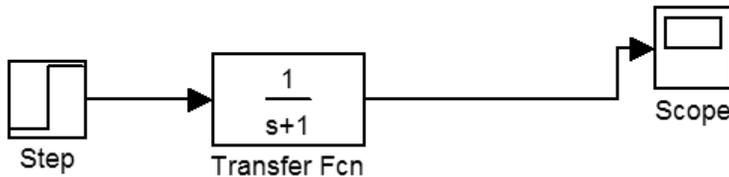


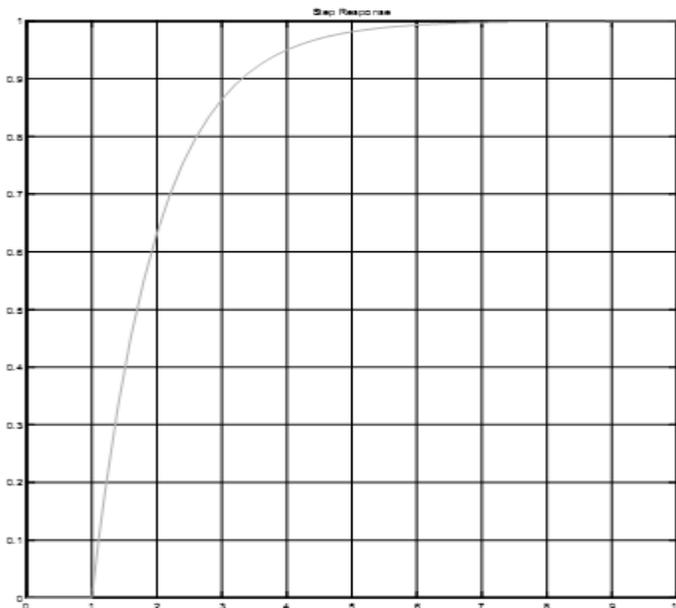
## The Process

ble. Right now, we will introduce simulink on transfer functions with the simple transfer function for the differential equation,  $dx(t)/dt + x(t) = u(t)$ . Our input signal of  $u(t)$  will be the step function. Remember how to transform this



DE into a transfer function: there is no  $s^2$ , so the first is the 1 in front of the first derivation and then a constant of 1, or  $1/(s^2 + 1)$ .

then repeat the design as shown below. {Expand "Simulink" and "Sources", then move the "step" into the open window; you will find the transfer function block



Type "simulink" in the MATLAB command window; in upper left select open a new; and then interconnect the step block to the transfer function by grabbing the little male connector and pulling to the female connector. Now, all we need is scope block to look at the output, which is under "Sinks"--drag it into the design window and interconnect.

Start the simulation under "Simulation" at the top of the design window; and double click on the Scope block so you can see the output. {If it does not look like the plot below, then click on the autoscale, it is a binocular symbol.}

### 16-3: The Transfer Function.

Much of the awe of the Transfer Function, besides with practical uses, will disappear as you remember how often in Circuit Analysis classes you used techniques like Norton's Theorem, Thevenin's Theorem, and the Superposition Theory to explain and characterize what was in a black box in terms of what it looked like and how it behaved at the input and output. In other words, the internal representation of the black box does not have to be the same as what is actually in the black box as for example in terms of how resistors, capacitors, and inductors--even power sources, in order for that representation to accurately model the internal behavior for input and output. For example you recall how in Theven's Theorem, at the output you measured the open circuit voltage and the closed circuit current to form a single power source and resistance to represent the black box behavior.

And if you find any or all of this theory too difficult, simply fall back on this initial definition of the Transfer Function,  $G(s)$ , as a black box diagram with a single input,  $U(s)$  and single output,  $Y(s)$  like in Figure 6-1.

### Figure 6-1: Block Diagram Representation of Transfer Function.

The symbols of representation like  $U$  for input and  $Y$  for output are common in most technical books, especially on Control Engineering; and the  $s$  for input or output is a function of  $s$  lets us know that a mathematical transform has taken place to convert the DE into an algebraic equation using the famous and classical LaPlace transform. In terms of what you have known and used in circuit analysis, if the black box was a simple amplifier, then the Output,  $Y$ , over the Input,  $U$ , would be the gain of the black box. A more precise definition of the transfer function,  $G(s)$ , would be the LaPlace transform of the output divided by the LaPlace transform of the input. Dwell on this Figure 6-1 for a while, as it is one of the most important concepts you will find for the practical application of MATLAB and Simulink to the solution of technical problems.

### 16-4: LaPlace Transform, parameters in $s$ .

Wikipedia starts their definition of the LaPlace Transform with the widespread areas of applications in math, physics, optics, electrical engineering, control engineering, signal processing, and probability theory, all applications we will

come to in chapters to follow as we have behind us some appreciation for the concepts and equations of the transfer function, the LaPlace Tranform, the Fourier Transform, and State-Space Equations. Both the LaPlace Transform, L, and the Fourier Transform are math techniques for the solution of differential and integral equations; but so detailed are many aerodynamic and control engineering problems that solutions are expedited with an up front conversion of DEs into functions of  $s$  through the LaPlace transform, then after solution of the large problems, to convert back from  $s$  parameters into DEs through integration. {Most often in the usage of standard tables of LaPlace transforms.} Wikipedia continues in their history and defintion of the LaPlace transform with the statement that these transformation are from the time-doman {recall that DEs often are integrated or differentiated with respect to time,  $t$ } to the frequency domain of angular frequency in radians/sec or cycles/sec.  $f(t)$  represents the original function in the DE form and as a function of time,  $t$ ; and  $F(s)$  represents the function transformed into algebraic variables of  $s$  with the LaPlace transform.

$F(s) = \int_0^{\infty} f(t) e^{-st} dt$  after it is multiplied by  $e^{-st}$  and  $dt$  which is integrate over time, or

Equation 5-1: Definition of the LaPlace Transform, L.

$$F(s) = \mathcal{L}\{f(t)\} = \int_0^{\infty} e^{-st} f(t) dt$$

$$F(s) = \mathcal{L}\{f(t)\} = \int_0^{\infty} (e^{-st} f(t)) dt$$

### 16-5: State-Space Variable Equations.

We naturally think of space variables as the  $x, y,$  and  $z$  coordinates of the cartesian coordinate system like where an object moves in three dimensional space with a precise location of  $x, y,$  and  $z$ . Well, to combine such space coordinates with state variables in a little more complex, but is so feasible in the descrip-

tion of many mechanical, electrical, and control systems in terms of both space and state, where the state variables describe the state or states inside the black box. For example, if the LCR circuit of the next section is under analysis, one state would be the magnetic energy stored in the inductor and another state would be the electric energy stored in the capacitor. Before we start applying the State-Space equations to such an LCR circuit, let us show the two state-space equations in their general equation form as shown in Equations 6-2.

Equations 6-2a,b: The State-Space Equations.

$$\dot{x} = Ax + Bu$$

$$y = Cx + Du$$

where you immediately recognize  $x$ ,  $y$ , and the derivative of  $x$ ,  $\dot{x}$ , as the physical parameters of space. You even recognize  $y$  as the standard symbol for output and  $u$  for input, and are leading naturally to start looking for the State variables thus hitting on  $A$ ,  $B$ ,  $C$ , and  $D$  as the variables of state.

It is impossible to stress how important these equations for state-space variables are in computer and engineering applications; and if we had like in other chapters one DE of chapter emphasis, for this chapter it would be the two State-Space Equations of Equatons 6-2. Of course only the top of the two equations is a DE, which could be written as  $\dot{x} = Ax + Bu$ ; and the second equation is recognized as a simple linear equation of to solve for  $y$  with the variables of  $x$  and input,  $u$ , and the constants of  $C$  and  $D$ .

Literature on the subject, like in the currently popular Wikipedia, stresses how state variables are "user-defined" which also serves to tell us that there are several ways to define the internal states for any given system or black box.

It is easy to think too much of State-Space. Once again, we can easily go from a DE like  $\dot{x}(t) + x(t) = 0$  {with initial condition of  $x(0) = 1$  and a solution required over the interval 0 to 10} to the State-Space format of:

$$\dot{x}' = Ax(t) + Bu(t) \qquad y(t) = Cx(t) + Du(t)$$

and it is understood that  $x$  is the "state";  $u(t)$  the input; and  $y(t)$  the output. Therefore the first of the two equations above is called the state equation, and the second the output equation. Now we start plugging our DE into the standard state-space format by allowing  $y(t) = x(t)$ ; therefore:

## The Process

$$dx(t)/dt = -1x(t)$$

$$y(t) = x(t)$$

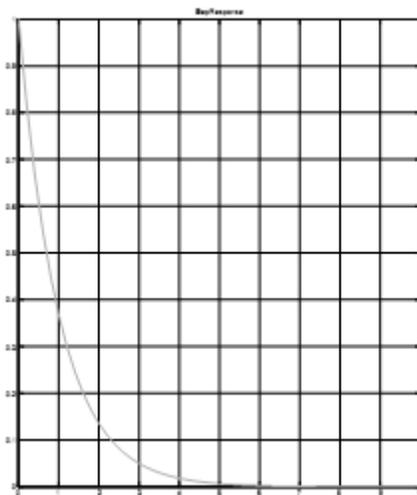
and  $A = -1$ ,  $C = 1$ ,  $B = 0$ , and  $D = 0$  {recall that  $u(t)$  is also 0.

It feel good about a solution, let us solve it by Simulink with a model block as follows.



This time in Simulink, and with a new model window, grab the State-Space block and a scope block, interconnect, and double click on the State-Space block. Fill in  $A$  thru  $D$  as agreed above, respectively -

1,0,1, and 0 and with an initial condition of 1. Then run Start under Simulation. {Oh, don't forget to hit "Apply" before you close the State-Space window for Block Parameters.}



**NOTE:** Aren't you impressed that you can so easily with Simulink visualize a DE, and even more so that you can easily get into the State-Space format?

### 16-6: Solution of Second Order DE by State-Space.

It is a little more challenge to setup the Simulink of the following DE, and especially to first put it in standard State-Space for-

mat.

$$2x'' + 2x' + 4x = 3u(t)$$

and once again  $u(t)$  is the unit step input function, this time the time interval is 10 seconds, or 0 to 10 seconds; and we have two initial conditions of  $x(0) = 2$  and  $x'(0) = -1$ . We will use the same block model as shown above; but this time, however, our  $A, B, C, D$ , and initial conditions will be quite different. It is also a little more complicated to put this second order DE into standard state-space

## The Process

form. The state is a state vector of two components,  $x_1$  and  $x_2$ , so that each will have a state equation:

$$x_1' = Ax_1 + Bu$$

and

$$x_2' = Ax_2 + Bu$$

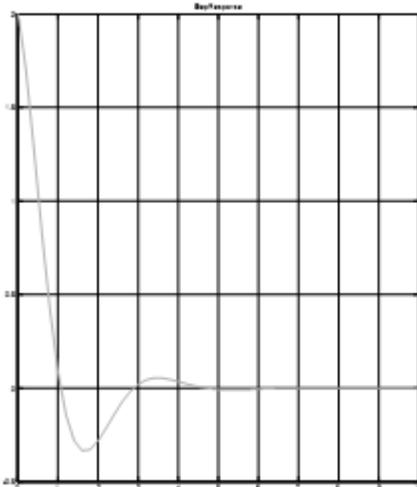
Define {you can do anything by definition}  $x_1 = x$  and  $x_2 = x_1'$  with the first state equation of  $x_1' = x_2$ ; and the second state equation,  $x_2'$ , is found by substituting  $x_1$  and  $x_2$  into the original DE. {Actually for  $x_2$  put  $x_1'$ , making the second derivative of  $x_1$ , or  $x_1'' =$  to the first derivative of  $x_2$ , or  $x_2'$ ; therefore the DE now becomes:  $2x_2' + 2x_2 + 4x_1 = 3u(t)$ , and the state-space form is:

$$x_1' = x_2$$

and

$$x_2' = -4x_1 - 2x_2 + 3u(t)$$

and the matrix format would be:



$$\begin{array}{r} x_1 \\ u(t), \\ x_2 \end{array} y(t) = \begin{bmatrix} 0 & 1 \\ -4 & -2 \end{bmatrix} \begin{array}{r} x_1 \\ x_2 \end{array} + \begin{bmatrix} 0 \\ 4 \end{bmatrix} u(t)$$

where you see  $A = [0 \ 1; -4 \ -2]$ ,  $B = [0, 4]$ ,  $C = [1 \ 0]$ , and  $D = 0$ . Thus use the same block diagram, but fill in the new appropriate  $A, B, C$ , and  $D$  and with the initial conditions of  $[2 \ -1]$ . And by double clicking again on the scope and using the autoscale, you see the plot on the left with a neat little dip and then a hump before leveling off.

### Electric Circuit.

These concepts and techniques of engineering math become quite clear as we apply them to an electric circuit like the one shown in Figure 6-2. You can see by the skematic symbols for  $v_{in}$ ,  $L$ ,  $R$ , and  $C$  that it is a simple series LCR circuit with a single power source,  $v_{in}$ , an inductor,  $L$ , a single resistor  $R_1$ , and a capacitor,  $C$ . {The same circuit of chapter 1.} If a unit step input were applied to the LCR circuit of Figure 6-2 instantly the capacitor would act like a short and the inductor like an open, indicating the stored energy state of the capacitor as  $v$