

## Simulink Tool in Matlab Environment

Solijonov Jasurbek Valeriy ugli,

*student of Fergana state university*

**Annotation:** *This article discusses an interactive tool for modeling, simulating and analyzing dynamic Simulink systems and its capabilities with a system of differential equations*

**Key words:** *Tool, Simulink, Matlab, model, differentiation, transformation, command, block, source*

The name MATLAB is short for Matrix Laboratory, and the MATLAB package was originally developed as a means of accessing the LINPACK and EISPACK libraries for matrix computing. The MATLAB package was created by Math Works about twenty years ago.

Currently, MATLAB is a powerful and versatile tool for solving problems arising in various fields of human activity. The spectrum of problems that can be studied using MATLAB and its extensions (Toolbox) covers: matrix analysis, signal and image processing, mathematical physics problems, optimization problems, financial problems, data processing and visualization, work with cartographic images, neural networks, fuzzy logic and more. Many Toolbox applications include graphical user interface applications that provide quick and intuitive access to basic functions. Simulink package, supplied with MATLAB, is intended for interactive modeling of nonlinear dynamic systems consisting of standard blocks.

Simulink is an interactive tool for modeling, simulation and analysis of dynamic systems. It provides the ability to build graphical block diagrams, simulate dynamic systems, explore system performance and improve projects. Simulink is fully integrated with MatLab, providing immediate access to a wide

range of analysis and design tools. Simulink also integrates with Stateflow to simulate the behavior caused by events.

Simulink contains:

- An extensive library of blocks for the construction of graphical models of systems.
- Simulink graphical debugger for interactive localization and diagnosis of errors in the model.
- Model Browser for hierarchical navigation.
- Search dialog box for searching models and libraries.
- Custom blocks that can incorporate existing custom C, Ada, MatLab, and Fortran codes.

When developing large models, you can use:

- Linear, nonlinear, continuous and discrete in time, with a variable transmission speed of signals, with the specified execution conditions and hybrid systems.
- Models can be grouped hierarchically for efficient viewing of components or subsystems.
- Simulink data objects allow you to create application-specific data types MatLab for Simulink models.
- A special Simulink graphical user interface for monitoring and editing Simulink data objects.
- Library browser for convenient block selection.
- Simulations can be run from the MatLab command line, either interactively or in batch mode.

Also present in Simulink:

- Support matrix signals and operations.
- The bitwise logical operator logically masks, reverses, or splits an integer signal into bits.

Any dynamic system is described by a system of differential equations. Having decided such a system, you can go to the transfer function of the dynamic system. The transfer function is the ratio of the output Laplace images

the parameter to the input with zero initial conditions:  $W(s) = \frac{Y(s)}{X(s)}$ , where  $s$  is the operator

Laplace. Consider a simple differential equation:

$$a_1 \dot{y}(t) + a_2 y(t) = b_1 x(t).$$

Apply the Laplace transform to it, that is, the differentiation operator is replaced by multiplication by  $s$ . We get:

$$sa_1 Y(s) + a_2 Y(s) = b_1 X(s).$$

After simplification, we obtain the transfer function

$$W(s) = \frac{Y(s)}{X(s)} = \frac{b_1}{a_1 s + a_2}$$

Simulink allows you to solve numerically a system of differential equations.

Creating a model: Consider an RC-chain and build its model.

$$u_{BX}(t) = i(t)R + u_{CB}(t),$$

$$i(t) = \frac{C du_C(t)}{dt} = \frac{C du_{CB}(t)}{dt}$$

$$u_{BX}(t) = RC \frac{du_{CB}(t)}{dt} + u_{CB}(t)$$

After applying the Laplace transform we get:

$$U_{BX}(s) = RCs U_{CB}(s) + U_{CB}(s),$$

$$W(s) = \frac{Y(s)}{X(s)} = \frac{1}{RCs + 1}$$

To create a model in the Simulink environment, you must perform a series of actions sequentially:

1. Create a new model file using File → New → Model, or using the button on the toolbar (hereinafter, using the “/” symbol indicates the program menu items that must be sequentially selected to perform the specified action).

2. Arrange the blocks in the model window. To do this, open the appropriate section of the library (for example, Sources - sources). Next, pointing the cursor to the desired block and clicking on the left mouse button, drag the block into the created window. The mouse key must be pressed.

To delete a block, select a block (point at its image with the cursor and press the left mouse button), and then press the Delete key on the keyboard. To change the block size, you need to select a block, set the cursor to one of the corners of the block and, by pressing the left mouse button, change the block size (the cursor will turn into a double-sided arrow). If a block needs to be rotated (say, it is in the feedback loop), then it is necessary to select this block and press the Rotate block in the Format menu.

3. Further, if required, you need to change the block parameters set by the “default” program. To do this, double-click the left mouse button, pointing the image of the block with the cursor. A window for editing the parameters of this block will open. When specifying numerical parameters, it should be borne in mind that a decimal point should be used as the decimal separator and not a comma. After making changes, close the window with the OK button. In Figure 4, as an example, a block modeling the transfer function and the window for editing the parameters of this block are shown.

4. After installing on the diagram all the blocks from the required libraries, you need to connect the schema elements. To connect the blocks, you need to point the cursor to the “exit” of the block, and then press and, without releasing the left mouse button, draw a line to the input of another block. Then release the key. In case of a correct connection, the image of the arrow at the input of the block changes color. To create a branch point in the connecting line, move the cursor to the intended node and, by pressing the right mouse button, draw a line. To delete a line, you need to select a line (in the same way as for the block), and then press the Delete key on the keyboard.

After drawing up the calculation scheme, it is necessary to save it as a file on disk by selecting the menu item File→ Save As in the scheme window and specifying the folder and file name. It should be borne in mind that the file name must not exceed 32 characters, must begin with a letter and cannot contain Cyrillic characters and special characters. The same requirement applies to the file path (to the folders in which the file is saved). Upon subsequent editing of the scheme, you can use the menu item File→ Save. When you restart the program, use the File menu → File– Open in the library browser window or from the main window MatLab.

#### **USED LITERATURE:**

1. *Simulink modeling / Moscow State Institute of Electronic Technology (Technical University) / Moscow 2007*
2. <http://csaa.ru/vychislitel'naja-sreda-matlab/>