Variables

Variable is a named memory cell, to which a value can be written or from which a value can be read out. Variables can be created by means of **var**, **const**, **init**, **output**, **input** key words or automatically according to an expression assigned to the variable, either those can be obtained from the submodel Global Parameters. Variables can have one of the following standard types.

Type identifier Data type description

**integer** Integer significant 32-bit number

**double** Real 64-bit number

**complex** Complex 128-bit number. It is provided with output format a+{-}bi, where a – is for real part and b is for complex part. Complex number can be set as (a,b)

**intarray** (Row-vector) array for integers

**array** (Row-vector) array for real numbers

**carray** (Row-vector) array for complex numbers

**matrix** Matrix of real numbers

**cmatrix** Matrix of complex numbers

**boolean** Binary 1-bit value, can be 0 or 1

**color** Color – is equivalent to **integer** type

**point** Geometric point – is equivalent to complex number, has output format  
(a, b), where a is for the abscissas and b is for the ordinates.

**string** String of symbols

***Note.*** Hereinafter vector is understood as an array (row-vector) of numbers, while column-vector is a matrix consisting of several rows and one column.

Beside standard types of variables there can be variables of a complex type – *record.*

*The record* is declared during identification of the variable data type according to the following format:

<field name 1>{:<field type 1>}{=<field initial value 1>}{…}

**Example.**

|  |  |
| --- | --- |
|  | **var** newrec : (x = 0.0 , y = 0.0); |

***Note.*** Fields inside a complex-type declaration are defined in the same way as **var** key word.

Access to the field of a complex variable is ensured by means of operator “ **->** ” according to the following format:

<record name> **->** <record field name>

**Example.**

|  |  |
| --- | --- |
|  | A = newrec**->**x; |

The following system variables are available in the “Programming language” block:

|  |  |  |
| --- | --- | --- |
| Variable | Data type | Purpose |
| **time** | Real (double) | Simulation time |
| **stepsize** | Real (double) | Step of integration |
| **goodstep** | Binary (Boolean) | “Good” step flag |
| **getderi** | Binary (Boolean) | Flag of calculation of derivative values |
| **setstepflag** | Binary (Boolean) | Flag of forced step assignment |
| **newstepvalue** | Real (double) | Forced step value in case **setstepflag = 1** |

goodstep variable possesses value 1 if calculations are executed in the end point of the step of integration and at the same time the condition for specified accuracy is met; in all other cases goodstep variable possesses value 0. getderi variable possesses value 1 during calculation of the matrix of Jacobi by the method of numerical differentiation; in all other cases goodstep variable possesses value 0.

The following special variables are accessible in the animation module:

|  |  |
| --- | --- |
| Name | Purpose and example |
| **GROUPCOLOR** | Group background color |
| **GLOBALCOLOR** | Main window background color |
| **PAINTSTEP** | Flag to be set if the window image is redrawn. |
| **CONTAINER\_NAME** | Name of the graphics container, in which the script is written. |
| **SCALEX** | Set of scaling of the container in X-axis. |
| **SCALEY** | Set of scaling of the container in Y-axis. |
| **KLINE** | Set of scaling of lines inside the container. |
| **CHANGE** | Flag of forced redrawing of graphics container. |
| **PARENTOBJECT** | Reference to an object-owner of the container (to operate the object from inside of the container). |
| **old\_project\_name** | Name of previous file loaded into the window. |
| **old\_link\_id** | Name of the last reference identifier loaded into the window |
| **last\_active\_screen\_id** | screen\_id of the last active window |
| **screen\_id** | Identifier of the editor window – a random string assigned in the script. |
| **last\_active\_project\_id** | Window reference of the last active project. |
| **SYSTEM\_DB\_ROOT** | Storing path for different data files installed for graphics shell. |
| **DYNAMIC\_PROPERTY\_ITEMS** | Variable for helping selection through the script (for scripts inside the block properties) |
| **last\_active\_hwnd** | Indicator of the last active window (handl) |
| **restartstep** | Step for storing restarts |
| **time** | Simulation time |
| **submodel** | Indicator of submodel parent-block |

Variables can be set using **var, const, input, output, init** key words.