

1. Record Nr.	UNINA9910953434403321
Autore	Karris Steven T
Titolo	Introduction to Simulink with engineering applications / / Steven T. Karris
Pubbl/distr/stampa	[Fremont, Calif.], : Orchard Publications, 2011
ISBN	1-283-25284-8 9786613252845 1-934404-22-5
Edizione	[3rd ed.]
Descrizione fisica	1 v. (various pagings) : ill. (some col.)
Disciplina	518
Soggetti	Engineering mathematics - Data processing Computer simulation - Computer programs
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Title from title screen.
Nota di bibliografia	Includes bibliographical references and index.
Nota di contenuto	Intro -- Simulink 2011 TOC All Chapters -- Simulink 2011 Ch 01 -- Simulink 2011 Ch 03 -- Chapter 3 -- 3.1 An Application of Newton's Second Law -- 3.2 Ball Thrown Vertically Upward -- 3.3 Barge Pulled by Horse Teams -- 3.4 Rated Speed of an Automobile on a Highway Curve -- 3.5 Train Traveling on a 3 Per Cent Grade -- 3.6 Spring with Upper End Fastened to a Rigid Support -- Simulink 2011 Ch 04 -- Chapter 4 -- 4.1 Rise of a Liquid in a Capillary Tube -- 4.2 An Application of a Simplified Form of the Bernoulli Equation -- 4.3 Takeoff Speed for an Aircraft -- 4.4 An Application of the Bulk Modulus -- 4.5 Calculation of Fluid Flow in an Open Channel -- Simulink 2011 Ch 05 -- Chapter 5 -- 5.1 Chemical Solutions -- 5.2 Heat Flow -- 5.3 Cantilever Beam Deflection -- 5.4 Tractrix Curve -- 5.5 Bessel Differential Equation -- 5.6 Van der Pol Differential Equation -- 5.7 Simple Pendulum -- 5.8 Simple Oscillator -- 5.9 Zero-Order Hold and First-Order Hold as Reconstructors -- 5.10 Direct Form Realizations of a Digital Filter -- 5.11 Series Form Realization of a Digital Filter -- 5.12 Parallel Form Realization of a Digital Filter -- 5.13 3-bit Up / Down Counter -- 5.14 4-bit Ring Counter -- 5.15 Mass-Spring-Dashpot -- 5.16 Cascaded Mass-Spring System -- 5.17 Mechanical Accelerometer -- 5.18 Feedback Control Systems -- 5.19 Electric Circuit in Phasor Form -- 5.20 Application of the Superposition Principle -- 5.21 Discrete Time

Integration with Variable Amplitude Input -- 5.22 S-Function Example 1 -- 5.23 S-Function Example 2 -- 5.24 Concluding Remarks --
 Simulink 2011 Appendix A -- Appendix A -- A.1 MATLAB® and Simulink® -- A.2 Command Window -- A.3 Roots of Polynomials -- A.4 Polynomial Construction from Known Roots -- A.5 Evaluation of a Polynomial at Specified Values -- A.6 Rational Polynomials -- A.7 Using MATLAB to Make Plots -- A.8 Subplots.
 A.9 Multiplication, Division, and Exponentiation -- A.10 Script and Function Files -- A.11 Display Formats -- Simulink 2011 Appendix B -- Appendix B -- Masked Subsystems -- this appendix presents an overview of masked subsystems, and a step-by-step procedure to create custom user interfaces, i.e., masks for Simulink subsystems. --
 B.1 Masks Defined -- A mask is a custom user interface for a subsystem. A masked subsystem conceals the subsystem's contents, and it appears to the user as an atomic block with its own icon and parameter dialog box. However, a masked subsystem provides only graphical, not... -- B.2 Advantages Using Masked Subsystems -- A masked subsystem allows us to -- 1. Replace the parameter dialogs of a subsystem and its contents with a single parameter dialog with its own block description, parameter prompts, and help text. -- 2. Replace a subsystem's standard icon with a custom icon that shows its purpose. -- 3. Prevent accidental modification of subsystems by concealing their contents behind a mask. -- 4. Placing a masked subsystem in a library. We can also mask S-Function and Model blocks. -- B.3 Mask Features -- Masks can include any of the following features: -- Mask Icon - The mask icon replaces a subsystem's standard icon, i.e., it appears in a block diagram in place of the standard icon for a subsystem block. Simulink uses MATLAB code that we supply to draw the custom icon. We can use any MATLAB drawing c... -- Mask Parameters - Masked subsystems allow us to define a set of user-specified parameters. Simulink stores the values of these parameters in the mask workspace as the value of a variable whose name you specify. These associated variables allow us to ...
 Mask Parameter Dialog Box - The mask parameter dialog box contains controls that enable a user to set the values of the mask's parameters and hence the values of any internal parameters linked to the mask parameters. The mask parameter dialog box rep... -- Mask Initialization Code - The initialization code is MATLAB code that you specify and that Simulink runs to initialize the masked subsystem at critical times, such as model loading and the start of a simulation run (see Initialization Pane). You can... -- Mask Workspace - Simulink associates a workspace with each masked subsystem that you create. Simulink stores the current values of the subsystem's parameters in the workspace as well as any variables created by the block's initialization code and par... -- A block parameter expression can refer only to variables defined in the mask workspaces of the subsystem or nested subsystems that contain the block or in the model's workspace. -- A valid reference to a variable defined on more than one level in the model hierarchy resolves to the most local definition. -- For example, let us suppose that model M contains masked subsystem A, which contains masked subsystem B. Also, let us suppose that B refers to a variable x that exists in both A's and M's workspaces. In this case, the reference resolves to the value ... -- A masked subsystem's initialization code can refer only to variables in its local workspace. -- The mask workspace of a Model block is not visible to the model that it references. Any variables used by the referenced model must resolve to workspaces defined in the referenced model or to the base (i.e., the MATLAB) workspace. -- B.4 Creating a Masked Subsystem -- It is best

to illustrate the creation of a masked subsystem with an example. -- Example B.1 -- The Simulink model in Figure B.1 below implements the quadratic equation .

Figure B.1. Simulink model for Example B.1 -- To create a subsystem, we replace the Unknown x block with an Inport block, we encircle all blocks except the Inport and Outport blocks, and from the Edit drop menu we select Create Subsystem. The model now appears as shown in Figure B.2. -- Figure B.2. The model in Figure B.1 shown as a subsystem block -- To see the contents of the Subsystem in Figure B.2, we double-click the Subsystem block and now the model appears as shown in Figure B.3. -- Figure B.3. The contents of the subsystem block -- With the Subsystem block in Figure B.2 selected, from the Edit drop menu we click on the Mask Subsystem and the Mask Editor window appears as shown in Figure B.4. With the Icon tab selected as shown, we position the text cursor in the Icon Drawing co... -- Figure B.4. The Mask Editor window for Example B.1 -- Figure B.5. The masked subsystem with an imported image -- We right-click on the Subsystem block in Figure B.5, and from the drop menu we select Edit Mask. From the Mask Editor window which appears, we select the Parameters tab shown in Figure B.6 below. -- Figure B.6. The Parameters tab for the Mask Editor window -- We select the Add tool and the Mask Editor window now appears as shown in Figure B.7. -- Figure B.7. The Mask Editor window for specifying the attributes of the masked parameters -- The Mask Editor in Figure B.7 is used to specify the attributes of the masked parameters. The Prompt column under Dialog parameters is used as a text label to describe the parameter. For our example we enter Constant a, Constant b, and Constant c. Th... -- Figure B.8. The Masked Editor with the equation constants specified. We right-click on the masked subsystem block shown in Figure B.5, Page B-5, and in the Function Block Parameters dialog box we enter the values 1, -5, and 6 for the variables a, b, and c respectively, as shown in Figure B.9. -- Figure B.9. The Function Block Parameters window with the values of the constants -- With the variables defined as above, the masked subsystem implements the quadratic equation -- and the roots of this equation are and . Our model is tested for the first root as shown in Figure B.10. -- Figure B.10. -- The Mask Editor also contains the Initialization tab that allows us to enter MATLAB commands that initialize the masked subsystem, and the Documentation tab that lets us define or modify the type description and help text for a masked subsystem. Thes... -- Simulink 2011 Appendix C -- Appendix C -- C.1 Inport, Outport, and Subsystem -- C.2 Ground -- C.3 Terminator -- C.4 Constant and Product -- C.6 Bus Creator and Bus Selector -- C.7 Mux and Demux -- C.8 Switch -- C.9 Sum -- C.11 Relational Operator -- C.14 Integrator -- C.15 Unit Delay -- C.16 Discrete-Time Integrator -- C.17 Data Types and Data Type Conversion -- C.18 Vector Concatenate -- Simulink 2011 Appendix E -- Appendix E -- E.1 Saturation -- E.2 Saturation Dynamic -- E.3 Dead Zone -- E.4 Dead Zone Dynamic -- E.5 Rate Limiter -- E.6 Rate Limiter Dynamic -- E.7 Backlash -- E.8 Relay -- E.9 Quantizer -- E.10 Hit Crossing -- E.11 Coulomb and Viscous Friction -- E.12 Wrap to Zero -- E.13 Nonlinear Systems - Describing Functions -- Appendix F -- F.3 Tapped Delay -- F.4 Discrete-Time Integrator -- F.5 Discrete Transfer Fcn -- F.6 Discrete Filter -- F.7 Discrete Zero-Pole -- F.8 Difference -- F.9 Discrete Derivative -- F.10 Discrete State-Space -- F.11 Transfer Fcn First Order -- F.12 Transfer Fcn Lead or Lag -- F.13 Transfer Fcn Real Zero -- F.14 Discrete FIR Filter. F.15 Memory.

contains five chapters on engineering applications and 20 appendixes describing all Simulink functional blocks followed by illustrative examples.
