REPORT DOCUMENTATION PAGE

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (From - To)
	Final Technical Report	05/01/2015-08/15/2017

4. TITLE AND SUBTITLE
Development of Universal Controller Architecture for SiC Based Power Electronic Building Blocks

5a. CONTRACT NUMBER

5b. GRANT NUMBER
N00014-15-I-2346

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Dr. Herbert Ginn

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
University of South Carolina

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Office of Naval Research

10. SPONSOR/MONITOR'S ACRONYM(S)
ONR

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for Public Release; distribution is unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The objective of this project was to develop a Universal Controller Architecture suitable for SiC based Power Electronic Building Block (PEBB) converters. Recent developments in SiC power devices are beginning to yield PEBBs with far greater switching frequencies than Si based devices resulting in an order of magnitude reduction of the control time scales as compared to converter systems utilizing conventional IGBT based PEBBs. In addition there have also been advancements in highly modularized converter systems with hundreds of PEBBs, such as the Modular Multilevel Converters. Both of those trends present the need to evaluate architecture tradeoffs and communication requirements for hardware realizations of the Universal Controller Architecture for control of power electronic converters and systems of converters. The control network should be designed to have minimal round-trip latency and maximal scalability.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF: Unclassified			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON Dr. Herbert Ginn
a. REPORT	b. ABSTRACT	c. THIS PAGE			19b. TELEPHONE NUMBER (include area code) 803-777-5598

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39.18

Final Report

Development of Universal Controller Architecture for SiC Based Power Electronic Building Blocks

30 October 2017

SUBMITTED BY

DR. HERBERT L. GINN, PI
DEPT. OF ELECTRICAL ENGINEERING
UNIVERSITY OF SOUTH CAROLINA

DR. JASON BAKOS, Co-PI
DEPT. OF COMPUTER SCIENCE AND ENGINEERING
UNIVERSITY OF SOUTH CAROLINA

Award Number	N000141512346
Title of Research	Development of Universal Controller Architecture for SiC Based Power Electronic Building Blocks
Principal Investigator	Herbert L. Ginn
Organization	University of South Carolina

Technical POC

Dr. Herbert L. Ginn
Dept. of Electrical Engineering
University of South Carolina
Columbia, SC 29208
ginnhL@cec.sc.edu
phone: 803-777-8045

Administrative POC

Danielle McElwain
Sponsored Awards Management
901 Sumter Street
Columbia, SC 29208
dmcelwai@mailbox.sc.edu
phone: 803-777-1119

Table of Contents

1. Scientific and Technical Objectives

The objective of this project was to develop a Universal Controller Architecture suitable for SiC based Power Electronic Building Block (PEBB) converters. Recent developments in SiC power devices are beginning to yield PEBBs with far greater switching frequencies than Si based devices resulting in an order of magnitude reduction of the control time scales as compared to converter systems utilizing conventional IGBT based PEBBs. In addition there have also been advancements in highly modularized converter systems with hundreds of PEBBs, such as the Modular Multilevel Converter (MMC). Both of those trends present the need to evaluate architecture tradeoffs and communication requirements for hardware realizations of the Universal Controller Architecture for control of power electronic converters and systems of converters. The control network should be designed to have minimal round-trip latency and maximal scalability.

In order to accomplish the primary project objective a key secondary objective was to conduct a study to determine the most appropriate communication architecture and routing for networked PEBB control systems. In doing so we have considered all of the control layers from the lowest hardware layers up to ship-wide system control. The appropriate partitioning and interface requirements between the various control layers is considered for power electronic systems with particular focus on the system and application control layers for systems of power electronic converters so that the minimum set of application level control interfaces is compatible across all power electronic controllers. This provides flexibility of system design and operation for distributed converter systems so that a more flexible energy management capability is achieved for naval electric distribution systems.

2. Approach

In a ship-wide PEBB-based power distribution system, control and measurement modules are spatially distributed. While modules that form the control system for a single converter may be somewhat co-located, modules at the application level of control and above will be distributed throughout the overall system. Therefore, it is generally not feasible to connect them all directly into a single central controller. Instead, it is more practical to distribute control among the modules within converters and at layers above individual converter control, such as zonal or bus level controls. Using a multi-hop network, each control module should contain a small integrated router that can both serve as a network interface and serve as an intermediate forwarding point for other messages sent among other control modules. In these types of networks, the worst-case message latency is determined by the longest possible path between two control modules. This worst-case latency serves as a constraint for the overall control system design. As such, both the physical topology of the communication network and the routing algorithm are important considerations for the system design.

FPGA based designs are used with increasing frequency for power electronic converter designs due to modular design approaches and performance requirements imposed by increasing switching frequencies. Recent generations of FPGAs include multiple high performance serial transceivers that support speeds in excess of 10Gbit/sec. This opens up possibilities beyond the simple ring communication network topology common up to now in power electronic control systems. Other network topologies were explored in this project given that inclusion of multiple Gigabit serial links per control node is viable.

In performing control module development at any control level there are several design challenges that must be addressed. The key design focus areas and approach toward each at the beginning of this project were specified as follows:

1. Bounding Control Loop Delay
The stability and performance of the PEBB modules is affected by the delay between when measurements are taken and when updated references are received from the controller. Since each level of the PEBB hierarchy is connected in a local topology, transitioning packets between levels also contributes to the delay. Each FPGA-based endpoint will be designed to contain multiple integrated gigabit channel interfaces. In a ring topology only two channels are used. By using additional channels alternative topologies can be explored. The delay can be reduced by exploring alternative network topologies at each control level and between control levels considering a range of system scales.

2. Library of Reusable IP Blocks
Recurring control design costs is reduced by development of a module library of commonly-needed measurement and control behaviors that can be easily added to the FPGA design for each PEBB module. Use of standard bus architectures such as the Xilinx-standard AXI4 bus for interoperability and integration, allows them to be easily instantiated into a design using the latest FPGA platform tools (ex: Xilinx Vivado). These modules include channel interfaces and associated error control and routing logic, measurement and control interfaces, and online diagnosis and system health monitoring modules. Diagnosis modules will incorporate softcore microcontrollers and associated software that can be individually contacted on each module to assess performance counters and access low level status information during runtime.

3. Fault Tolerance
Each channel connecting modules at every PEBB level utilizes a MAC layer protocol (Xilinx Aurora), which includes 8b/10b encoding that facilitates channel synchronization. We will add support for forward error correction (FEC) at the link layer to guarantee recovery from communication errors.

4. Application Layer Partitioning
For system level design, a standardized interface or set of interfaces between the application level and system level control functions is needed. An objective of the research is to determine the appropriate partitioning and interface requirements between the system and application control layers so that the minimum set of system level to application level control interfaces is compatible across all power electronic controllers. The application level control dictates the operation of a power electronics system in order to meet the mission determined by the system level control. In order to meet the research objective, the components and function of both the system layer of control and the application layer of control must be determined for the various classes of power electronic systems and their applications. The commonalities across applications will be explored in order to define the minimum set of interface requirements.

5. Top System Control Layer Considerations
A Shipboard Wide Area Network (SWAN) has been developed for the LPD-17 transport class as well as DDG-1000. There is also a high probability that future ships will have a similar SWAN. The SWAN design is expected to be an open architecture employing current network technology and commercial equipment. Connection of the PEBB-1000 controllers over a SWAN provides an

infrastructure that supports further development and validation of the interface between the power system converters' network and top system control. A minimum of two control systems will be connected to a three-node SONET/ATM network that will represent the SWAN. Performance restrictions due to network communication bandwidth and latency times will be investigated. Tradeoffs between communication requirements and variations of the system level to application level control interface architecture will be explored. This activity will aid in optimizing the partitioning and interface requirements of the control boundaries at the topmost system interface.

Significant progress was made in each of the focus areas as outlined in the accomplishments section and described in detail in the expanded accomplishments section below.

3. Accomplishments

The project focus areas are interrelated challenges that must be overcome to achieve the overall objective. In order to organize execution along a project timeline the work was conducted under four tasks. Major accomplishments achieved during the course of this research project are presented below by task.

Task 1. Technology transfer of USC modular digital control platform to CPES
The latest design of the Universal Controller Architecture at the time of the project start was used as an initial design point. It was also shared with researchers at CPES to assist with ongoing PEBB-1000 control efforts. Detailed design documents, notes and schematics were sent to CPES.

Task 2. Evaluation of digital control platform partitioning and module development below the application level
Communication topologies and protocols have been explored utilizing a new Kirtex-7-FPGA platform with support for six 12.5 Gb/s bidirectional communication channels. Re-usable IP blocks have been designed for the FPGA module interfaces. A design was developed as an FPGA-based system-on-chip (SoC), consisting of a Microblaze soft-core microcontroller and network interface supporting four optical bidirectional 10 Gb/s channels using the Xilinx Aurora communication protocol. Communication between FPGAs was tested successfully at 10GBit/sec.

The FPGA boards were interconnected to form a control network to achieve closed loop control among the control nodes within a Power Electronic Building Block based system. Each FPGA-based endpoint contains multiple integrated Gigabit channel interfaces allowing for various topologies to be explored. Several topologies were evaluated and a 2D Torus was identified as the best compromise for further development. In this topology each node is connected to four neighboring nodes via bidirectional communication links.

In addition to the network topology, routing protocols are needed for the FPGA intercommunication. We started from the knowledge base in the distributed computing area and modified those routing protocols based on the fact that the determinism needed for real-time feedback loops via the communication links imposes different requirements on the routing than for standard distributed computing applications for general purpose distributed computing systems. Power electronic control systems consist of multiple control loops and levels or layers of control within a hierarchy. Achieving the minimum bandwidth requires that the routing algorithm equally distribute the communication data transfer load at critical bottleneck locations in the power electronics control system. Such bottlenecks occur at nodes located at control layer boundary interfaces designated as ingress/egress nodes To achieve this, we proposed "Hub Routing",

4

comprised of a set of pre-computed static routes between each node and the ingress/egress node (layer boundary point), where each packet follows a path that keeps its location on the grid closest to the straight line between the node and the ingress/egress node. In Hub Routing the packet follows a path closet to the shortest distance. The minimum channel bandwidth utilization was compared for both standard X-Y type routing and the proposed Hub Routing for various network sizes. Hub routing optimally balances the communication load among the channels, and was shown to reduce the minimum bandwidth requirement for a given power electronic control network size as compared to X-Y routing.

Results from the study show that the selected topology and developed routing protocol result in round trip latencies in the low single digit microsecond range for networks of 25 (5x5) PEBB nodes. Moreover, it allows increased system scaling within the available bandwidth as compared to the typically used X-Y routing.

Task 3.   System to application layer interface definition development and upper layer module development for the modular digital controller

Our proposed FPGA design for use in the application layer is decomposed into two mostly isolated subsystems. One of these systems is designed for real-time control and control network routing and the other for non-real time instrumentation and monitoring. The two subsystems are isolated and share only one common peripheral, an on-chip BRAM that holds the controller state. Both processors have local on-chip memory from which they execute their respective program code, both processors have independent interrupt controllers, and both processors have independent timers.

We characterized the network performance of the 10 Gbps communication infrastructure including all of the overhead of sub-systems that provide a flexible platform for application control. The test numbers obtained using the 2D Torus network configuration along with the developed Hub Routing method have shown that the Application layer of control can function as the most fundamental System layer within a distribution system comprised of many power electronic converters. As an initial test of these conclusions, a hardware test for a single ship distribution system zone fed by two Power Converter Modules (PCMs) was successfully conducted in which the Application control layer of the PCMs was moved to the Zonal system control layer. Specifically, in-zone voltage regulation and power sharing functions were moved to the Zonal system control layer.

Task 4. Evaluation of system to application level partitioning and communication

In order to include a study of the interface between the power electronic converter network and the uppermost ship system control network a three multiplexer node OC-48 synchronous optical network (SONET) representing a portion of a shipboard system level communication network was procured and commissioned. Several candidate network configurations have been successfully provisioned. Testing has been conducted to determine which configuration has the lowest level of latency and jitter. In order to do this we have employed an existing control platform with standard Ethernet communication and modified it to serve as a delay measurement baseline by sending communication packets around the OC-48 network along with time markers. It was determined that the SONET ring only introduces approximately 200μs of maximum delay into the top layer of system control communication. However, up to 50ms is required if there is a break in the optical ring network. Because SONET is a ring topology this fault recovery is not able to be reduced.

4. Expanded Accomplishments

4.1 Technology transfer of USC modular digital control platform to CPES

The University of South Carolina developed a previous generation of the Universal Controller prior to the start of this project for use with a Modular Multilevel Converter prototype and PM-1000 converters in the USC Energy Routing Lab. It is comprised of three modules: a PEBB level adapter, a FPGA based lower level controller (Hardware Manager), and a DSP based high level controller. The FPGA module and a functional block diagram are shown in Fig. 1, and the DSP controller module and its functional block diagram are shown in Fig. 2.
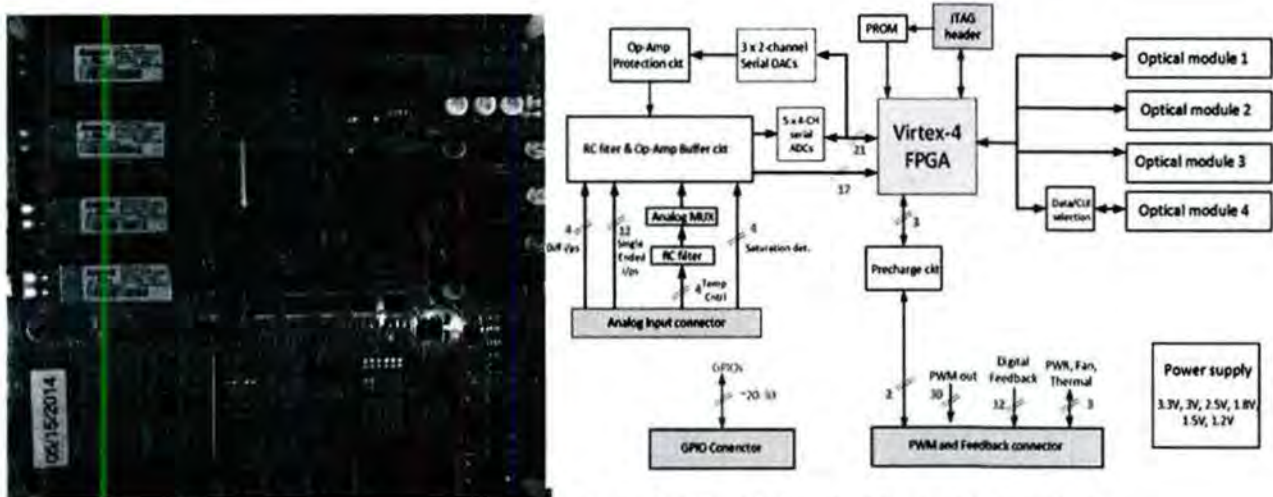

Figure 1. FPGA based lower level controller with gigabit serial interface.

The FPGA modules utilize four high speed serial transceivers at 1.25GB per second and are connected via fiber optics in a ring configuration. Control layers above and below the FPGA boards are connected in a star configuration as shown in Fig. 3. Although the control modules described above were developed for standard Si based PEBBs and utilized a traditional ring topology they provided a starting point for the developments in this project. It was also provided to the Center for Power Electronic Systems (CPES) at Virginia Tech as a reference design for development of the PEBB-1000 control platform.
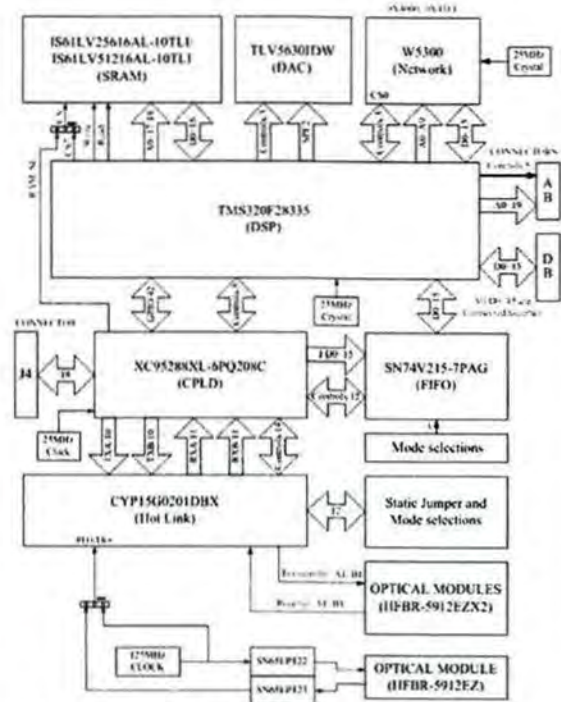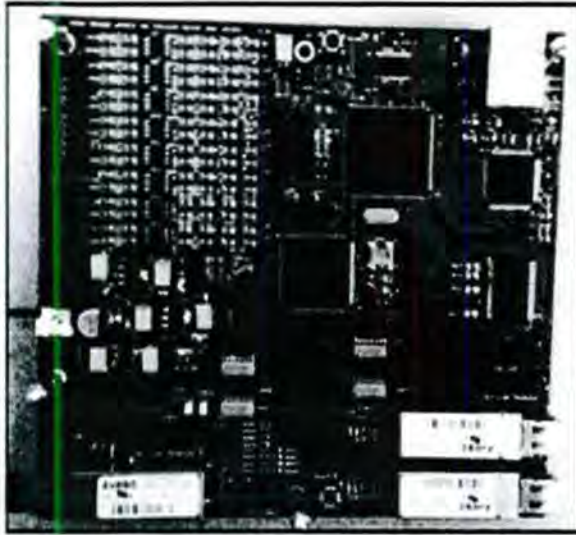
6

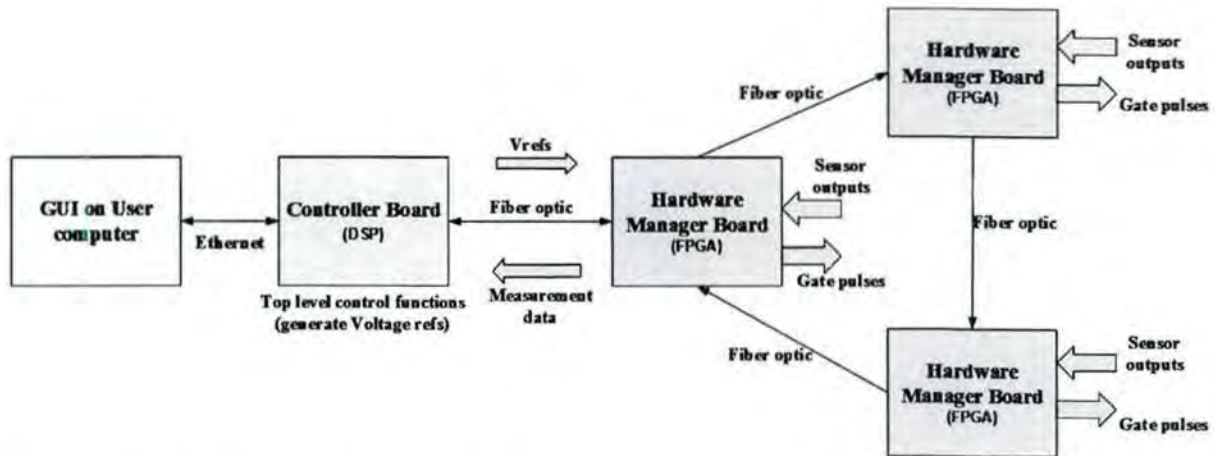Figure 2. DSP based high level controller photo and block diagram.

Figure 3. Universal controller modules developed using 1.25GB/s fiber optic communication in the MMC control configuration.

4.2. Evaluation of digital control platform partitioning and module development below the application level

Each PEBB control module collects measurements from the attached power electronics and encodes and transmits the measurements and control data over the multi-hop control network to other control nodes either within the same control hierarchy layer or across a layer boundary as dictated by the control loops in

7

operation. Each node will later receive a corresponding control message from other nodes or layers. Since each operating control loop is deterministic, each control node must complete these tasks according to a fixed control period. In addition, the control system must also route messages on behalf of PEBB control modules on their path to or from other locations in the control network as needed.

The control system is constrained by the communication latency imposed both by the network (in terms of worst-case path length) but also the on-chip overheads of processing and forwarding packets. Longer worst case delays will constrain the minimum control period for a given control layer. Likewise, the effective channel bandwidth limits the maximum size/scale of the network, since larger networks have more overlapping routing paths requiring more channel bandwidth. Like other network technologies, the effective bandwidth is dependent on the packet size.

FPGA-based control node

Our interface node demonstrator platform is a Xilinx KC705 board with a Kintex-7 325T FPGA and attached quad-SPF+ transceiver FMC module, shown in Fig. 4. It was selected due to networking and expansion capabilities. This platform is capable of connecting directly to PEBB hardware managers or other PEBB control level interfaces via an expansion connector, assuming of course that an appropriate adapter board is designed and implemented for a given PEBB. The FPGA boards provide interconnection capabilities allowing the formation of a control network to achieve closed loop control among the nodes within a PEBB based system.

Our core FPGA design is structured as an FPGA-based system-on-chip (SoC), consisting of a soft-core microcontrollers for management, monitoring, and debug functionality and an on-chip router and network interface supporting four optical bidirectional 10 Gb/s channels using the Xilinx Aurora link-layer protocol.

Figure 4. FPGA based control platform used in testing.

4.2.1 Network Topology and Routing

Each FPGA-based endpoint contains multiple integrated gigabit channel interfaces allowing for various communication network topologies to be explored. A primary control network design

consideration is to have minimal round-trip latency and maximal scalability with a reasonable number of channels per control node. Communication delay imposed on the controllers can be reduced by exploring alternative network topologies at each level and between levels. Several options were explored. As shown in Fig. 5a, in a ring topology the worst case delay of control and measurement messages will scale according to the number of modules participating in the ring (n). The delay can be reduced by exploring alternative network topologies at each level and between levels. As shown in Fig. 5b and Fig. 5c, a bidirectional ring will reduce the worst case delay to n/2 but will require four channels for most modules, while a 2D torus will reduce the latency further without requiring more channels but increasing the routing complexity. A 3D torus will further reduce latency while requiring 6 channels per module. Therefore, a 2D torus was selected as the best compromise for further development. In this topology each node is connected to four neighboring nodes via bidirectional links.
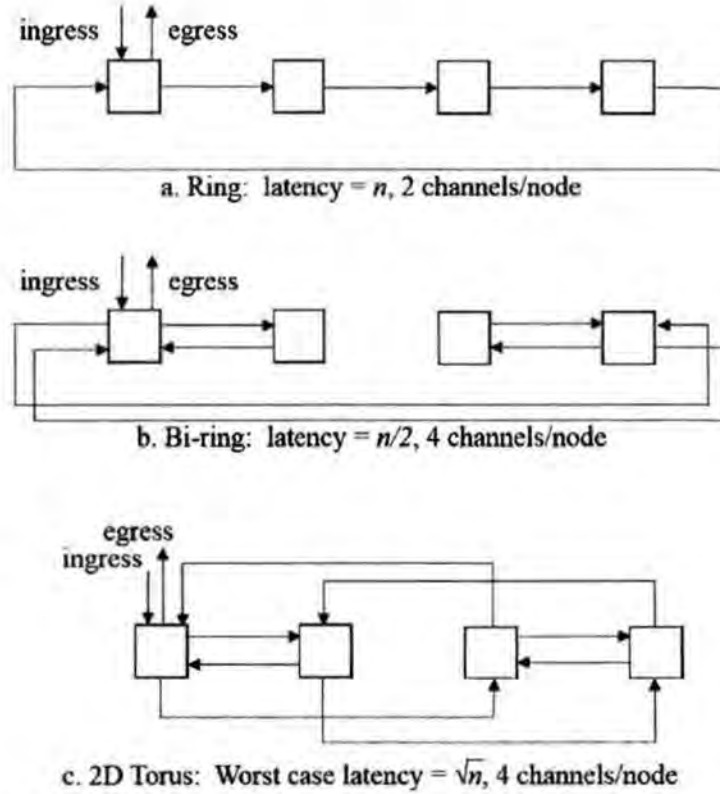

a. Ring: latency = n, 2 channels/node

b. Bi-ring: latency = $n/2$, 4 channels/node

c. 2D Torus: Worst case latency = $\sqrt{n}$, 4 channels/node

Figure 5. PEBB communication network topologies evaluated.

In a 2D torus of width w and height h, a message sent between nodes having addresses (x_1, y_1) and (x_2, y_2) has the following offsets in both dimensions:

$$\Delta x = \min\left(\left((x_1 - x_2)\ mod\ w\right), \left((x_2 - x_1)\ mod\ w\right)\right) \quad (1)$$

$$\Delta y = \min\left(\left((y_1 - y_2)\ mod\ h\right), \left((y_2 - y_1)\ mod\ h\right)\right), \quad (2)$$

and requires a routing distance of $\Delta x + \Delta y$ hops with $\frac{(\Delta x + \Delta y)!}{\Delta x! \Delta y!}$ possible paths. The longest possible path is $\frac{1}{2}w + \frac{1}{2}h$ hops.

9

Much of the current work in routing and topologies for multi-hop networks on FPGAs focus on networks-on-chip where a single FPGA contains all the routers comprising the network. In this case the router must be as compact as possible [2,3]. These networks typically use non-minimal deflection-routing to avoid the need for buffers in the router. Deflection routing allows packets to follow non-minimal routes when the outgoing ports on the minimal path(s) are currently occupied with other traffic, as opposed to buffering in the router. Because deflection routing increases latency and timing uncertainty it is not appropriate PEBB control applications. Well-known algorithms developed for distributed computing also generally employ non-minimal routing to maximize throughput, often at the cost of latency [4]. These networks are also generally designed for dynamic traffic patterns, as opposed to the static patterns assumed for controller networks. Work that focuses on multi-FPGA systems often focus on exploration of network topologies and not specific routing algorithms, and often do not explicitly consider the overheads contributed by the on-chip processors that interact with the network [5,6]. Here we have explored the most appropriate routing methods for PEBB based control systems where latency considerations is the driving factor.

Controller Traffic Pattern #1: All-to-One/One-to-All

All-to-One/one-to-All traffic assumes that a single node serves as an egress/ingress point to the adjacent control layers and that all nodes will transmit measurement or control data to the egress node and subsequently receive control signals from the ingress node, and that this control loop may have a latency no greater than the control layer sampling period. Assuming the design parameters shown in Table 1 and assuming that the parameter values chosen will not exceed the maximum bandwidth of any single channel, each packet will experience a round-trip latency of

$$latency_{roundtrip} = 2 \cdot \left(\frac{latency_{Aurora} + latency_{route}}{freq_{FPGA}} + \frac{size_{packet} \cdot 8}{bw_{Aurora}} \right) \cdot \left(\frac{1}{2}w + \frac{1}{2}h \right) \tag{3}$$

Table 2 shows minimum round trip latencies for the default parameter values.

Table 1: Design parameters

Parameter	Variable	Expected value
Maximum latency of the Aurora links	$latency_{Aurora}$	53 clock cycles
Packet size	$size_{packet}$	100 bytes
Routing latency	$latency_{route}$	1 clock cycle
Link bandwidth	bw_{Aurora}	10 Gb/s
FPGA user clock frequency	$freq_{FPGA}$	156.25 MHz
Network size	n	100 nodes
Network order, $n = o^2$	o	10 nodes

The simplest routing scheme for multi-hop networks is X-Y, or dimension-ordered, routing, in which the network routes packets in the X dimension until the packet reaches a node that is vertically aligned to the destination and then routes in the Y dimension [1]. X-Y routing is simple to implement and is guaranteed to follow minimal length routes. However, for the traffic pattern for PEBB control networks, where all nodes periodically send one packet and receive one packet from the ingress/egress node, the north and south channels into the ingress/egress node carry more traffic than the east and west channels. During the control period (which we set to $latency_{roundtrip}$), both the north and south channels will

10

experience $\frac{w \cdot h - w}{2}$ packet traversals while the east and west channels will only experience $\frac{w}{2}$ packet traversals. Thus the maximum-loaded channels will require a maximum channel utilization of:

$$bw_{utilization} = \frac{size_{packet} \cdot 8 \cdot \frac{w \cdot h - w}{2}}{latency_{roundtrip}} \tag{4}$$

Achieving the minimum bandwidth requirement requires that the routing algorithm equally distribute the load on the ingress/egress node's channels where load is greatest ($\frac{w \cdot h}{4}$ packets per channel per control period). To achieve this we propose "hub routing", comprised of a set of pre-computed static routes between each node and the ingress/egress node, where each packet follows a path that keeps its location on the grid closest to the straight line between the node and the ingress/egress node. The distance between a given node at location (x_0, y_0) and a straight line ($ax + by + c = 0$) is computed in the traditional way, i.e. $\frac{|ax_0 + by_0 + c|}{\sqrt{a^2 + b^2}}$. We can implement hub routing using a small routing table on each node comprised a $\frac{w \cdot h}{4} \times$ 3-bit table in each router.

Table 2: Minimum round trip latencies.

Network size	Round trip latency
5x5	4.3 us
10x10	8.5 us
20x20	17.0 us
30x30	25.6 us
40x40	34.1 us
50x50	42.6 us

Figure 6 shows an example of the packet's path using X-Y routing versus Hub routing. In X-Y routing, a packet traveling between the two black nodes travels in the X-dimension before the Y-dimension, while in Hub routing the packet follows a path closet to the straight line shown by the dashed line. Hub routing guarantees minimal bandwidth requirement:

$$bw_{utilization} = \frac{size_{packet} \cdot 8 \cdot \frac{w \cdot h}{4}}{latency_{roundtrip}} \tag{5}$$

Table 3 compares the minimum channel bandwidth utilization for both X-Y and Hub Routing. X-Y routing requires more than the available 10 Gb/s bandwidth when scaling the network to 30x30, while the Hub routing supports network sizes up to 40x40.

Minimum Link Bandwidth

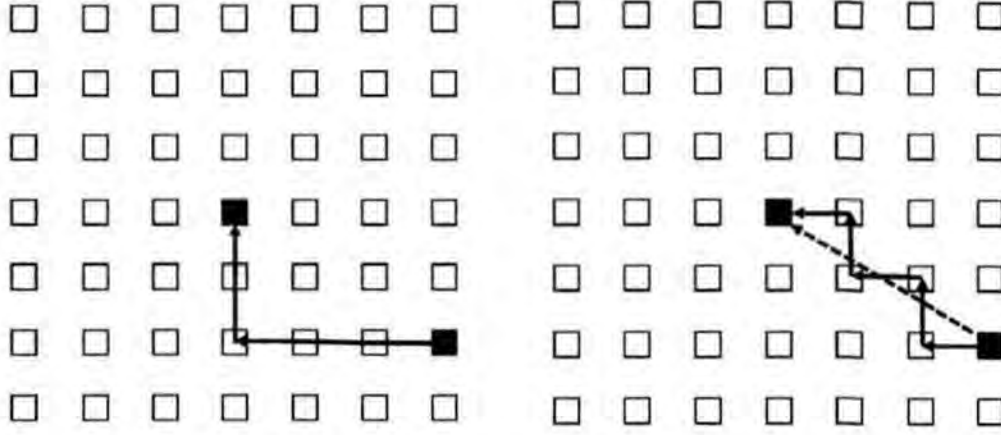| Network size	$bw_{utilization}$ (Gb/s): XY	$bw_{utilization}$ (Gb/s): Hub
5x5	1.9	1.2
10x10	4.2	2.3
20x20	8.9	4.7
30x30	13.6	7.0
40x40	18.3	9.4
50x50	23.0	11.7

Figure 6. X-Y routing (left), Hub routing (right).

Controller Traffic Pattern #2: All-to-All

In order to generalize the routing study for traffic patterns at higher levels of control such as application or system layers where any node may need to communicate with all other possible nodes in a PEBB based system the *All-to-All* scenario was also considered. A *k*-ary 2-cube network topology such as a 2D Torus of width w and height h has $w \cdot h \cdot (w \cdot h - 1)$ potential source-destination pairs, with each pair having $\frac{(\Delta x + \Delta y)!}{\Delta x! \Delta y!}$ possible minimum-hop paths. X-Y routing selects only one possible path but leads to load imbalance due to all paths not originating on the same row entering the destination from the Y-dimension channels. Hub-based routing selects only one path for any source-destination pair where $\Delta x \neq \Delta y$, leaving $2^{\lceil \frac{\Delta x + \Delta y}{2} \rceil}$ possible paths for the $h \cdot w \cdot \min(h, w)$ source-destination pairs where $\Delta x = \Delta y$.

An example of this effect is shown in Figure 7, in which there are there are $(\Delta x + \Delta y)/2$ binary decision points along the route, yielding an ambiguity of 16 possible valid paths among the 70 possible minimal paths. We refer to these as ambiguous routes. For a network of size 5x5 as shown in the Figure, there are 125 ambiguous routes among the 600 possible routes (21%). This effect does diminish for larger network sizes. For example, a 10x10 network has 10% ambiguous routes and a 100x100 network has only 1% ambiguous routes.

Despite this, ambiguous routes reduce effective network capacity for more complex routing patterns. For fan-in/fan-out traffic patterns, the network may route ambiguous paths deterministically without causing any load imbalance relative to the channels connected to the ingress/egress node and thus will not affect the network capacity. For more complex patterns, this ambiguity will cause significant

12

imbalance throughout the network. This is especially true for topologies not structured as a regular grid, as well as networks having failed or unconnected channels.

As part of this project we have developed a preliminary algorithm for achieving load balance in this type of environment. The algorithm uses a branch-and-bound search to resolve ambiguous hub routes for a network of a given topology.
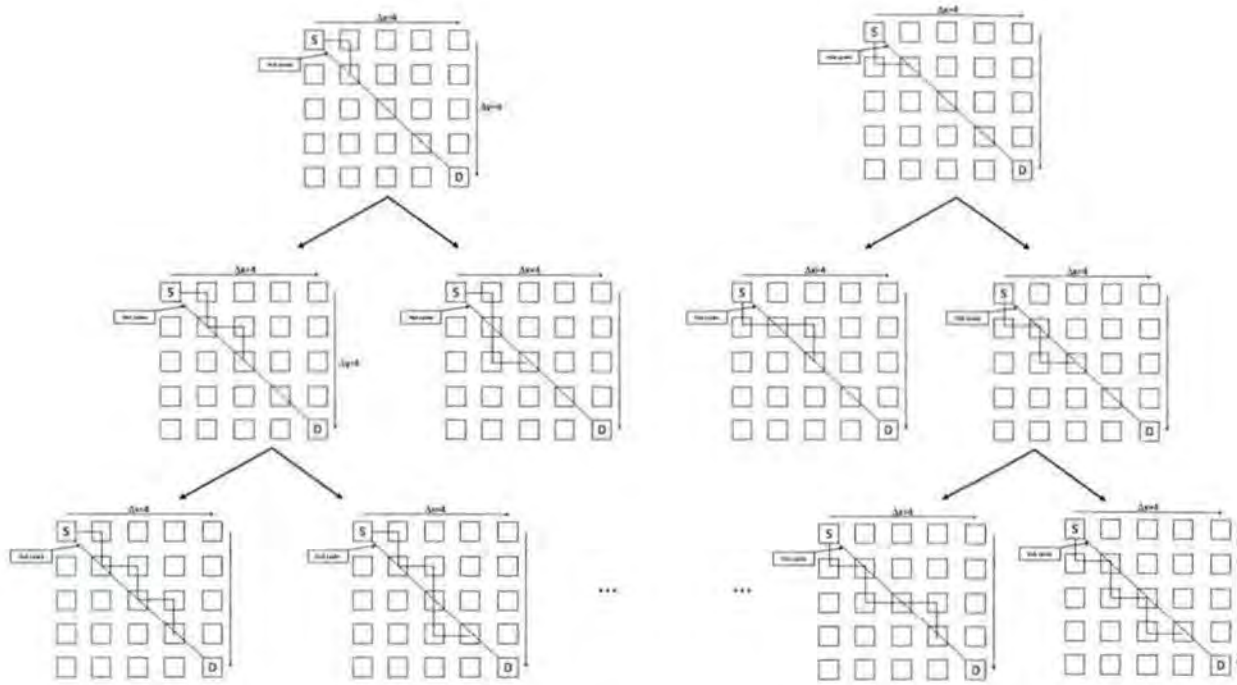

Figure 7: Route ambiguity for Hub-based routes when $\Delta x = \Delta y = 4$

4.3 System to application layer interface definition development and upper layer module development for the modular digital controller

A fully functioned controller for PEBB based systems targeted at application layer control was designed for the FPGA nodes and centered around the network architecture and routing method described above in Section 4.3. Including all of the necessary control peripherals typically used within application control allows for characterization of the communication latency and bandwidth utilization including all of the additional overhead that provides flexibility of function for the application layer.

4.3.1 System-on-Chip Design

Figure 8 shows a block diagram of the design we programmed into the FPGA. The design is logically split into two subsystems mastered by a separate Microblaze microntroller: the controller subsystem and the monitor subsystem. The two subsystems are isolated and share only one common peripheral, an on-chip BRAM that holds the controller state. Both processors have local on-chip memory from which they execute their respective program code, both processors have independent interrupt

controllers, and both processors have independent timers (the monitor processor uses its timer for the TCP/IP stack). The TCP/IP stack stores its data on off-chip DRAM.

The controller subsystem performs the control and routing tasks on behalf of the module and is optimized for latency and determinism. To minimize the amount of unpredictable delays, we took the following steps: (1) store the microcontrollers's software and data in on-chip memory, as opposed to off-chip memory, which has substantially higher latency, (2) limit the set of interrupts to only the four DMA interrupts corresponding to the four DMA modules connected to the Aurora interfaces (which only interrupt the processor when a packet arrives from any of the Aurora interfaces) and a timer interrupt (which interrupts the processor when it is time to collect measurements and transmit a message to the zone controller), and (3) place the interrupt controller in fast mode, in which the interrupt controller passes the handler address directly to the processor without any software intervention.

We use a non-real time 1 Gb/s Ethernet interface for monitoring and control of the module. The Ethernet subsystem runs as a fully-custom hardware IP module in the FPGA logic fabric but its TCP/IP stack runs in software. The TCP/IP stack is heavyweight and imposes unpredictable loads on the microcontroller, but when running on its own microcontroller it cannot interfere with the control subsystem.
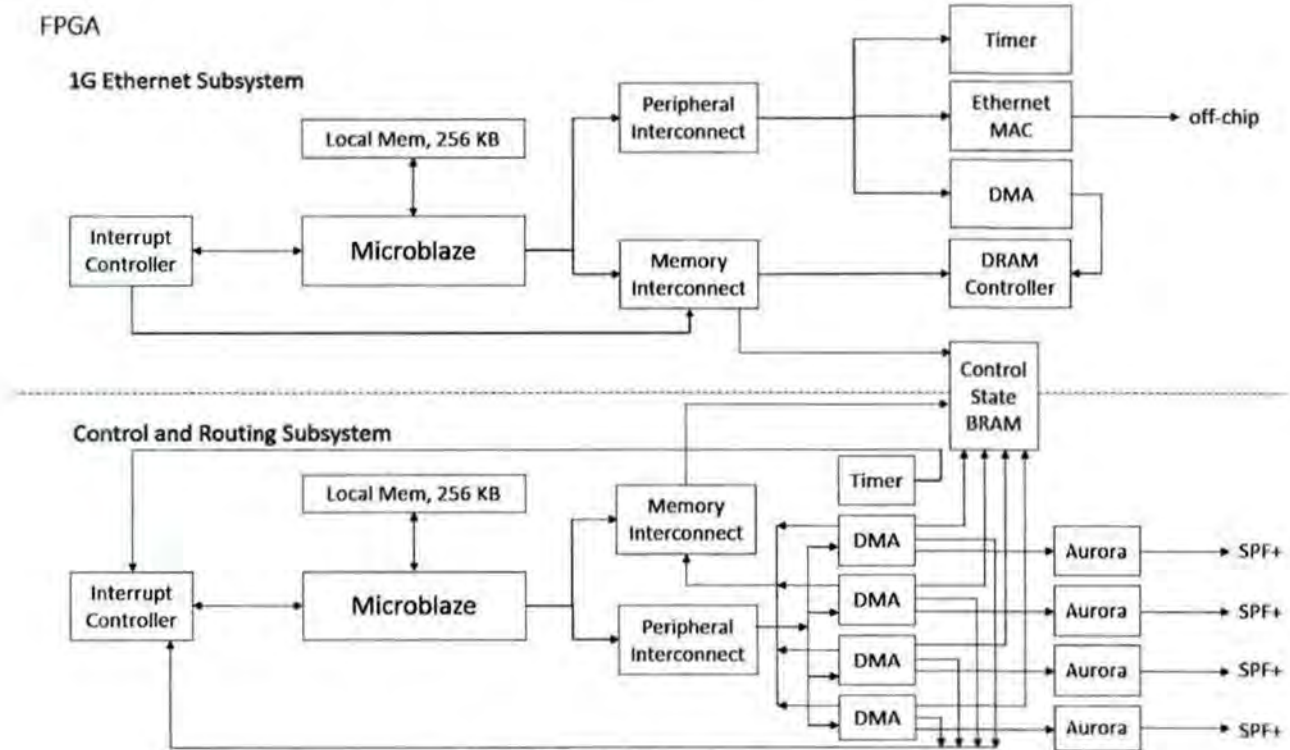

Figure 8. Top-level Design.

4.3.2 Platform Evaluation

Latency

In order to evaluate the internal latency of controller subsystem we set up an experiment with a single board having a loopback cable from channel 0 to channel 1. The software would transmit one packet every control period, and the DMA interrupt handler measured the round-trip delay. This measurement includes the latency contributions from the transmitting DMA engine, the transmitting Aurora interface,

the optical transmission latency, the receiving Aurora interface, the receiving DMA engine, and the interrupt controller. These values represent the effective channel latency for one hop.

Figure 9 shows the distribution of packet latencies over 1 million packet transmissions for a 32-byte packet and a 4 KB packet. Note that the Y-axis of the histograms is plotted on a logarithmic scale. For the 32-byte packet, 18.3% of the packets experienced 1150 to 1200 cycles of latency and 81.6% of the packets experienced 1200 to 1250 cycles of latency. On the Microblaze's 100 MHz clock, 1200 cycles is equivalent to 12 us, while the transmission time of a 32 byte packet on a 10 Gb/s channel is 25.6 ns (note that our clock rate is less than the example parameters listed in Table 1). For the 4 KB packet, 99.9% of the packets experienced 1250 to 1300 cycles of latency, against a 3.2 us expected transmission time. The ~100-cycle latency difference between the 32-byte and 4 KB packet size is equivalent to approximately 1 us, caused by the higher transmission time for the larger packet. These results indicate that the packet size has little relative effect on the end-to-end transmission latency, since a 128X increase in packet size required only a 5 to 10% increase in latency. Note that because the platform overheads are 3.9X to 468X that of the channel transmission time.
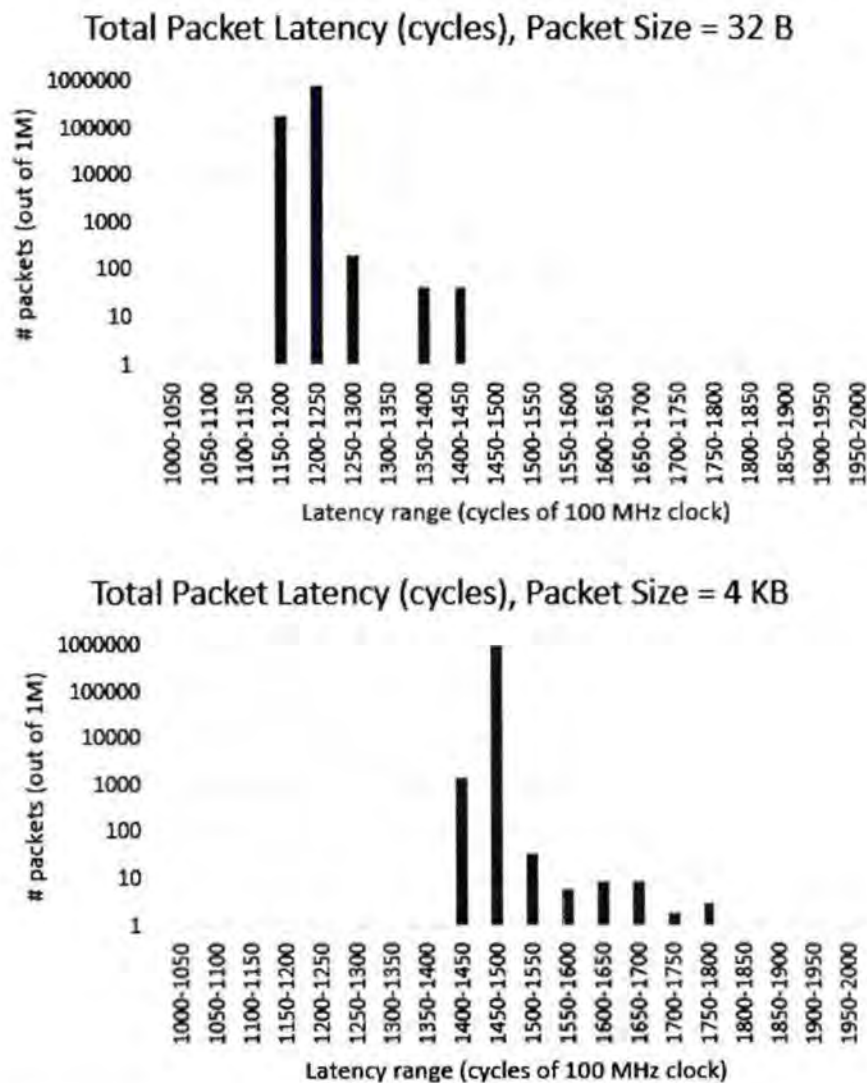

Figure. 9. Observed packet transmit latency for 32 byte packets (top) and 4 Kilobyte packets (bottom). These results include packet transmission time over the 10 Gbps link (~3 cycles for a 32 byte packet and ~328 cycles for a 4KB packet) and all controller design overheads. Note that the Y-axis is logarithmic.

Bandwidth

To evaluate the effective channel bandwidth, we added a transmit command to the DMA handler that causes the software to transmit a new packet immediately after receiving a packet. We used a 2000-cycle timer interrupt to gather statistics. Figure 10 plots the effective bandwidth of the channel, in Megabits per second, versus the packet size. The 32-byte packet size uses 38 Mbps of the channel capacity, the 512-byte packet size uses 614 Mbps, the 4 KB-packet size uses 3.2 Gbps, and the 8 KB-packet size uses 6.5 Gbps. Our observed bandwidth is even lower than Table 4 suggests, since the processor must also periodically call the timer interrupt handler, which calculates and records performance statistics. In this test we lose additional performance because we only allow for up to one in-flight packet. In future work we will incorporate more descriptor-based DMA and/or flow control to allow for multiple simultaneous in-flight packets to improve effective bandwidth for smaller packet sizes.
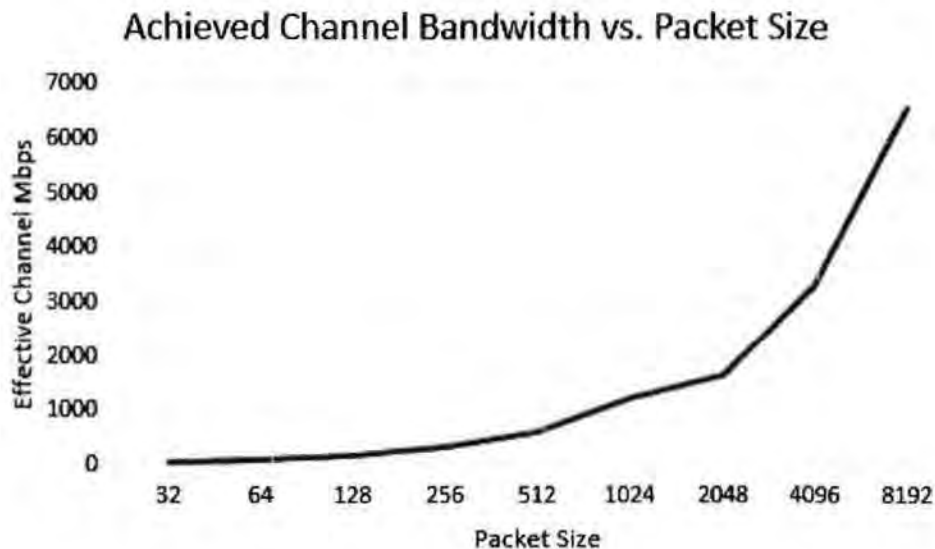

Figure 10. Observed Aurora channel bandwidth versus packet size.

Traffic Batching

To achieve higher bandwidth for smaller packet sizes we measured the effective bandwidth achieved by batching a set of smaller packets into a larger burst, requiring the Microblaze to interact with the DMA controller only after each burst.

Figure 11 shows these results in gigabits per second, which indicate that burst performace is largely independent of packet size and yield a roughly equivalent bandwidth to a single large-packet, as a 2KB burst achieves ~1.5 Gbps and a 16 KB burst achieves ~5 Gbps, mostly consistant with the large packet sizes of Fig. 10. Figure 12 plots effective bandwidth in gigabits per second achieved by batching an increasing number of packets and is consistent with the trend in Fig. 10 for increasing a one-packet transmission of increasing packet size.
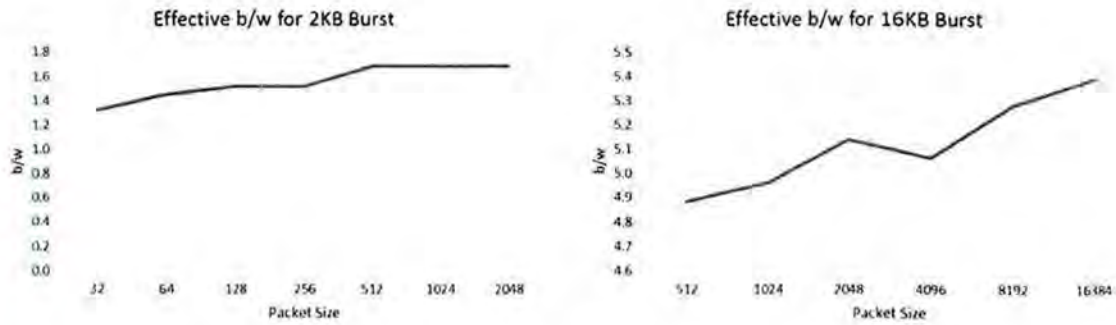
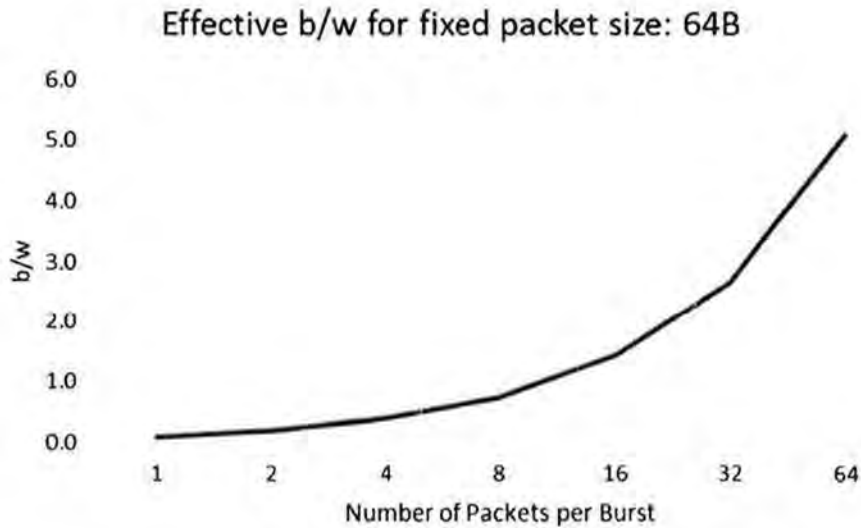Figure 11. Observed Aurora channel bandwidth versus packet size.

Figure 12. Observed Aurora channel bandwidth versus packet size.

Figure 13 plots the effective bandwidth in gigabits per second of bursts of a fixed number of packets and shows that effective bandwidth is bounded by packet size, at least for small packets. In this case, bandwidth levels off at 5 Gbps at a 128 packet batch size.

4.3.3 Multiple Controller Validation Test

The results summarized in Figures 9 and 10 indicate that the application layer of control can function as the most fundamental system layer within a Power Electronics Based Power Distribution System. As an initial verification of these conclusions, a hardware test for a single zone fed by two PCMs was successfully conducted in which the application control layer of the PCMs was moved to the zonal system control layer. In-zone voltage regulation and sharing functions were moved to the system layer as depicted in Fig. 14.
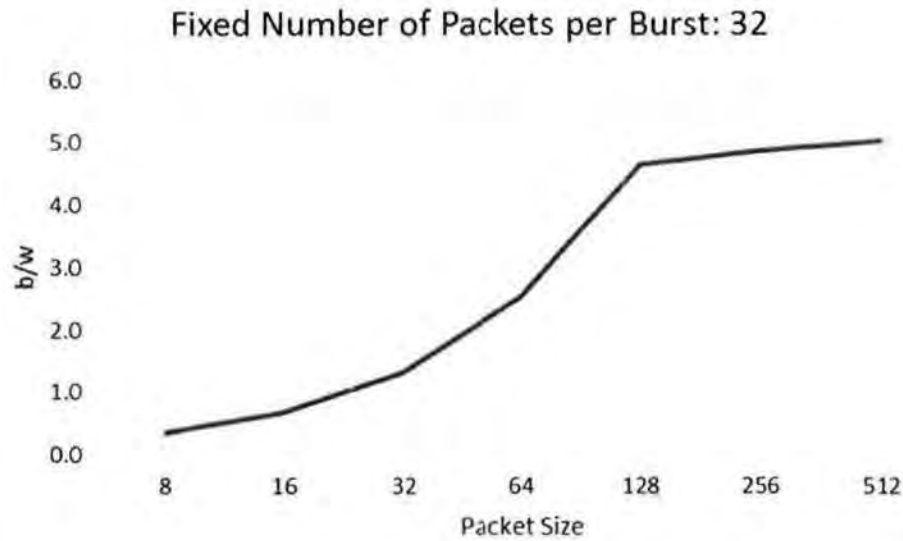
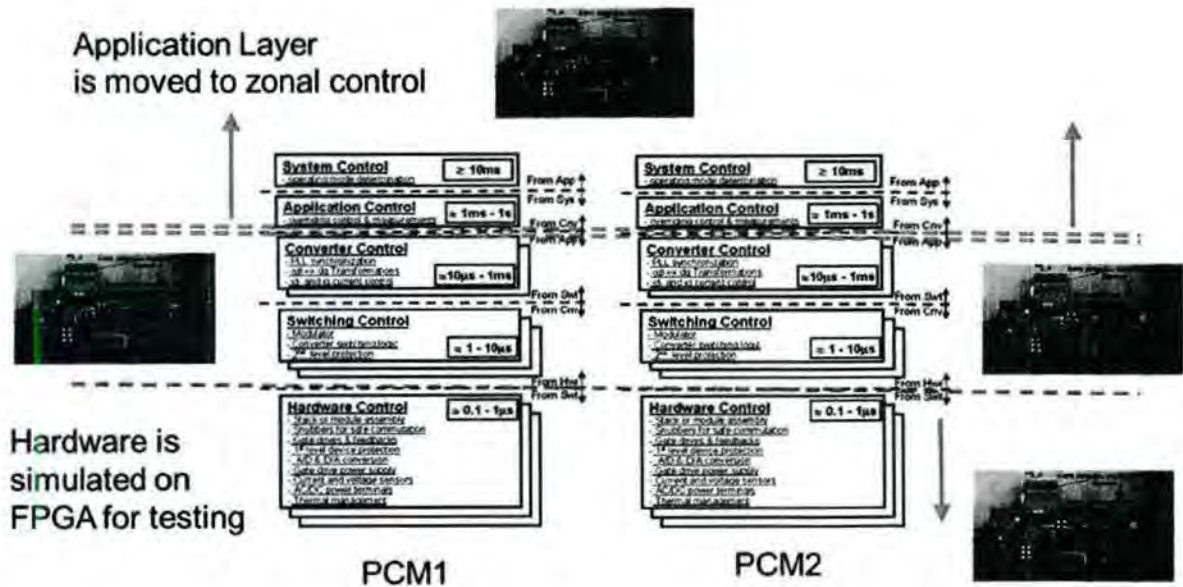Figure 13. Observed Aurora channel bandwidth versus packet size.

Figure 14. Experimental setup realizing the converter application layers as a zonal system control.

4.4 Evaluation of system to application level partitioning and communication

A study of the interface between the power electronic converter network and the uppermost ship system control network was conducted utilizing a three multiplexer node OC-48 synchronous optical network (SONET). This system is a Time Division Multiplexing Network (Alcatel-Lucent DMXtend 1665), which can be configured for dedicated or critical services in order to support real-time control applications. The multiplexer system under test supports a wide array of wideband and broadband transport, such as traditional SONET transport, Ethernet over SONET (EoS), among others. In order to test or characterize the network, different configurations and strategies were made to apply control traffic management or protection while using SONET.

Time Division Multiplexing (TDM) is a data communications method that interleaves multiple data streams over the same physical medium, giving each data stream a predefined, fixed-length time slot for using the physical medium. All sub-channels have unique time slots on the physical medium. Some of the key advantages of TDM are its guaranteed bandwidth and deterministic data delivery times. TDM systems are therefore naturally suited to support applications that stream data steadily rather than send data in irregular bursts [8] (regular Ethernet data). New systems offer both technologies resulting in packet transport solution with a native Ethernet interface, which would allow the use of the existing deployed SONET infrastructure.

SONET-based networks provide a robust communication network by providing healing properties. The ring's services can be automatically restored following a link failure or degradation in the network signal. This is done using the automatic protection switching (APS) protocol. The time to restore the services in case any disruption on the SONET specified to be less than 50 milliseconds, with proper configuration.

4.4.1 TDM Synchronization

Three different modes of synchronization for the TDM can be configured depending on its physical equipment and/or service packs (service cards active in the shelf) installed. These configurations are mentioned below. It is important to consider that this system is a synchronous communication system and it has to be well synchronized across all nodes and its time reference must be very accurate.

Although its internal generator clock is very precise (ST3), SONET is designed to operate in a network that complies with recommendations stated in GR-436-CORE, where each chain consists of one PRS-primary reference source and up to 16 SONET Minimum Clock (SMC) at the end. Fig. 15 shows a basic synchronization example, where clock reference of each node is common to a Stratum 1 clock and data transmitted between nodes may vary in time (plesiochronous), but it might be adjusted by the time it is received.
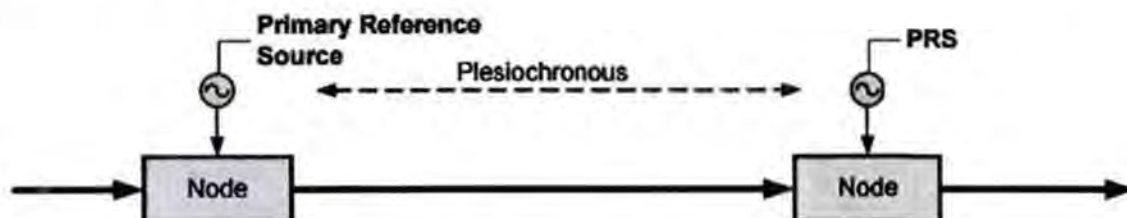

Figure 15. Recommended SONET network Synchronization

4.4.2 Configurations

Three main configurations have been tested. Private line (point-to-point), Ethernet bridging (transparent mode), and Ethernet tagging mode. Observations, advantages and main purpose are provided in more detail in Appendix A. Below key points for each configuration are highlighted.

Ethernet Private Line (point-to-point Ethernet)

The Private Line mode (also known as no tag or repeater mode) is used to establish simple point-to-point connections between two ports with no Ethernet switching functions applied. Private line mode can be used to provide either a full rate or sub-rate (fractional rate) dedicated Ethernet link across SONET networks [9]. No preferential treatment for high priority packets is provided.

802.1D Ethernet for transparent bridging

In Transparent Mode, port tags (which are actually VLAN tags with a provisionable TPID-Tag Protocol ID value) are used to separate traffic for different customers. A port tag is added to each incoming frame at the ingress LAN port. The port tag contains a provisionable customer ID and priority level to direct the Ethernet frames to its correct destination.

802.1 Q Ethernet Tag mode

802.1Q establishes a standard method for inserting virtual LAN membership information into Ethernet frames. This standard was developed to address the problem of broadcast or multicast traffic utilizing more bandwidth than necessary. In 802.1Q Mode, a circuit pack can be provisioned to use an incoming frame's VLAN tag, to add a VLAN tag associated with the port for untagged frames, or to drop an incoming frame if its VLAN tagging does not meet provisioned specifications. The priority bits in an incoming frame's VLAN tag can also be used to affect the handling of the frame. VLANs provide inherent security to the network by delivering the frames only within the destination VLANs and to specific recipient within the destination VLAN.

4.4.3 Data and Analysis

Two physical configurations are used to characterize the network, direct DSP-boards connection and the other through SONET network employing two switches at the TDM-node at the customer side ports (boundary ports). Fig. 16 is the direct connection between DSP boards through a cross-over cable. This configuration is taken as a reference measurement for characterization. Here constant data is sent from client and acquired at the server board. Latency for data transmitted is observed and recorded for further analysis. Minimum and maximum time delay for transmission as well as the jitter is captured in real-time while data is flowing – frames are 70 byte length with 16 bytes of data over TCP protocol.
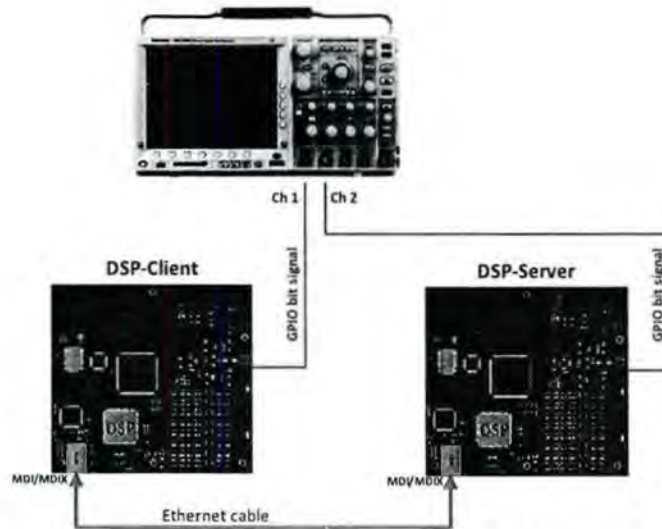
Figure 16. Direct communication between DSP for latency observation

Figure 17 shows the scheme with the DUT-devices under test connected on a peripheral port 1 at Node 1 and Node 2. This connectivity is employed on every test to observe the latency of the system when the three configurations are provisioned.
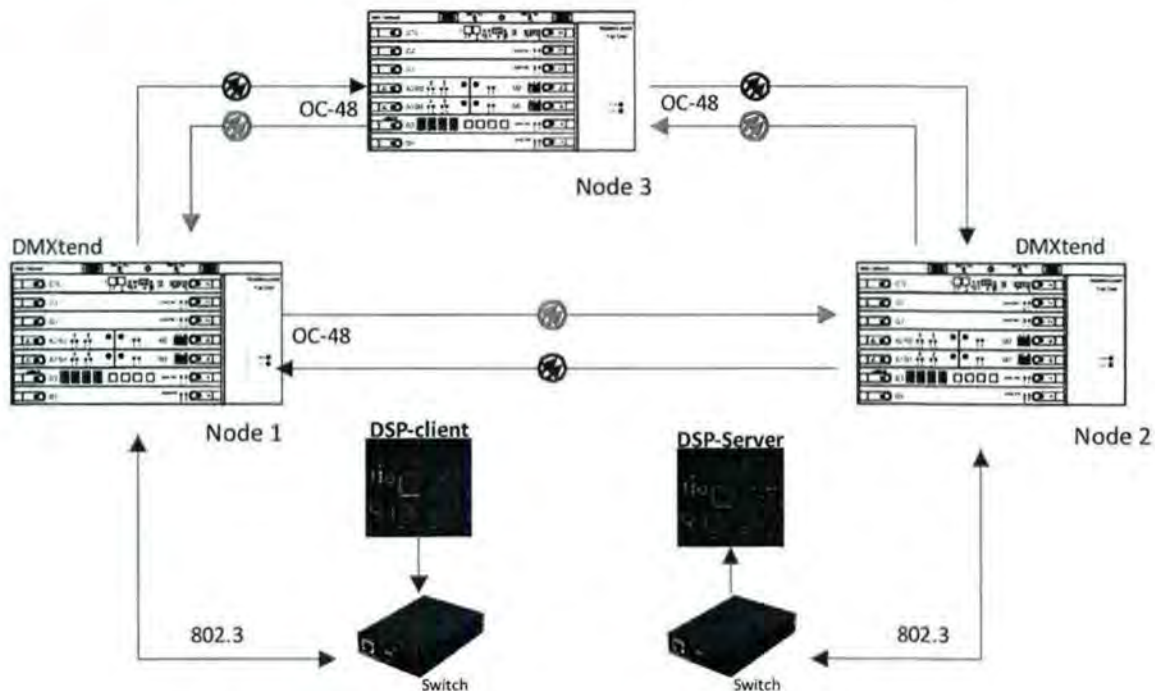

Figure 17. Main test organization. DSP boards over SONET network with Ethernet switches

In Table 4 the first two rows show the latency and jitter acquired using the direct connection. Analyzing the table information and metering the capability of each configuration it is possible to state that configuration 802.1Q and Private Lines Non-switched are the two best options for handling critical OT-operational technology information, or real-time control information. Out of those two the *802.1TAG or 802.1Q* configuration is considered as the most convenient because it has traffic management control, quality of service for communication improvements, and complies with

security. It also provides one of the lowest time delays compared with the other switched modes. It is also considered as a replacement for traditional private lines because of bandwidth scalability, increment or decrement in bandwidth, as the network requirements change. Private lines non-switched, can be also considered because it is stated that for private networks bandwidth is dedicated and the performance will not be affected negatively by other traffic. However, no modifications in bandwidth and the number of services are accepted once it is configured. Another drawback for this arrangement is that the delay and jitter computed is slightly different than the observed when Switched Mode is deployed.

The only configuration that has lower delay and jitter than the preferred two modes identified above is the 802.1W or Ethernet link with a spanning tree protection, but this configuration is not considered because its protection depends on configuring a spanning tree and high priority paths selected for automatic restoration.

The data frame employed in the communication test is 70 bytes long, it has a header length of 20 bytes for Internet Protocol version 4 (IPV4) for a total of 54 bytes and a TCP segment length of 16 bytes data. Test configuration is made through a Unidirectional Protected Switched Route (UPSR) with a maximum bandwidth of 2.5Gbps with the capability of being segregated for different transfer rates over SONET on STS-1 tributaries (51.84 Mbps, which payload is roughly 49.5 Mbps). Actual transfer rate is set to two STS-1 to obtain an average speed of 100Mbps on channel 1 at Node 1 and Node 2. Node 3 is working as passing through with full bandwidth enabled (48 tributaries or 2.5Gbps) and no traffic management.

Table 4: Observed latency and Jitter

Service Connection	Traffic Management mode	Delay (μ seconds)		Jitter (μ seconds)
		Minimum	Maximum	
Direct connection (DPS-DSP)	N/A	15.61	16.06	0.45
DSP-Sw1 to Sw2-DSP	N/A	29.51	30.7	1.19
Private Line non-Switched	NOTC[1]	77.49	185	107.51
	PORT[2]	77.5	186.1	108.6
Transparent (Switched)	NOTC[1]	95.53	200.4	104.87
	PORT[2]	96.08	201.5	105.42
802.1TAG or 802.1 Q	NOTC[1]	82.23	187.1	104.83
802.1W Spanning tree Prot.	NOTC[1]	76.96	184.0	107.04
Multiple point Private Line	NOTC[1]	203.2	382.3	179.1

N/A Non applicable
[1] No traffic control at the customer or boundary port
[2] Traffic management is handled in the same way it is received at the incoming port, single conditioner

Figure 18 depicts the DSP-boards for SONET network test where these boards are sending information at a constant rate and known data every 30 milliseconds for an undefined time.
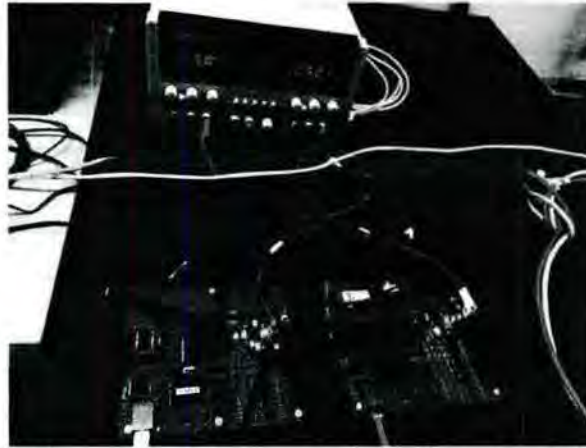
Figure 18. Communication boards, TCP- Client and TCP-Server

Figure 19 shows the equipment under test, on the left side one can observe the client and server boards which both are connected through the SONET network. On the right side, there are three Time Division Multiplexers to build the synchronous optical network.
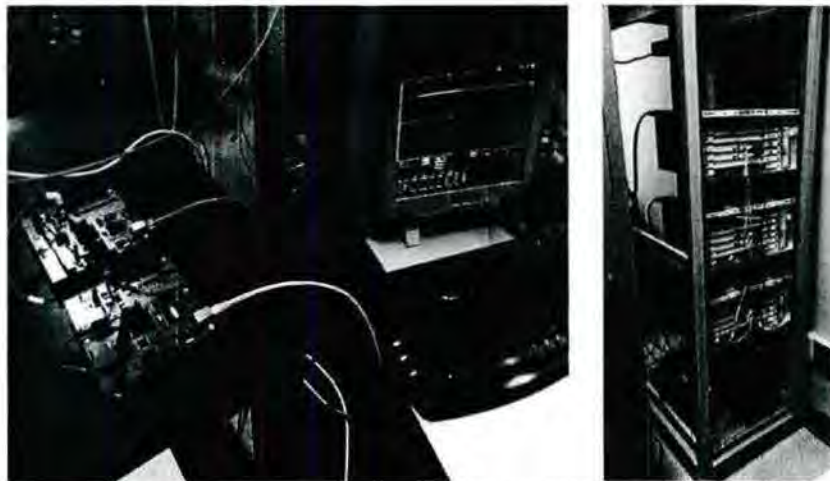

Figure 19. Communication set up for network characterization

Analysis of Jitter is performed considering three different scenarios. First, direct connection between DSP-boards. The second test employs the SONET network with a private line non-switched configuration. Third, SONET connection and configuring the multiplexers with 802.1Q or Tagging mode structure and default Tag for inbound untagged traffic. Each test is observed and recorded with ten thousand samples at 500KS/s. A DPOJET tool [9] available on the oscilloscope is used to measure the time interval error (TIE). The measurement tool measures the time difference between the reference signal edge and the signal edge. The measured difference is the time interval error. After measuring a significant samples of time intervals and their related timing errors, the standard deviation and peak values can be resolved. This statistical information is the TIE "jitter". TIE is very useful, especially when looking for real-time communication, because it maintains a record of error versus time [10]. From this record, accumulated phase error measurements are possible.

Figure 20 shows latency for direct connection between DSP boards, here also a histogram and standard deviation are configured in order to acquire Time Interval Error.

23

Figure 20. Latency over channel 1 for direct configuration (DSP to DSP)

From Fig. 21 we can observe the actual deviation (accumulated Jitter) from the ideal reference signal over all periods. The histogram present an accumulated incidence of errors [11] (Gaussian distribution) that are very close to zero microseconds, while other spread values range between 0.04 and 0.9 micro seconds.
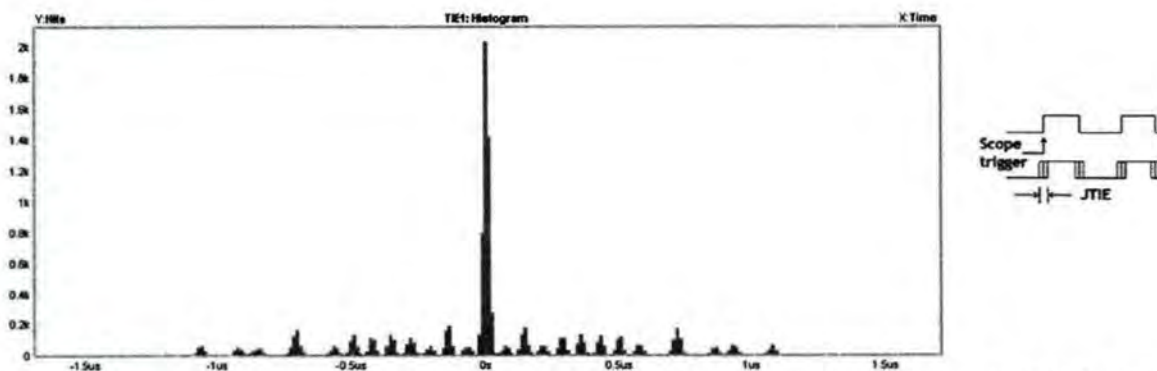

Figure 21. Time Interval Error Histogram when DUT is direct connected (DSP-DSP)

When packets are sent over SONET network two occurrences are observed. One is the increment in latency introduced by the multiplexed communication which uses a STS-1 signal. Second, there is a displacement in time caused by the justification pointer or synchronization issues. This pointer is an adjustment mechanism to accommodate expected phase differences between the SONET and the virtual tributaries. This issue exposes a constant displacement time of the received signal that increases from 87 microseconds to 187 microseconds. Figure 22 shows the delay for an 802.1Q mode configuration on the SONET nodes. Time interval error is arranged in this test with the aim of the utility tool on the scope to quantify the error deviation (Jitter).

Figure 22. Latency over channel 1 for 802.1Q mode connection

Figure 23 shows the histogram when information is sent over SONET using 802.1Q configuration. The Time interval Error (Jitter) is recorded over a span of 10,000 samples. Error is significantly high when closed to 5 microseconds. In the same manner, the acquired histogram of the system sets the peak-to-peak amplitude of the time interval error near to 20 microseconds. Figure 24 shows the histogram for the time interval error observed when private line non-switched is configured. It is possible to observe that the error distribution is close to the one acquired on Tag mode, but some other significant peaks are observed at 5 microseconds. Peak-to peak jitter is determined between 20 microseconds.
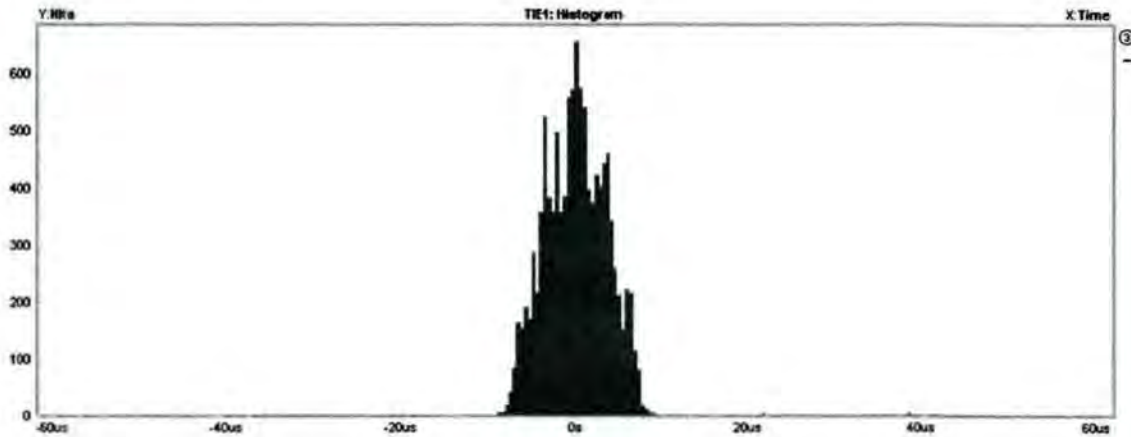

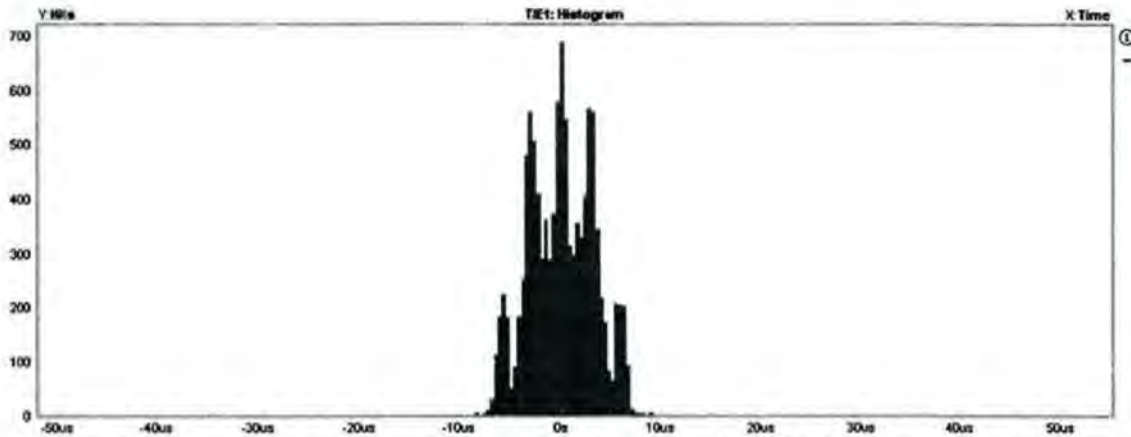Figure 23. Time Interval Error Histogram when 802.1 TAG mode connection is employed

Figure 24. Time Interval Error Histogram when Non-Switched Private Line connection is configured

The slip in time. Is shown in Fig. 25 and initial time is observed at 84 microseconds, right after the recorded delay, then a maximum time is captured at 186 microseconds.

The SONET network was broken in a single point by disconnection of a main fiber optic cable numerous times in various places to test the healing (restoration) time of the network for both 802.1Q and Ethernet Private Line configurations. Figure 26 shows the healing time of the network as compared to the maximum specified 50ms. The private line configuration in most cases provided faster restoration of a network path.

Considering the results of latency and advantages tested for different configurations, Ethernet Private Lines were configured to provide dedicated bandwidth (with or without

protection), QoS, and security for critical data transport applications. When private lines are configured SONET layer protection switching guaranteed a lower restoration time. On the other hand, 802.1Q configuration has noticeable advantages including not only shared networks and dedicated LANs, but also traffic management. Making it possible to avoid any traffic disruption and deviation to incorrect destination and SONET protection. Finally, to suppress the frame slip it is recommended to provide a primary reference source to synchronize every multiplexer on the SONET network, this reference should have a better quality in the range of stratum 2 (ST2) or higher since a master-slave synchronization system has a single primary reference clock to which all other clocks are phase-locked, within a hierarchical structure.
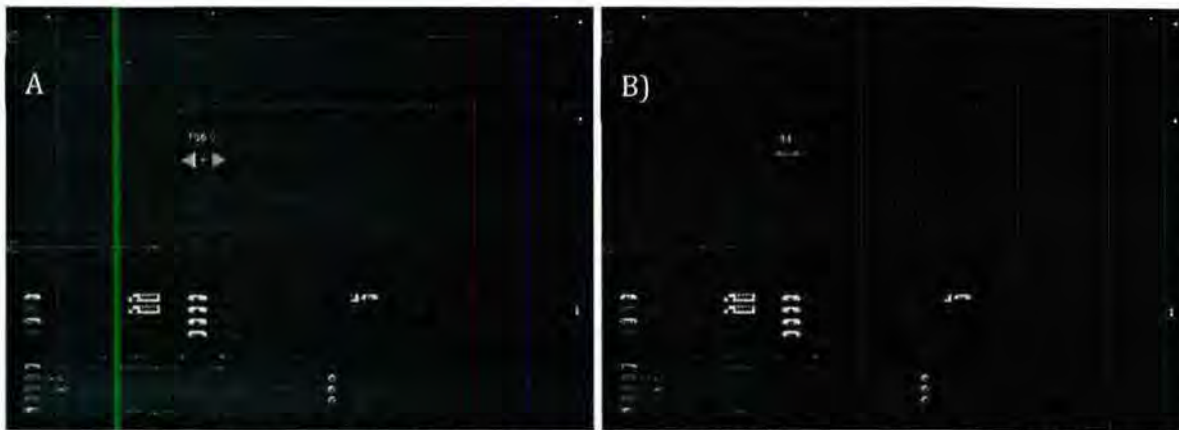

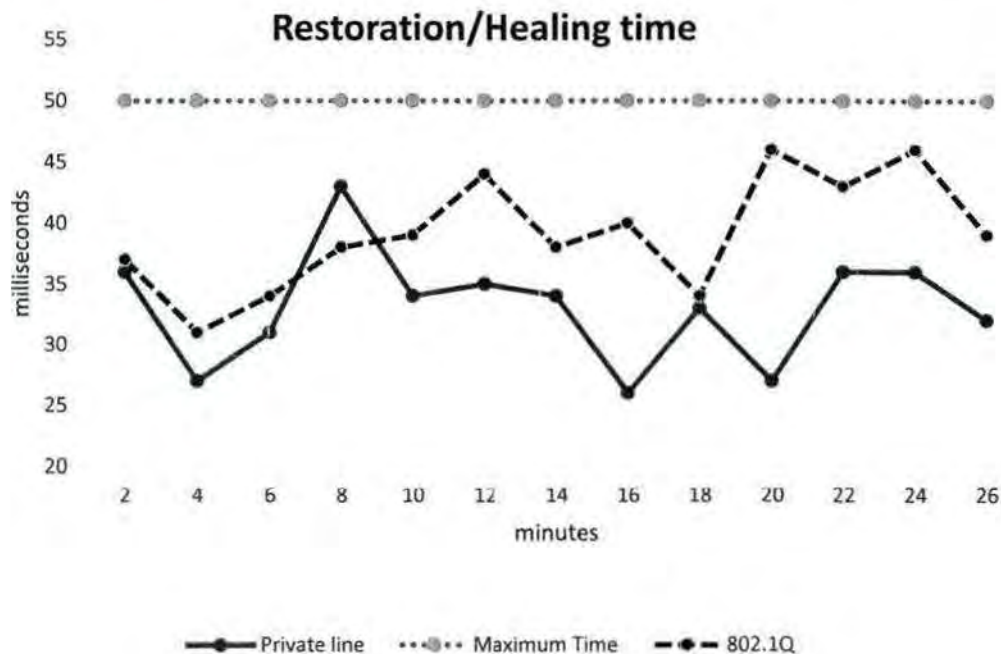Figure 25. Signal analysis: A) Maximum delay and B) Minimum delay over SONET network

Figure 26. SONET healing time for the two most promising configurations.

5. Productivity

A) Refereed Journal Articles:
- M.R. Hossain and H.L. Ginn III, "Real-Time Distriubed Coordination of Power Electronic Converters in a DC Shipboard Distribution System," IEEE Transactions on Energy Conversion, Vol. 32, No. 2, June 2017, pp. 770-778.
- Ginn, H.L., Hingorani, N., Sullivan, J.R., Wachal, R., "Control Architecture for High Power Electronics Converters", Proceedings of the IEEE, Vol. 103, No. 12, Dec. 2015, pp. 2312-2319.

B) Non-Refereed Significant Publications: None

C) Books or Chapters: None

D) Technical Reports: None

E) Workshops and Conferences:
- Ginn, H.L. Bakos J, Panchenko I, "Control system communication architecture for power electronic building blocks", 2017 IEEE Electric Ship Technologies Symposium (ESTS), Aug. 2017, pp. 544 – 550.
- Ginn, H.L. Bakos J, De La O, A., Panchenko I., "Control System Communication Architecture for Power Electronic Building Blocks", ONR Control Workshop, Office of Naval Research, Philadelphia PA, Aug. 2017.
- Ginn, H.L. Bakos J., "Development of Universal Controller Architecture for SiC Based Power Electronic Building Blocks", ONR Control Workshop, Office of Naval Research, Columbia SC, Feb. 2017.
- Ginn, H.L., "The Universal Controller Architecture and System Level Converter Coordination", ONR Control Workshop, Office of Naval Research, Tallahassee FL, Nov. 2015.
- Ginn, H.L., "Example of Controller Implementation for MMC Converters", Workshop on Control Architectures for Modular Power Conversion Systems, Office of Naval Research, Arlington VA, April 2015.
- Ginn, H.L., "Control Partition and Recommended Architecture", Workshop on Control Architectures for Modular Power Conversion Systems, Office of Naval Research, Arlington VA, April 2015.

F) Patents: None

G) Awards/Honors: None

6. Award Participants

Participants	Male non-minority	Female non-minority	Male minority	Female minority
PIs	2	0		
Undergrad Students	1	1		
Grad Students	3			
Staff				

7. Works Cited

[1] W. Dally and B. Towles. Principles and Practices of Interconnection Networks. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2003.

[2] N. Kapre and J. Gray, "Hoplite: Building austere overlay NoCs for FPGAs," Proc. 25th International Conference on Field Programmable Logic and Applications (FPL).

[3] Nachiket Kapre, "Implementing FPGA Overlay NoCs Using the Xilinx UltraScale Memory Cascades," Proc. 25th IEEE International Symposium on Field-Programmable Custom Computing Machines, 2017.

[4] Arjun Singh, William J. Dally, Amit K. Gupta, Brian Towles, "GOAL: a load-balanced adaptive routing algorithm for torus networks," Proceedings of the 30th annual international symposium on Computer architecture 2003.

[5] Trevor Bunker, Steven Swanson, "Latency-Optimized Networks for Clustering FPGAs," Proc. 21st Annual International IEEE Symposium on Field-Programmable Custom Computing Machines, 2013.

[6] Andrew G. Schmidt, William V. Kritikos, Rahul R. Sharma, Ron Sass, "AIREN: A Novel Integration of On-Chip and Off-Chip FPGA Networks," Proc. 17th IEEE Symposium on Field Programmable Custom Computing Machines, 2009.

[7] Y. Song, Y. Wang, P. Bull and J. D. Reiss, "Performance Evaluation of a New Flexible Time Division Multiplexing Protocol on Mixed Traffic Types," in Advanced Information Networking and Applications (AINA), Taipei, Taiwan, 2017.

[8] A. Mazzeo, C. Iliopoulos and R. Kline, "Performance characteristics for delivering Ethernet private line service in a multi-vendor SONET environment," in Optical Fiber Communication Conference, and National Fiber Optic Engineers Conference. , Anaheim, CA, 2006.

[9] Tektronix, "DPOJET, Jitter and Eye Diagram Analysis Tools," Tektronix, Inc., 04 2013. [Online]. Available: http://www.av.it.pt/medidas/data/Manuais%20&%20Tutoriais/18%20-%20Real%20Time%20Oscilloscope%2020Ghz/CD2/Documents/DPOJET.pdf.

[10] M. Miller and M. Schnecker, "A Comparison of Methods for Estimating Total Jitter Concerning Precision, Accuracy and Robustness," LeCroy Corporation, Chestnut Ridge, NY, 2007.

[11] K. Mochizuki and K. I. , "Data-Dependent Effects on Jitter Measurement," in Optical Fiber Communication and the National Fiber Optic Engineers Conference, Anaheim, CA, 2007.

[12] Alcatel-Lucent, "Applications and Planning Guide, Issue 2, release 10.0," Nokia, 2013.

[13] Alcatel-Lucent, "User Operations Guide, Issue2, Release 10.0," Nokia, 2013.