# Xcos for very beginners

# Table of content

## Introduction

## Become familiar with Xcos

## Annexes

## Introduction

### About this document

The purpose of this document is to guide you step by step in exploring the various basic features of Xcos tool included in Scilab for a user who has never used a hybrid dynamic systems modeler and simulator. This presentation is intentionally limited to the essential to allow easier handling of Xcos.

The examples, diagrams and illustrations are made with Scilab 5.4.1. You can reproduce all those examples from this version.

### Install Scilab

Scilab is open source software for numerical computation that anybody can freely download. Available under Windows, Linux and Mac OS X, Scilab can be downloaded at the following address: http://www.scilab.org/

You can be notified of new releases of Scilab software by subscribing to our notification channel at the following address: http://lists.scilab.org/mailman/listinfo/release

### Mailing lists

To ease the exchange between Scilab users, dedicated mailing lists exist (list for the education world, international list in English). The principle is simple: registrants can communicate with each other by e-mail (questions, answers, sharing of documents, feedbacks...).

To browse the available lists and to subscribe, go to the following address: http://www.scilab.org/communities/user_zone/mailing_list

### Complementary resources

Scilab website has a dedicated section on Scilab use (http://www.scilab.org/en/resources/documentation) with useful links and documents that can be freely downloaded and printed.

You can also find in the same section a similar document entitled "Scilab for very beginners" which is available for download.

## Become familiar with Xcos

Numerical simulation is nowadays essential in system design process. Complex phenomena simulation (physical, mechanical, electronics, etc.) allows the study of their behavior and results without having to conduct costly real experiments. Widely used in the world of industry, the future generation of engineers and scientists are trained since secondary school to the concepts of modeling and simulation.

Xcos is Scilab tool dedicated to the modeling and simulation of hybrid dynamic systems including both continuous and discrete models. It also allows simulating systems governed by explicit equations (causal simulation) and implicit equations (acausal simulation). Xcos includes a graphical editor which allows to easily represent models as block diagrams by connecting the blocks to each other. Each block represents a predefined basic function or a user-defined one.

### General environment

After launching Scilab, the environment by default consists of the console, files and variables browsers and a command history.
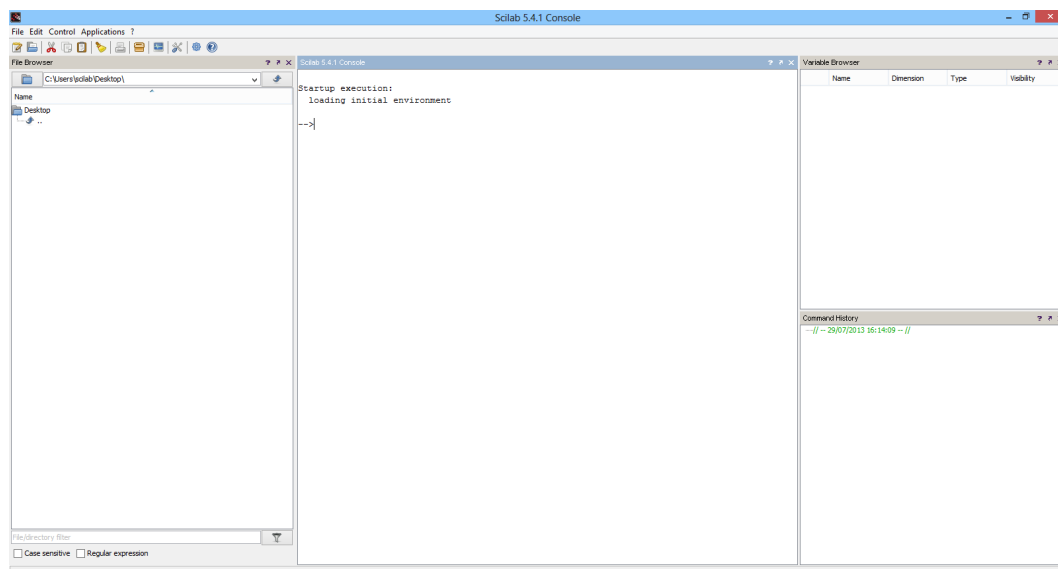


In the console after the prompt " **-->** ", just type a command and press the Enter key to obtain the corresponding result.
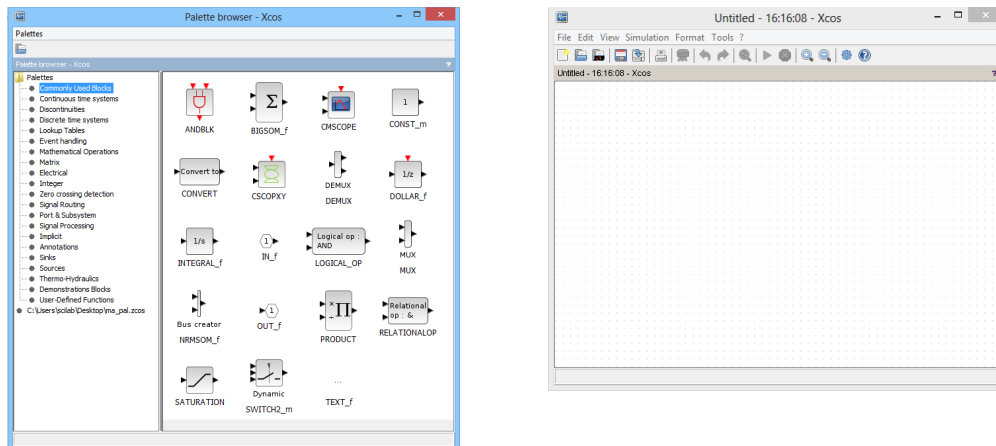
Xcos can be launched:

- From the toolbar, via the ⬚ icon, or
- From the menu bar, in Applications / Xcos, or
- From the console, in typing:

```
-->xcos
```

Xcos opens by default with two windows:

- A palette browser which provides a set of predefined blocks,
- An editing window which is the working space to design diagrams.
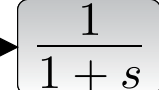


To design a diagram, just select blocks in the palette browser and position them in the editing window (click, drag and drop). Blocks are then connected to each other using their different ports (input, output, event) in order to simulate the created model.

## Example of a simple diagram design

We are going to explain how to design from scratch a model of continuous-time system represented by a first-order transfer function.

Launch Xcos.

As seen before, Xcos opens by default with the palette browser and an editing window. In the palette browser, we are going to use the following blocks:

| Designation | Representation | Sub-palette |
|---|---|---|
| Step |  | Sources / STEP_FUNCTION |
| Continuous transfer function | $$\frac{1}{1+s}$$ | Continuous time systems / CLR |
| Clock |  | Sources / CLOCK_c |
| Visualization |  | Sinks / CSCOPE |

Arrange the blocks in the editing window. To connect the input and output ports to each other, click on the output (black arrow) of the STEP-FU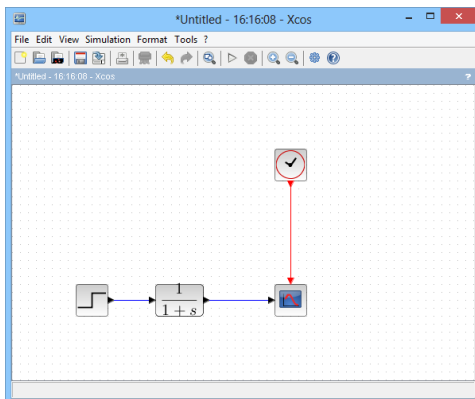NCTION block  and in maintaining the mouse button pressed, connect it to the input port of the CLR block (a green highlighted square appears to indicate that the link is correct) as described in the images below:
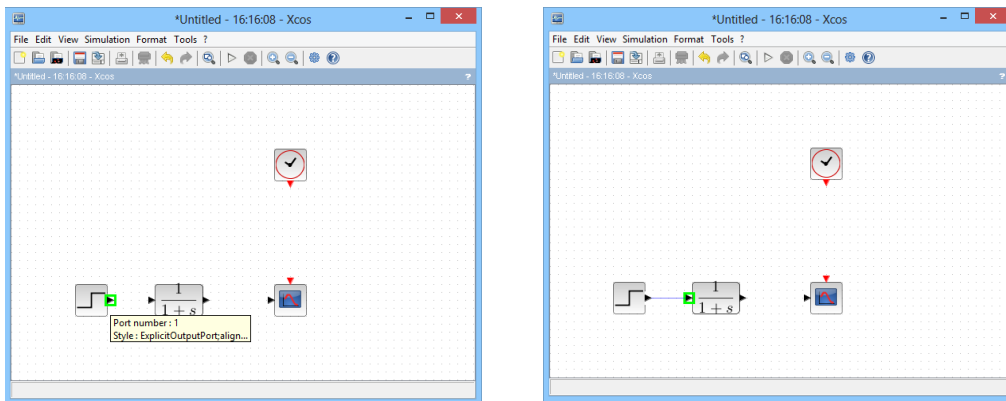


Release to complete the link.

Complete the connections between the different blocks to achieve this result:



It is possible to improve the general look of your diagram in using the blocks alignment options (Format Menu/Align blocks) and the links style (Format Menu/Link style). At any time, blocks can be moved or repositioned by selecting them and by maintaining the mouse button pressed while moving them. Release the blocks in the desired position.

Simulation is launched by clicking on the  icon (or from the Simulation Menu/Start) and can be stopped by clicking on the  icon (or from the Simulation Menu/Stop).

A new window is displayed (scope) with the simulation running. At the bottom of the diagram editing window, a statement indicates that the simulation is in progress:
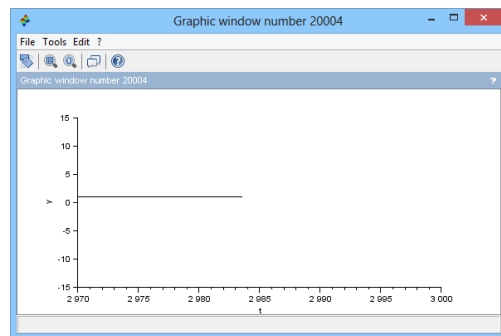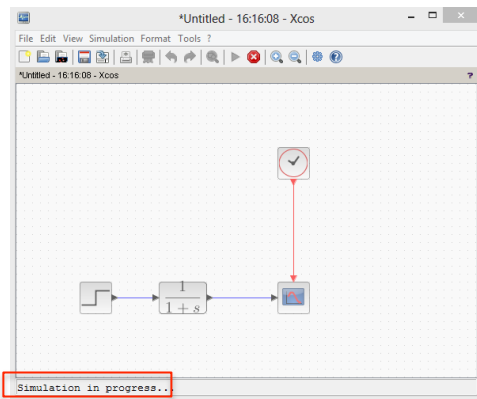
You can see that the simulation time is quite long (you may have needed to stop the simulation while running) and that the response is flat. Thus, we choose to modify the parameters of CLR block $\dfrac{1}{1+s}$ and of the simulation.

A "Context" containing Scilab script allows easy use of functions and variables in Xcos blocks. We are going to use this context to set the blocks parameters for diagram simulation.

Click on Simulation/Set Context in the menu bar and define the following variables:

- K = 1
- Tau = 1
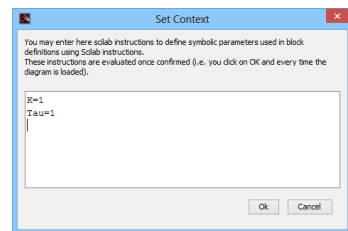


You can now use these variables to set up the diagram blocks.

Double-click on CLR block. A dialog box opens with the default settings of the block. Modify these settings as follows:

- Numerator: K
- Denominator: 1+Tau*s



The new transfer function is displayed on the block:



If necessary, enlarge the block so that the display fits in it.

We are now going to set up the simulation and the blocks to visualize the time response of the system to a stepe. For this, we limit the simulation time to 5 seconds (Simulation Menu/Setup) in modifying the final integration time.

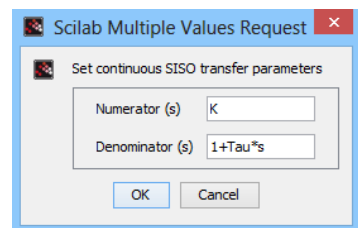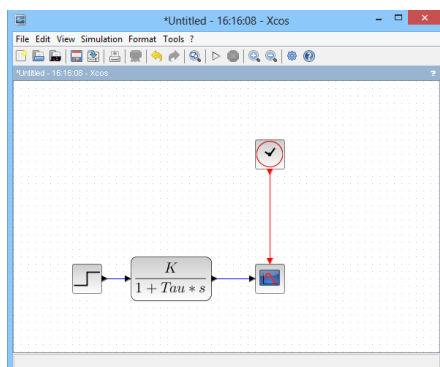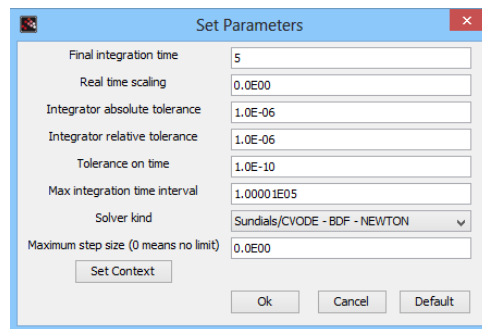Double-click on CSCOPE block to set up the display of values between 0 and 1.2, then the scope refresh period to 5 seconds. To do it, change the following settings:

- Ymin: 0
- Ymax: 1.2
- Refresh period: 5
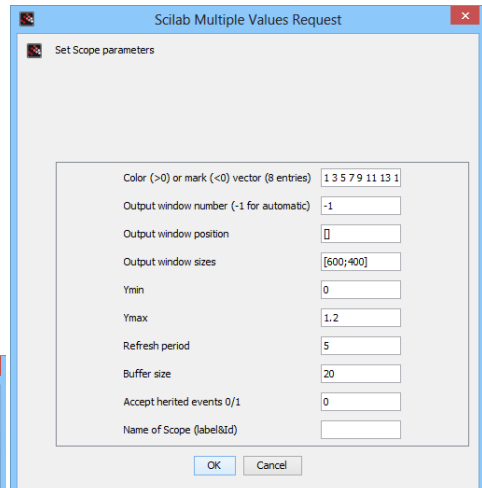
Restart the simulation and view the result:
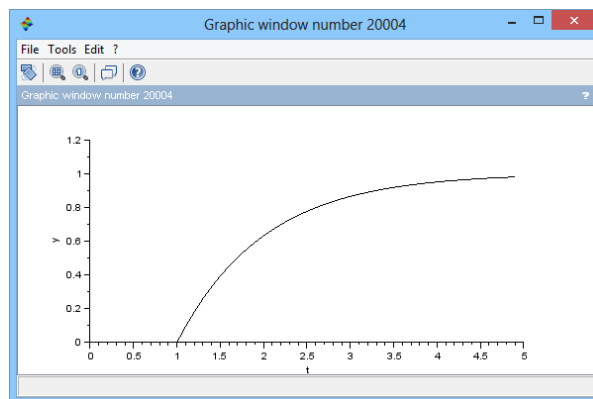
## Superblocks

To ease the understanding of certain diagrams, it is often useful to use superblocks or composite blocks. A superblock contains a part of a diagram and blocks representing its inputs and outputs. The superblock can be handled as a single block within the parent diagram.
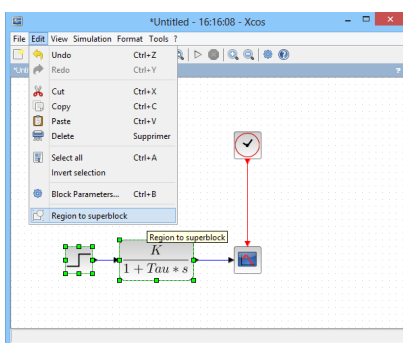
After designing a diagram and selecting the part of the diagram (or sub-diagram) that we want to gather into a block, the creation of the superblock is done from the menu Edit/Region to superblock.

The selection is now a block which content can be displayed by double-click. A new editing window opens with the initial selected blocks.
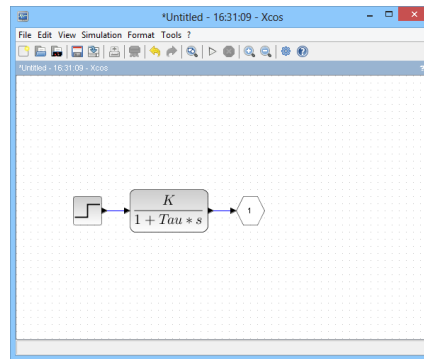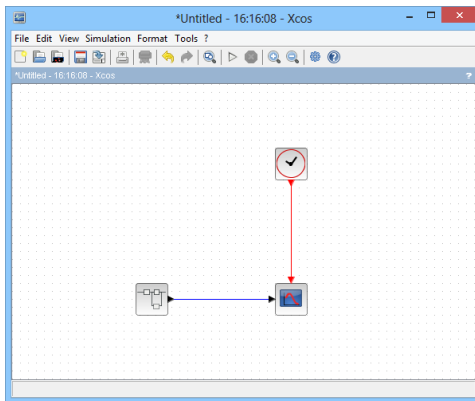
It is also possible to hide the created superblock to disable access to the subdiagram. To do so, right-click on the superblock, then on Superblock Mask/Create.

We can also make some sub-diagram configuration settings available in a single setup interface by a right-click on the superblock, then Superblock Mask/Customize.

Then just add the parameters you want to make available.

This presentation was intentionally short and many other possibilities for simulating systems exist with many available blocks. To continue to handle easily Xcos, we invite you to visit the many diagrams examples available in Xcos demonstrations by clicking on the menu ?/Xcos Demos.

## Annexes

### Menu bar

The useful menu bar of Xcos is the one of the editing window.

#### File Menu

- **New diagram** (Ctrl+N under Windows and Linux / Cmd+N under Mac OS X)
  Open a new Xcos editing window.
- **Open** (Ctrl+O under Windows and Linux / Cmd+O under Mac OS X)
  Load a Xcos file including a diagram or a palette in .zcos or .xcos format.
- **Open file in Scilab current directory**
  Load a Xcos file including a diagram or a palette from Scilab working directory in .zcos or .xcos format.
- **Recent files**
  Provide the recently opened files.
- **Close** (Ctrl+W under Windows and Linux / Cmd+W under Mac OS X)
  Close the current diagram if several diagrams are opened. Quit Xcos if only one diagram is opened. The auxiliary windows such as the palette browser are also closed in closing the last diagram.
- **Save** (Ctrl+S under Windows and Linux / Cmd+S under Mac OS X)
  Save the changes to the diagram. If the diagram has not been previously saved in a file, it will be proposed to be saved (see Save As).
- **Save as** (Ctrl+Shift+S under Windows and Linux / Cmd+ Shift +S under Mac OS X)
  Save the diagram or the palette with a new name. The diagram takes the name of the file (without the extension).
- **Export** (Ctrl+E under Windows and Linux / Cmd+E under Mac OS X)
  Export an image of the current Xcos diagram in standard formats (PNG, SVG, etc.).
- **Export all diagrams**
  Export images of the diagram and the content of its superblocks.
- **Print** (Ctrl+P under Windows and Linux / Cmd+P under Mac OS X)
  Print the current diagram.
- **Quit Xcos** (Ctrl+Q under Windows and Linux / Cmd+Q under Mac OS X)
  Quit Xcos.

#### Edit Menu

- **Undo** (Ctrl+Z under Windows and Linux / Cmd+Z under Mac OS X)
  Cancel the last operation.
- **Redo** (Ctrl+Y under Windows and Linux / Cmd+Y under Mac OS X)
  Restore the last operation canceled.
- **Cut** (Ctrl+X under Windows and Linux / Cmd+X under Mac OS X)
  Remove the selected objects of a diagram and keep a copy in the clipboard.
- **Copy** (Ctrl+C under Windows and Linux / Cmd+C under Mac OS X)
  Put a copy of the selected objects in the clipboard.
- **Paste** (Ctrl+V under Windows and Linux / Cmd+V under Mac OS X)
  Add the content of the clipboard in the current diagram.

- **Delete** (Delete)
  Erase the selected blocks or links. When a block is erased, all its connected links are also erased.
- **Select all** (Ctrl+A under Windows and Linux / Cmd+A under Mac OS X)
  Select all the elements of the current diagram.
- **Inverse selection**
  Reverse the current selection.
- **Block Parameters** (Ctrl+B under Windows and Linux / Cmd+B under Mac OS X)
  Set the selected block (see the block help to obtain more information on its setup).
- **Region to superblock**
  Convert a selection of blocks into a superblock.

### View Menu

- **Zoom In** (Ctrl+Plus numeric keypad under Windows and Linux / Cmd+Plus numeric keypad under Mac OS X)
  Enlarge the view of 10 %.
- **Zoom Out** (Ctrl+Minus numeric keypad under Windows and Linux / Cmd+Minus numeric keypad under Mac OS X)
  Reduce the view of 10 %.
- **Fit diagram or blocks to view**
  Adjust the view to the window size.
- **Normal 100 %**
  Scale the view to its default size.
- **Palette browser**
  Show / Hide the palette browser.
- **Diagram browser**
  Display a window which lists the global properties of the diagram and of all the objects it contains (blocks and links).
- **Viewport**
  Show / Hide a complete overview of the current diagram. With viewport, the user can move the working area on a particular part of the diagram.

### Simulation Menu

- **Setup**
  Modify the simulation parameters.
- **Execution trace and Debug**
  Set the simulation in a debug mode.
- **Set Context**
  Enable to enter Scilab instructions to define variables or functions that can be used in setting up diagram blocks.
- **Compile**
  Compile the diagram.
- **Modelica initialize**
  Enable the initialization of the variables in the acausal diagram subsystem.
- **Start**
  Launch the simulation.
- **Stop**
  Stop the simulation.

- **Rotate** (Ctrl+R under Windows and Linux / Cmd+R under Mac OS X)
  Rotate the selected block(s) 90° counterclockwise.
- **Flip** (Ctrl+F under Windows and Linux / Cmd+F under Mac OS X)
  Reverse the positions of input and output events positioned above and below of a selected block.
- **Mirror** (Ctrl+M under Windows and Linux / Cmd+M under Mac OS X)
  Reverse the positions of regular input and output positioned on the left and on the right of a selected block.
- **Show / Hide shadow**
  Show / Hide the shadow of the selected block.
- **Align blocks**
  By selecting several blocks, it is possible to align them on the horizontal axis (left, right and center) or on the vertical axis (top, bottom and middle).
- **Border color**
  Change the border color of the selected blocks.
- **Fill color**
  Change the fill color of the selected blocks.
- **Link style**
  Modify a link style
- **Diagram background**
  Change the background color of the diagram.
- **Grid**
  Enable / Disable the grid. Thanks to the grid, blocks and links positioning is easier.

**Tools Menu**

- **Code generation**
  Allow the generation of the simulation code of the selected superblock.

**? Menu**

- **Xcos Help**
  Open the help on Xcos functioning, palettes, blocks and examples.
- **Block Help**
  Open the help on a selected block
- **Xcos Demos**
  Open examples of diagrams and simulate them. The user can, if he wants to, modify those diagrams and save them for a future use. (Be careful, the execution of some demonstration diagrams needs to have a C compiler installed on your machine. Please refer to the page 15).

## Available palettes

- **Commonly Used Blocks**
  More used blocks.
- **Continuous time systems**
  Continuous blocks (integration, derivative, PID).
- **Discontinuities**

Blocks whose outputs are discontinuous functions of their inputs (hysteresis).

- **Discrete time systems**
  Blocks for modeling in discrete time (derivative, sampled, blocked).
- **Lookup Tables**
  Blocks computing output approximations from the inputs.
- **Event handling**
  Blocs to manage events in the diagram (clock, multiplication, frequency division).
- **Mathematical Operations**
  Blocks for modeling general mathematical functions (cosine, sine, division, multiplication).
- **Matrix**
  Blocks for simple and complex matrix operations.
- **Electrical**
  Blocks representing basic electrical components (voltage source, resistor, diode, capacitor).
- **Integer**
  Blocks allowing the manipulation of integers (logical operators, logic gates).
- **Port & Subsystem**
  Blocks to create subsystems.
- **Zero crossing detection**
  Blocks used to detect zero crossings during simulation. These blocks use the solvers capabilities (ODE or DAE) to perform this operation.
- **Signal Routing**
  Blocks for signal routing, multiplexing, sample/blocked.
- **Signal Processing**
  Blocks for signal processing applications.
- **Implicit**
  Blocks for implicit systems modeling.
- **Annotations**
  Blocks used for annotations.
- **Sinks**
  Output blocks used for graphical display (scope) and for data export (file or Scilab).
- **Sources**
  Blocks of data sources (pulse, ramp, sine wave) and for reading data from Scilab files or variables.
- **Thermo-Hydraulics**
  Blocks representing basic thermo-hydraulic components (pressure source, pipes, valves).
- **Demonstrations blocks**
  Blocks used in demonstration diagrams.
- **User-Defined Functions**
  User-blocks to model a behavior (C, Scilab or Modelica simulation function).

## Install a C compiler

For some systems simulation (acausal systems containing for example hydraulic or electrical blocks), it is necessary to have a C compiler installed in the machine.

### Under Windows

Install MinGW module from Scilab, Applications Menu / Module manager – ATOMS / Windows Tools category. MinGW module will make the link between Scilab and GCC compiler (which you

also have to install separately). Follow the procedure detailed in the module install window which will guide you step by step in the install of MinGW and GCC compiler.

### Under Linux

GCC compiler is available by default under Linux OS. Just check that the compiler is installed and up to date (via Synaptic, Yum or any other package management system).

### Under Mac

Download Xcode via the App Store (Mac OS ≥ 10.7) or via the CD supplied with the computer (Mac OS 10.5 and 10.6). For earlier versions, see the Apple website. Confirm the possibility to use the compiler out of Xcode environment. To do so, after launching Xcode, go to "Settings", then "Downloads" and in the "Components" tab, select the "Check for and install updates automatically" box and install the "Command Line Tools" extension.

Naturally, if you already have an installed C compiler in your machine, you do not have to install another one. To check that Scilab has detected a compiler, use the command that returns %T if a compiler is installed:

```
--> haveacompiler()
```