LEVERAGING SERIAL DIGITAL INTERFACES STANDARDIZATION: THE RASTA REFERENCE ARCHITECTURE FACILITY AT ESA

Session: Spacewire onboard equipment and software

Long Paper

Aitor Viana Sanchez, Gianluca Furano, Massimiliano Ciccone, Farid Guettache, Claudio Monteleone, Chris Taylor

ESA - European Space Technology Centre (ESTEC)

ESA/ESTEC P.O. Box 299 / 2200AG Noordwijk ZH, the Netherlands

Manuel Prieto, Ignacio Garcia Tejedor

University of Alcala

Ctra. Madrid-Barcelona, km. 33.6, 28871 Alcala de Henares, Madrid

E-mail: aitor.viana.sanchez@esa.int, gianluca.furano@esa.int, massimiliano.ciccone@esa.int, farid.guettache@esa.int, claudio.monteleone@esa.int, chris.taylor@esa.int, manuel.prieto@aut.uah.es, ngarcia@aut.uah.es

ABSTRACT

This paper presents an overview of the internal R&D project in ESA named Reference Avionics System Test-bed Activity (RASTA). This activity aims to benefit from interface standardization to provide a hardware/software reference infrastructure where all the incoming R&D activities can be integrated, coming out with a single and generic test environment instead of dedicated environments for each activity.

Within RASTA, all the present and future digital serial interfaces used in space are included, like CAN bus, MIL-STD-1553 and Spacewire. RASTA is constituted by building blocks, such as: LEON2 based on-board computer, Telemetry and Telecommand control unit and a Mass Memory Unit, and the communication technology available so far may be selected between cPCI and Spacewire bus.

One of the latest activities is the Modular Advanced Mass Memory Architecture (MAMMA). This unit tends to be an intelligent unit fully based on Spacewire, embedding several services trying to be as autonomous as possible from the on board computer and scalable enough to serve different missions. The block device mapping over Spacewire allows adding another redundancy layer on the mass memory. This architecture will exploit the future capabilities of active Spacewire backplanes.

Some outputs are already available for the space community, such as: Basic SW and SW drivers (CAN/1553/SpW, TT&C), OS abstraction layer, SW libraries (e.g. CFDP). In the future, more SW libraries will be available (e.g. PUS library) and more HW units integrated in the environment (e.g. intelligent RTUs, low speed control bus)