PRELIMINARY DESIGN OF A REAL-TIME HARDWARE ARCHITECTURE FOR eRHIC*

R. Michnoff#, P. Cerniglia, M. Costanzo, R. Hulsart, J. Jamilkowski, W. Pekrul, Z. Sorrell, C. Theisen,
Brookhaven National Laboratory, Upton, NY 11973, USA

Abstract

The 3.8 km circumference Relativistic Heavy Ion Collider (RHIC) at BNL has been in operation since 2000. An electron-ion collider (eRHIC), which is in the design phase, plans to use one of the existing ion rings and new electron rings constructed in the existing tunnel to provide collisions of up to 21.2 GeV electrons with up to 100 GeV gold ions, 250 GeV polarized protons, as well as other species. Many new real-time systems will be required to satisfy the needs of eRHIC, including over 2000 beam position monitors, 1000 beam loss monitors, 18 current monitors, feedback systems, controls for about 10,000 power supplies, machine protection system, new beam timing systems, and more. The selected architecture must be flexible, expandable, cost-effective, reliable, and easy to maintain. Interface with existing and new accelerator timing systems is required, and compatibility with existing infrastructure and equipment must be maintained. Embedded modules based on the Xilinx Zynq gate array, with direct Ethernet connection and on-board Linux, housed in multi-slot chassis (VME, VXS, VPX, TCA, etc.) is under consideration. Preliminary design concepts for the architecture will be presented.

CONTROLS ARCHITECTURE HISTORY

Hardware architectures for accelerator controls have gone through many transformations as technology has evolved. In the 1970's, custom field-busses were designed to control remote devices in accelerators. Engineers at BNL's Alternating Gradient Synchrotron (AGS) designed a system called Datacon – a multidrop field bus capable of withstanding the electrically noisy accelerator environment [1,2]. A central computer typically controlled all of the devices. At AGS, the then state-of-the-art PDP-8 and PDP-10 were used.

As hardware evolved, real-time controls became more distributed. Intel's Multibus architecture became a standard in the 1980s for systems developed at the AGS. A typical Multibus system consisted of a custom enclosure with a backplane card cage housing processor, memory and I/O modules. A locally resident RMX deterministic real-time operating system was used. These units worked very well, but although the processor and some I/O modules were common, a major disadvantage was that each system required custom cable assemblies and connector panels to external devices. This made it very difficult to maintain spares because each system required a dedicated spare chassis.

*Work supported by Brookhaven Science Associates, LLC under Contract No. DE-SC0012704 with the U.S. Department of Energy.
#michnoff@bnl.gov

Then in the 1990's, VME-bus became a popular platform at AGS and for the RHIC project that was under construction at that time [3]. System maintenance greatly improved with the use of VME because modules could easily be replaced when failures occurred. Swapping VME boards is very simple compared to fully encased systems like Multibus and custom designed enclosures.

More recently, manufacturers and accelerator hardware engineers are commonly incorporating direct Ethernet connectivity into systems. This has provided tremendous advantages for system designers. However, custom chassis have again become popular, which can increase the complexity of maintenance and spare inventory.

THE eRHIC APPROACH

The present plan for eRHIC real-time hardware systems is to use the Xilinx Zynq family system-on-a-chip (SoC) devices (or similar newer components that become available), which combine large programmable gate arrays with embedded ARM processors all on the same integrated circuit. A suite of modules will be developed and housed in multi-slot backplane chassis such as VME, VXS, openVPX, or TCA, with each module containing a direct Ethernet connection for communication to the higher-level control system. An operating system such as Linux will reside locally on each module.

Gbit serial links will be used for communication between modules via either front panel connections or via Gbit serial lanes on the backplane. This high-speed communication across multiple modules is essential for providing the real-time operation required for many accelerator systems. Details for the communication links are not yet fully defined, but standard protocols will be used where feasible. A draft design is provided in fig. 1.

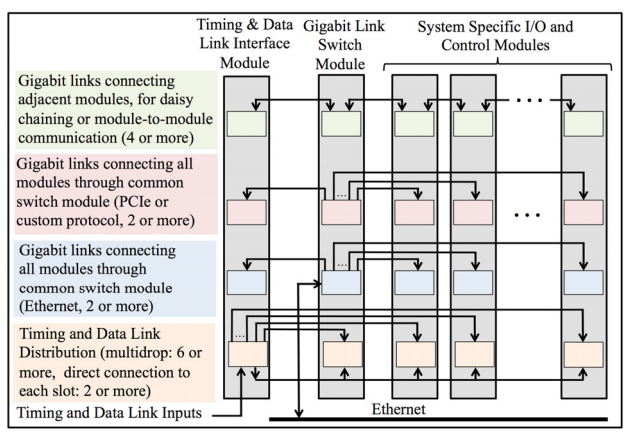

Figure 1: Draft design of inter-module communication. Communication via backplane is most desirable; some paths may be via front or rear panel connections.

ISBN 978-3-95450-148-9

IS A DETERMINISTIC REAL-TIME OPERATING SYSTEM ESSENTIAL?

Deterministic real-time operating systems have been vital for many years in order to satisfy time-critical system functions. Processor modules in VME-bus systems currently installed in the BNL Collider-Accelerator complex use the real-time operating system vxWorks [4]. Task priorities are carefully planned to ensure that time critical functions are properly handled and interaction between hardware modules within the chassis is well orchestrated.

With newly available components like the Xilinx Zynq SoC, the majority of the real-time aspects are now performed within the gate array code, and the on-chip processor running Linux or similar OS acts like a window into the hardware that simply passes data to and from higher level workstations. Although Linux task priorities (or nice values as they are called) must still be carefully planned, true real-time functionality at the Linux level is not as important anymore. And with a low cost SoC installed on every hardware module, systems can be more distributed than ever; so each processor performs somewhat dedicated tasks. This is a major advancement!

THE PROTOTYPE

A first prototype using the Zynq XC7Z030 SoC with on-chip dual ARM Cortex A-9 CPUs has been developed (V301) for installation in a VME chassis (Fig. 2). The VME backplane use is limited to providing 5V power to the module and receiving machine timing and data link signals bussed over user-defined P2 pins from an in-house designed module (V208, Fig. 3). A VME interface is not provided; instead a direct front panel Ethernet connection is used for Control System communication. One of the on-chip CPUs is running Linux, and the second CPU, although not currently used is available if required.

The V301 is a fully contained beam position monitor system with flexibility for measuring several beam species including protons, ions, and electrons. This module is not only intended as a prototype for eRHIC, but will also be used as a next generation BPM system for the existing and other future machines within the BNL Collider-Accelerator (C-A) complex.

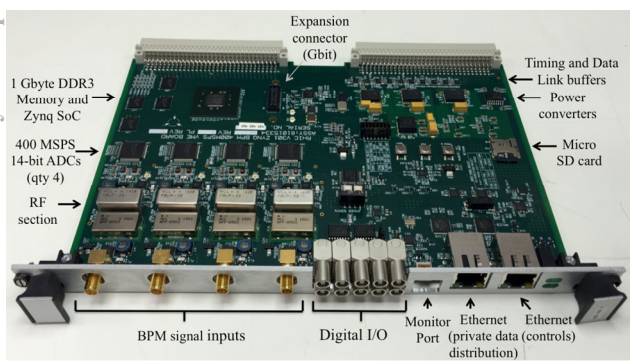

Figure 2: Photograph of V301 module.

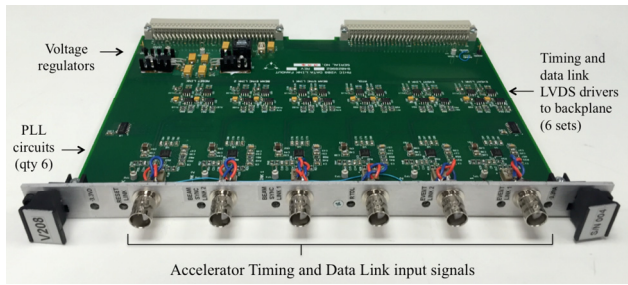

Figure 3: Photograph of V208 Timing and Data Link interface module.

Additional modules under consideration to be developed in the near future include:

- A multi-channel analog input module.
- A simple carrier module with connector to gate array pins to allow custom I/O boards to be easily developed into the Zynq architecture.

These new modules will include the digital section that has already been designed. The goal is to use a common footprint for the Zynq SoC, memory, and power converters so that new boards can rapidly be developed.

MAJOR DESIGN REQUIREMENTS

Some of the major design requirements for eRHIC real-time hardware systems include:

- Minimization of equipment rack space
- Easy replacement of operational modules
- High reliability
- Simple software configuration of modules
- Simplified interface to machine timing and data links
- High speed processing of I/O signals
- Two or more Ethernet connections on each module
- Embedded operating system on each module with tightly coupled interface to real-time hardware, and with control system software objects resident locally
- Gigabit communication between modules
- Ability to use commercially available modules
- Ability to share custom hardware modules between other facilities

Each of the above items is addressed below, including explanations of how the requirement will be satisfied.

The requirements in this list are not unique for eRHIC, but can be considered common for all modern accelerator control systems.

Minimization of Rack Space

Since the cost for buildings is high, minimizing the floor space required to house electronics seems obvious. While this requirement is not always at the forefront of our minds when engineering systems, higher and higher density electronic components with smaller footprints have allowed this requirement to be satisfied without tremendous thought.

Providing increased capability into each hardware module is part of satisfying this requirement, but system packaging details must also be carefully considered.

ISBN 978-3-95450-148-9

With more and more embedded systems becoming available with direct Ethernet connections, many manufacturers and accelerator electronic design engineers have developed custom fully integrated enclosures for packaging of systems.

While this architecture works well when only a few units are needed, for systems like eRHIC that will require over 2000 BPMs, rack space requirements would be huge for so many individual units.

Therefore, the hardware architecture design for eRHIC will be based on a multi-slot chassis system like VME, VXS, openVPX, or TCA with many slots in order to minimize rack space requirements.

One might argue that using a multi-slot chassis can be very expensive if only a few slots are used. For these applications, chassis with fewer slots are an option.

The multi-slot chassis architecture will also satisfy other design requirements as will become evident in additional sections below.

Easy Replacement of Operational Modules

Minimizing the time required to replace failed hardware modules during operations is quite important in order to limit machine downtime.

For the present RHIC BPM system, which has been in operation since 2000, two hardware boards (one for each BPM plane) are installed in a custom 3U rack-mount chassis. Maintenance is quite difficult because replacing a failed hardware module requires removing the chassis from the equipment rack, removing many screws to access the interior of the enclosure, and then removing more screws and connections to access the board.

A multi-slot chassis system with modules like the V301 will provide easy replacement by simply disconnecting external cables and removing two front panel screws.

High Reliability

As with all accelerator systems, equipment must be robust and highly reliable. Radiation induced failures must be considered in the design.

Simple Software Configuration

For most front end computers and embedded systems with direct Ethernet connections, configuring the network settings usually requires a cumbersome method like local logon via a serial port or local monitor and keyboard to configure parameters in either flash or disk files.

With possibly thousands of modules at eRHIC having direct Ethernet connections, simplifying the configuration process is essential.

The prototype V301 module provides the required simple configuration. The network IP settings and host name are stored in files on a removable micro SD card, which also contains a boot file. When the module boots up, the hosts file and /etc/network/interfaces network configuration file are copied from the SD card to the local ram disk. Then a remote server is mounted, and the host name is used to point to a directory on the server that defines the software modules to be automatically loaded, including the gate array file. The technique of loading software modules from the server has been in use at RHIC for many years. The major difference here is the use of a removable micro SD card.

The beauty of using the micro SD card is that it can be set up externally, allowing a desktop computer to be used to easily pre-configure the micro SD cards. In addition, if an operational hardware module fails and requires replacement, the micro SD card can simply be removed from the failed module and installed in the replacement module, thereby preventing the need to reconfigure the new module using a direct connection.

Auto-configuration using DHCP is another option being considered.

Simplified Interface to Timing and Data Links

Several serial Manchester encoded 10 MHz and 14 MHz timing and data links exist in the Collider-Accelerator complex to distribute important machine data and provide synchronized machine timing triggers [5,6]. The eRHIC hardware architecture must provide a simple interface to these links.

Existing hardware in the C-A complex typically provides a direct hardware connection to the links, thus requiring that each module have several PLL circuits (one for each link), which although not complicated, requires a fairly large PC board footprint (Fig. 3).

For eRHIC, a goal is to further simplify the link distribution by connecting one copy of each link to a single module in the multi-slot chassis, and distributing the signals to all modules via the backplane.

A custom designed module (V208, Fig. 3) has already been developed with 6 phase lock loop (PLL) circuits to transmit the clock and data signals for each link over user defined P2 pins on the VME backplane using LVDS multi-drop pairs. An overlay module is installed on the rear of the backplane to bus each clock and data pair to all slots on the VME backplane. A simple interface to a gate array on each module is then possible.

High Speed Processing of I/O Signals

With 4 400 MSPS analog to digital converters resident on the V301 module, high speed processing of signals is essential. The Zynq family of SoCs is fully capable of providing the required processing speeds.

Ethernet Connections

Two or more Ethernet connections are required to be available on each module. One will be used for communication with the higher-level control system using standard Ethernet protocols and the other will be used to provide a private data distribution network.

Embedded Operating System with Resident Control System Software Objects

The latest low-cost SoC components like the Xilinx Zynq family have made it possible to easily and inexpensively design modules that incorporate an ARM processor running an embedded Linux operating system

ISBN 978-3-95450-148-9

(OS) that is tightly coupled on the same integrated circuit to high speed programmable logic. The eRHIC real-time hardware architecture plans to take full advantage of this capability.

Modules will be designed using the Zynq or similarly available SoC. The on-chip ARM processor will run a version of Linux that will be responsible for executing the low-level control system software objects (called ADOs in the RHIC control system, and IOCs in EPICS).

A dedicated Ethernet connection on each module will be used to communicate with higher-level software using the currently available software infrastructure.

The V301 prototype has been designed to perform as described here, and is currently operating exactly as intended.

Gigabit Communication Between Modules

In order to develop a system with building block modules that can be used in a variety of system applications, a method for high speed communication between modules is essential. Parallel busses like VME are no longer adequate for many systems, and it is unreasonable to think that a universal module can be designed to accommodate the I/O needs for every system.

The specific details for satisfying this requirement are a work in progress. Some options under consideration include:

- Custom designed Gbit link between modules using external board-to-board connections
- Ethernet hardware protocol with custom defined software packet structure to eliminate Ethernet layer overhead but allow commercially available Ethernet components and network switches to be used for private networks. This approach has successfully been used for the RHIC 10 Hz global orbit feedback system.
- Ethernet or PCIe communication via a backplane like openVPX through a custom designed switch module located in the center slots per the openVPX standard.

The V301 includes an on-board connector with direct connection to the Zynq MGT multi-Gbit pins (Fig. 2). This interface will be used to prototype high-speed serial communication between modules.

Using Commercially Available Modules

The present RHIC Control system utilizes many commercially available VME modules. The ability to use commercially available modules in eRHIC is highly desirable. This is another reason why a multi-slot chassis system architecture is planned.

Sharing of Modules

Similar to using commercially available modules, there is also a strong desire to be able to share modules between groups within the C-A complex as well as with other institutions. In order to make this viable, standards must be developed and adhered to.

CONCLUSION

A general architecture for eRHIC real-time hardware has been designed and a very successful prototype system using existing VME chassis has been developed.

The basic hardware architecture, which uses a SoC with embedded processor running an operating system, with the processor being tightly coupled to the on-chip gate array, is not unlike other recent developments in the C-A complex and other facilities.

Previous SoC system developments like the C-A low-level RF system [7] and the 10 Hz global orbit feedback system [8] have been constructed in self-contained chassis. The main difference with the proposed eRHIC architecture is the use of multi-slot chassis in order to house the large number of modules that are anticipated, to provide easy maintenance and board swapping, to allow the integration of commercially available modules, and to share hardware module designs with other institutions.

The present prototype system has been based on VME, mainly because VME chassis are abundant at the C-A complex. A more modern chassis and backplane design such as openVPX will likely be selected to provide Gbit serial link communication between modules via the backplane. However, of the chassis backplane configurations that have been researched to date, none contain an architecture that perfectly suits the needs for eRHIC and accelerator systems in general. The best option may be to develop a custom backplane, possibly using the VPX standard, which provides the capability to customize backplanes.

Since sharing modules with other institutions is highly desirable, this would be a perfect time for the accelerator hardware community to collaborate our efforts and together select a common backplane design.

REFERENCES

[1] R. Frankel, “Informal Datacon System Report,” BNL AGS Division Tech. Note No. 88, December 6, 1971

[2] V.J. Kovarik, “Signal Specifications and Transceiver Operation for DATACON II System,” BNL EP&S Division Tech. Note No. 58, March 14, 1973

[3] J. Morris, T. D’Ottavio, “Status Report for the RHIC Control System,” ICALEPCS 2001, San Jose, CA (2001)

[4] L.T. Hoff, “Real-Time Scheduling of Software Tasks,” ICALEPCS 1995, Chicago, IL (1995)

[5] B. Oerter, C.R. Conkling, “Accelerator Timing at Brookhaven National Laboratory,” PAC 1995, Dallas, Texas (1995)

[6] T. Kerner, C.R. Conkling, B. Oerter, “V123 Beam Synchronous Encoder Module,” PAC 1995, Dallas, Texas (1995)

[7] T. Hayes, K.S. Smith, “A Hardware Overview of the RHIC LLRF Platform,” PAC 2011, New York (2011)

[8] R. Michnoff, et. al., “RHIC 10 Hz Global Orbit Feedback System,” PAC 2011, New York (2011)

ISBN 978-3-95450-148-9