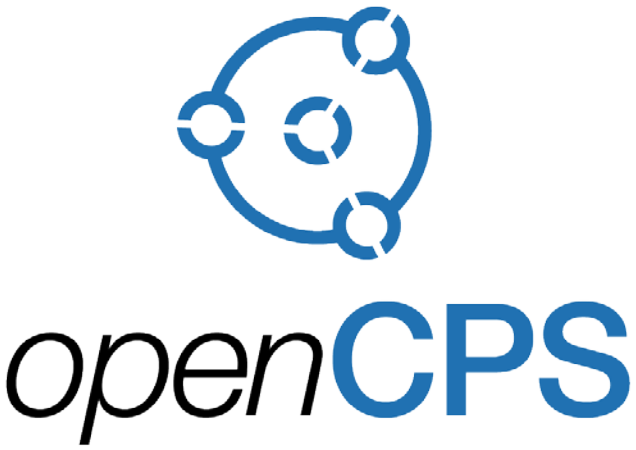
D5.2	Full support for synchronous and state machines features
Access[1]:	**PU**
Type[2]:	**Prototype**
Version:	**0.1**
Due Dates[3]:	**M12**



Open Cyber-Physical System Model-Driven Certified Development

Executive summary[4]:

This deliverable details the OpenModelica support of OpenModelica state-machines and synchronous language features. Full support for these features have been achieved in the OpenModelica compiler, however, we are working on resolving some small remaining issues such as better results verifications for tests and better graphical handling of state-machines.

[1] Access classification as per definitions in PCA; PU = Public, CO = Confidential. Access classification per deliverable stated in FPP.
[2] Deliverable type according to FPP, note that all non-report deliverables must be accompanied by a deliverable report.
[3] Due month(s) according to FPP.
[4] It is mandatory to provide an executive summary for each deliverable.

Deliverable Contributors:

	Name	Organisation	Primary role in project	Main Author(s)[5]
Deliverable Leader[6]	Adrian Pop	SICS East	WP5 Leader	X
Contributing Author(s)[7]	Bernhard Thiele	LIU	WP3 Leader	
Internal Reviewer(s)[8]	Robert Braun	LiU/SKF	T2.4, T5.6 Leader	
	Magnus Eek	Saab	Project Coordinator	

Document History:

Version	Date	Reason for Change	Status[9]
0.1	3/11/2016	First Draft Version	Draft

[5] Indicate Main Author(s) with an "X" in this column.
[6] Deliverable leader according to FPP, role definition in PCA.
[7] Person(s) from contributing partners for the deliverable, expected contributing partners stated in FPP.
[8] Typically person(s) with appropriate expertise to assess deliverable structure and quality.
[9] Status = "Draft", "In Review", "Released".

CONTENTS

ABBREVIATIONS

List of abbreviations/acronyms used in document:

Abbreviation	Definition
FMI	Functional Mock-up Interface
FMU	Functional Mock-up Unit
M&S	Modelling and Simulation
N/A	Not Applicable
SotA	State of the Art

1 STATE-MACHINES AND SYNCHRONOUS FEATURES SUPPORT IN OPENMODELICA

Modelica 3.3 introduced dedicated built-in language support for state machines that was inspired by semantics known from Statechart and mode automata formalisms. The specification describes the semantics of these constructs in terms of data-flow equations that allows relating them to the Modelica DAE representation which is the conceptual intermediate format of Modelica code after instance creation. Modelica 3.3 also introduced language support for synchronous features such as clocks and clock partitions.

This deliverable details the support of state-machines and Modelica synchronous features in OpenModelica [1]. The state machine implementation in OpenModelica is based on [2].

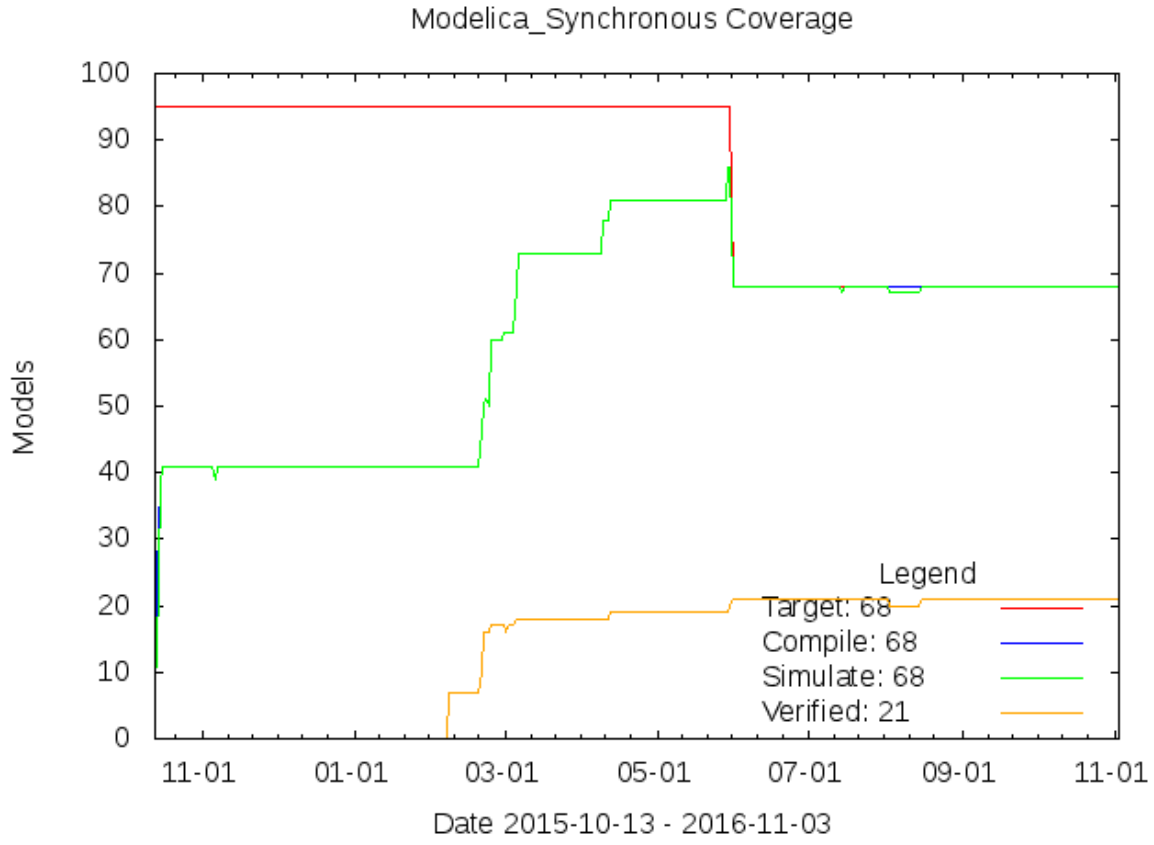


Figure 1. OpenModelica coverage of Modelica_Synchronous library

At the start of the OPENCPS project OpenModelica had partial support for state-machines and synchronous features. During 2016 full support for Modelica synchronous language features has been implemented, see Figure 1. OpenModelica can now compile and simulate all 68 example models from the Modelica Synchronous library. Although the prototype works partly, there are some issues with the high precision handling of clock events that still need to be solved in order to be able to verify the 47 test models that fail verification (the results differ when compared with results produced by other tools such as Dymola) in the Modelica Synchronous library. To ensure there are no regressions the OpenModelica testing framework (https://test.openmodelica.org/libraries/trend.html) runs coverage tests of libraries each night on top of Hudson (https://test.openmodelica.org/hudson/). Hudson (http://hudson-ci.org/) is a continuous integration service that can run jobs to test different aspects of a system, in this

case library coverage. Also, to ensure compiler robustness we run extensive tests (~3000) on each OpenModelica commit.

During 2016, support for state-machines has been improved to achieve full support for Modelica state machines; the OpenModelica compiler can compile and simulate models using these features. Support for graphical editing of state machines in the OpenModelica Connection Editor (OMEdit) is in the works. Users can now use OMEdit to display (see Figure 2), create and modify state-machines and transitions. Full support for transitions between states is work in progress and upon finalization this deliverable will be updated at M24.

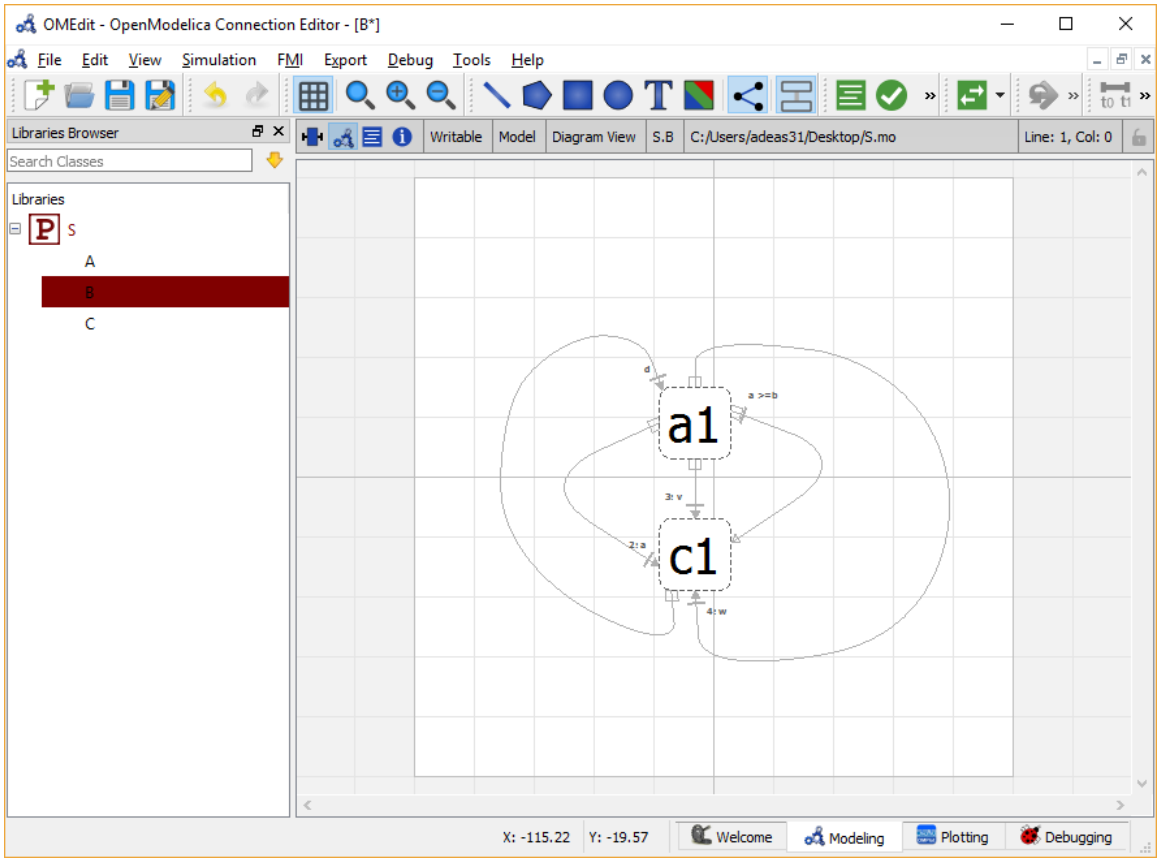


Figure 2. State-machines in OMEdit.

REFERENCES

[1] OSMC, “OpenModelica - an open-source modeling and simulation environment for Modelica,” Open Source Modelica Consortium, www.openmodelica.org, Linköping, 2016.

[2] B. Thiele, A. Pop and P. Fritzson, “Flattening of Modelica State Machines: a practical symbolic representation,” in *Proceedings of the 11th International Modelica Conference: Versailles, France, September 21-23, 2015 / [ed] Peter Fritzson, Hilding Elmqvist, Linköping: Linköping University Electronic Press, 2015, 255-263 p.*.