Contribution ID: **33** Type: **Poster**

Development of an Universal ROC

Thursday, September 23, 2010 4:00 PM (2 hours)

Since 2007 we design and develop a ROC (read-out controller) for FAIR's data-acquisition. While our first implementation solely focused on the nXYTER, today we are also designing and implementing readout logic for the GET4 which is supposed to be part of the ToF detector and the CBM-XYTER which is supposed to be used in various other detectors like the STS or the TRD detectors. Furthermore we fully support both, Ethernet and Optical Transport, as two transparent solutions. The usage of a strict modularization of the Read Out Controller enables us to provide an Universal ROC, where front-end specific logic and transport logic can be combined as flexible as possible.

Summary

Since 2007 we design and develop a ROC (read-out controller) for FAIR's data-acquisition. In the beginning, we were focusing solely on the STS detector - especially on interfacing the nXYTER chip. The according beam tests at GSI in September 2008 and in September 2009 showed that this development was very successful.
Our first ROC setup consisted of the nXYTER read-out logic and a PowerPC based Ethernet transport logic. This setup is called the nXYTER Starter Kit, which turned out to be very helpful in small laboratory setups, where physicists need quick access to the nXYTER and its measured data. The Ethernet interface keeps the communication as simple as possible. It can be interfaced using a commodity PC with no exotic hardware requirements.
However, in the final experiment thousands of ROCs will be used and all those ROCs might generate a huge amount of data. In this setup Ethernet will not be an option. Thus, a second transport logic, the Optical Transport, has been implemented. Today, we fully support both, Ethernet and Optical Transport, as two transparent solutions.
While our first implementation solely focused on the nXYTER, today we are also designing and implementing readout logic for other front-end chips. Those chips are the GET4 which is supposed to be part of the ToF detector and the CBM-XYTER which is supposed to be used in various other detectors like the STS or the TRD detectors.
To keep the re-usability as high as possible, we did split the ROC into two fully independent modules: the readout logic (RL) and the transport logic (TL). Both modules are connected to each other via a well defined interface: a standard bus interface for slow control, a FIFO interface for the hit-data transport, and a special "deterministic latency message" (DLM) interface for synchronization. Therefore the design and implementation of the modules can be done independently - even by several developers. The control software running on the PC benefits from the modularization as well. The put/get-protocol which is used to communicate with the ROC is transparently encapsulated by the drivers for the optical transport or the Ethernet transport respectively.
In conclusion the modularization of the Read Out Controller and the consequential separation of the control software into different layers, enables us to provide an Universal ROC, which offers quick access to a long-run tested transport logic. This leads to a significant increase of stability and reliability compared to a development from the scratch. Beyond that, the modularization concept allows us to easily add a new readout logic for further FEE setups.

Primary author: Mr MANZ, Sebastian (Heidelberg University)

Co-author: Mr ABEL, Norbert (Heidelberg University)

Presenter: Mr MANZ, Sebastian (Heidelberg University)

Session Classification: POSTERS Session

Track Classification: Programmable Logic, design tools and methods