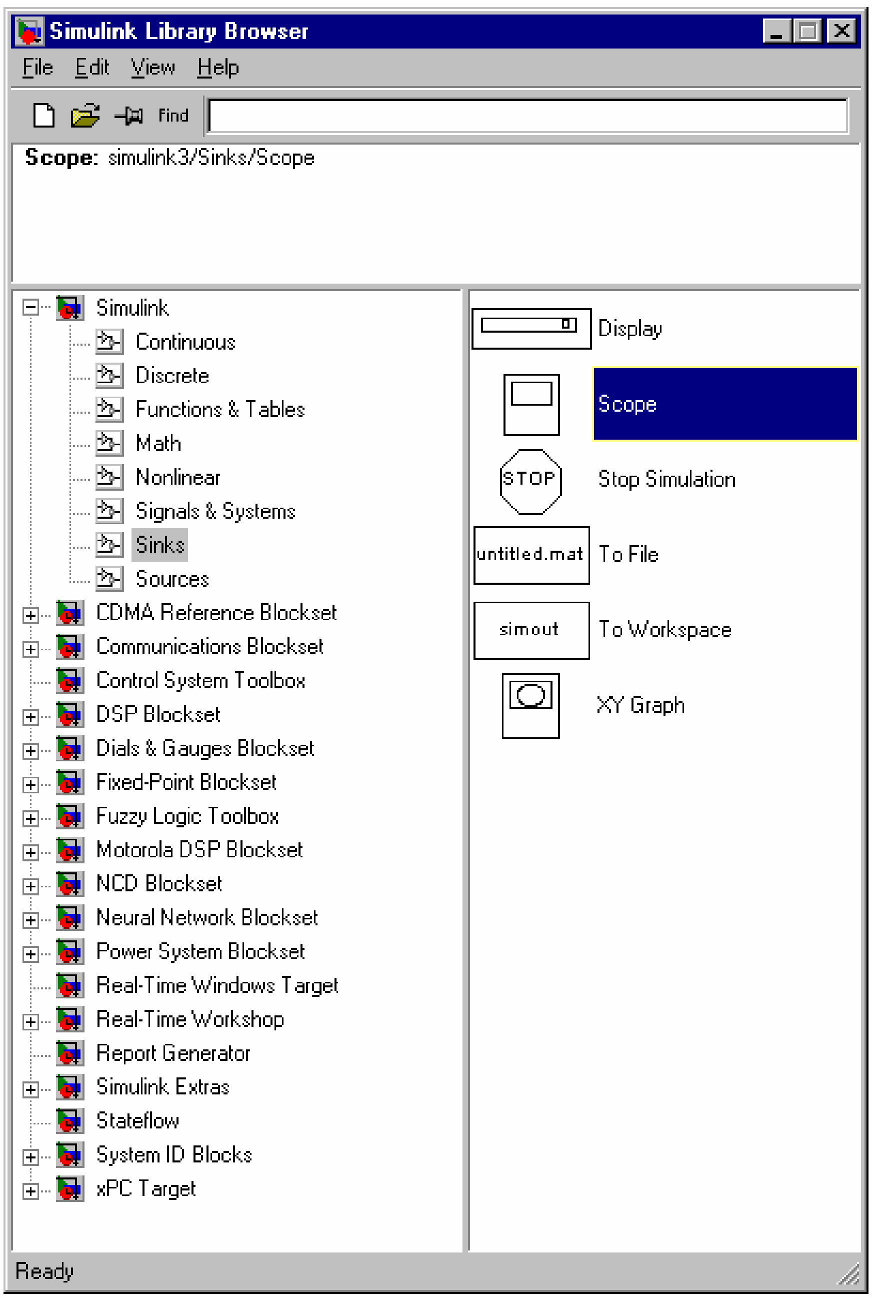
Simulink Help

In the Matlab command window write *simulink*.

The window that has opened is the *Simulink Library Browser*. It is used to choose various Simulink modules to use in your simulation.

From this window, choose the *File* menu, and then *New* (*Model*). Now we have a blank window, in which we will build our model. This blank window and the library browser window, will be the windows we'll work with.

We choose components from the library browser, and then drag them to our work window. We'll use only the Simulink library (also called toolbox) for now.

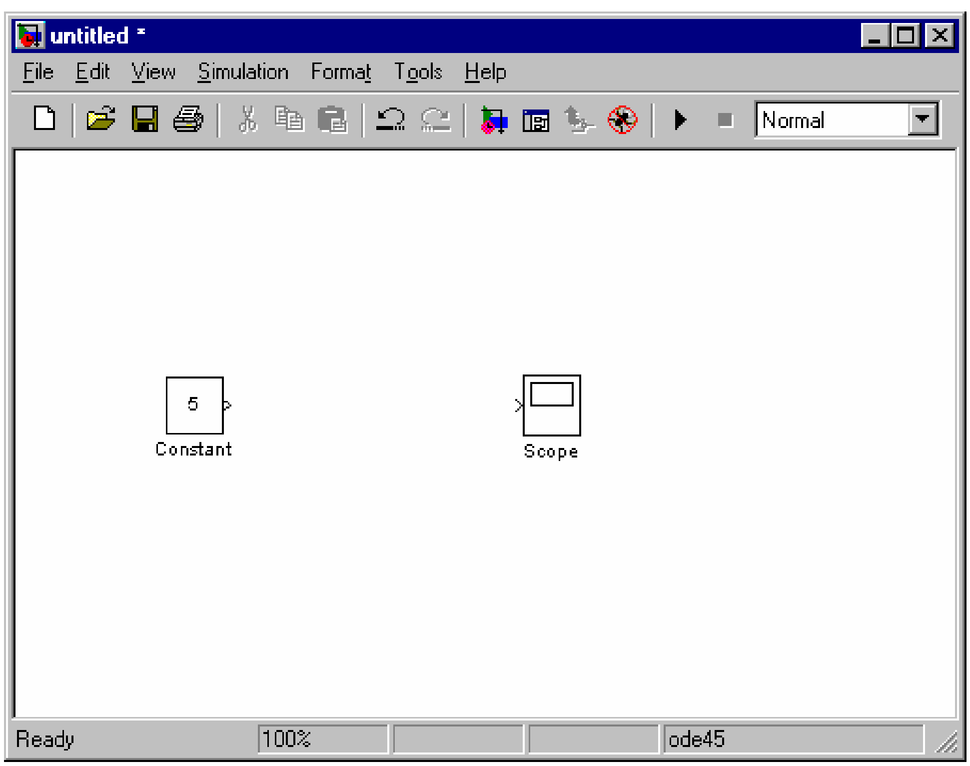


As we can see, the Simulink library is divided into several categories:

1. Continuous – Provides functions for continuous time, such as integration, derivative, etc.
2. Discrete – Provides functions for discrete time.
3. Funcitons & Tables – Just what the name says.
4. Math – Simple math functions.
5. Nonlinear – Several non-linear functions, such as switches, limiters, etc.
6. Signals & Systems – Components that work with signals. Pay attention to the mux/demux. The rest is generally not interesting.
7. Sinks – Components that handle the outputs of the system (e.g. display it on the screen).
8. Sources – Components that generate source signals for the system.

Let's build our first simple simulation. Drag the *Constant* component from the Simulink library (*Sources)* to the work window, and then drag the *Scope* module (*Sinks)* in the same manner. Note that the library viewer is used **only** to drag the components to the work window. The rest of the work is being made in the work window.

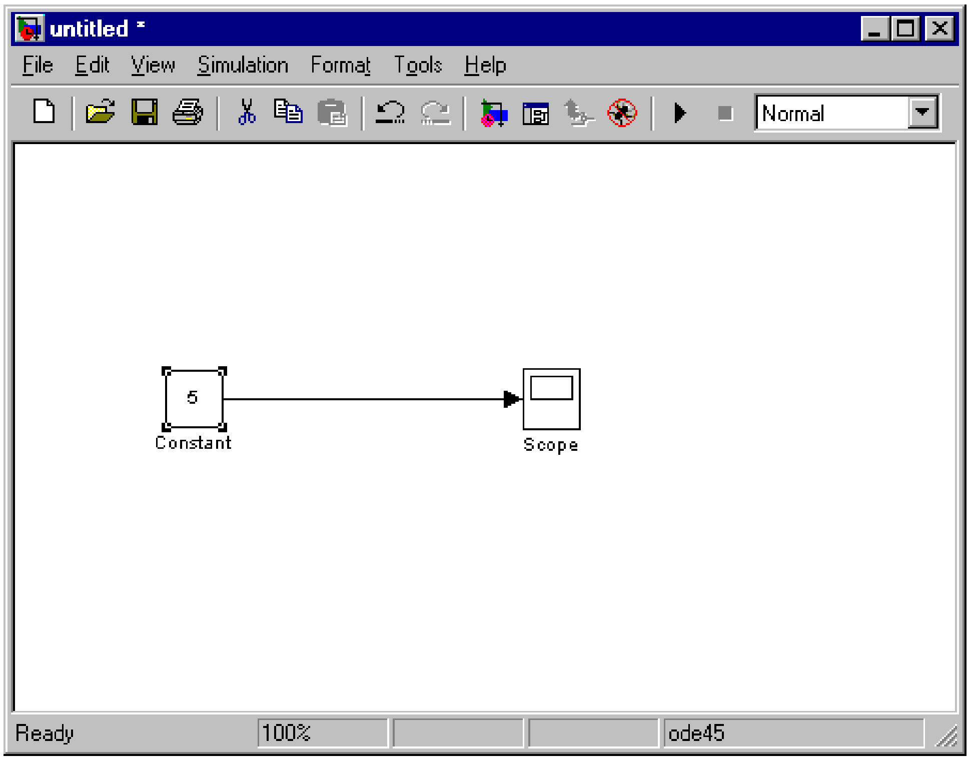
Your work window should now be looking something like this:

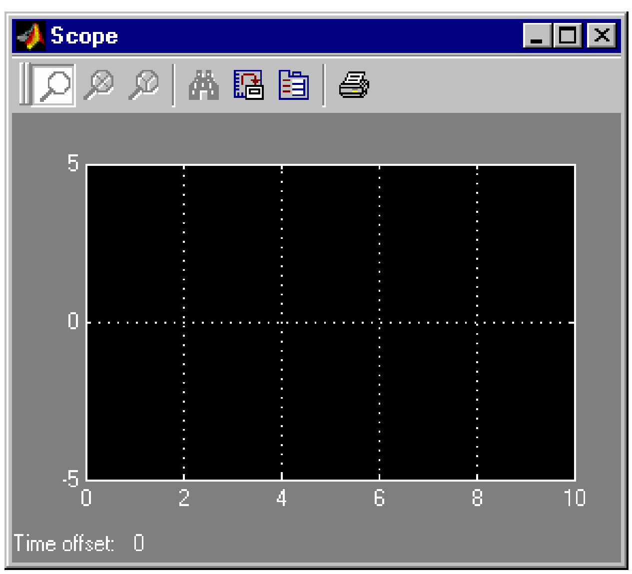


Now, click the little triangle (output port) to the right of the *Constant* module, and while holding the mouse button down, drag the mouse to the left side of the scope (input port) and then release it. You should see a pointed arrow being drawn.

Double click the *Constant* module to open its dialog window. Now you can change this module's parameters. Change the constant value to 5.

Double-click the scope to view its window. This should look something like this:

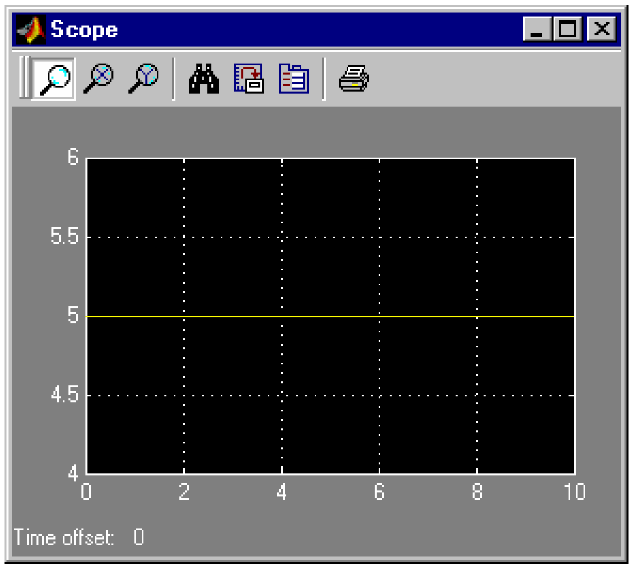




You can choose *Simulation Parameters…* from the *Simulation* to change the time limits for your simulation.

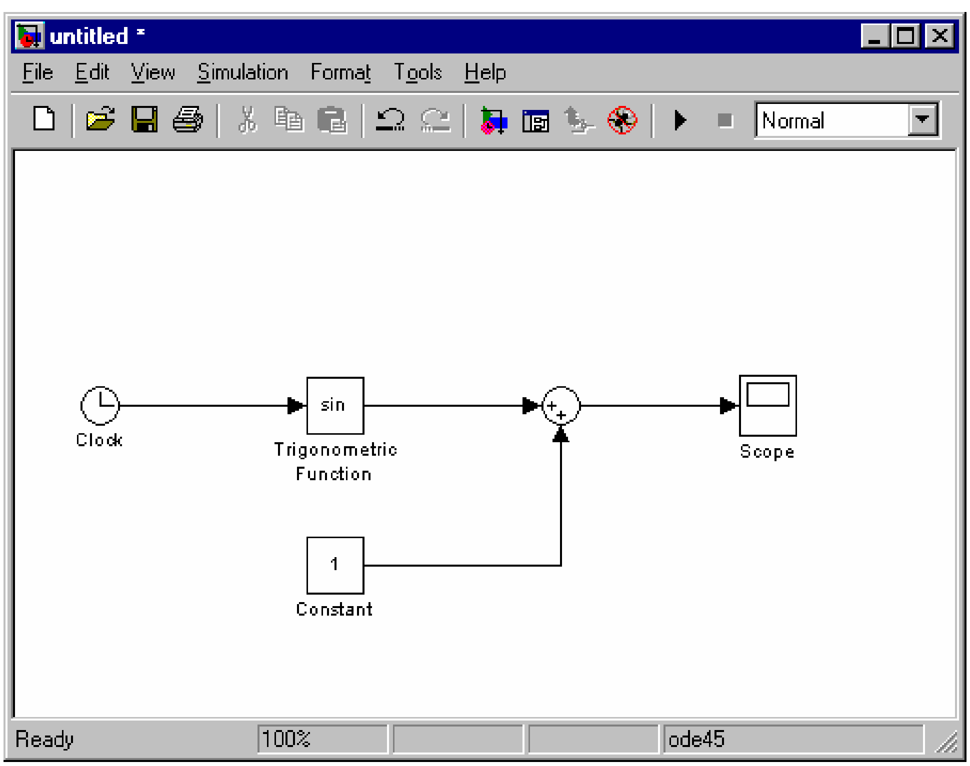
Choose *Start* from the *Simulation* menu (or press Ctrl+T, or click the play button on the toolbar) to start the simulation. Now this is a rather silly simulation. All it does is output the constant value 5 to the graph (the x-axis represents the simulation time).

Right-click on the scope's graph window and choose *Autoscale* to get the following result:

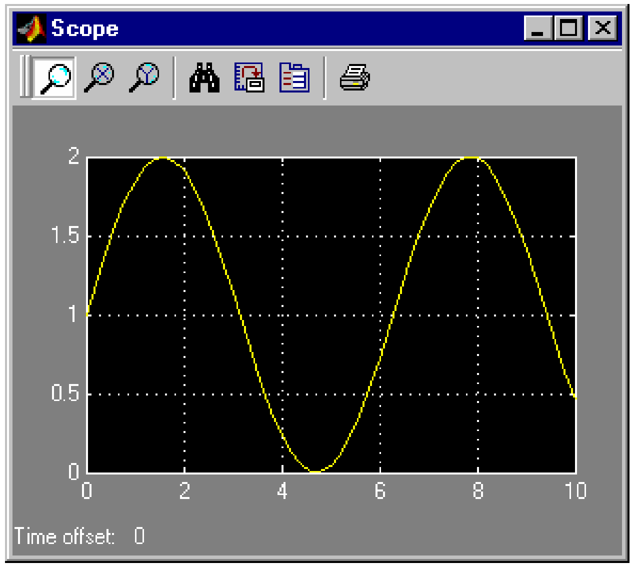


Now let's try something a little bit more complicated (but just a little).

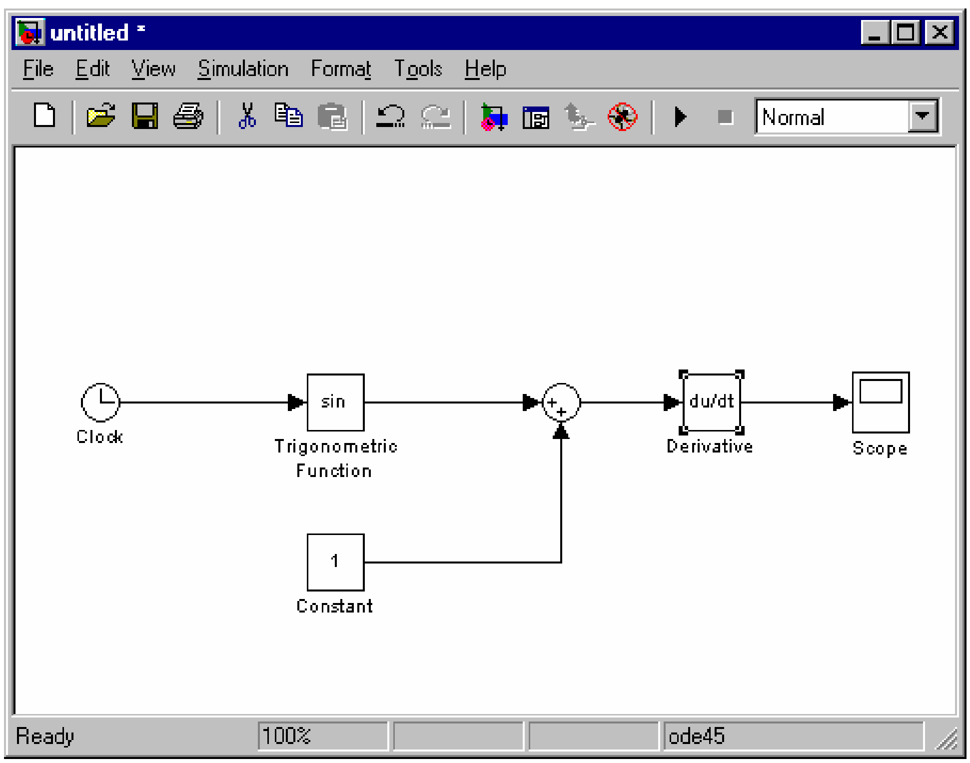
First, build the following system (you can find the *Clock* module in the *Sinks* category. The *Trigonometric Function* and *Sum* modules reside in the *Math* category):



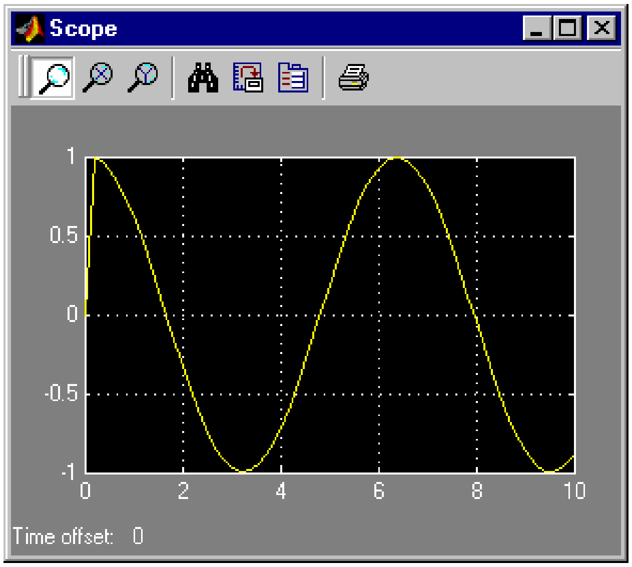
This should produce the following output:



Now, let's see if the derivative is really a cosine. Build the following system (the *Derivative* module is located in the *Continuous* category):



Now let's see what the scope tells us:



We can see that this is indeed a cosine, but something is wrong at the beginning. This is because at time 0, the derivative has no prior information for calculation (there's no initial value for the derivative), so at the first time step, the derivative assumes that its input has a constant value (and so the derivative is 0).

Let's look at something different. Let's say that we have a differential equation that we want to model. The equation is:

$$A' = 0.5 \cdot A^2$$

$$A_0 = 0.5$$

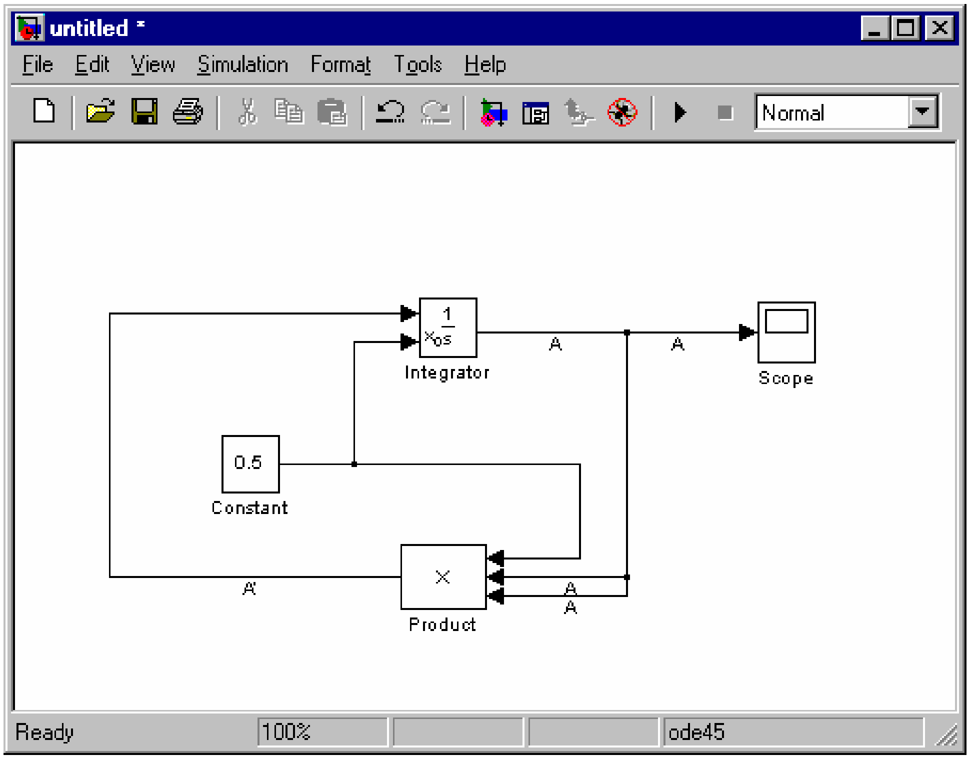
How can we solve this numerically using Simulink?

We'll notice 2 simple facts:

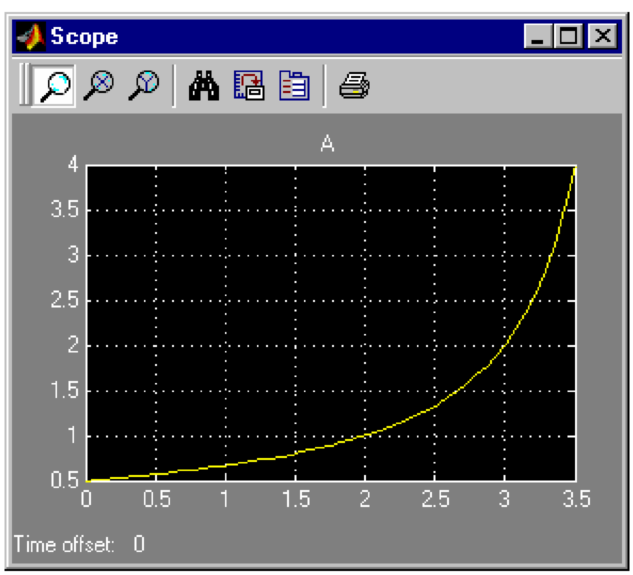
1) If we have A, then we have A' (multiplication).
2) If we have A', then we have A (integration).

So it seems as if we're stuck in a loop. In order to get A, we need A', but in order to get A', we need A. We can get out of this loop by using the initial condition. We know that $A_0 = 0.5$, so now we can calculate A' and then recalculate the new A, and so on.

In our simulation, this will look like this (double-click the *Integrator* and choose *Initial Condition Source: External*. You may have to double-click or right-click some other modules and lines. Note that pressing ctrl when clicking the mouse button on a line, allows you to split it into 2 lines):



Set the simulation stop time to 3.5 seconds (the solution goes to infinity), and see the results in the scope:



That's all for now. To get additional help (which is HIGHLY recommended) regarding Simulink, or any other Matlab component, write *helpdesk* in the Matlab command window.

All the images provided here are from Matlab 6.