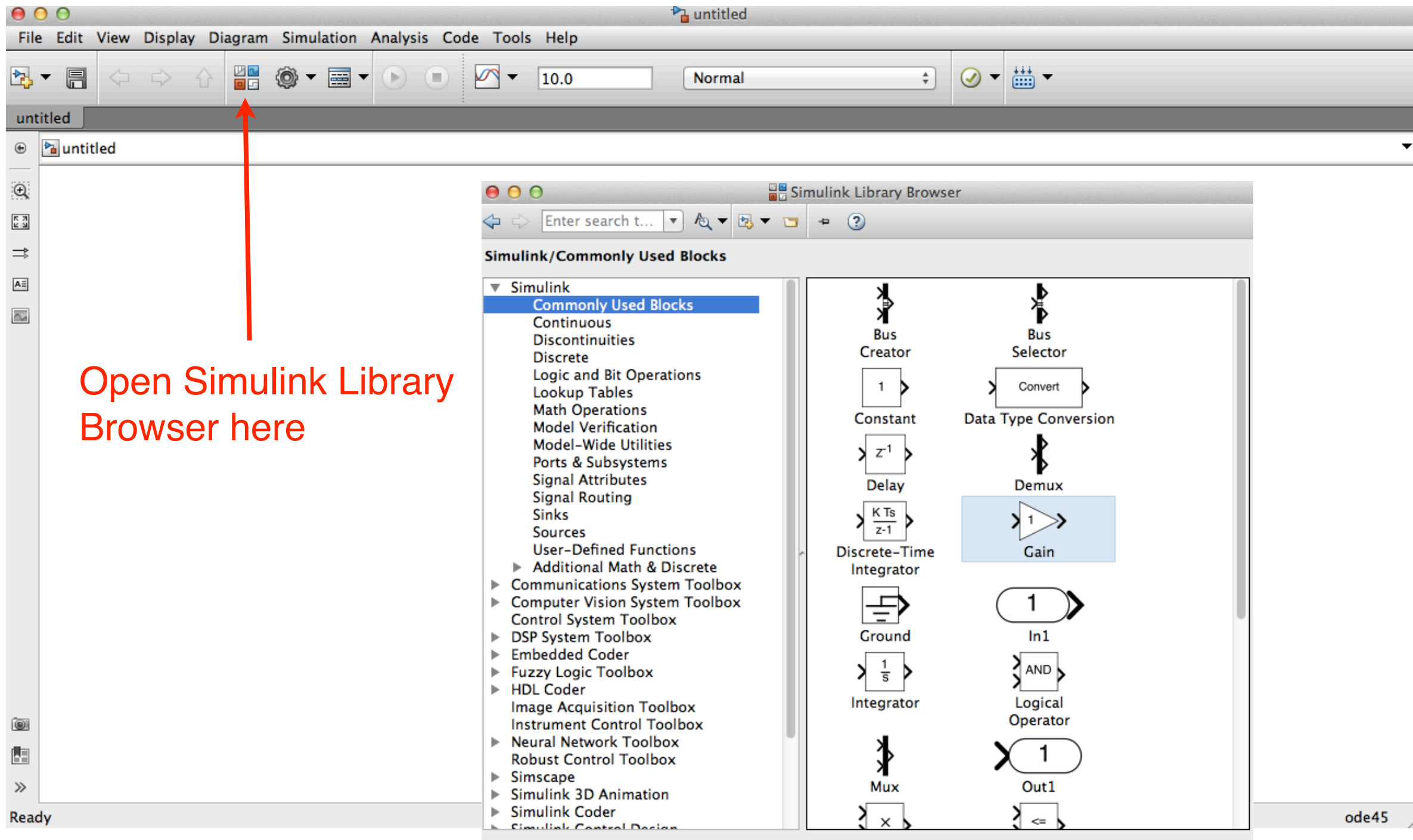
Open a blank Simulink model:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

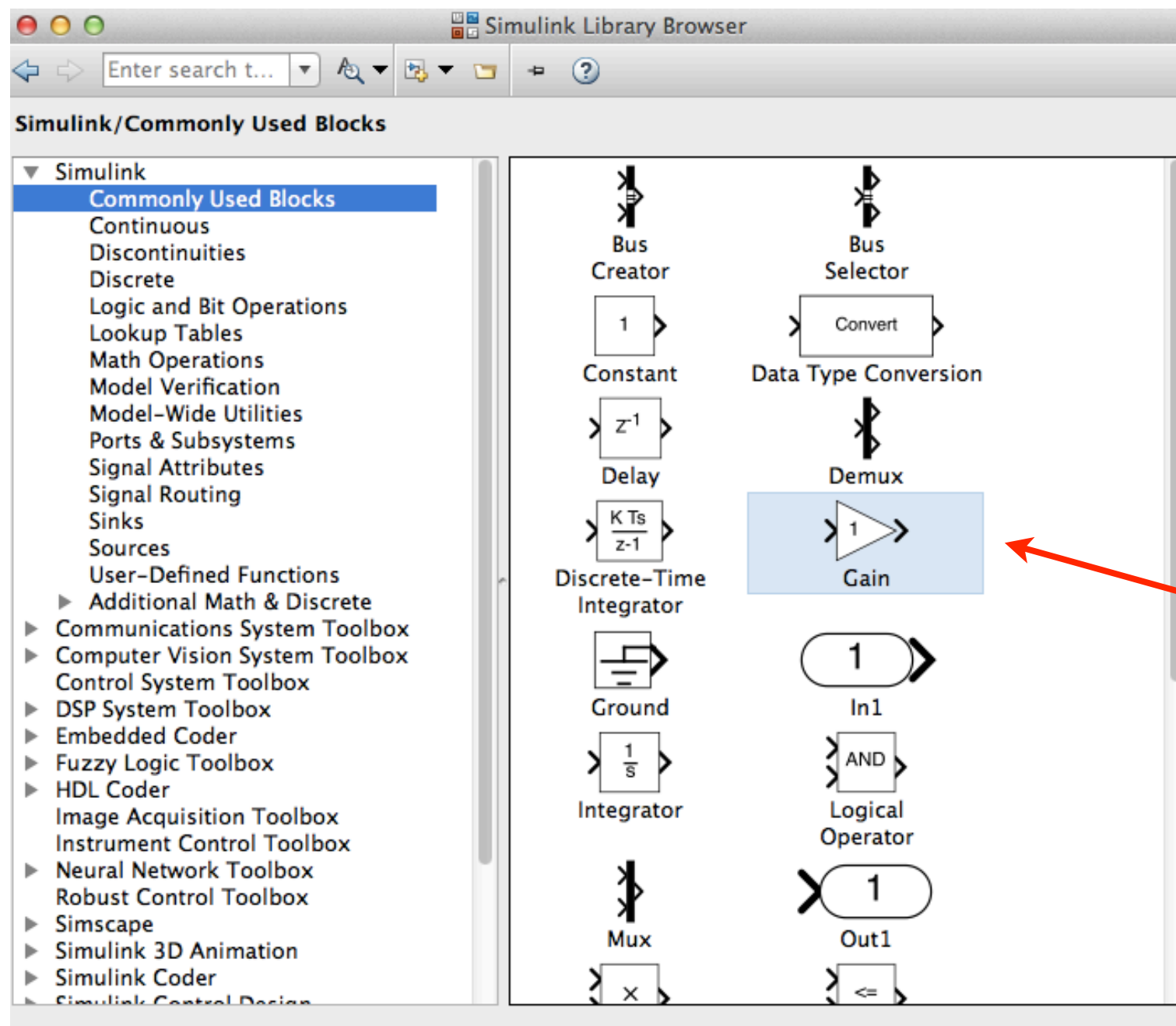
Open the Simulink Library Browser:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

We will start by adding a gain block, which will represent the mass of the system:

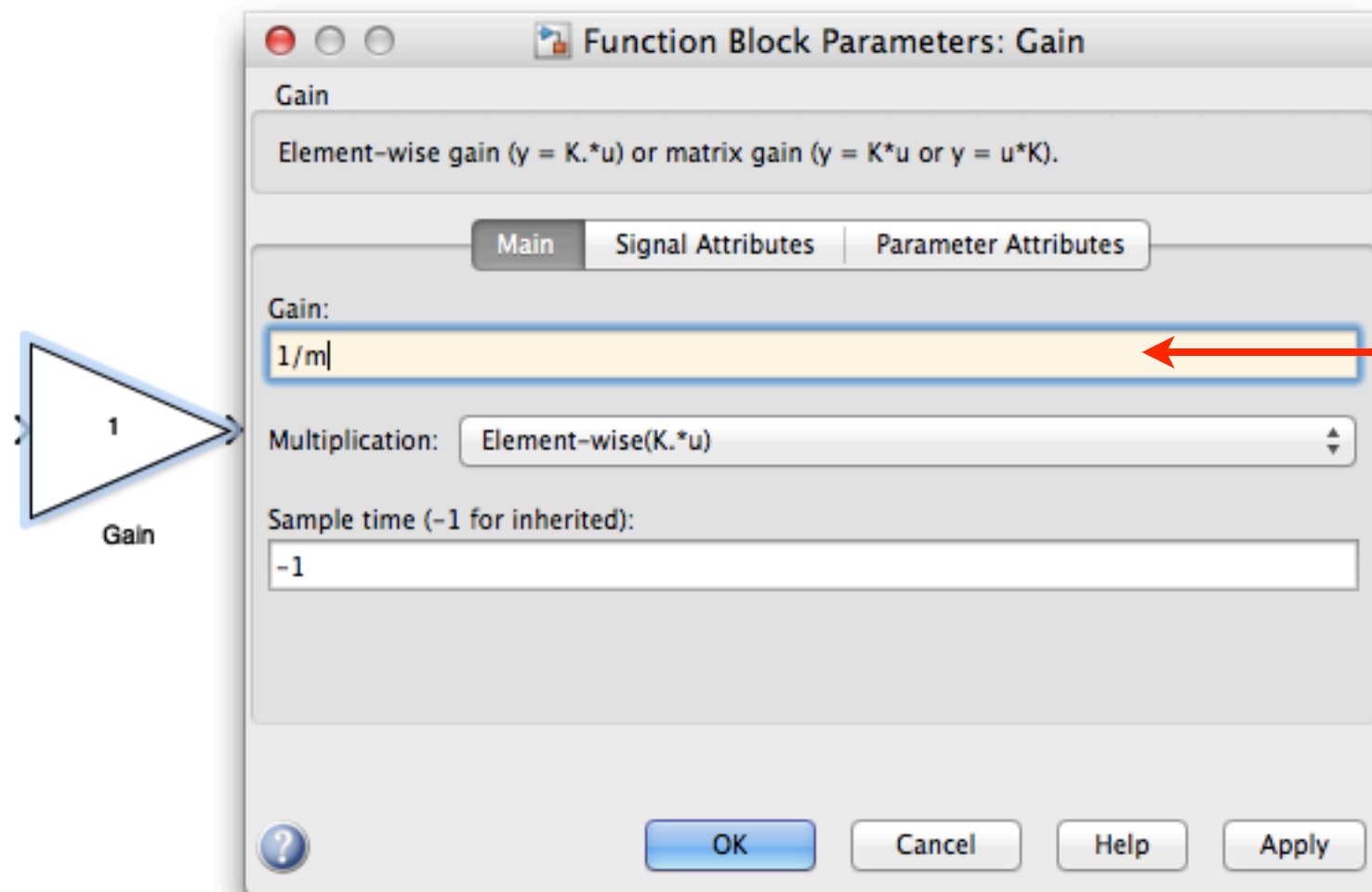


Click and drag onto your Simulink model

# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

We will start by adding a gain block, which will represent the mass of the system. Double click on the block to access the parameter setting menu:



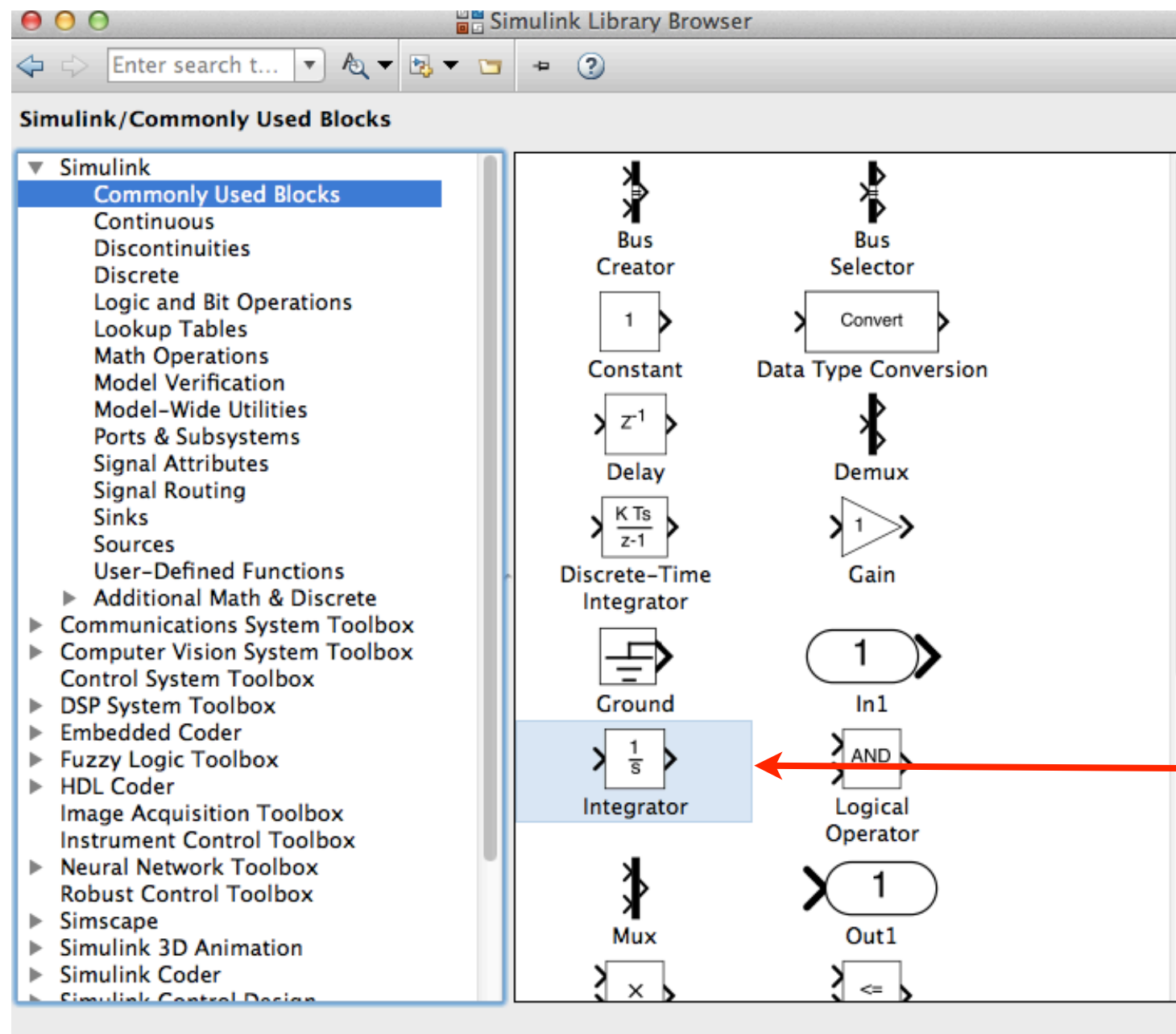
Set the gain parameter to be 1/m



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

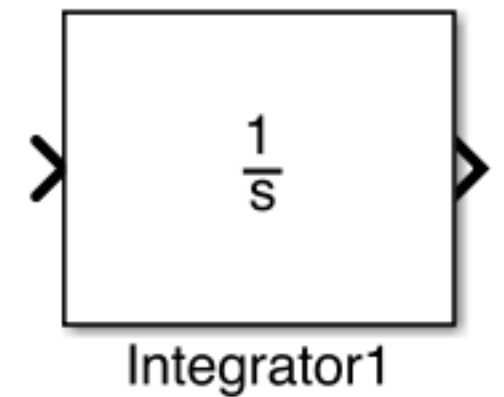
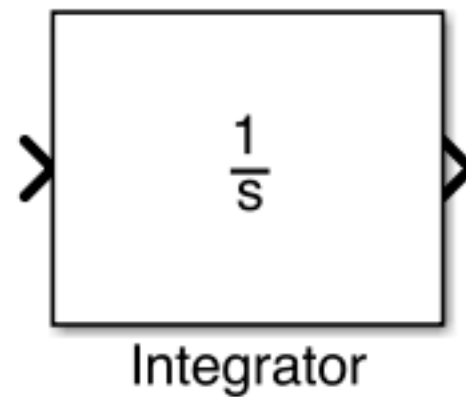
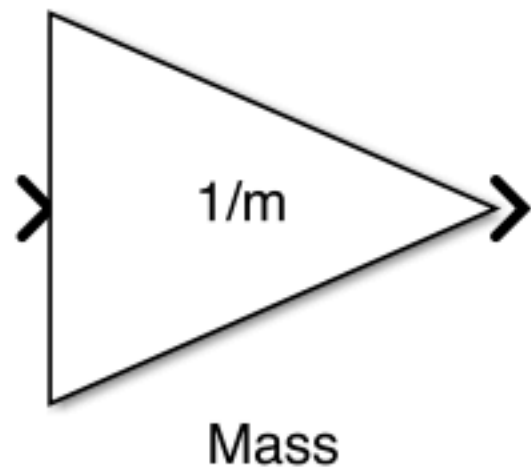
The output of this gain block is acceleration - therefore need to add integrator blocks to produce velocity and position:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

Click and drag two of these blocks onto your model:

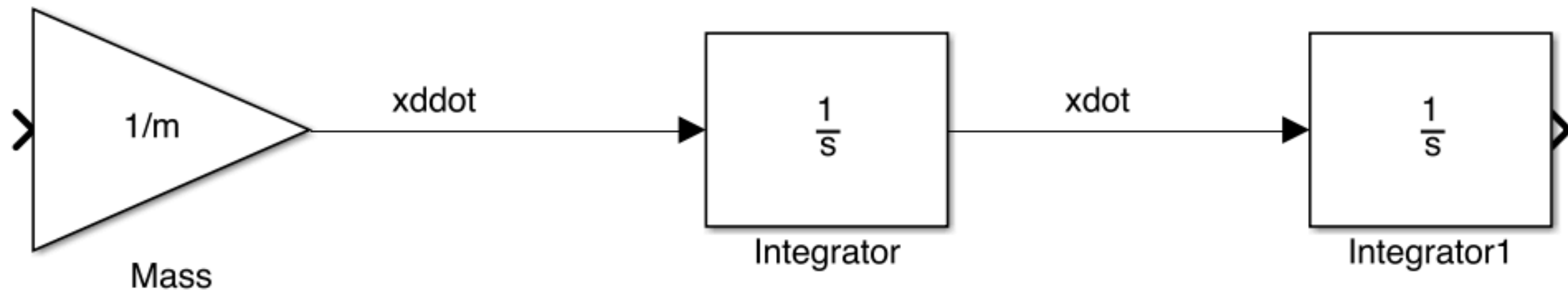
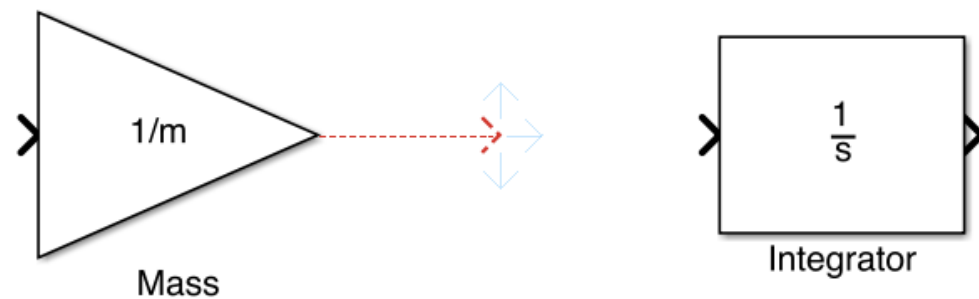




# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

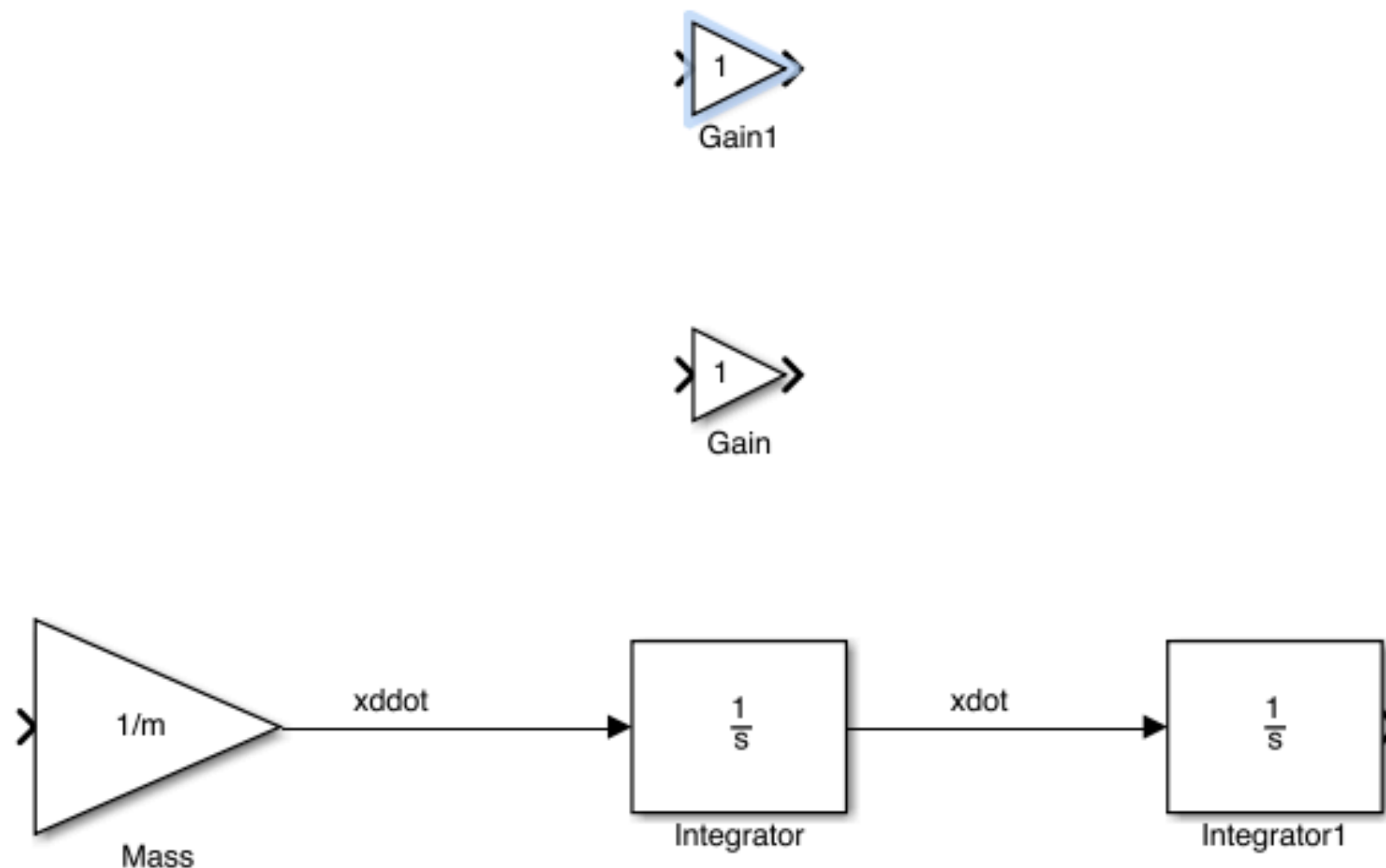
Now want to click and drag arrows to connect the blocks:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

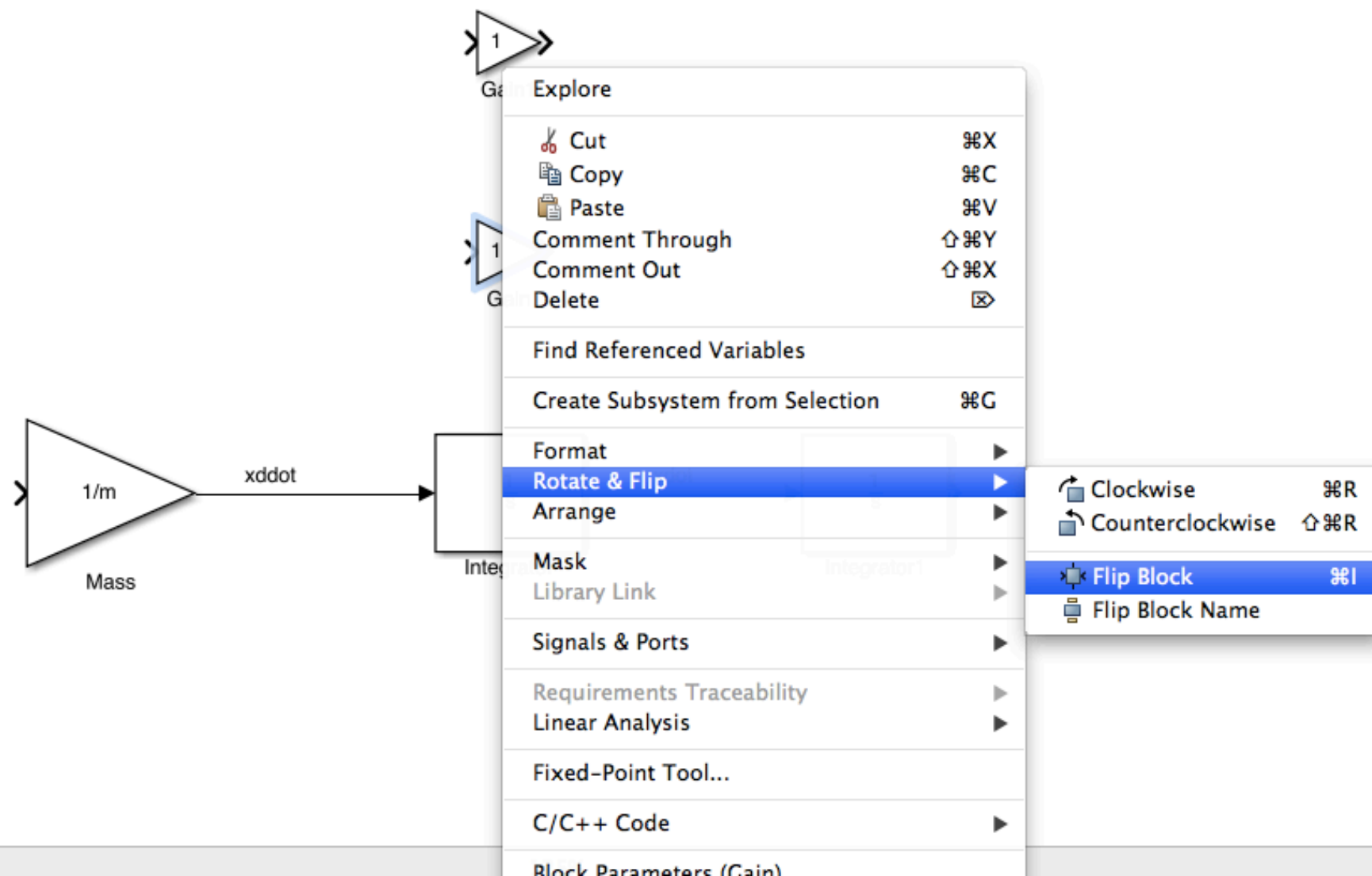
Now place two additional gain blocks onto your model - these will represent the spring stiffness and damping components of the physical model:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

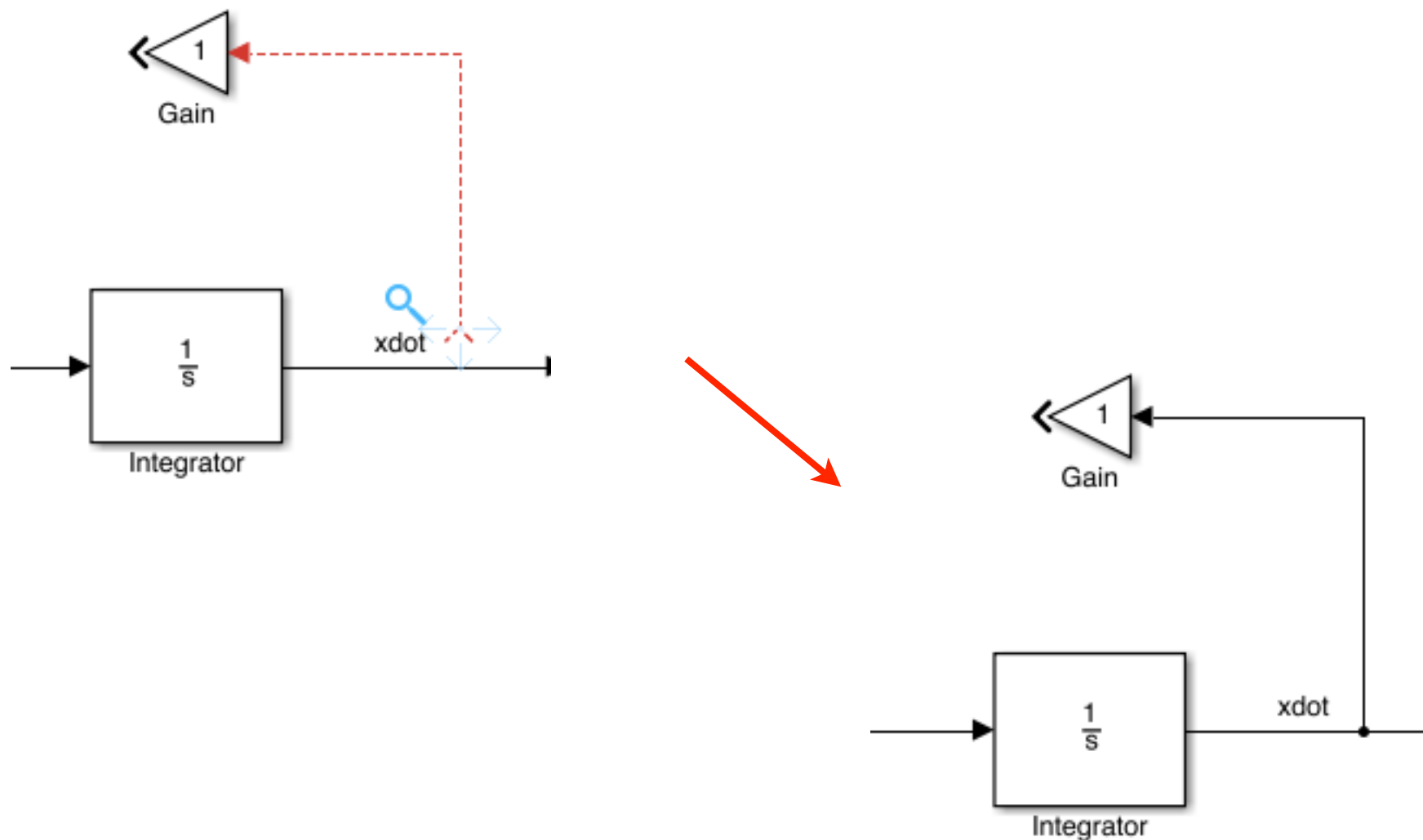
Need to flip these two gain blocks so that they follow the flow of the feedback loops. Right click on the gain block and select the Rotate & Flip option:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

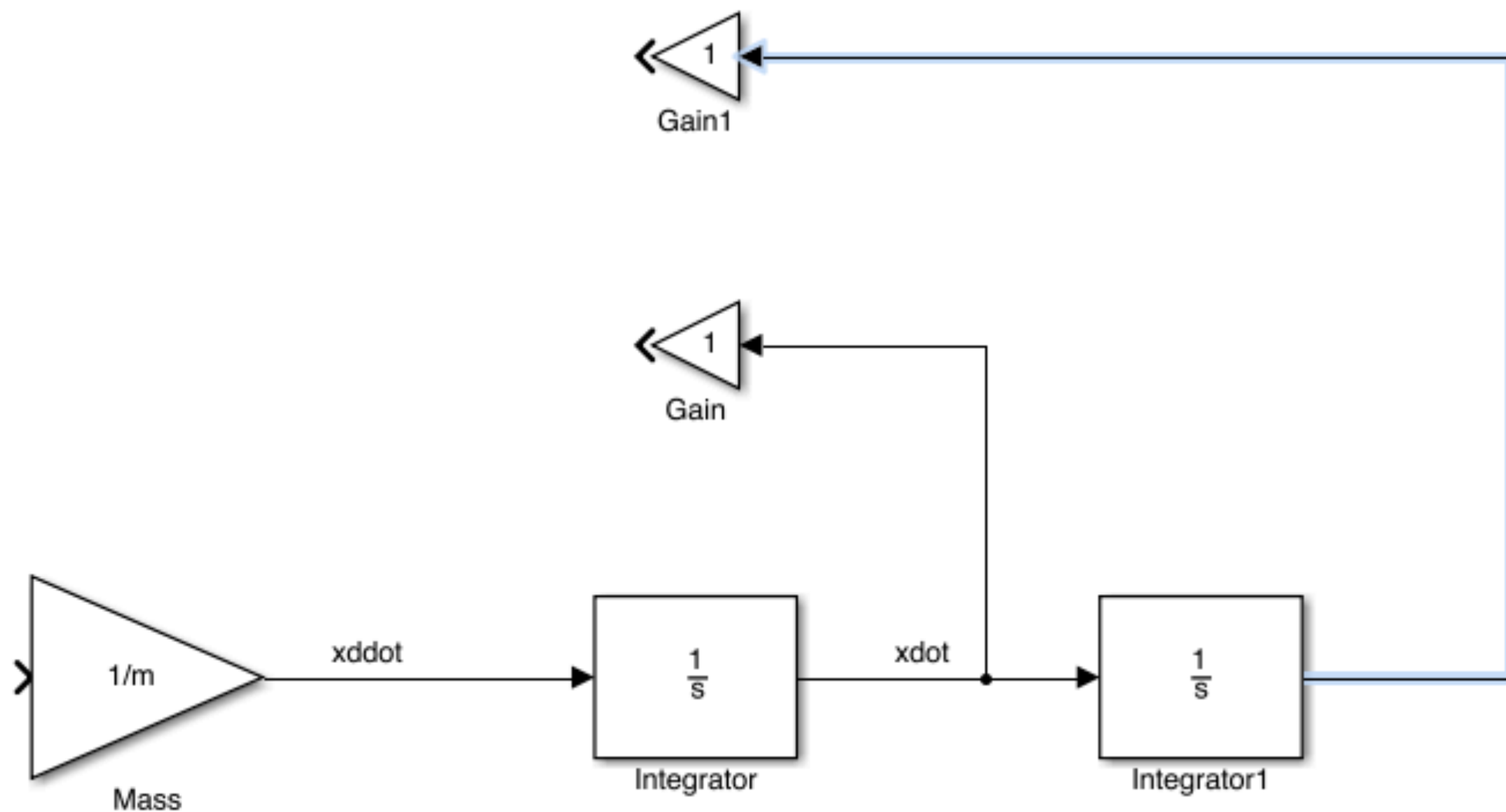
Now drag arrows to connect these gain blocks to the existing signal paths:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

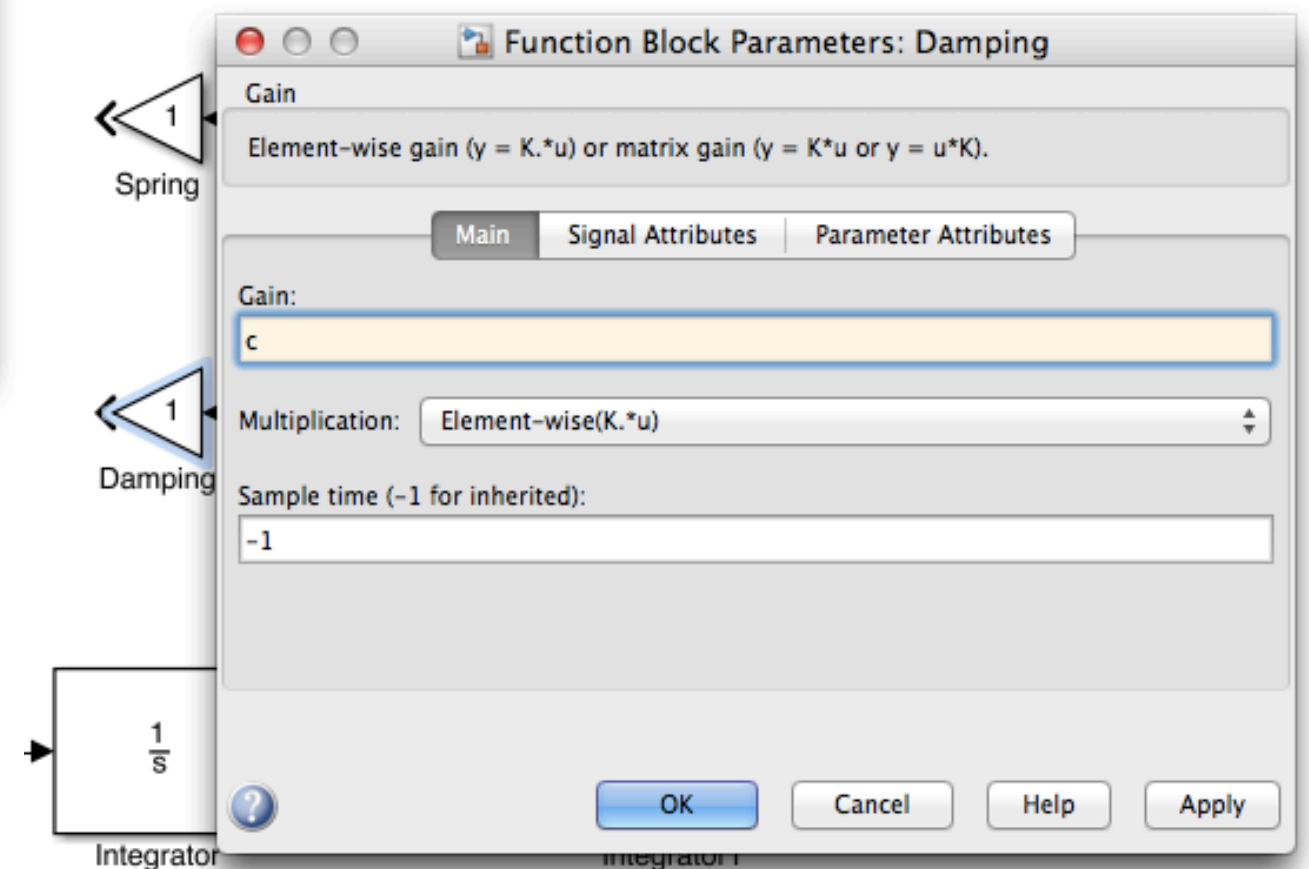
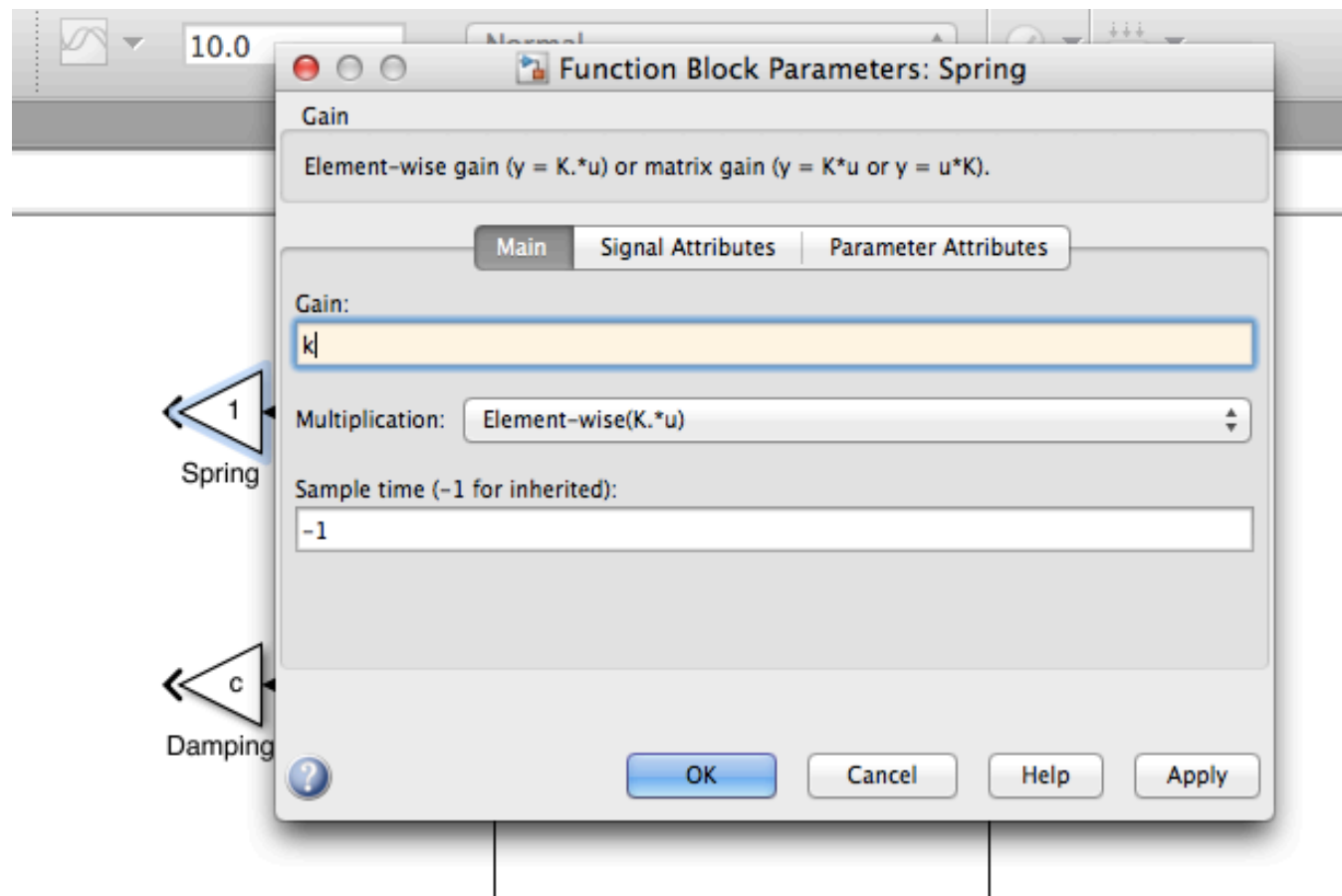
Now drag arrows to connect these gain blocks to the existing signal paths:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

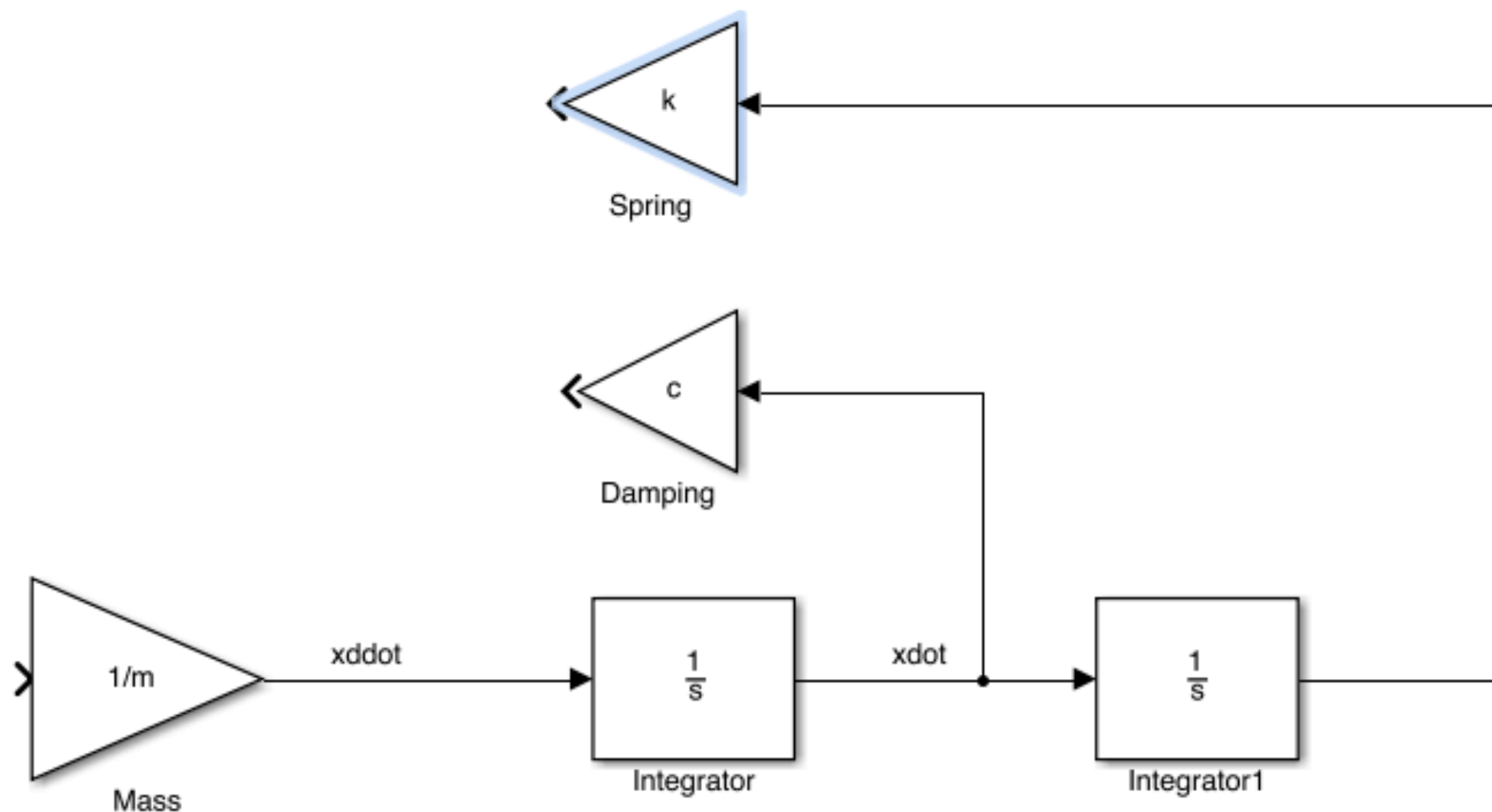
Now double-click to set the parameters for the spring and damper blocks:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

The outputs of these gain blocks are the spring force and damping force - these need to be connected as inputs to the system:

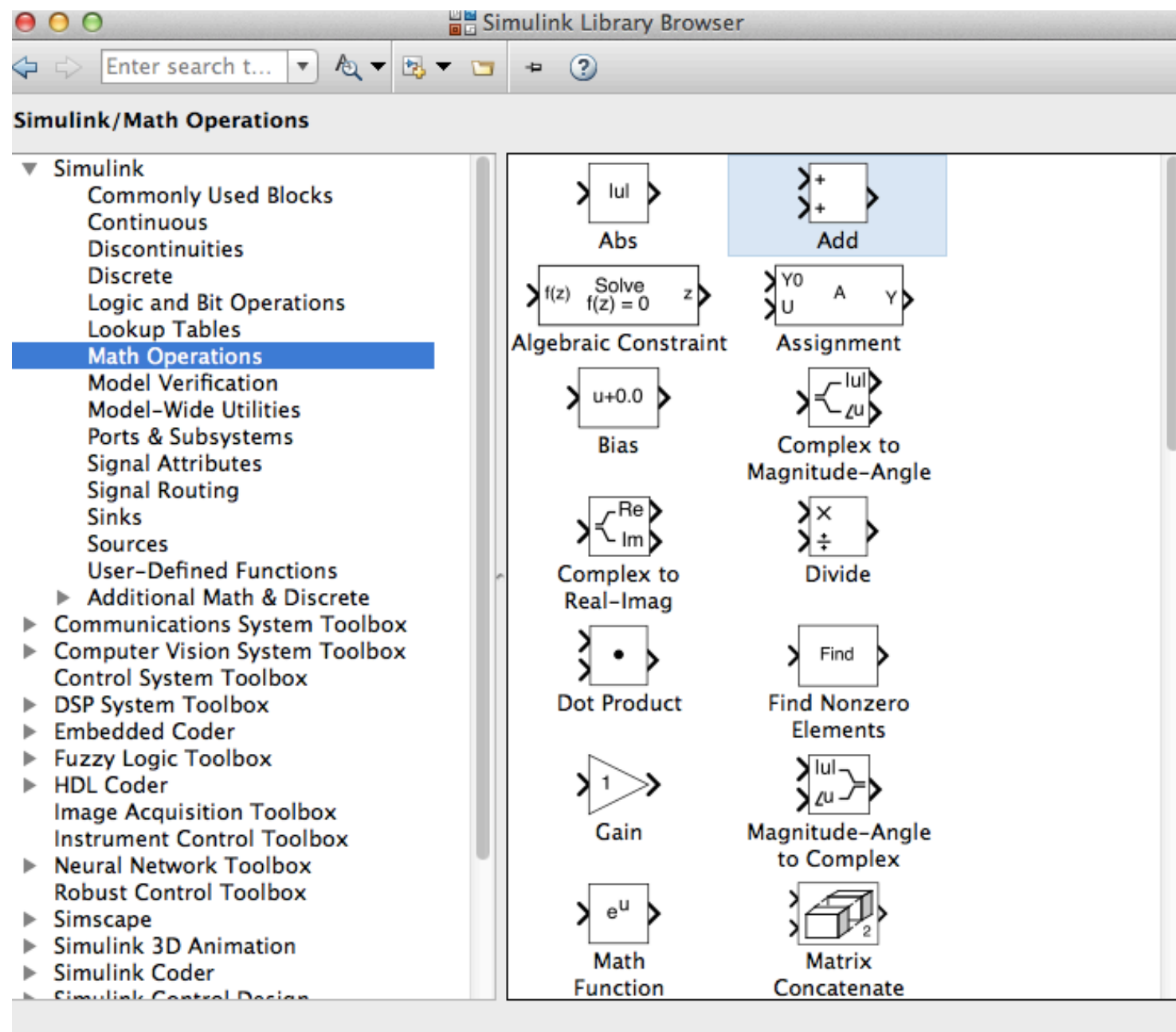




# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

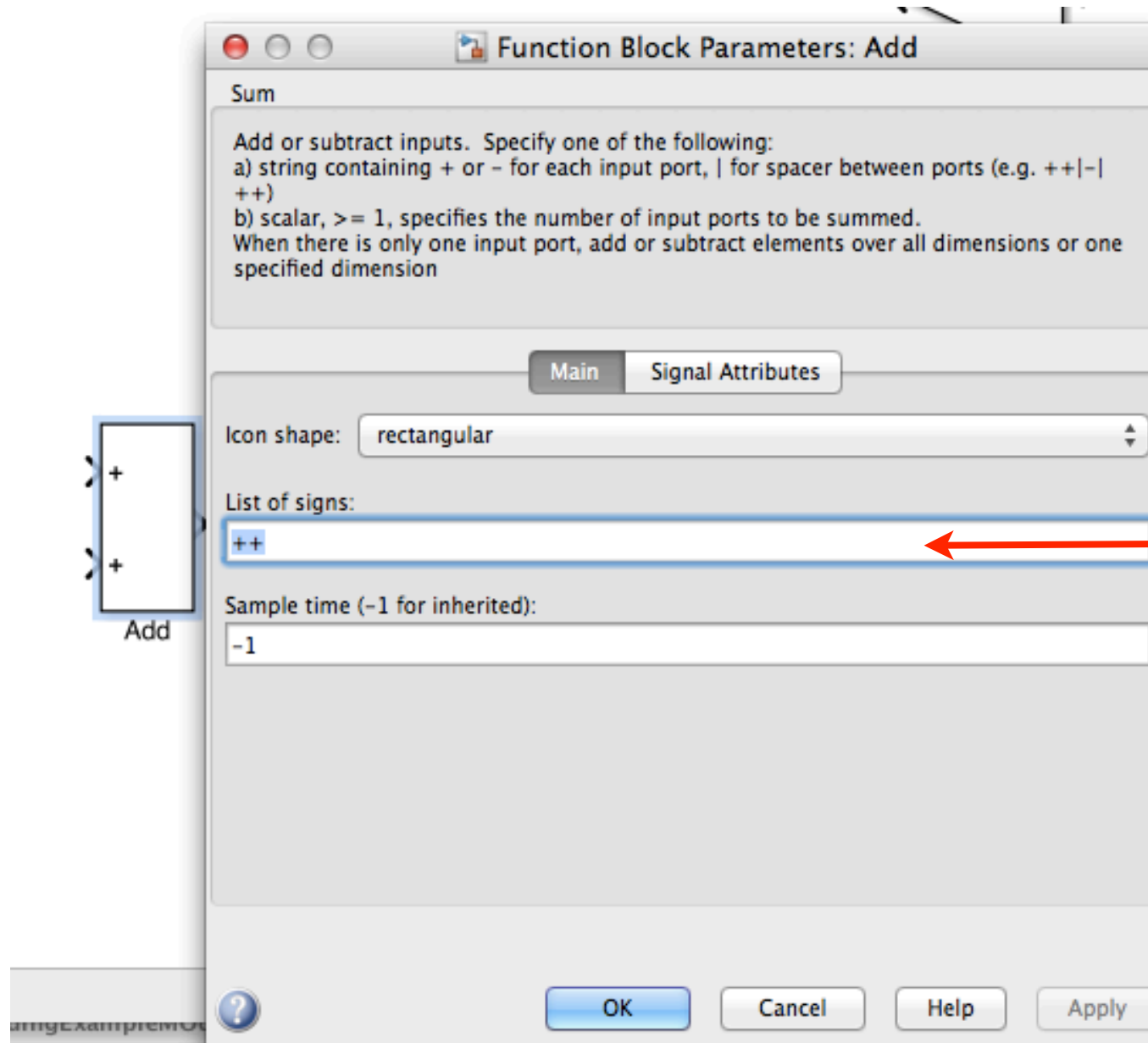
Select an Add block from the library browser - this will encode the arithmetic in our differential equation:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

Double-click on the block to set the arithmetic of the Add block:

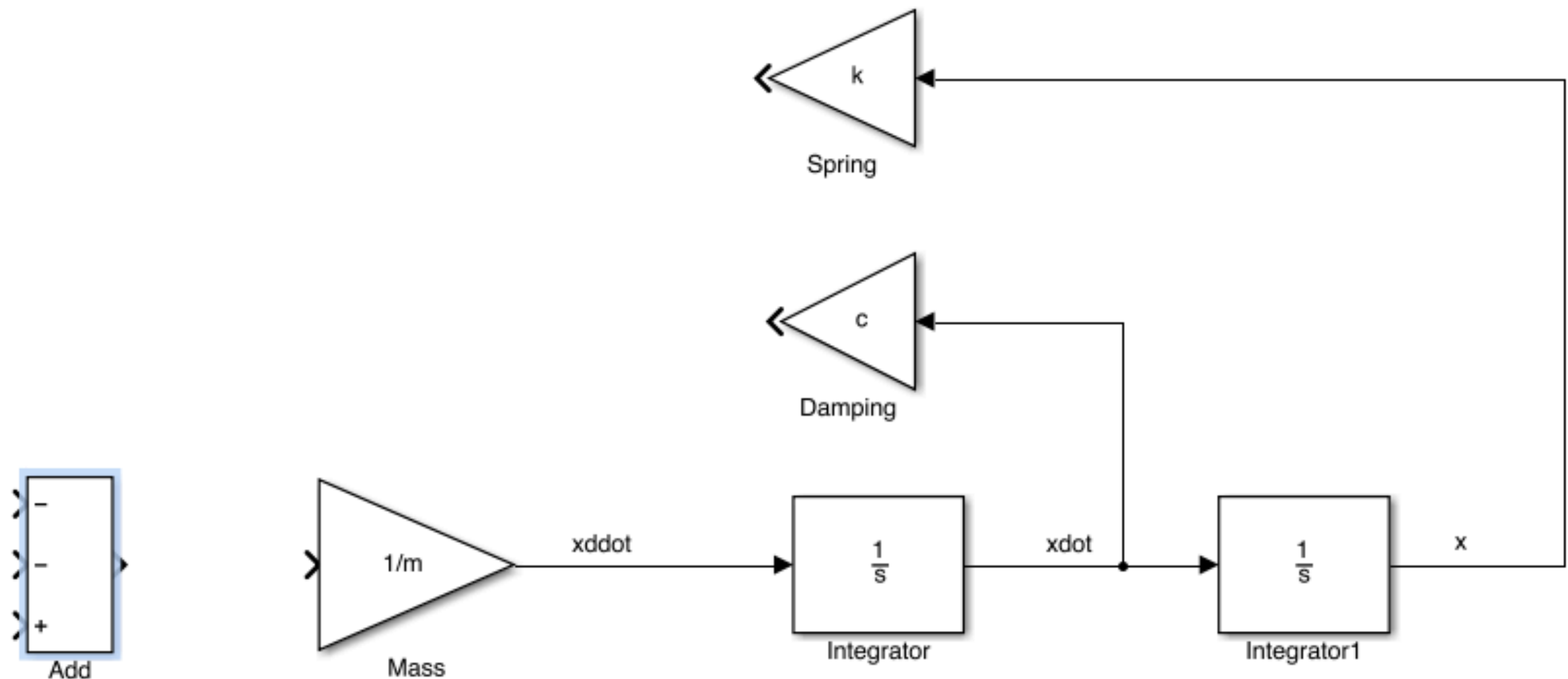


Edit the List of Signs to  
be: --+

# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

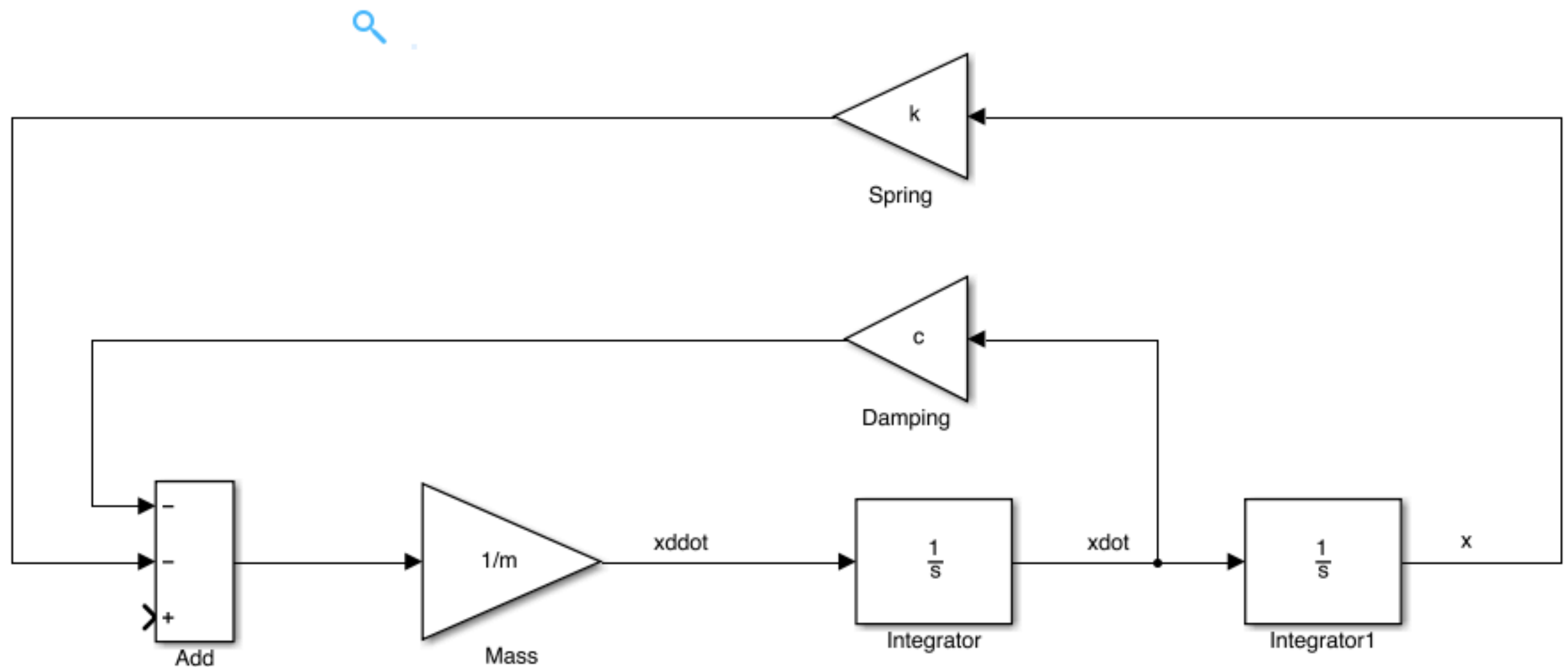
This modification is now displayed on the block within the model:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

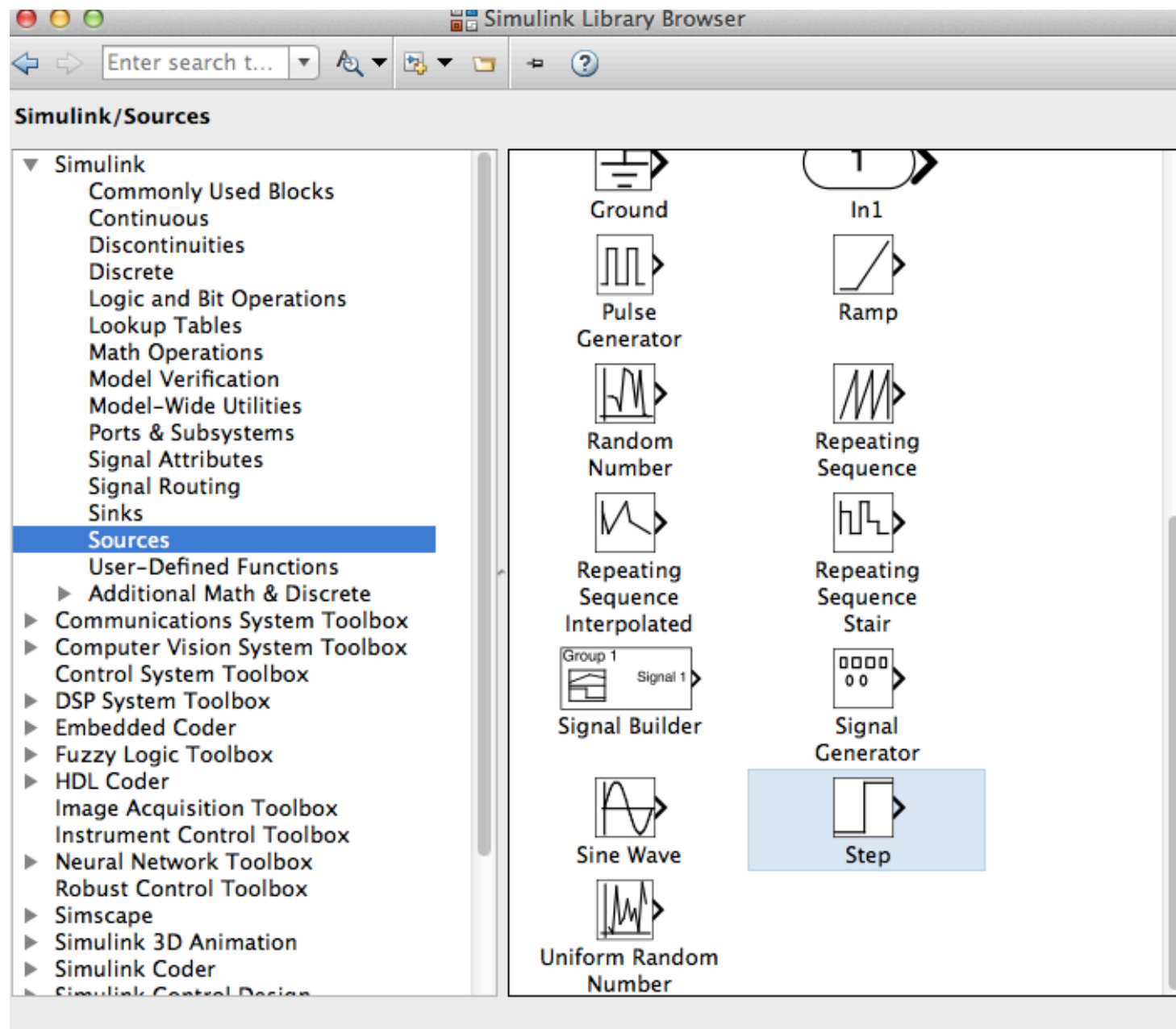
Connect the gain blocks to the add block - these forces are now being applied to the mass:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

However, we also want to apply an external loading to the model. We will use a source block to do this - in this case, the step function:

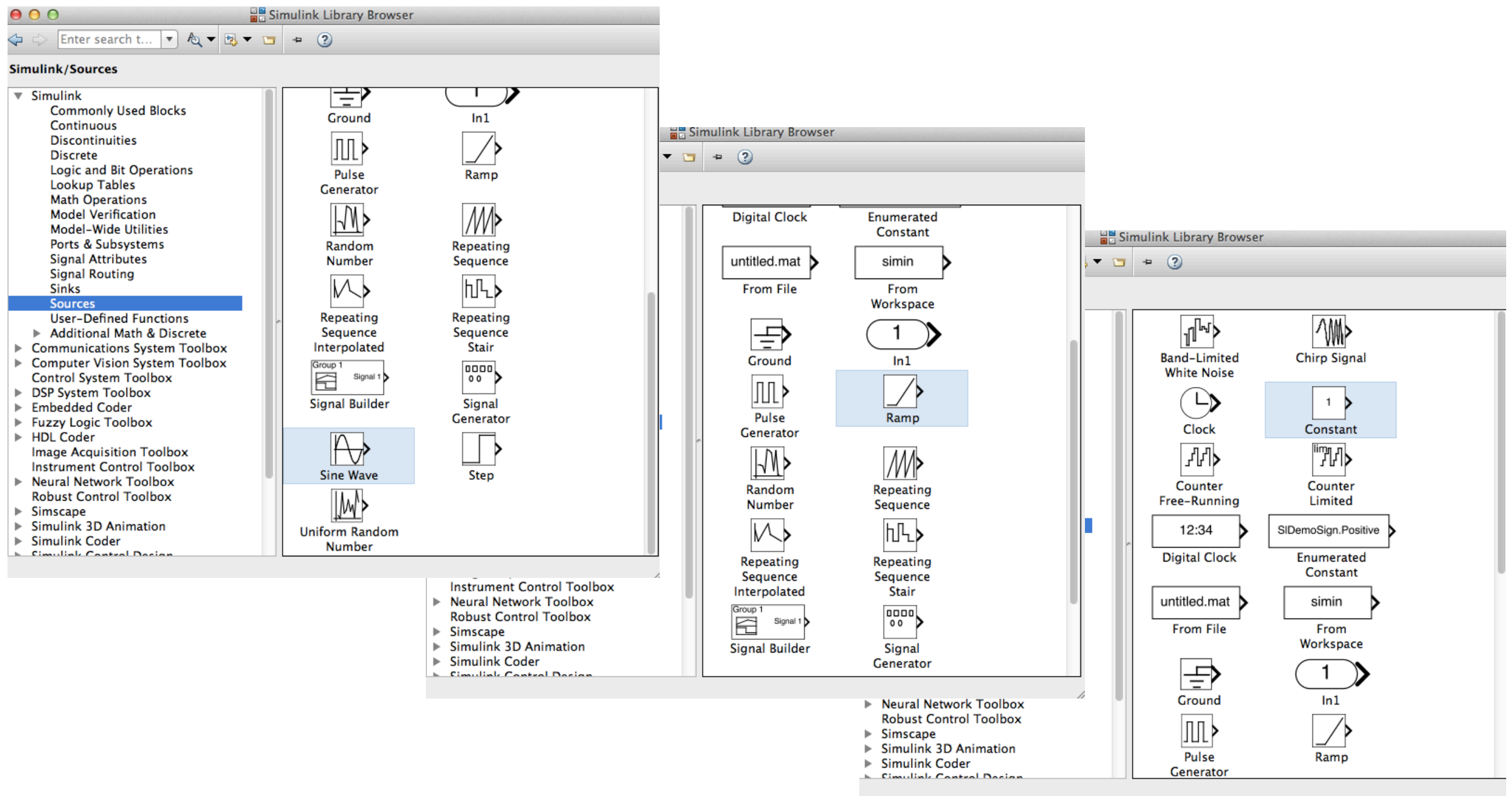




# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

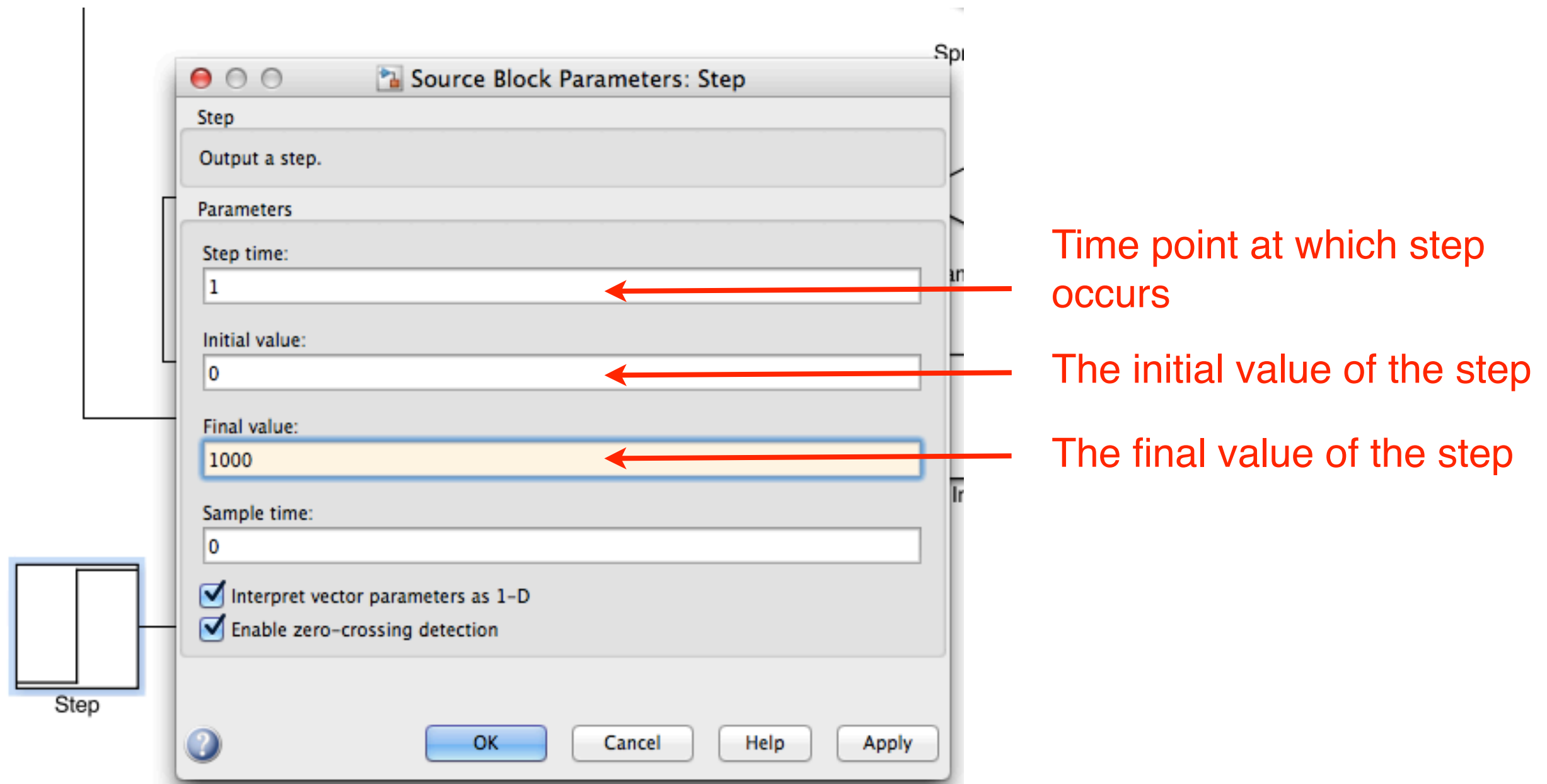
Other source blocks are available, such as Sine Wave, Ramp, and Constant:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

Connect this to the Add block and double-click to set the parameters:

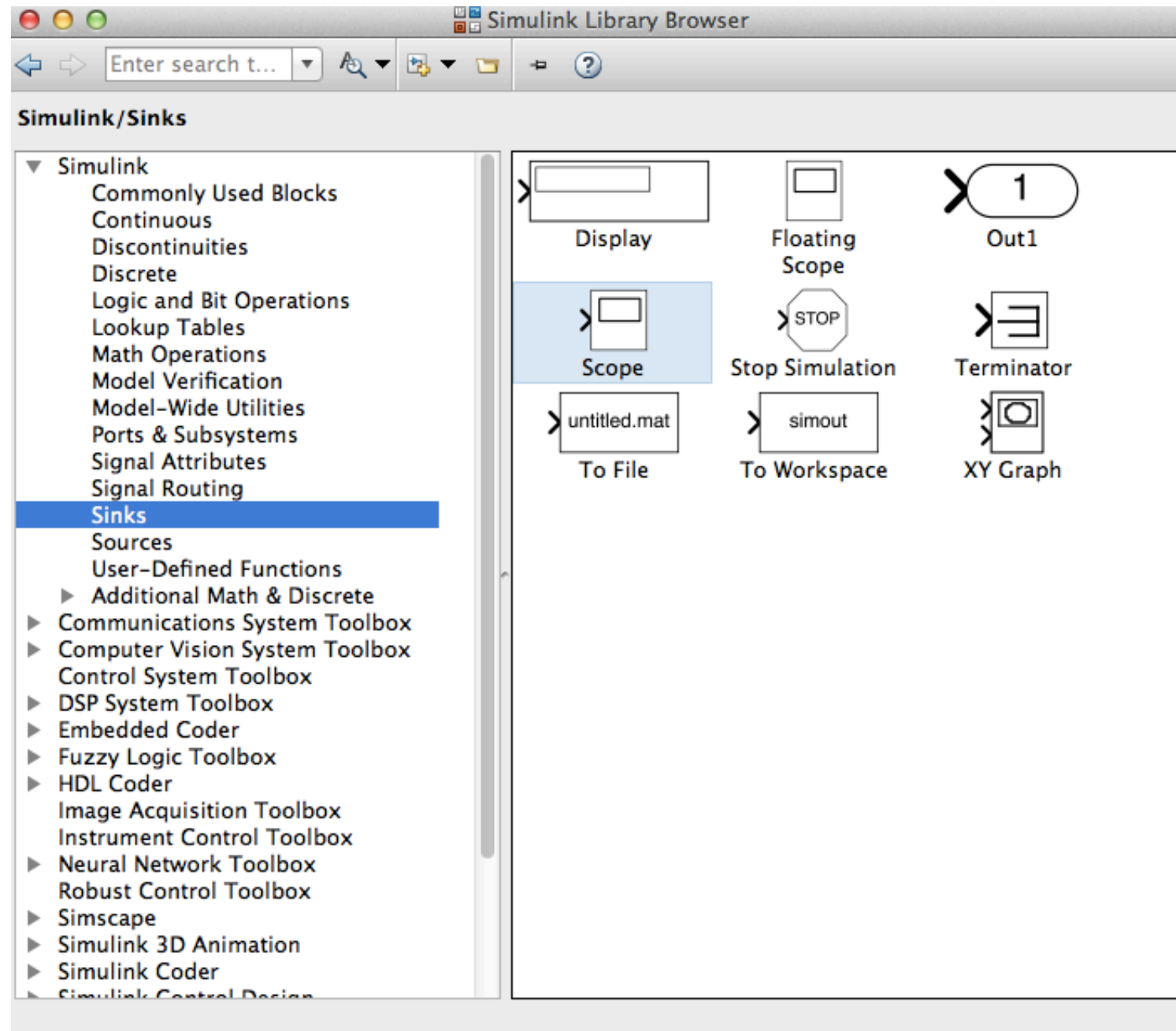




# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

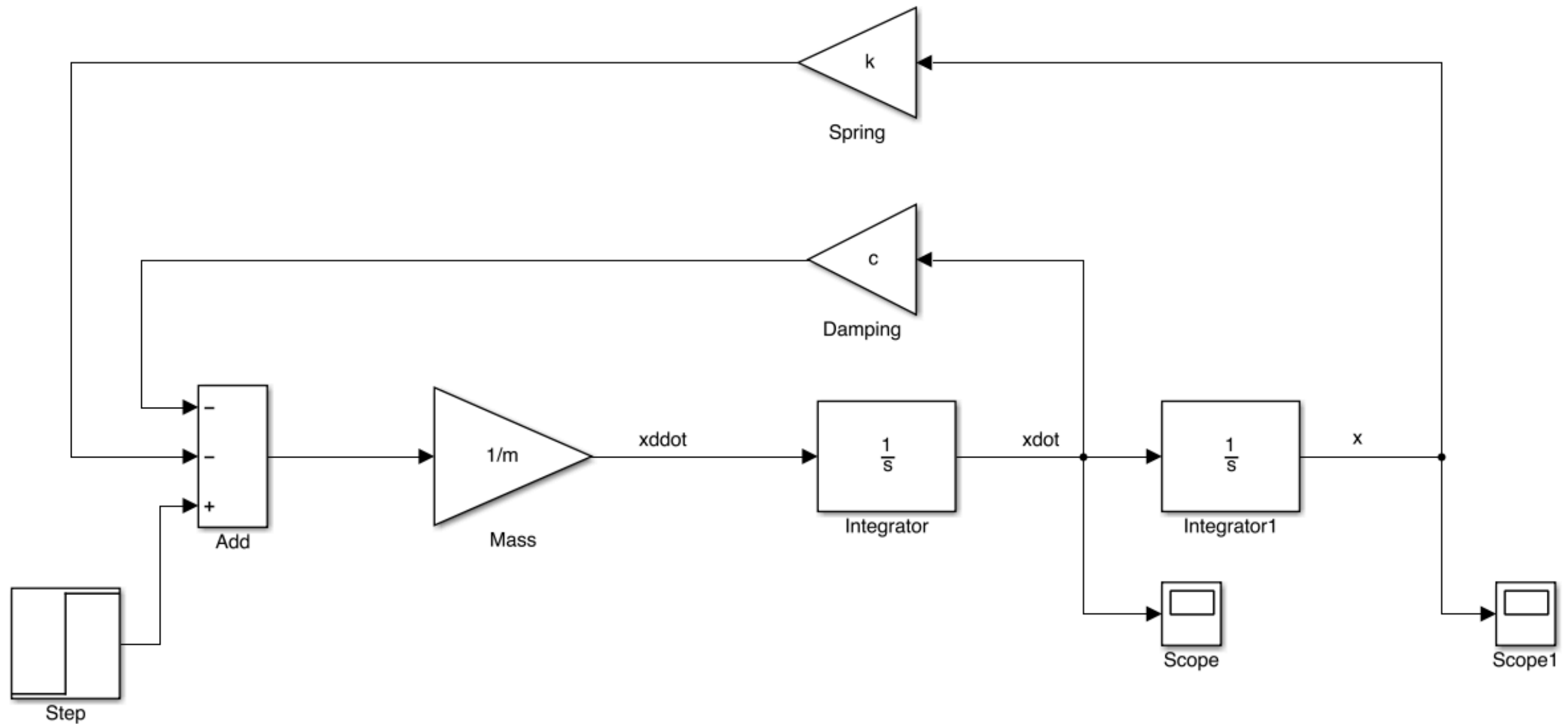
We want to add some way of visualising the results of this simulation. One method is to add a Scope block to the signal of interest:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

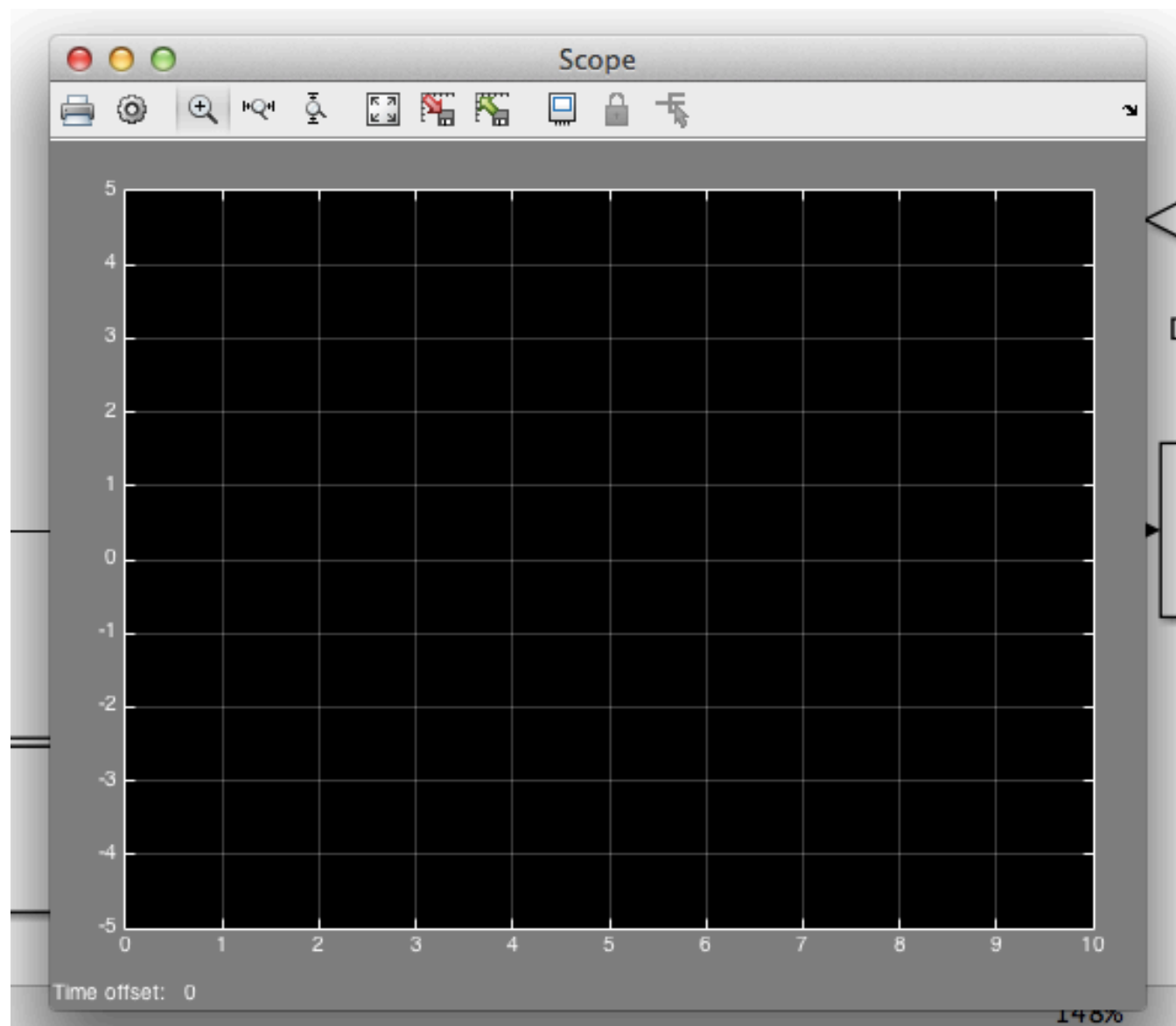
In this case add a two scopes, one to the velocity signal, and one to position:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

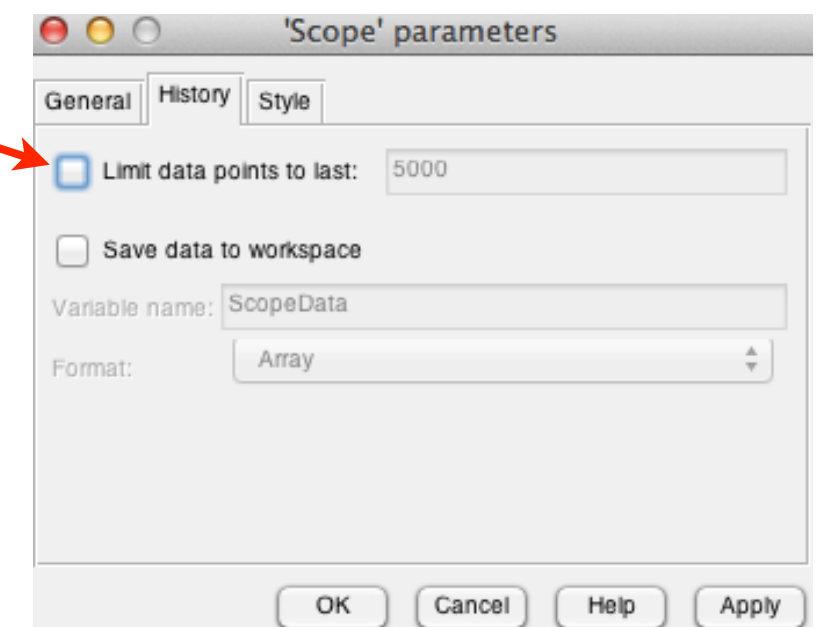
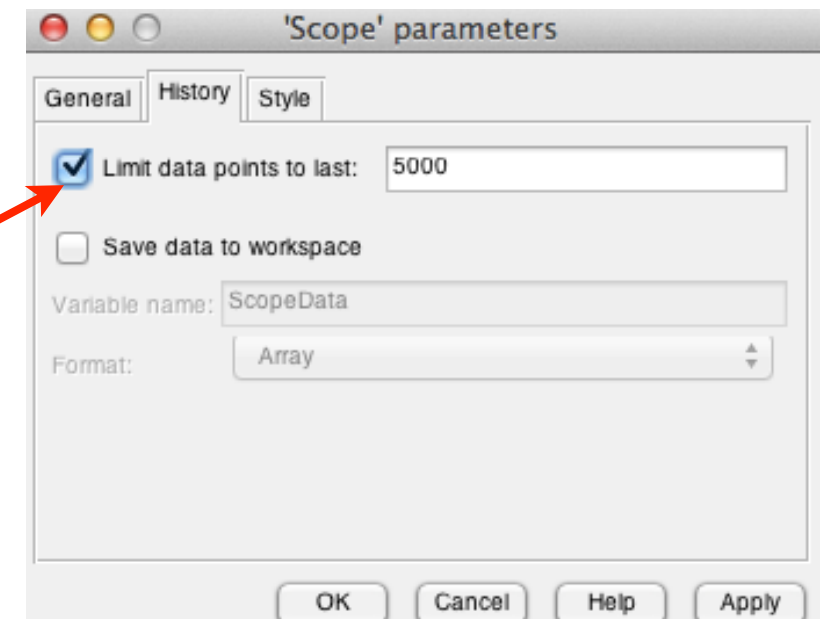
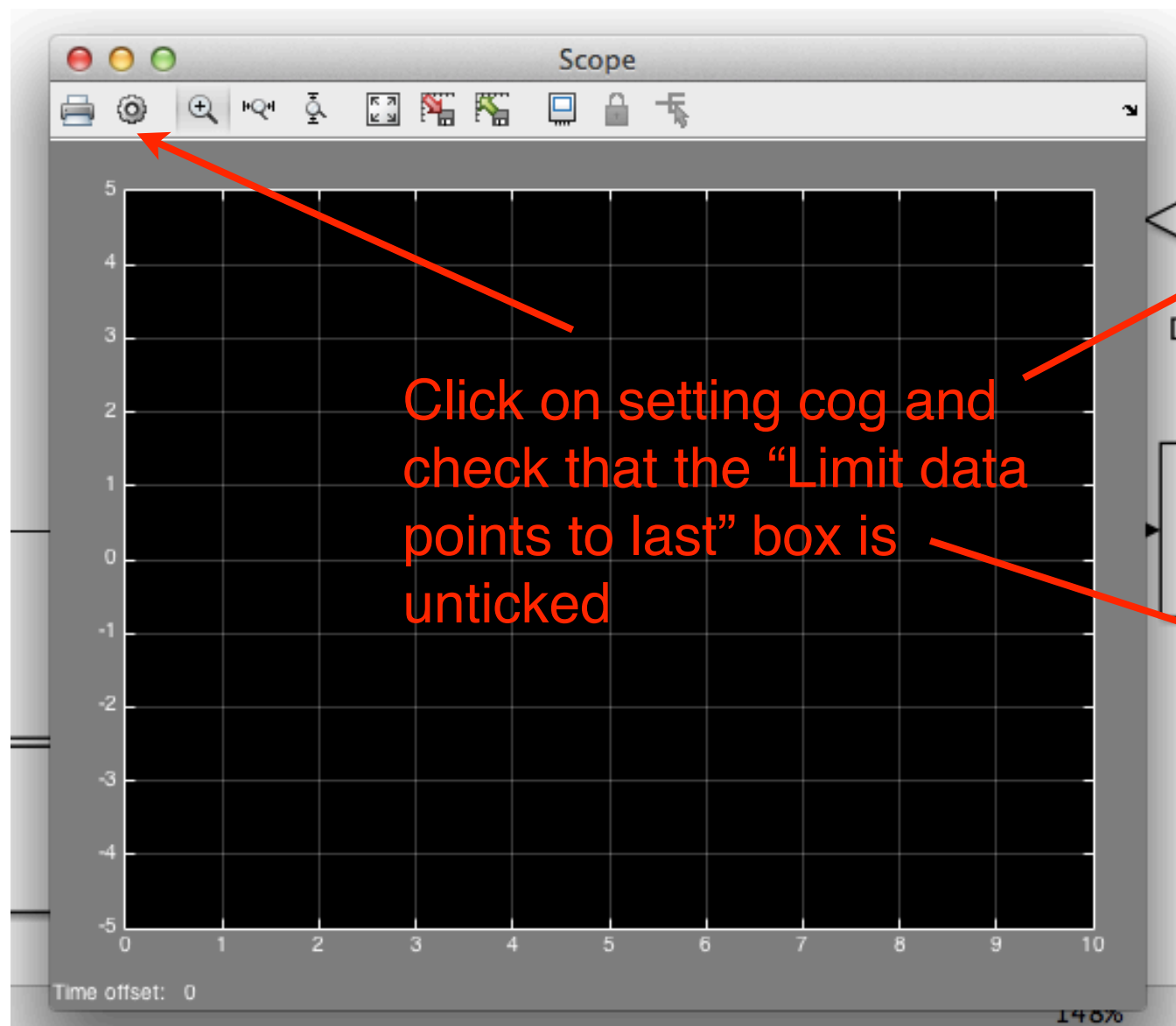
Double-clicking on the Scope opens a window showing the axes and current data plotted:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

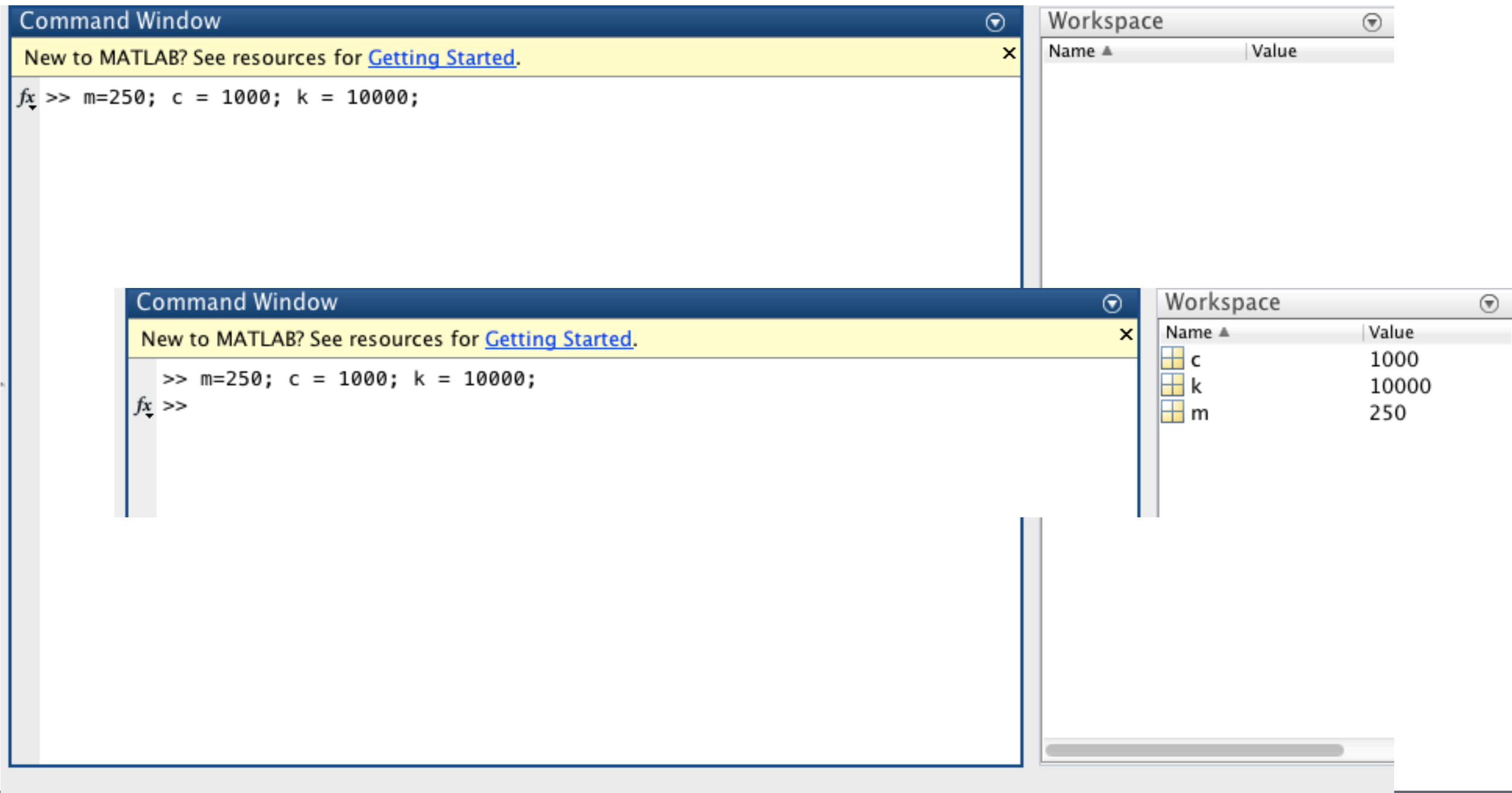
Double-clicking on the Scope opens a window showing the axes and current data plotted:



# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

To be able to run the model, we must set the parameter values. Do this by typing them into the Matlab Command Window:



The image displays two screenshots of the MATLAB environment, specifically the Command Window and the Workspace.

**Top Screenshot:** The Command Window shows the command `m=250; c = 1000; k = 10000;` being entered. The Workspace window is empty.

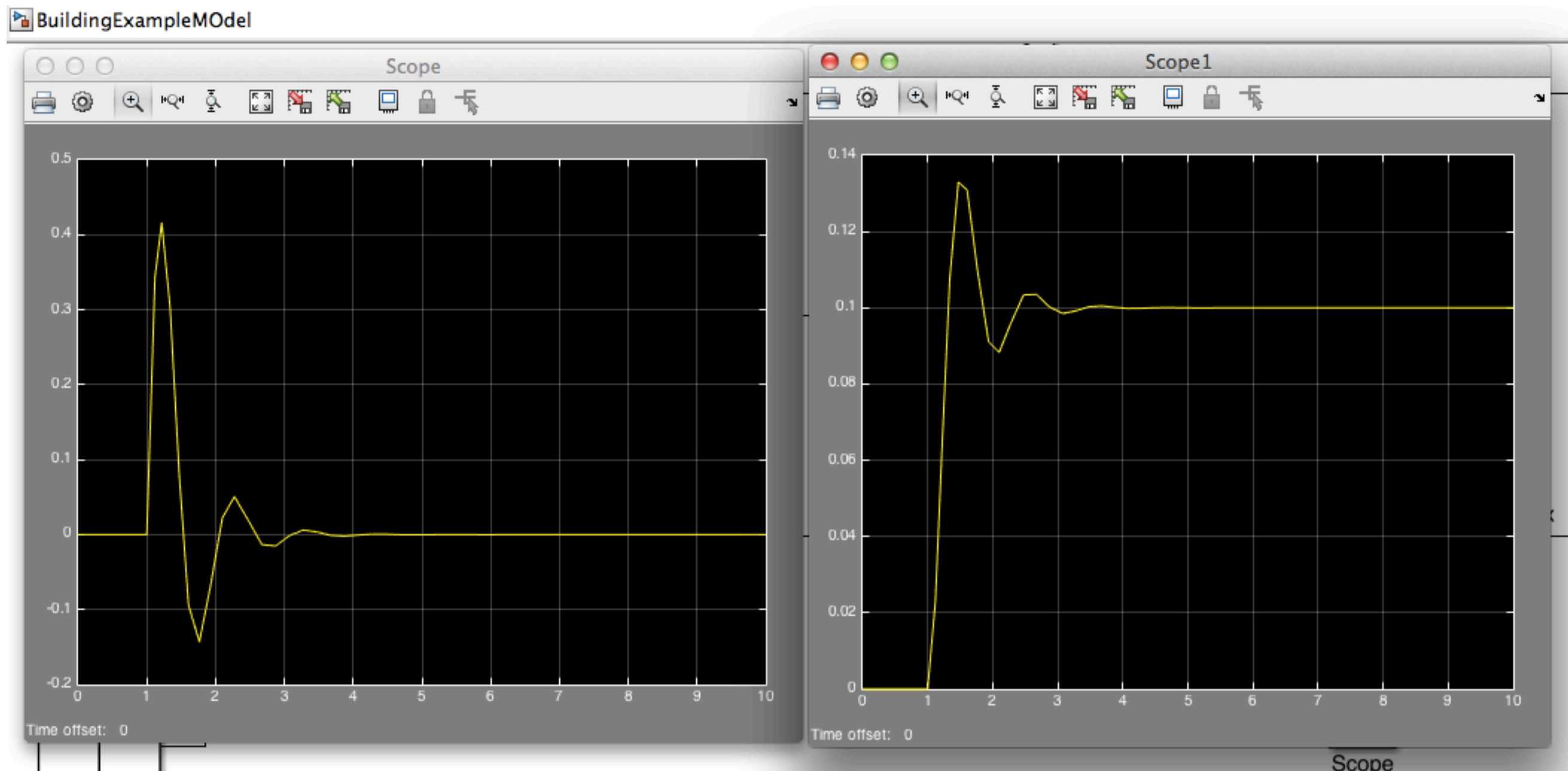
**Bottom Screenshot:** The Command Window shows the same command `>> m=250; c = 1000; k = 10000;` followed by a new prompt `>>`. The Workspace window now displays the assigned values for the variables `c`, `k`, and `m`.

Name ▲	Value
c	1000
k	10000
m	250

# BUILDING A SIMPLE MODEL

## Spring-Mass-Damper In Simulink

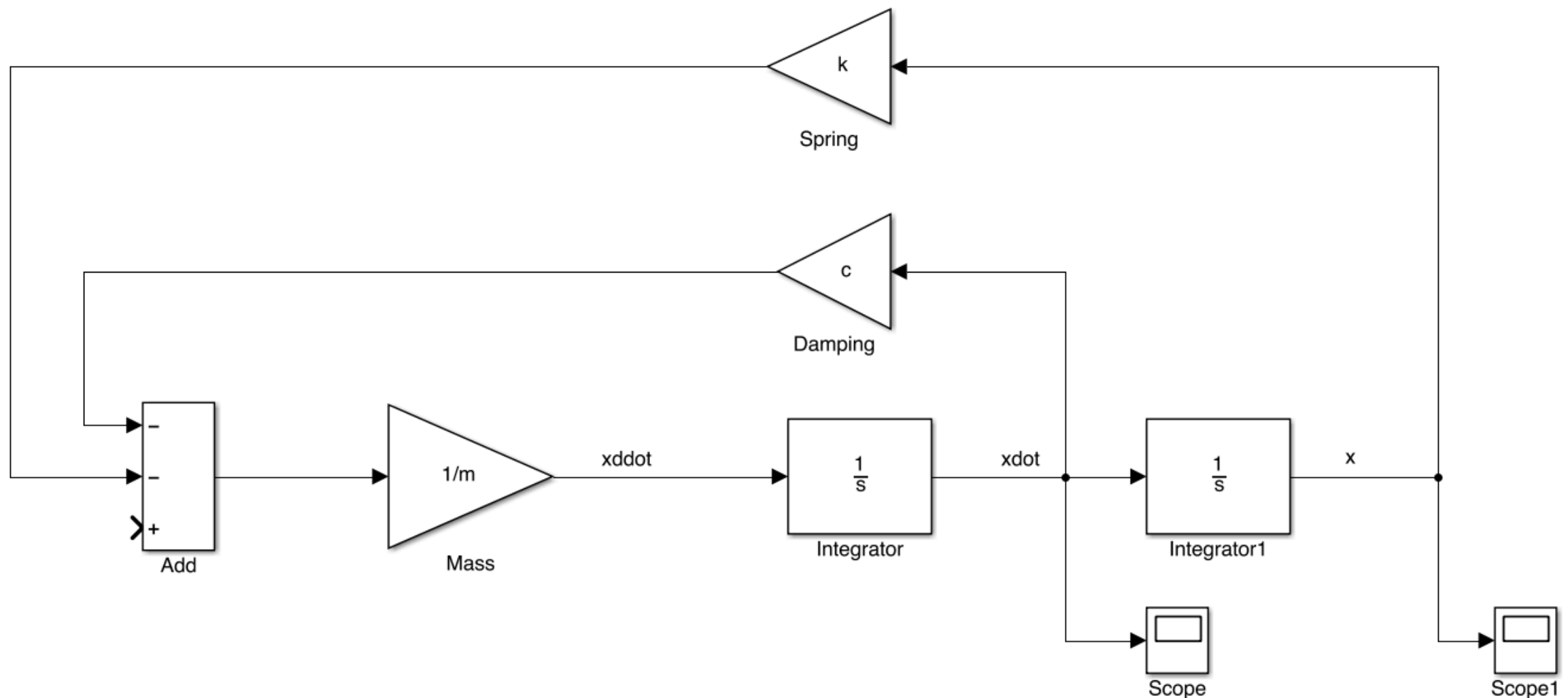
Run the model by clicking the Play button at the top of the model and view the results:



# RUNNING SIMULATIONS

## Setting Initial Conditions

In this example, we will run the model without a forcing function, but instead have a non-zero initial condition specified on the displacement

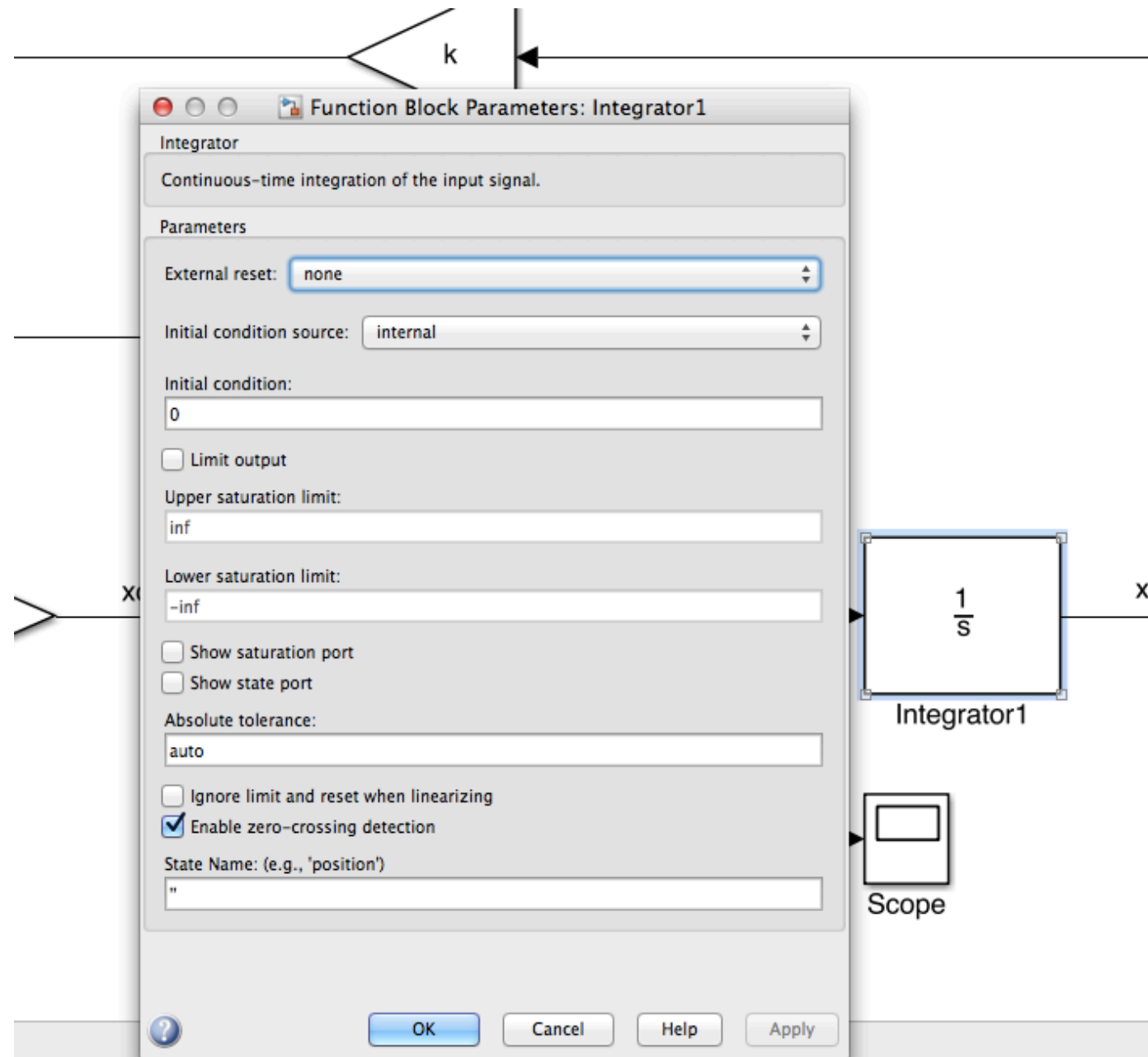




# RUNNING SIMULATIONS

## Setting Initial Conditions

As the initial condition is on position, we must set this value in the integrator block that is outputting position:



# RUNNING SIMULATIONS

## Setting Initial Conditions

As the initial condition is on position, we must set this value in the integrator block that is outputting position:

The image shows the 'Function Block Parameters: Integrator1' dialog box in Simulink. The 'Initial condition source' is set to 'internal', and the 'Initial condition' is set to '1'. A red arrow points from the text 'Set the initial condition equal to 1' to the 'Initial condition' field. The dialog box also shows options for 'Limit output', 'Upper saturation limit', 'Lower saturation limit', 'Show saturation port', 'Show state port', 'Absolute tolerance', 'Ignore limit and reset when linearizing', 'Enable zero-crossing detection', and 'State Name'. The block diagram on the right shows an integrator block labeled 'Integrator1' with a transfer function  $\frac{1}{s}$  and an output signal 'x'. A scope block is connected to the output of the integrator block.

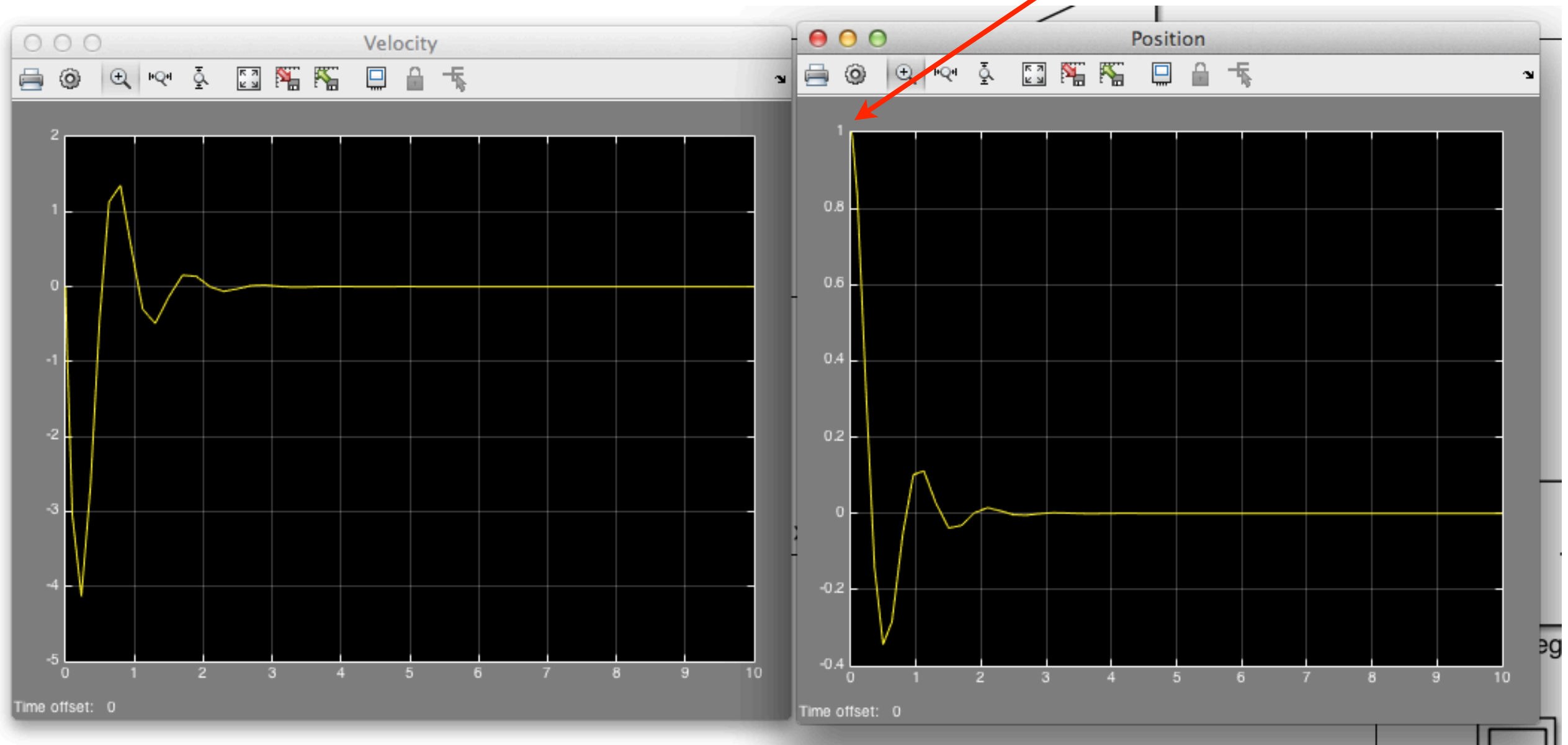
Set the initial condition equal to 1

# RUNNING SIMULATIONS

## Setting Initial Conditions

Running this model, produces the following outputs:

Initial condition satisfied



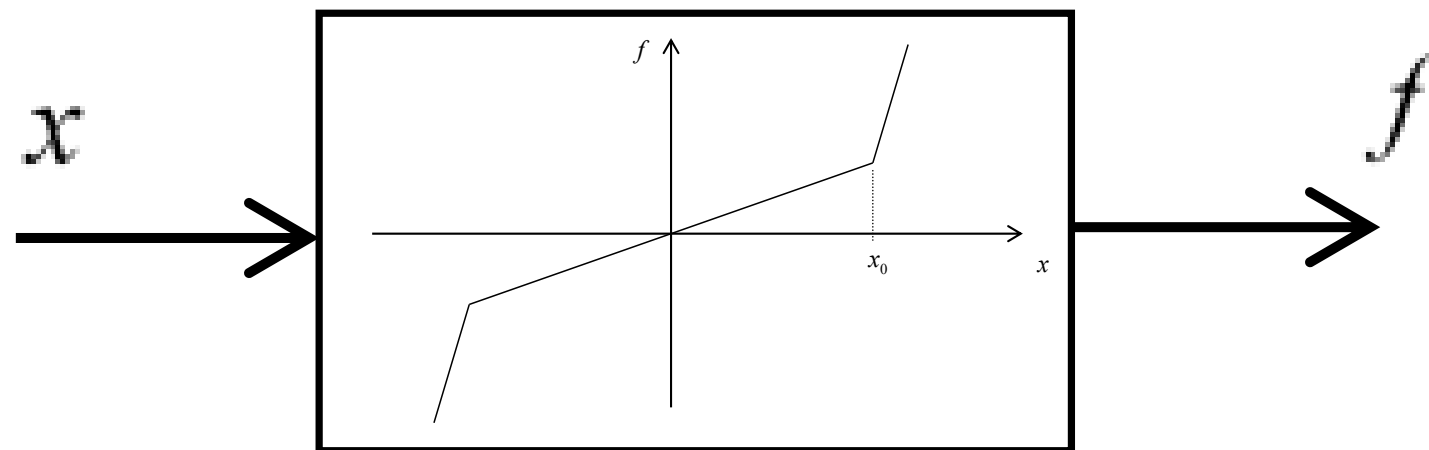
# NONLINEARITIES

## Using The Lookup Table In Simulink

### Nonlinear parameter/component

Hardening or softening of spring stiffness

In Simulink this can be represented by a look-up table



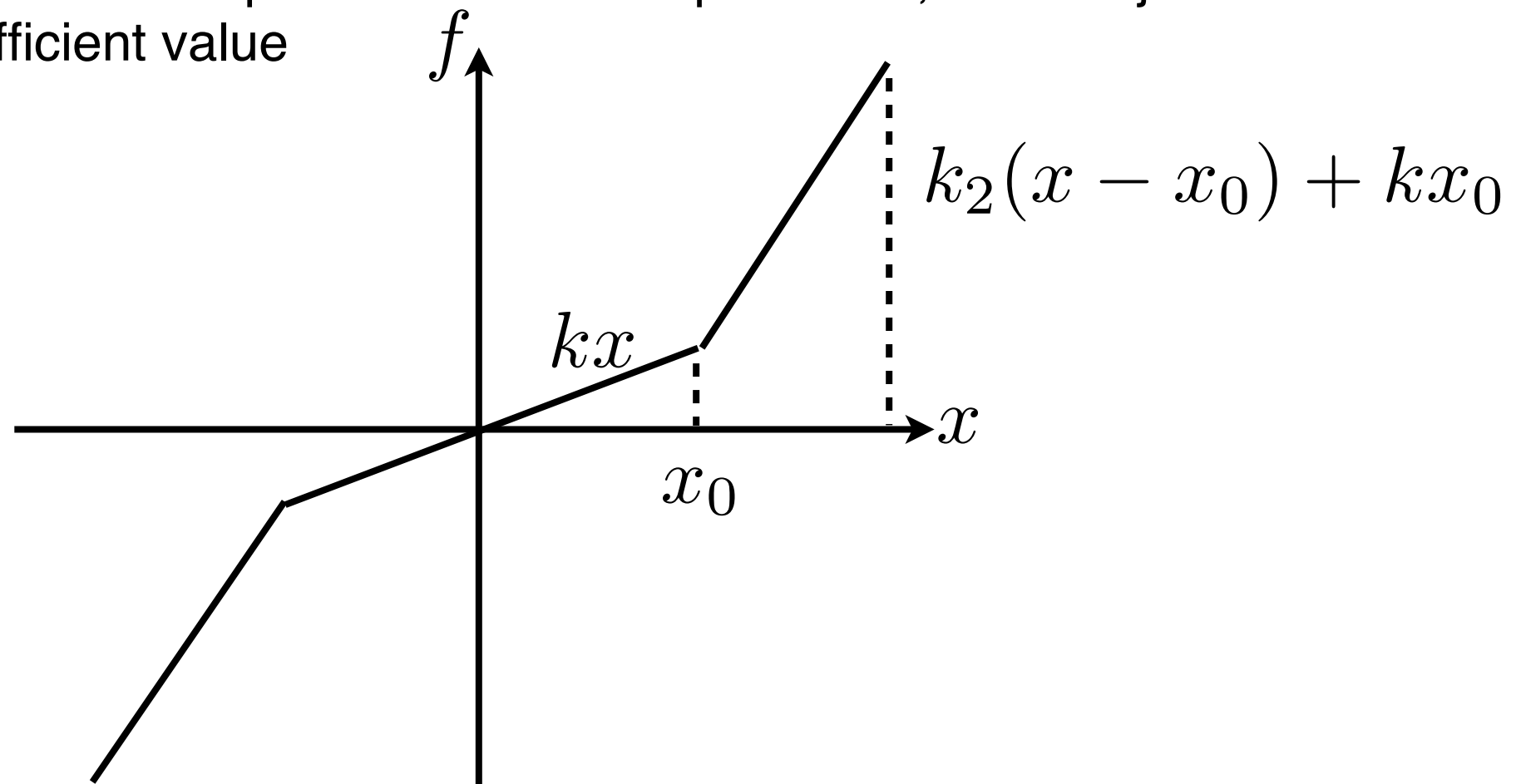
# NONLINEARITIES

## Using The Lookup Table In Simulink

Discrete data points that characterise the function are specified in the look-up table

When supplied with an input, the look-up table block will either interpolate in between the specified data points or extrapolate beyond them

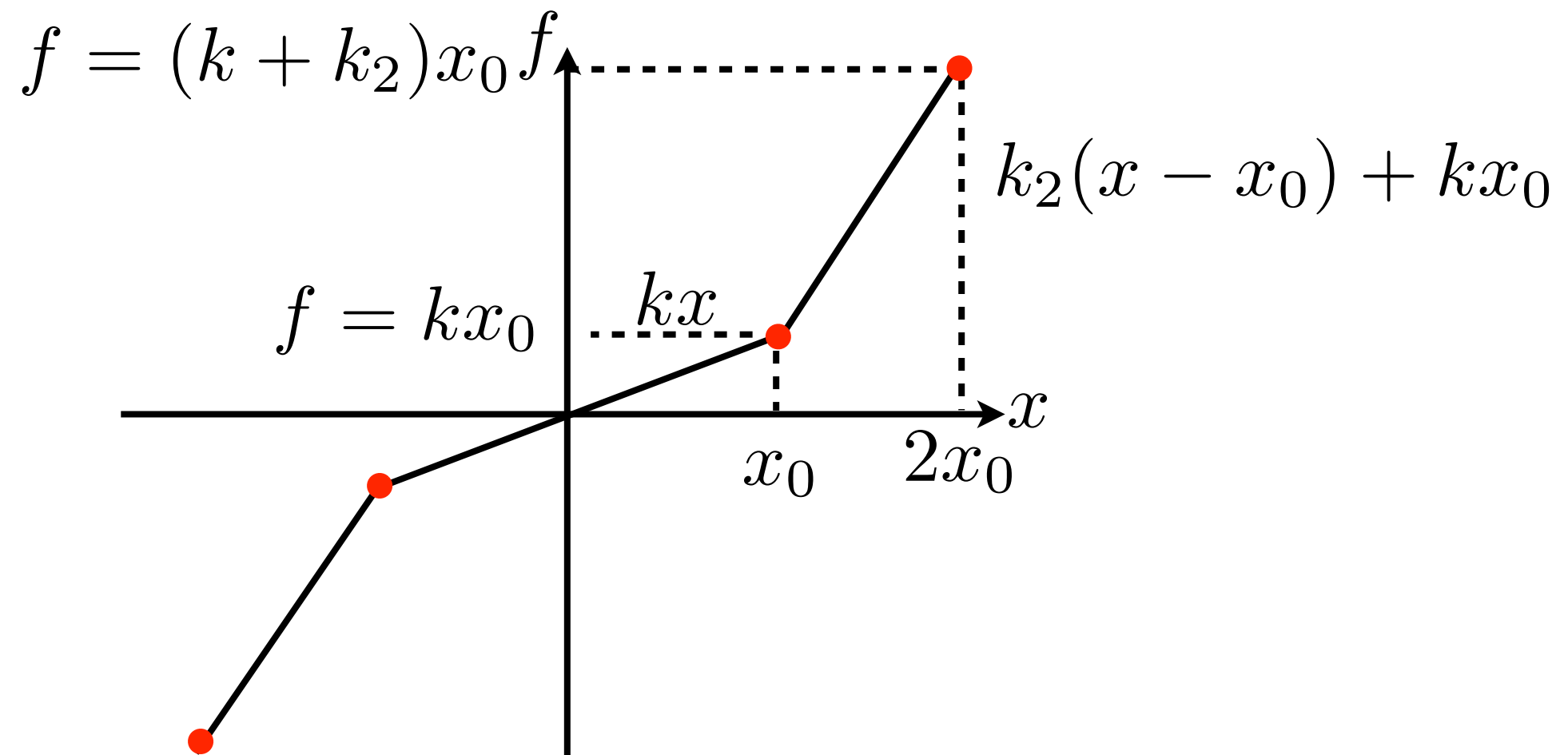
These look-up tables will replace the gain block currently used in the model - note that the output of the lookup table is the force produced, and not just the interpolated coefficient value



# NONLINEARITIES

## Using The Lookup Table In Simulink

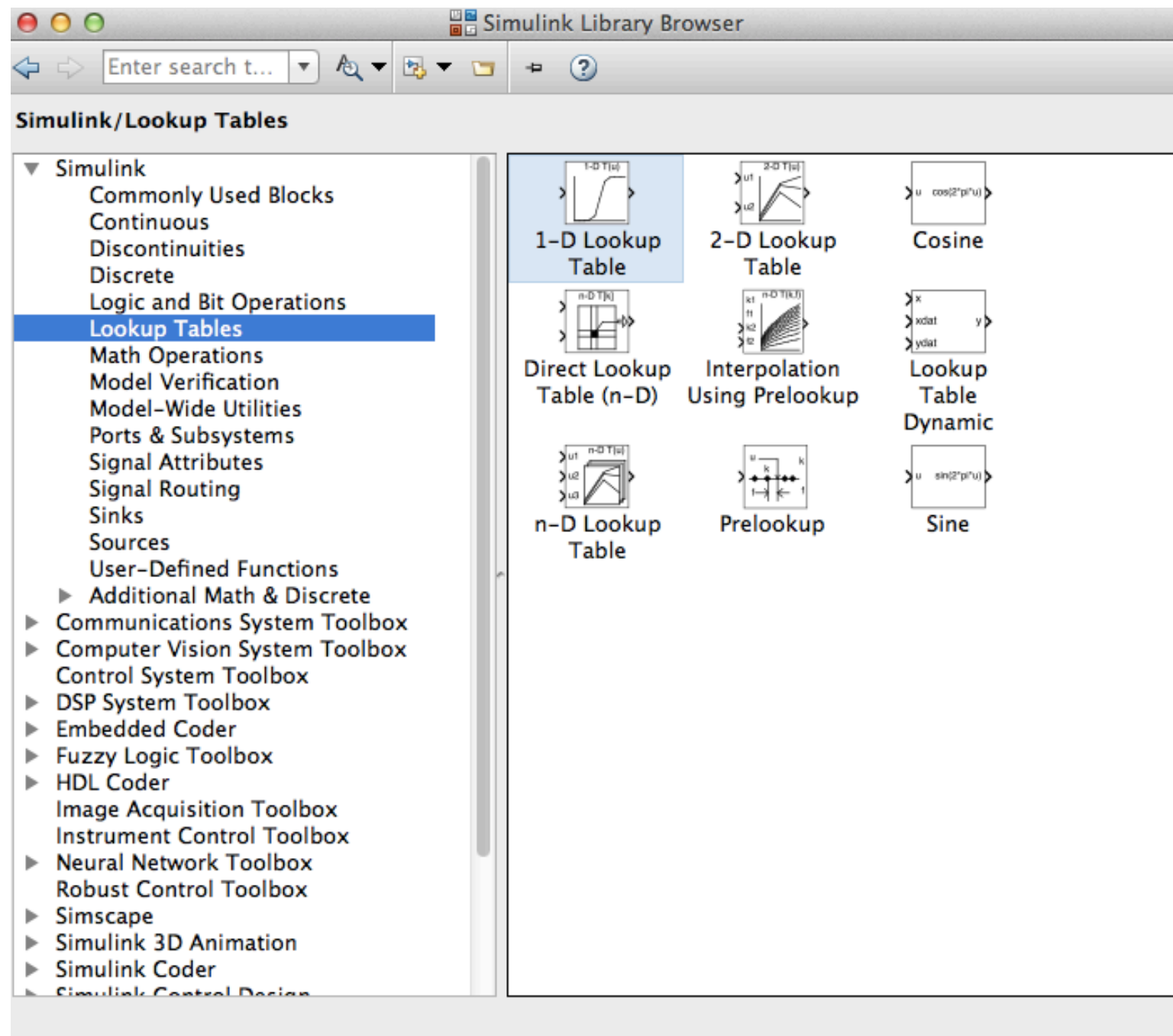
Just need to specify the coordinates of the four red dots to characterise this function:



# NONLINEARITIES

## Using The Lookup Table In Simulink

Discrete data points that characterise the function are specified in the look-up table

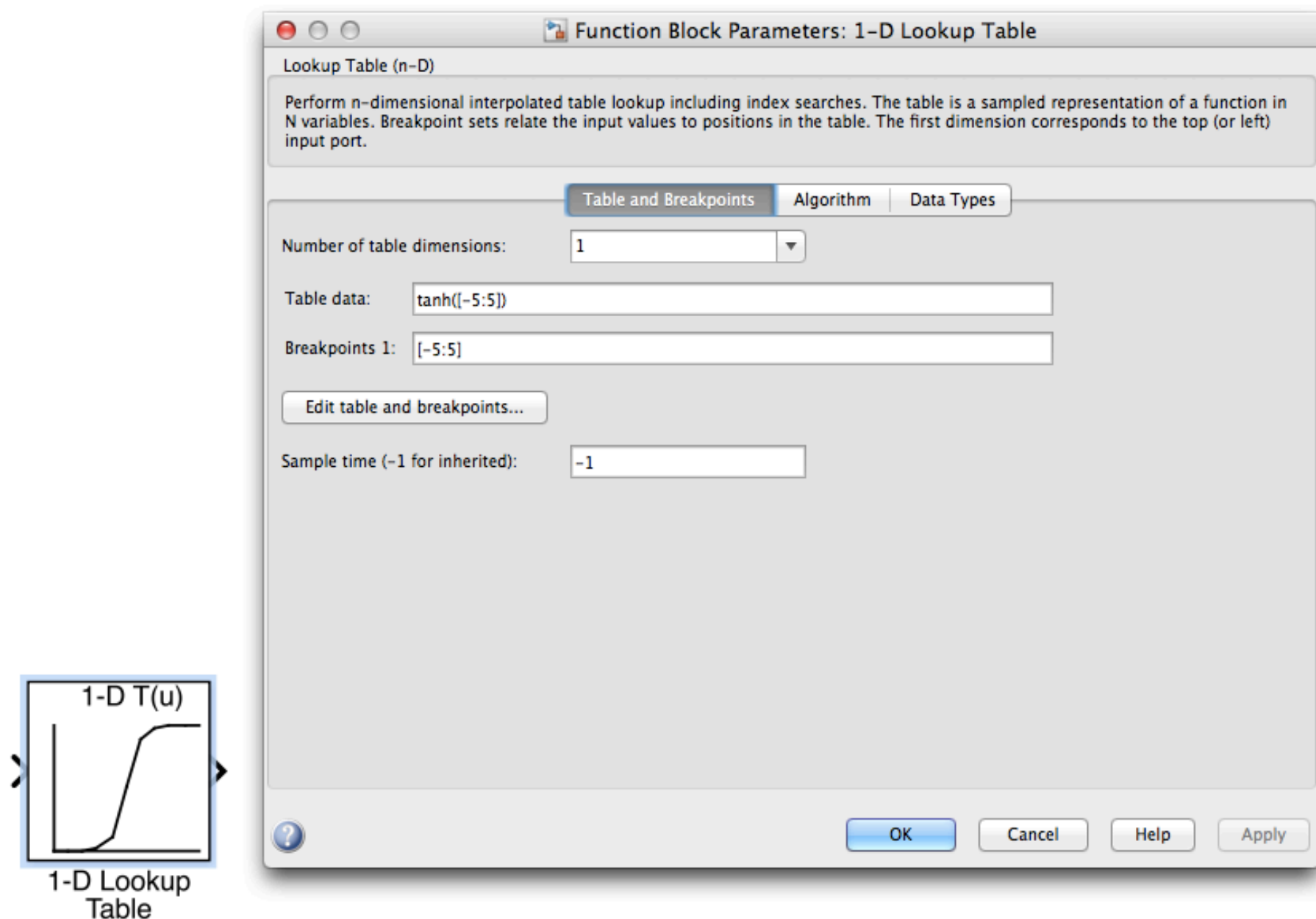




# NONLINEARITIES

## Using The Lookup Table In Simulink

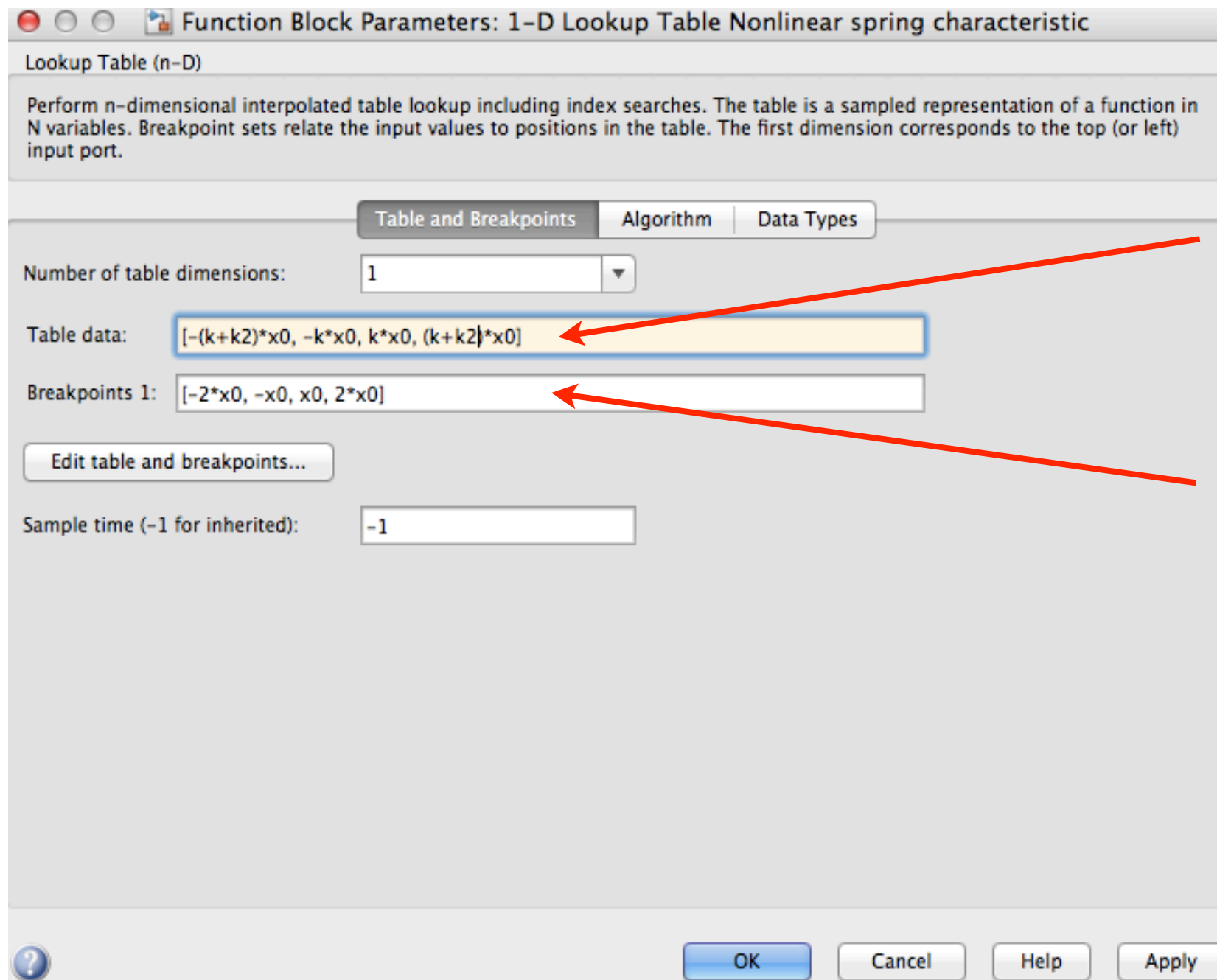
Drag and drop into your model and double-click to open the parameter setting menu:



# NONLINEARITIES

## Using The Lookup Table In Simulink

Now enter the data points as two Matlab vectors



These are the y values that will be outputted

These are the x coordinates at which the function is changing value

