



Contribution ID: 46

Type: Oral

## IPbus: A Flexible Ethernet-based Control System for xTCA Hardware

*Wednesday, 24 September 2014 14:25 (25 minutes)*

The xTCA standards define pathways for control communication, but no specific hardware control protocol is defined. The IPbus suite of software and firmware implements a reliable high-performance control link for particle physics electronics, and has successfully replaced VME control in several large projects. In this talk, we outline the IPbus system architecture, and describe recent developments in the reliability, scalability and performance of IPbus systems, carried out in preparation for deployment of CMS upgrades before the LHC 2015 run. We also discuss lessons learnt in the integration of a complex distributed firmware and software system, and future directions.

### Summary

IPbus will be used for controlling the uTCA electronics in the CMS HCAL, TCDS, Pixel and Level-1 trigger upgrades. IPbus control has already been extensively used in the work of these upgrade projects so far, and final uTCA systems will be deployed in the experiment starting from Autumn 2014. IPbus is also being evaluated for use in the ATLAS and ALICE upgrades, as well as other particle physics experiments.

A tightly-integrated suite of software and firmware components has been developed to implement the IPbus protocol: the firmware core, a reference VHDL implementation of an IPbus server over UDP, decoding IPbus read/write requests within end-user hardware; uHAL, the C++/Python library providing an end-user API for IPbus reads and writes; and the ControlHub, a software application which arbitrates hardware access to each board from multiple clients.

Over the past two years we have developed a new reliable, higher-throughput version of the IPbus protocol, firmware and software. We have set up an IPbus test system with realistic network topology in the CMS electronics integration centre, in order to validate the reliability and performance of the IPbus control system.

The software has been optimised to increase the block write/read throughput towards the Gigabit Ethernet bandwidth, and to improve the scalability with the number of targets handled by each ControlHub instance. For 1 client and 1 target, the latency is about 250us for sequences of up to tens of transactions and the maximum block read/write throughput is 0.54Gbit/s; the throughput increases to 0.8Gbit/s for 3 or more targets. We have accumulated weeks of continuous high-throughput random writes and reads over IPbus, without any errors. We also investigated scenarios with network congestion in the MCH Ethernet switch for a full uTCA crate, and found that with appropriate configuration this congestion only has a small effect on the IPbus throughput (12% reduction).

Plans for future work include improving the monitoring of IPbus dataflows in large systems of hundreds of targets, and investigating further ideas for usability and performance improvements.

**Primary author:** WILLIAMS, Tom (University of Bristol (GB))

**Presenter:** WILLIAMS, Tom (University of Bristol (GB))

**Session Classification:** Systems, Planning, Installation, Commissioning and Running Experience

**Track Classification:** Systems