Developing a SMP2 compliant Hardware-In-the-Loop simulation framework

A. Trung, M. Gehre, P. Froehner, D. Della Ratta, N. Lambl, D. Lammers, H. Lindberg, M. Nizou, G. Robbers, I. Vukman, A. Weihusen

OHB System AG
Universitätsallee 27-29
28359 Bremen, Germany
Email: anh.trung@ohb.de, m.gehre@ohb.de, pamela.froehner@ohb.de

INTRODUCTION

With simulation used at various phases of a project to support a range of engineering, development and operational activities, it is beneficial to maximise the reuse of simulation software items among different simulator facilities to reduce engineering, development and validation effort. OHB has adopted the SMP2 (Simulation Modelling Portability 2) Standard [1] to facilitate model reuse among the different simulator facilities of a project, as well as from project to project. The use of SMP2 at OHB was first established with the development of the MTG SVF (Software Validation Facility) which used BASILES as its SMP2 simulation environment. This experience was then built upon to develop Rufos (RUntime FOr Simulation), OHB's own SMP2 compliant simulation infrastructure with a light-weight MMI (Man Machine Interface). At SESP 2015 the development of Rufos was presented and proposed as the baseline for all new simulator facilities at OHB [2]. This has now been extended to a common 'Software Base Simulator' approach, presented at SESP 2017 [3].

The SARah project is the first project that uses the new software base simulator approach for the following three simulator facilities:

- SVF, a simulation of the whole satellite which is used to develop and validate on-board software, illustrated by Fig. 1
- Assembly Integration & Verification Simulator (AIVS), used to simulate missing physical satellite equipment for EGSE, illustrated by Fig. 2
- Training, Operations and Maintenance Simulator (TOMS), used to validate Flight Control Procedures, train the flight control team and support operations

The SARah SVF and TOMS simulators will be delivered as desktop applications (i.e. they are software-only products). Unlike the SVF and TOMS, the AIVS must be a hardware-in-the-loop (HIL) simulator. It provides hardware interfaces, such as analogue inputs and MIL-STD-1553B busses, to the simulated equipment.

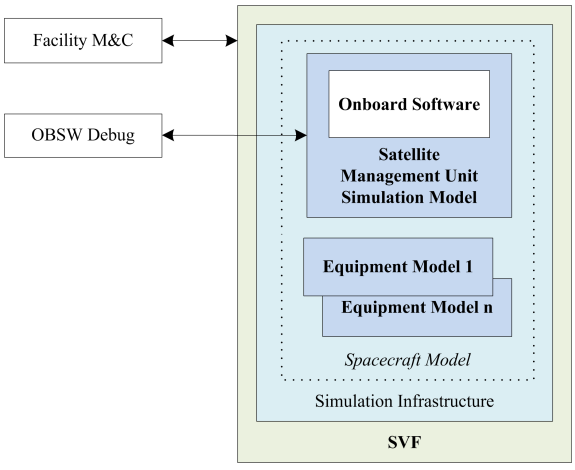

Fig. 1. SARah SVF

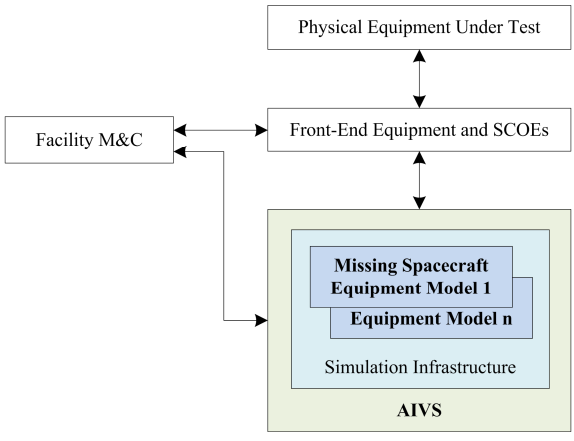

Fig. 2. SARah AIVS

The initial development of Rufos focused on its use for simulators delivered as desktop applications, without any specific support for use in HIL simulators. The only HIL simulation platform known to support SMP2 at that time was EuroSim.

For SARah, the HIL simulation platform chosen by the project for the AIVS was the SCALEXIO platform. Although SCALEXIO does not support SMP2 natively, OHB determined through analysis that developing SMP2 support was possible.

OHB has developed its own SMP2 compliant HIL simulation framework by adapting Rufos to the SCALEXIO platform and developing SMP2 compliant software-hardware interfaces. The development of this framework allows SMP2 equipment models (e.g. star tracker, gyroscope) to be reused between the SVF, TOMS and AIVS simulators of a project.

HARDWARE PLATFORM

The chosen hardware platform of the SARah AIVS is the SCALEXIO system supplied by dSpace. The SCALEXIO is a highly flexible HIL simulation platform that can be customised and extended for the desired hardware interfaces.

The SARah AIVS SCALEXIO is composed of a processing unit, I/O boards and electronics. The processing unit is the computing core. It is based on an industrial PC, with an Intel XEON processor and has a real-time operating system, QNX. The I/O boards and electronics were configured and built as per hardware interface requirements of the SARah AIVS.

The hardware interface types provided are:

- High voltage high power command (HV-HPC) receiver
- Analogue signal monitor (ASM) source
- ASM receiver
- Bi-level discrete monitor (BDM) source
- Bi-level switch monitor (BSM) source
- Variable electrical load
- RS-422 UART
- MIL-STD-1553B

Accompanying the hardware is a suite of software from dSpace to install on an external PC. This software contains the tools to support the development, configuration and operation of the SCALEXIO simulation.

The SCALEXIO processing unit executes a real-time application that is built from a ‘real-time model’. The real-time model is made of two parts: the ‘I/O model’ and the ‘behaviour model’. The interface between the two parts is called the ‘model interface’ and can be seen in Fig. 3.

The I/O model is implemented in dSpace ConfigurationDesk software, and defines the functions for measuring and generating I/O signals with access to the real-time hardware. For example, an analogue input can be created in the I/O model by linking the voltage measured at a specified channel of a specified board to a named variable of type Float64 in the model interface.

The behaviour model contains the algorithm of the controlled system. This means for the SARah AIVS, the behaviour model includes the SMP2 simulation environment including the equipment models. SCALEXIO supports the implementation of the behaviour model as a MATLAB/Simulink model, V-ECU (virtual Engine Control Unit) or FMU (Functional Mock-up Unit). As SMP2 is implemented in C++, and the FMU interface is implemented in C, an FMU is the most suitable method to implement the behaviour model for OHB. The variables defined in the model interface are input or output variables of the behaviour model.

The SCALEXIO software stack ensures that data between the I/O model and behaviour model is interchanged in accordance with the model interface.

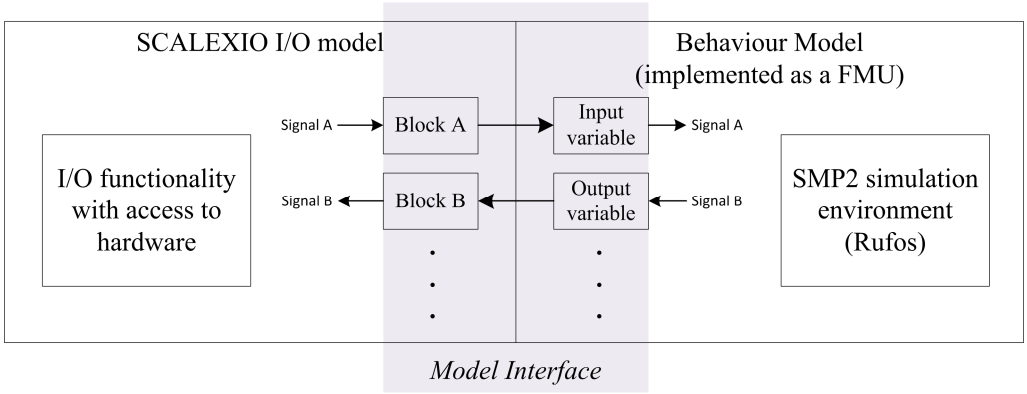

Fig. 3. SCALEXIO real-time model components in SARah AIVS

ADAPTING RUFOS

Rufos, OHB's SMP2 compliant simulation environment, was adapted so that it could run on the SCALEXIO, the SARah AIVS hardware platform.

Firstly, Rufos and its dependencies were ported to the 32-bit QNX 6.5.0 real-time operating system of the SCALEXIO processing unit. This work included:

- updating and patching the QNX toolchain for C+11 support
- patching and cross-compiling boost and python libraries (Rufos dependencies)
- cross-compiling Rufos

Changes were also made to ensure real-time performance, such as:

- adding a memory pool so that dynamic memory can be allocated with constant execution time
- removing system calls with non-bounded execution time
- removing writing of log files to disk

Secondly, Rufos was encapsulated in a FMU to provide the behaviour model for the SCALEXIO environment. A FMU is a model that implements the Functional Mock-up Interface (FMI) – a free, tool-independent standard that defines an interface for the coupling of simulation tools [4].

In accordance with the FMI standard, the Rufos FMU implements an initialisation function, a step function that is to be executed periodically, and getters and setters for input and output variables, which form the interface to the I/O model. The initialisation function initialises the SMP2 simulation environment and models (without a MMI). In the step function, the simulation time is advanced equal to the elapsed Zulu time (the real clock time based on the computer's clock) since the function was last executed and the simulator events that have been scheduled for the elapsed simulation time are executed.

The SCALEXIO software stack manages the periodic execution of the step function and synchronises the values of the FMU input and output variables with the I/O model between every step. Fig. 3 illustrates the relationship between the Rufos FMU and the I/O model.

SOFTWARE/HARDWARE INTERFACES FOR SMP2 MODELS

SMP2 equipment models were first developed for use in the SARah SVF and then reused in the SARah AIVS without modification or rework. This was possible due to the use of OHB Simulation Interfaces and the development of new SMP2 AIVS line models that allowed signal chains to be formed from the hardware interfaces to the SMP2 equipment models.

OHB Simulation Interfaces are standardised software-based interfaces which define how electrical interfaces, such as analogue signal monitor or bi-level discrete monitor, are to be simulated. These interfaces are similar in scope to those defined by the 'ISIS Training, Operation and Maintenance Interface Specification' [5] and comparable to SystemIF Ports

of Spacecraft Simulation Reference Architecture [6]. Equipment models include a reference to a corresponding OHB Simulation Interface for each electrical interface that is being simulated.

In the SVF and TOMS, line models connect one equipment model to another, to simulate an electrical connection in the satellite harness. These line models are SMP2 models that implement an OHB Simulation Interface on each end. Thus, a line model can be connected at each end to a compatible equipment model.

For the AIVS, AIVS line models were developed to facilitate the software/hardware interface. An AIVS line model implements an OHB Simulation Interface at one end, so that it can connect to an equipment model. The other end of the AIVS line model is connected to the I/O model (via FMU input and output variables), joining the signal chain to the physical equipment.

In the case of the AIVS M1553 line model, the M1553 line model is connected directly to the SCALEXIO M1553 hardware card instead of the I/O model. The AIVS M1553 line model uses the C++ API of the M1553 hardware driver directly and implements hardware interrupt handlers to provide the higher responsiveness required for MIL-STD-1553B communications.

Equipment models can be reused between the SVF and AIVS because the OHB Simulation Interfaces hide the implementation of the electrical interfaces from the equipment model. The equipment models depend only on the C++ standard libraries, SMP2 interfaces (provided by Rufos) and OHB Simulation Interfaces and are designed with SVF/AIVS interoperability from the start.

When an equipment model is added to the AIVS, the only additional work necessary is to configure the signal chain for each of its electrical interfaces. Each simulated electrical interface (via the OHB Simulation Interface) is linked to an instance of an AIVS line model, which is also linked (via FMU input and output variables in the model interface) to the I/O model. The I/O model defines a link to the SCALEXIO hardware. An example of this is illustrated in Fig. 5.

This results in output signals from an equipment model that control SCALEXIO hardware outputs, and inputs from the SCALEXIO hardware that are passed as inputs signals to the equipment model. This can be compared with the same model in the SVF, illustrated by Fig. 4.

dSpace ControlDesk can be used to test the signal chains, as it provides a method to monitor and inject values at the SCALEXIO model interface. Thus, the signal chain from equipment model to the SCALEXIO model interface can be tested without stimulating or measuring the hardware interface, and the signal chain from the SCALEXIO model interface to the hardware interfaces can be tested without using the behaviour model (i.e. FMU, SMP2 environment and models).

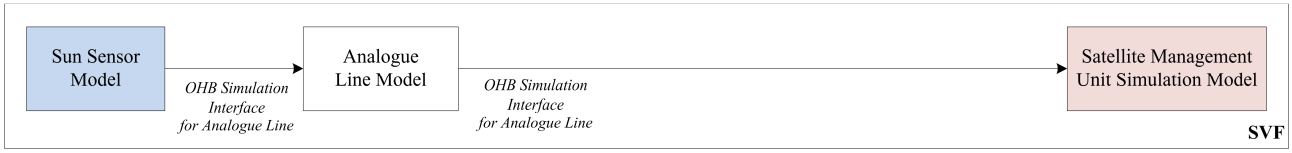

Fig. 4. Example signal chain in the SVF

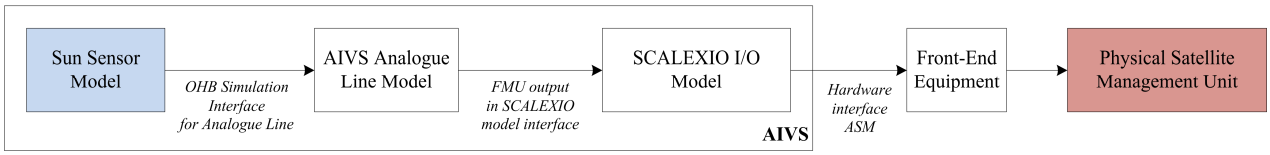

Fig. 5. Example signal chain with the AIVS

RESULTS

The SARah AIVS is currently under development. At present, some AOCS models have been included and more equipment models will be included when they have been developed. Rufos simulation runs in steps at 1000 Hz, meaning that simulation events are executed within 1 ms of their scheduled time.

Support for the following interfaces have been implemented:

- HV-HPC receiver, for pulse widths ≥ 50 ms
- ASM source
- ASM receiver
- BDM source
- BSM source
- Variable electrical load
- MIL-STD-1553B

Further development is needed to:

- implement support for RS-422 UART
- implement SMP2 log output via the EDEN protocol
- move TCP/IP communications to a separate thread (to avoid potential problems with real-time performance)

The AIVS will be used by EGSE for functional testing at the subsystem and satellite level, simulating missing physical equipment. Failures can be injected into the simulated equipment, allowing, for example, the testing of automatic switch-over by SCSW. Operation and monitoring of the AIVS can be performed in a similar way to other SCOE (Special Check-Out Equipment), through PUS packets carried over the EDEN protocol.

CONCLUSIONS

OHB has developed a SMP2 simulation framework for HIL simulators. The dSpace SCALEXIO system is used as the hardware platform and Rufos, OHB's SMP2 simulation environment, runs within the SCALEXIO software stack. AIVS line models allow hardware interfaces to be used by SMP2 equipment models.

Equipment models can be reused between software-only simulators (SVF, TOMS) and HIL simulators (AIVS) without modification, reducing the engineering, development and validation effort needed to develop the simulator facilities of a project.

REFERENCES

[1] European Space Agency, *SMP 2.0 Handbook*, Iss. 1 Rev. 2, October 2005.
[2] P. Froehner, A. Gamarra, M. Gehre, A. Weihusen, F. Hoffmann, D. Della Ratta, "MTF SVF: An excellent opportunity for assessing the SMP 2.0 Compatibility", *Proceedings of the Workshop on SESP 2015*, March 2015.
[3] A. Weihusen et al., "OHB's Software Base Simulator: Efficient Development of Software-Based Simulators by Re-Use of Generic Components", in press for *Proceedings of the Workshop on SESP 2017*.
[4] MODELISAR consortium, Modelica Association Project "FMI", *Functional Mock-up Interface for Model Exchange and Co-Simulation*, version 2.0, July 2014.
[5] ISIS AIV Project Team, *ISIS Training, Operation and Maintenance (TOMS) Interface Specification*, Issue 4, September 2011.
[6] Steinle T., Eisenmann H., *Spacecraft Simulation Reference Architecture – High level Architecture and Interface Requirements*, Issue 1.3, October 2010.