SESP 2010

Session:	Session: Avionics SCOEs (10)
Type:	Concurrent Session
Date:	Wednesday, September 29, 2010
Time:	09:00 - 11:00
Room:	Newton
Chair:	
Co-chair:	
Remarks:	

Seq	Time	Title	Abs No
1	09:00	Hardware-in-the-loop Multi-satellite Simulator for Proximity Operations	

Gaias, G.; D'Amico, S.; Boge, T.
DLR, GERMANY

This work describes a hardware-in-the-loop multi-satellite simulator for dealing with proximity operation scenarios. Its development was justified by the increase of interests in on-orbit servicing missions and its design philosophy lead to setting up a mission-independent test facility. Thanks to its flexibility, both applications involving two cooperating active satellites, and one active and one passive spacecraft can be investigated. The facility allows displaying the real-time real-scale 6DoF motion of the two satellites.
In its simplest form the simulator consists of a model part, developed in the Matlab/Simulink environment, and a hardware part represented by two industrial robots that constitute the European Proximity Operation Simulator (EPOS). In contrast to a traditional embedded software development approach where the focus lies in software engineering aspects, here the techniques of model-based design are applied. As a result emphasis is given to system engineering methods with a more efficient handling of complex systems. As shown in the paper model-based design automates code generation for the embedded system by eliminating the need to hand-code the guidance, navigation and control (GNC) algorithms. For EPOS, a model-based design method is used for the complete simulation software which comprehends the satellites dynamics and characteristics, the interaction with the space environment and the GNC functionalities. Further models of case-specific sensors and actuators can be included at need. Complex functions are included through either embedded Matlab functions or C/C++ S-functions which are then auto-coded with Real Time Workshop and executed under the real-time operating system VxWorks on the EPOS target processor. Benefits of this approach are expected in the field of robustness, predictability, scalability, efficiency and faster development schedules. Furthermore the whole application software can be simulated with faster-than-real-time speed. This means that a large number of aspects of the system behavior can be investigated before deployment on the hardware platform.
The EPOS Robot Hardware consists of a Kuka Robot KR 100 HA robot mounted on a linear slide and a Kuka Robot KR 240 robot mounted to the laboratory floor. The maximum realizable linear relative displacement is 25 m. Both robots possess 6DoF. The motion control of the hardware is performed by the Facility Monitoring and Control System (FMC) which is connected and synchronized to both Kuka Robot Control units. The FMC provides commands every 4ms. Despite this hardware constraint, the user is given the freedom to choose what time-step and what fixed-step numerical integration method better fits his specific application. Due to its structure, this multi-satellite

simulator can be used to support the development and rapid prototyping of the whole GNC system. The model part could also include some software-in-the-loop, represented by the real flight software that determines the maneuvers the spacecraft will perform. Moreover, the flight software could be embedded in a LEON3-FT board, working under RTEMS, thus realizing a processor-in-the-loop architecture. Further hardware, i.e. relative navigations sensors and actuators, could be included in the facility, thus taking the place of the models used so far. Case specific interfaces would be set up on purpose.
As a demonstration of the advanced simulation capabilities, this paper presents a representative test simulation of two spacecraft: an active one approaching a passive target to conduct on-orbit servicing tasks. The active satellite is asked to realize some proximity translational and rotational maneuvers, to emulate the behavior of a servicing spacecraft inspecting the target one. During the simulation, the desired relative attitude and positions are provided by an operator through a 6D mouse and a graphical interface. The chaser satellite demonstrates the capability of planning and realizing the maneuvers sought.

2 09:30 Flexible Hardware In the Loop Configuration in Spacecraft Test Benches

Neefs, M.; Haye, M.
Dutch Space B.V., NETHERLANDS

In recent space programs new development approaches have been adopted in order to optimise spacecraft integration and verification efficiency. A coherent set of test facilities is developed supporting the different phases in the verification program. The spacecraft simulator is an important common component present in all test configurations. The use of a single simulator throughout the project maximises reuse and simplifies validation of system performances. Characteristics measured in test bench configurations can easily be compared with expected results derived from earlier simulations. However deployment of a single simulator in a range of test configurations is not trivial. A flexible solution can only be achieved by careful design of the simulator and its external interfaces from the very start of the program. The presented paper will describe the design of the Real Time Simulator and Avionics SCOE in the Gaia program. The presentation will focus on the mechanisms used to configure the test system for the different Hardware In the Loop configurations used in the Gaia AVM and PFM verification campaigns.

The real time simulator in a spacecraft test bench must support different combinations of real and simulated spacecraft equipments. In the simplest HIL configuration only the on board computer is real and all other spacecraft units and their environments are simulated. During spacecraft integration real units are gradually integrated on the bench, replacing their simulated counterparts. Initially electrical integration is performed, later the units open and closed loop performances are verified. In closed loop tests the sensors are supplied with appropriate stimuli. Actuator commanding and status is acquired to allow the simulator to compute orbit and attitude propagation. The total number of possible test configurations is quite large especially when considering the unit redundancy used in most space systems. Changing a single spacecraft unit from simulated to real, requires changing various system configuration settings. The simulation model and its electrical interfaces simulated towards the spacecraft (such as MIL1553 response generation) must be disabled. If the unit involved is a sensor, another piece of software computing stimuli may need activation. As a consequence electrical interfaces involved in the sensor stimuli generation must be enabled. In some configurations power supply to the real unit must be activated using the PCDU simulation functions present in the EGSE. Furthermore simulation models depending on the

specific state of the unit must now retrieve data from the real unit interfaces instead of copying the information from the simulated unit. A similar set of configuration parameters must be altered when changing a spacecraft actuator model from simulated to real. To further complicate the matter additional limitations usually apply when setting up the test configuration due to functional or interface constraints in flight hardware or EGSE equipment.

Considering the described complexity it is clear that the simulation and test infrastructure should support the operator in the definition of the HIL configuration used in a specific test session. Leaving the configuration a manual task would lead to large risks on human errors with potentially serious consequences. In Gaia a flexible configuration mechanism has been developed that allows HIL configuration by changing one single parameter per unit. The mechanism defines different model modes for unit simulation models. Dynamic entry point activation and configurable data exchanges are used to implement the required flexibility in the simulator configuration. In the presentation the HIL configuration mechanism and the underlying design considerations will be explained. The described HIL mechanism has been build on top of existing functions available in the EuroSim real time simulator environment. However the concepts behind the mechanism are generally applicable for any real time simulator used in spacecraft test benches.

3 10:00 Real-time SpaceWire and PacketWire Interfaces as Developed for GAIA Verification

Hommes, F.
Dutch Space, NETHERLANDS

Existing check out equipment for SpaceWire and PacketWire bus link interfaces has always been focused on non real-time monitoring and simulation. For this reason these equipments are not easily combined with the Dutch Space hard real time SCOE's used in closed loop test configurations. For SpaceWire monitoring in particular, different approaches like wormhole routing to the network or additional electronics with a non-standard interface introduce differences which compromise the representativeness of the system under test.

The GAIA SCOE provided the opportunity to complete the development of a 'non intrusive' test-means which monitors SpaceWire and PacketWire traffic by looking at the voltage level of the individual signals. The HW involved was based on commercial FPGA IO cards and well proven IP cores, thus ensuring proper interface characteristics. The developed hardware made it possible to monitor and simulate the real time bus traffic on the cross strapped busses that connect EIU (Electrical Interface Unit) and the CDMU (Configuration and Data Management Unit) onboard the GAIA spacecraft. In the presented paper the design of the SpaceWire and PackWire interfaces and their use in the Gaia avionics test bench is discussed in more detail.

The main objectives of the project were:

To provide representative real time simulation of missing SpaceWire and PacketWire equipment towards the CDMU. Simulation and fully transparent monitoring of the on-board bus links, without added latency on time stamps or time tagged message generation. It should be possible to switch between monitoring a link and simulating a bus transceiver without HW reconfiguration. Integrated into the Dutch Space architecture To provide a unified solution for SpaceWire and PacketWire connections Cost effective and flexible for reuse in future projects

HW development for EGSE at Dutch Space is centered on extending the Dutch Space SCOE architecture with new electrical interface types to the S/C while maintaining flexibility and robustness to change through the use of FPGA's. Because the project is inline with these goals, the project could be partially funded by the Netherlands Space Office (NSO). The IP expertise of PacketWire and SpaceWire was provided by Saab and NLR respectively.

The system HW consists of the following parts: Active interface boxes are put either on a avionics test bench or on the spacecraft and provide a pickup for the bus signals and the switch between monitoring the on-board bus or simulating one of the bus transceivers in the SCOE Multiplex boxes which are also located on the test bench or S/C and select one of four redundant busses either under operator / test sequence control or automatically. The latter is achieved with a small CPLD in the box. Connecting cables between the test bench or S/C and the SCOE. Adapter boards in the SCOE which contain line drivers and switch logic to connect the monitored / simulated busses with the FPGA card. Commercial FPGA cards that contain the SpaceWire and PacketWire transceivers.

Alpha-Data PMC cards with Xilinx FPGA's were selected to host the SpaceWire and PacketWire IP cores. The electronics for AI boxes, MUX boxes and VME sized adapter boards were developed, verified, integrated into the GAIA check out system at Dutch Space.

At the moment the system is used on a daily basis on the GAIA avionics test bench and will be used soon when integrating the GAIA S/C in Toulouse. Although refinements to the system are planned, the development of the S/C bus monitor and simulation HW has achieved all its basic goals.

4 10:30 Test of ESP or AOCS - Are there real differences?

Haupt, Hagen; Ploeger, Markus; Spenneberg, Dirk; Bracker, Joerg
dSPACE GmbH, GERMANY

Hardware-in-the-loop (HIL) testing is a well established method in the development cycles of automotive as well as space electronic control systems. Nevertheless the high level of vehicle electronics, the high number of vehicle developers and the huge number of HIL test systems have given the automotive industry an advantage in experience of system setup and function.
This paper describes standard HIL use cases in the automotive field and how they can be transferred into the space systems area. Taking the HIL test of a satellite AOCS system as an example, the paper describes how HW and SW structures as well as test methods can be taken over from the automotive domain .

A well established HW/SW tool chain, which has been successfully deployed more than 1000 times in various projects is introduced.

On the hardware side this includes real time processing and I/O hardware as well as signal conditioning and failure insertion components. In addition the paper focuses on special HW demands to space test systems. Solutions for such requirements like fail safe design, self test capabilities, galvanic isolation or distributed I/O are described.

The Software of the toolchain can be divided into two parts.

The first part is the real-time software which includes realtime satellite and environment models and the related I/O software. The models

allows to simulate the orbital and attitude dynamics and the structural dynamics of the satellite, its environment (magnetic field, radiation pressure amongst others) the AOCS sensors (e.g. Sun Sensor, Star Tracker, Gyro) and the AOCS actuators (e.g. Reaction Wheel, Magnetotorquer, Thruster). The paper presents a Simulink® based real-time SW structure which is designed for easy use and well established in hundreds of automotive test systems. For space applications it includes ready to use satellite and environment models for LEO, MEO and GEO systems. Also for the models design and parameterization handling standards are presented.

The second part of the software are the test and experiment tools to define and control the test system behavior. The paper presents an experiment software which can be used easily to set up arbitrary Graphical user interfaces (GUIs) for manual operation of the Hardware-in-the-loop test system. It includes plotting functionality with synchronous data acquisition and complex experiment handling. For automated testing a powerful tool is presented which can be used both for self test procedures of the HIL system or for automated testing of the AOCS controller. Especially for testing the AOCS it can be benifical to have a 3D Real time animation which shows the position and attitude of the simulated satellite. An appropriate tool which allows this will be presented. A particular challenge is the testing of communication. The test system must enable the user to implement Bus monitoring functionality as well as transmit and receive blocks. Using the example of a MIL 1553 Bus it will be shown how an easy to use approach to implement the requested functionality in Simulink® can be used.

All the presented simulation and testing approaches and the corresponding tools have a long and successful heritage in the automotive industry which led to a very high maturity of the whole tool chain. We believe that this maturity makes them very attractive for the space domain. We showed that with special extensions for space applications it is possible to reuse this well-proven and cost-efficient off-the-shelf tool-chain. Therefore, we expect that our tool-chain will improve the test quality and at the same time reduce the time needed for verification and validation of electronic control units significantly. Furthermore, this tool-chain has been designed to be adapted rapidly to different domains, therefore we expect that a user can easily reuse our tool framework for different space projects.