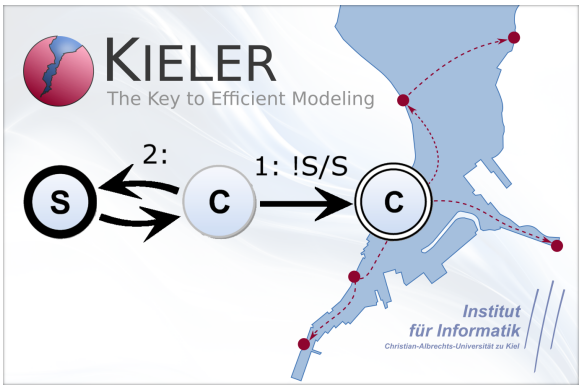
Quick Start Guide

This quick start guide does not include the KEITH product. For a KEITH quick start guide look here.

Starting to model SCCharts with the KIELER tool usually includes the following steps:



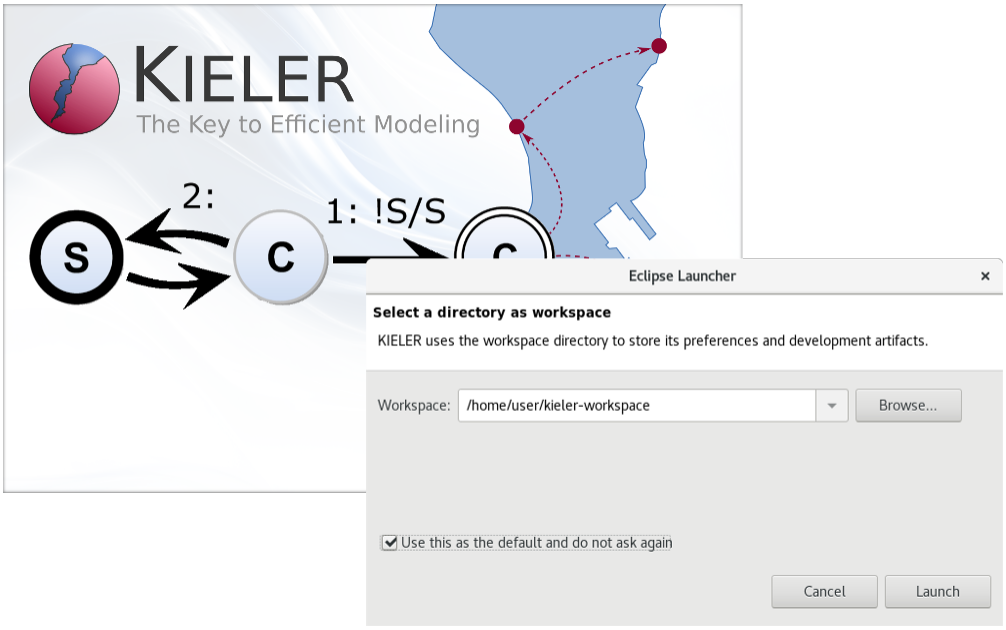
1 Installing KIELER
2 Modeling SCCharts
3 Compiling SCCharts
4 Simulating SCCharts

> **Hint**
>
> You can click on images to enlarge them.

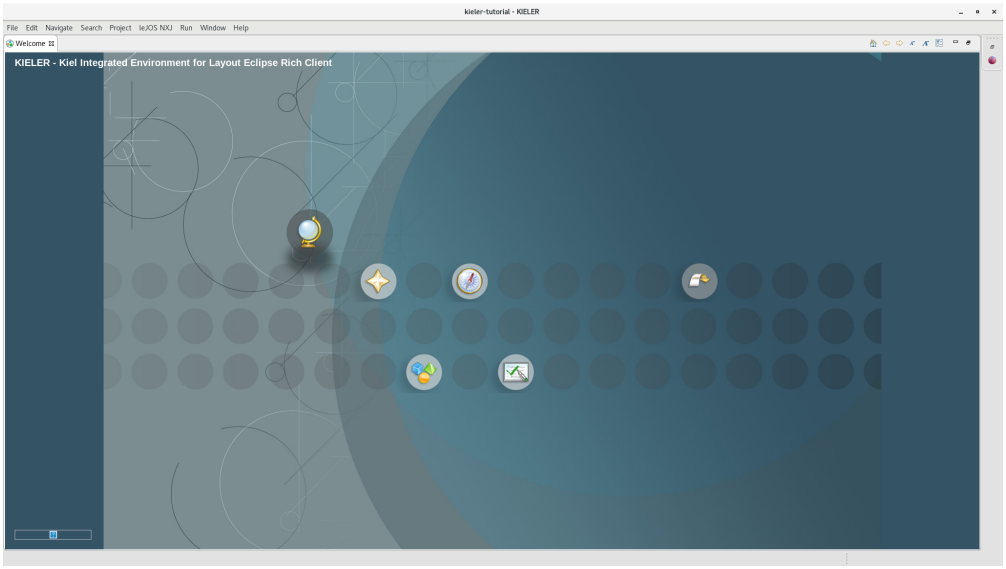
Installing KIELER

Download a suitable KIELER version for your operation system.
The zip bundles are fully compiled and ready-to-run, they do not need any further installation process. Just save and extract them to/at the location of your choice and start the KIELER executable.

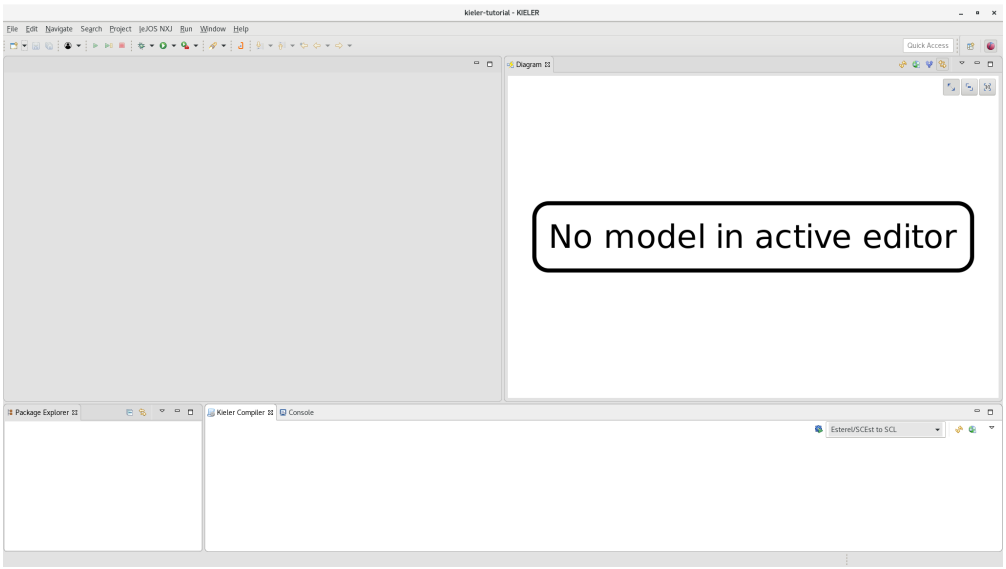
When you start the KIELER executable, you probably need to set a (new) workspace location of your choice.



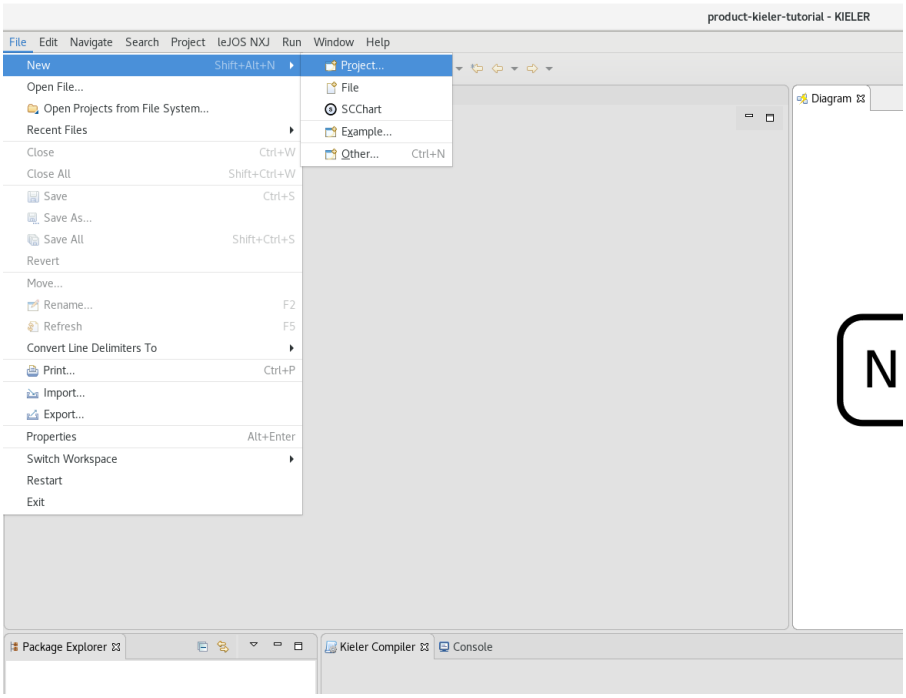
If you selected a new workspace location, you will see a welcome screen like this:



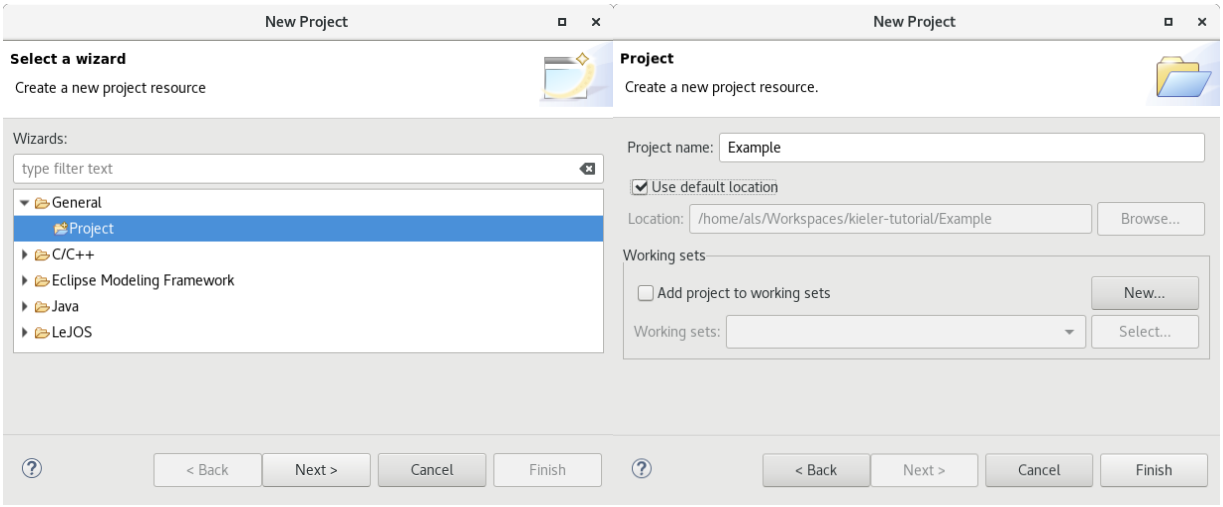
Dismiss this "Welcome" tab. After that you'll be all set and KIELER should look like this:



In order to prepare modeling, we must create a project. Right-click on the *File* entry of the top menu bar, select *New* and click on *Project*.



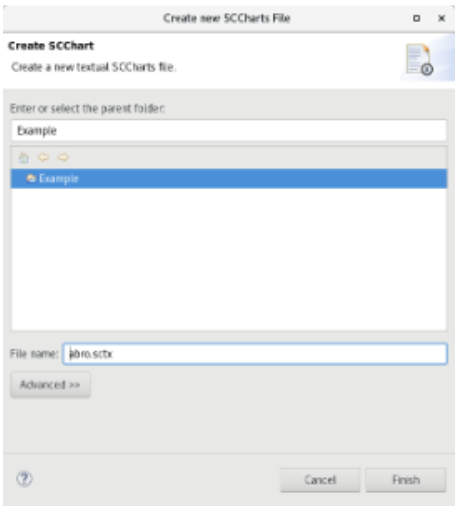
Select a *Project* in the *General* folder and click on next, then enter a project name, here *Example*, and click on *Finish*.



Now you have a project where you can create and model your SCCharts.

Modeling SCCharts

To create a new SCChart, you first need to create a new file. Click on *File New SCChart* in the top menu, similar to creating a new project. Select your project (where you want to create the new file) and set a file name. The creation dialog will automatically append the .sctx file extension. Note that since version 0.13 SCCharts are written in a revised syntax with a new file extension.



A new editor tab will automatically open and you can start modeling your SCCharts.
For this example you may copy paste the following code for the ABRO SCChart into your editor. You can find further examples and an overview about the SCCharts syntax in the sibling pages of this wiki.

```
scchart ABRO {
  input bool A, B, R
  output bool O

  initial state ABO {
    entry do O = false

    initial state WaitAB {
      region {
        initial state wA
          if A go to dA

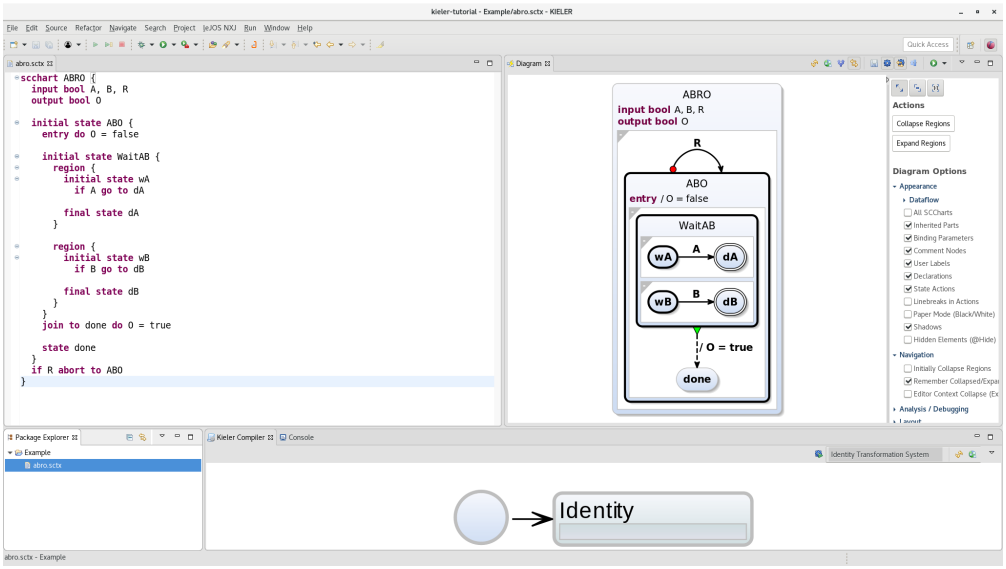
        final state dA
      }

      region {
        initial state wB
          if B go to dB

        final state dB
      }
    }
    do O = true join to done

    state done
  }
  if R abort to ABO
}
```

When you press the save button (or ctrl+S / cmd+S) the model is saved and the diagram view on the right automatically updates and presents the graphical representation of your ABRO SCChart.



The editor provides content assist (Ctrl+Space) to help you find the correct keywords or references to state or variables.

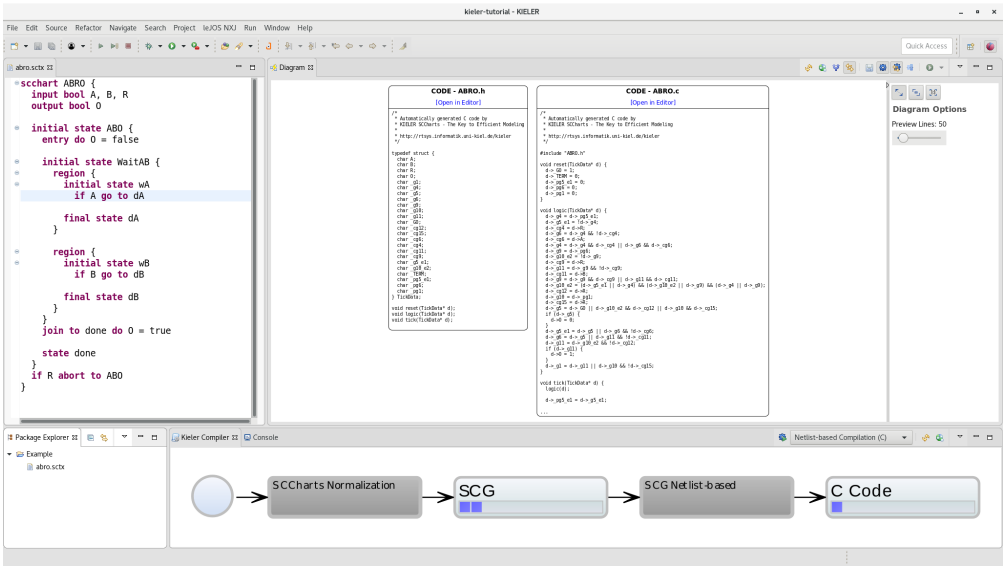
Compiling SCCharts

The KIELER Compiler (KiCo) provides different compilation systems to translate SCCharts into source code for different host languages.
KiCo is controlled via the Kieler Compiler view.

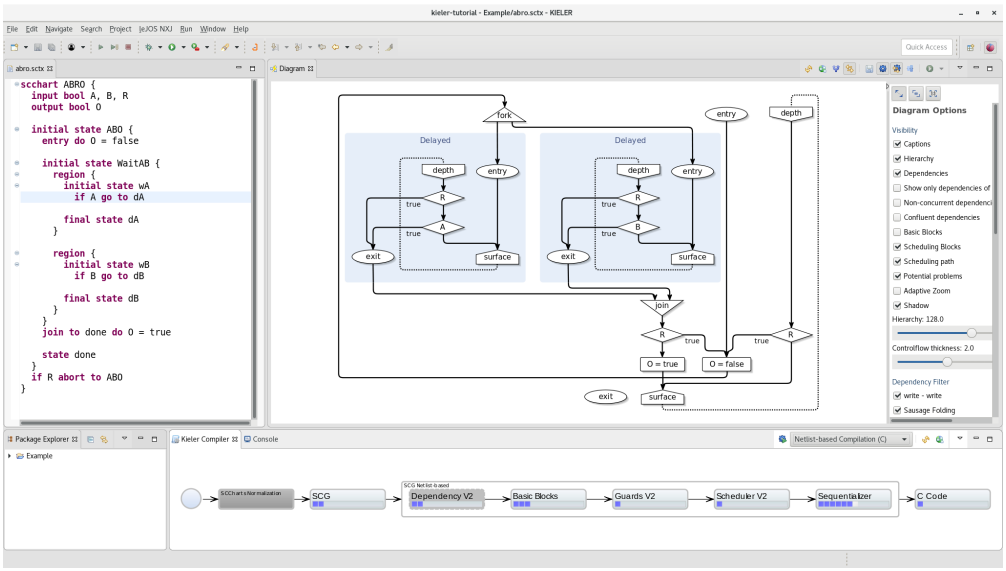


The combo box in the toolbar of this view allows to select a compilation system for the model in the active editor. In the screenshot, the *Netlist-based Compilation* into C code is about to be selected.
The view itself show a diagram of all the processors in the currently selected compilation system. Gray nodes are groups of processors which can be expanded by a double click.

The button before the combo box (highlighted by the red arrow in the screenshot) **starts the compilation** using the model in the active editor and the selected compilation system.
The KiCo view updates during compilation and shows all the available intermediate results per processor as blue boxes. When the compilation is finished the diagram view updates and shows the end result of the compilation (in this example the C code).



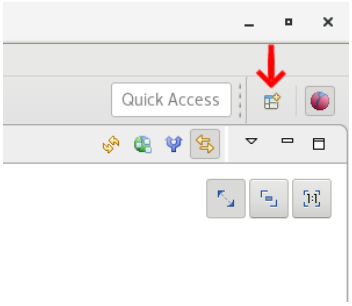
The intermediate results can be inspected by clicking on the processor itself (to select the last one) or by clicking on a blue box. The diagram view will display the selected intermediate result.
In the screenshot below the *SCG Netlist-based* group is expanded and the fist processor in this group, *Dependencies*, is selected. Consequently, the diagram view shows a Sequentially Constructive Graph (SCG) with dependencies.



Simulating SCCharts

SCCharts can be simulated to inspect their behavior.

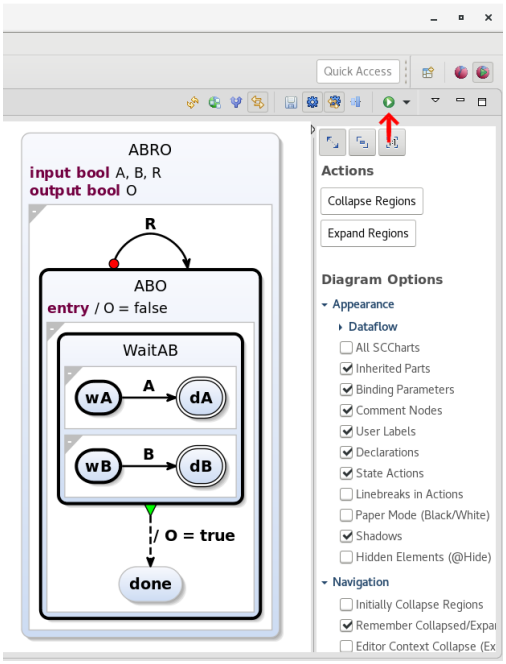
First, we recommend switching to the KIELER Simulation perspective. Click on the *Open Perspective* button in the top right corner.



In the then opening dialog, select the *KIELER Model Simulation* environment and click open. After this step you can switch directly between the two perspectives by clicking on their icon in the top right corner.

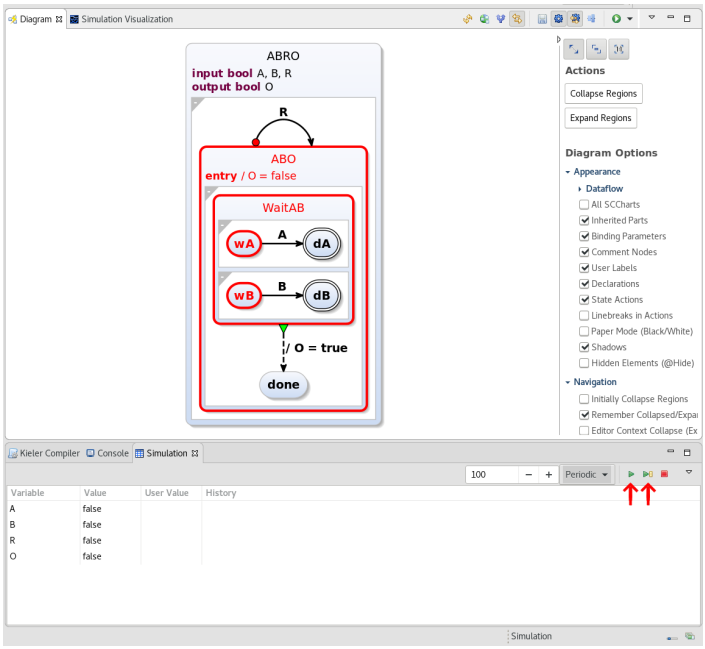


To start a simulation, click on the *Start Simulation* button in the toolbar of the diagram view.



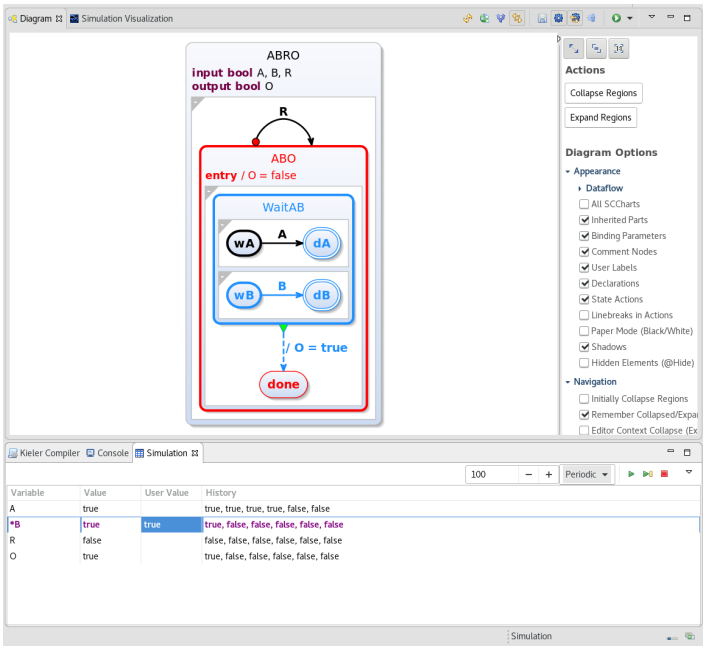
This button will start a new simulation for the model displayed in that diagram view. By default the simulation configuration uses a C code backend for compilation. This requires a GCC available in the system's PATH. If you are using Windows, you need to install some Unix environment (e.g. MinGW or Cygwin). If you want to use a Java backend, you can switch the simulation configuration by clicking on the small black triangle right next to the simulation start button.

When the simulation starts the simulation view will be brought to the top, where you can control the steps of the simulation. On the right site of the simulation view toolbar, there are buttons to stop (rigth) the simulation, perform a single step (middle), and let the simulation continuously play (left). The play mode can be selected by the combobox to the left. By default the periodic mode is select which, when started, performs a step every X milliseconds, which can be adjusted in the field to the left.



The table in the body of the view shows the value of each (IO) variable in the current tick. You set own values (e.g. inputs) by typing the value in the *Us er Value* field. The value will be send to the model in the next tick/step.

The SCCharts simulation is also visualized in the model. States that are active in the current tick are highlighted in red. States that were left and/or traversed are highlighted in blue. Note that this visualization is only an approximation may not represent the actual state of the SCChart.


Diagram
Simulation Visualization
ABRO
input bool A, B, R
output bool O
R
ABO
entry / O = false
WaitAB
wA
A
dA
wB
B
dB
/ O = true
done
Actions
Collapse Regions
Expand Regions
Diagram Options
Appearance
Dataflow
All SCCharts
Inherited Parts
Binding Parameters
Comment Nodes
User Labels
Declarations
State Actions
Linebreaks in Actions
Paper Mode (Black/White)
Shadows
Hidden Elements (@Hide)
Navigation
Initially Collapse Regions
Remember Collapsed/Expar
Editor Context Collapse (Ex
Kieler Compiler
Console
Simulation
100
Periodic
Variable
Value
User Value
History
A
true
true, true, true, true, false, false
*B
true
true
true, false, false, false, false, false
R
false
false, false, false, false, false, false
O
true
true, false, false, false, false, false
Simulation