

EMPHYSIS – D6.9 Prototype of the FMI monitoring and debugging tool integrated into the OpenModelica environment

ITEA 3

Version 1, (Release) February 2021

Adeel Asghar and Lennart Ochel

Linköping University and RISE SICSEast

Project Terms and Definitions

FMI	Functional Mock-up Interface
FMU	Functional Mock-up Unit
TLM	Transmission line modeling
SSP	System Structure and Parameterization
OMEdit	OpenModelica connection editor

Table of Contents

1	INTRODUCTION	4
2	IMPLEMENTATION.....	6
3	AVAILABILITY OF THE PROTOTYPE	8

1 Introduction

The Functional Mock-up Interface (FMI) is a free standard from Modelica Association that allows model exchange and co-simulation of models between different simulation tools. The models are stored in the form of a container called Functional Mock-up Unit (FMU). OMSimulator is a subsystem of the OpenModelica tool suite and provides a simulation engine for FMI-based co-simulation which supports ordinary (i.e., non-delayed) and TLM connections. OMSimulator models are represented in the form of System Structure and Parameterization (SSP) standard. OpenModelica connection editor (OMEdit) is used as a graphical user interface for OMSimulator models.

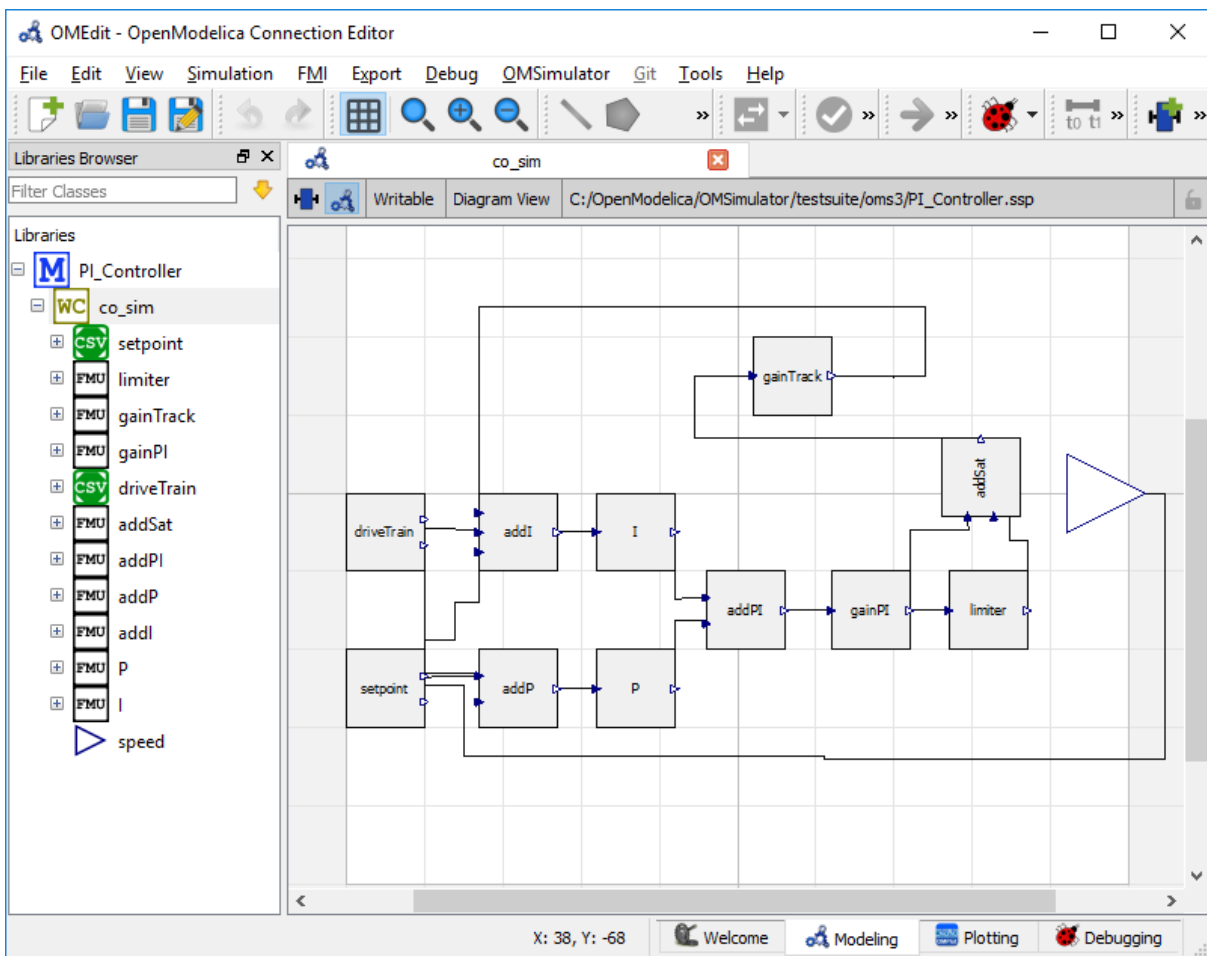


Figure 1: OMSimulator PID Controller model in OMEdit

In the context of T 6.9, a debugger prototype is developed in OMEdit for the debugging of OMSimulator models consisting of FMUs. The capability of debugging consists of enabling the user to suspend the execution of running simulation, inspect/visualize variables values while simulating, modify inputs and tunable parameters, let the simulation execution to progress, etc.

This effort requires some architectural changes to OMEdit's integration of OMSimulator and it requires the development of extensions in both OMEdit and OMSimulator.

The architectural changes are mostly related to the intend of running problematic model, i.e. debugging, that might crash the process they are running in. Therefore, the graphical user interface and the actual simulation must be separated into different processes.

The extensions are related to the new functionality in terms of visualizing results during the simulation and interacting with the simulation. In the future it will also comprises conditional debug points etc.

2 Implementation

The debugger is implemented as a visualization in the form of real time interactive simulation. The debugger uses the ZeroMQ messaging library for the exchange of data between OMEdit and OMSimulator. A typical run scenario is shown below.

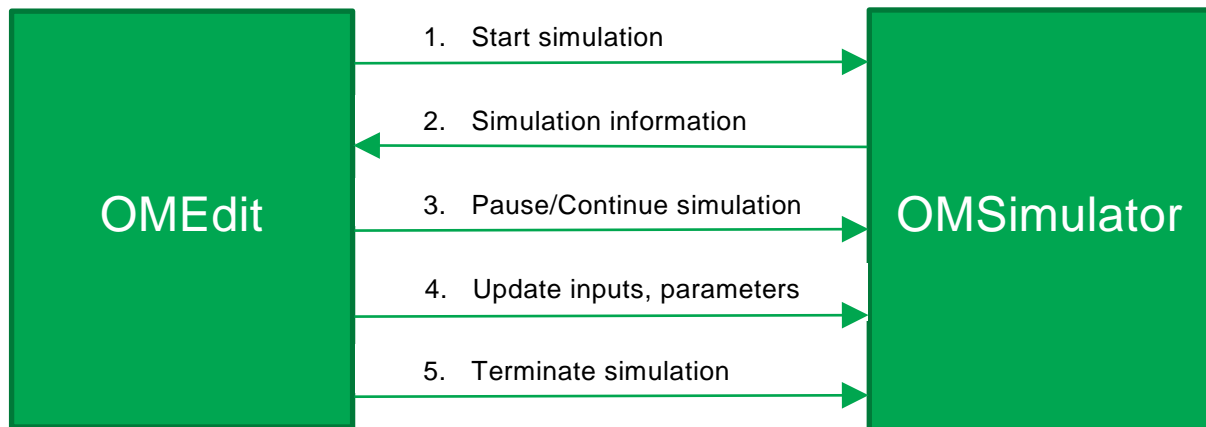


Figure 2: Data exchange between OMEdit and OMSimulator

1. The user starts the simulation of OMSimulator model containing the connected FMUs. OMEdit hosts the TCP/IP server and starts listening on a random port number. Then it will start the OMSimulator process with that port number.
2. OMSimulator connects to the TCP/IP server with the port number using the ZeroMQ and starts sending the simulation information to OMEdit. The simulation information is sent back in the form of json containing the simulation progress, variables data etc. This information is sent continuously during the whole simulation process.
3. The user can pause/continue the simulation from graphical user interface.
4. Once the simulation is paused it is possible to update the inputs and tunable parameters and continue the simulation.
5. The user can terminate the simulation at any time.

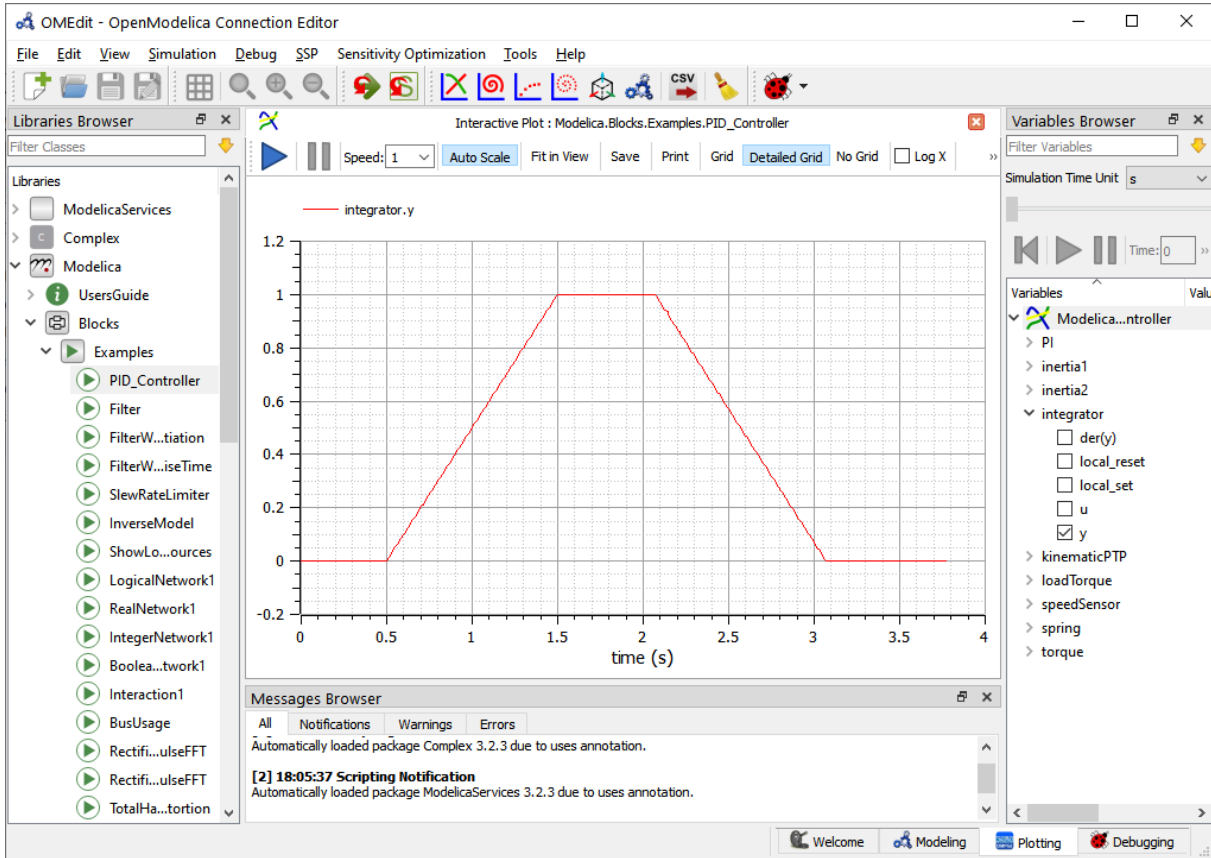


Figure 3: Interactive simulation in OMEdit

3 Availability of the prototype

The prototype of the FMI debugger is scheduled to be part of the upcoming 1.17 release.

All releases can be downloaded through <https://openmodelica.org>.