Exploring Mellanox Bluefield SmartNICs as Accelerators for Heterogeneous Architectures

Brody Williams*, Liliana Aguirre-Esparza†, Wendy Poole‡, Stephen Poole‡
*Texas Tech University †New Mexico State University ‡Los Alamos National Laboratory

Motivation

Limited advances in processor technology in recent years have forced researchers to explore alternative techniques to provide continued system performance improvements and facilitate further scaling. Many resulting approaches have shifted away from the strictly CPU-centric approach used in the past in favor of more heterogeneous architectures. These architectures often employ added computational components to support offloading computation from the CPU. Frequently, these components are also paired with their own local memory in order to minimize performance degradations associated with data movement.

In this work, we propose an extension to the heterogenous architecture paradigm using Mellanox Bluefield SmartNICs. These devices combine state of the art network controllers together with 16 ARM cores into a device that provides unique potential. Herein, we explore the feasibility of utilizing these SmartNICs as accelerators capable of offloading both communication routines, as well as computational kernels, from the CPU.

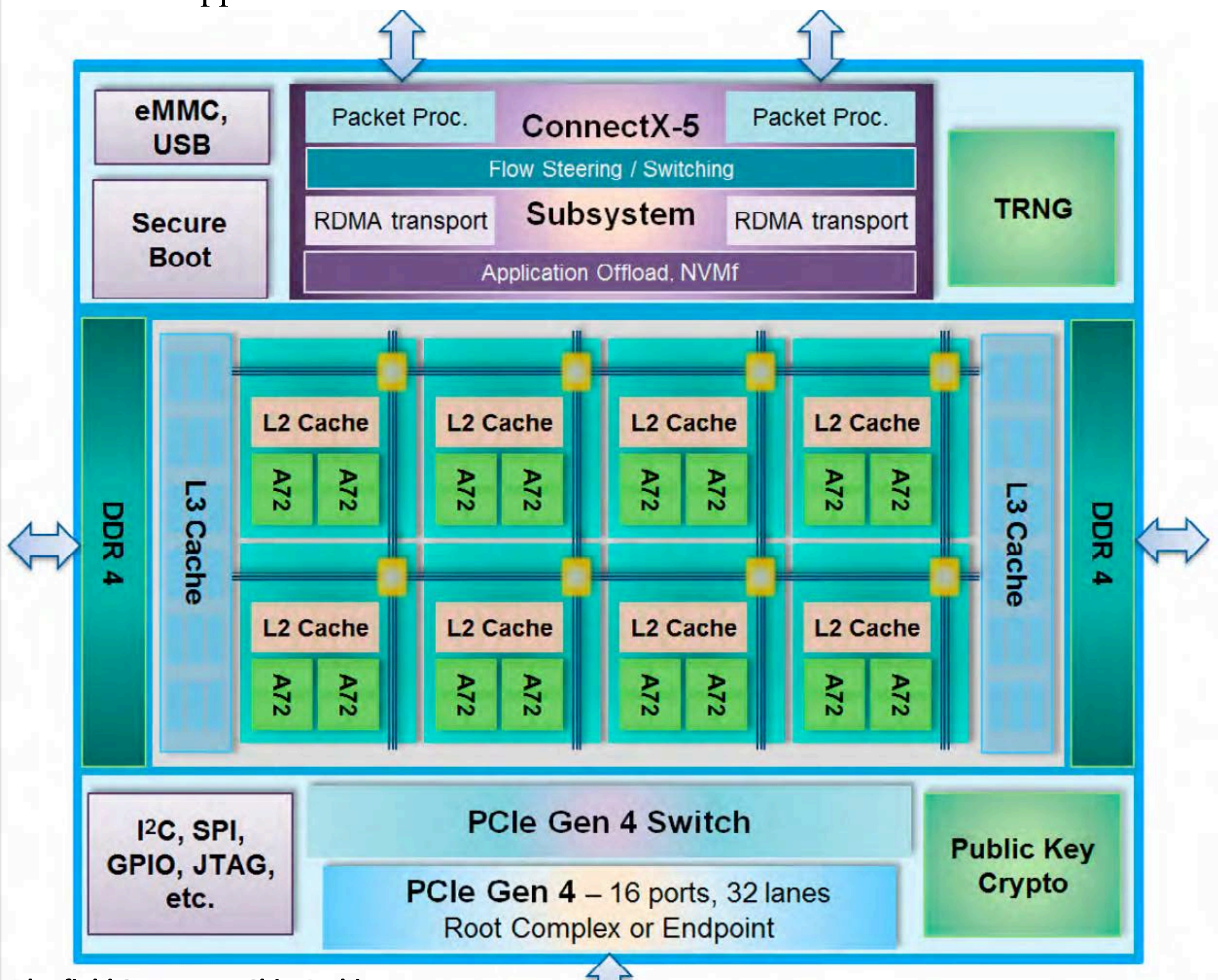
Bluefield System on Chip Architecture

16 ARMv8 Cortex-A72 Cores
- Three-level cache hierarchy
- SkyMesh coherent interconnect
- 128b ARM Neon SIMD execution unit per core

Connect X-5 Subsystem
- Dual Virtual Protocol Interconnect (VPI) ports
- Ethernet/Infiniband at 100Gbps per port
- RDMA & NVMe-oF support

Integrated PCIe Switch
- 32 bifurcated PCI 4.0 lanes
  - Configurable as 2x16/4x8/8x4/16x2
  - Speeds up to 200Gbps

Memory Controllers
- Supports two channels of 256 GB DDR4 DRAM at 1333MHz



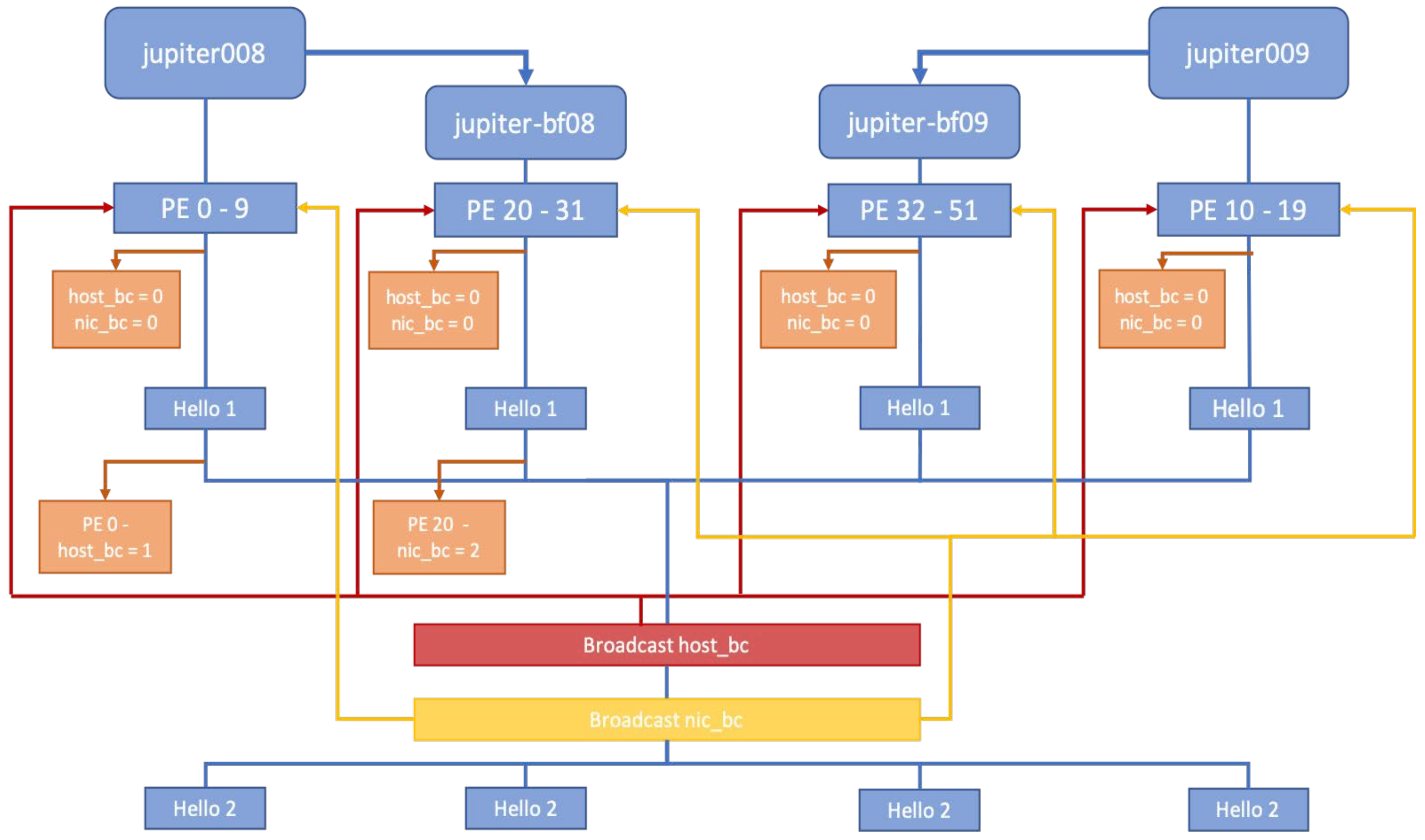
Bluefield System on Chip Architecture.
http://www.mellanox.com/related-docs/npu-multicore-processors/PB_Bluefield_SoC.pdf

Communication Experiment

In order for the SmartNICs to perform effectively as accelerators at any useful scale, they need to be able to communicate with other devices. Therefore, as a necessary prerequisite to any application performance optimization attempt, we first conducted an experiment to determine compatibility between the ARM cores onboard each SmartNIC and prominent distributed-address space programming paradigms.

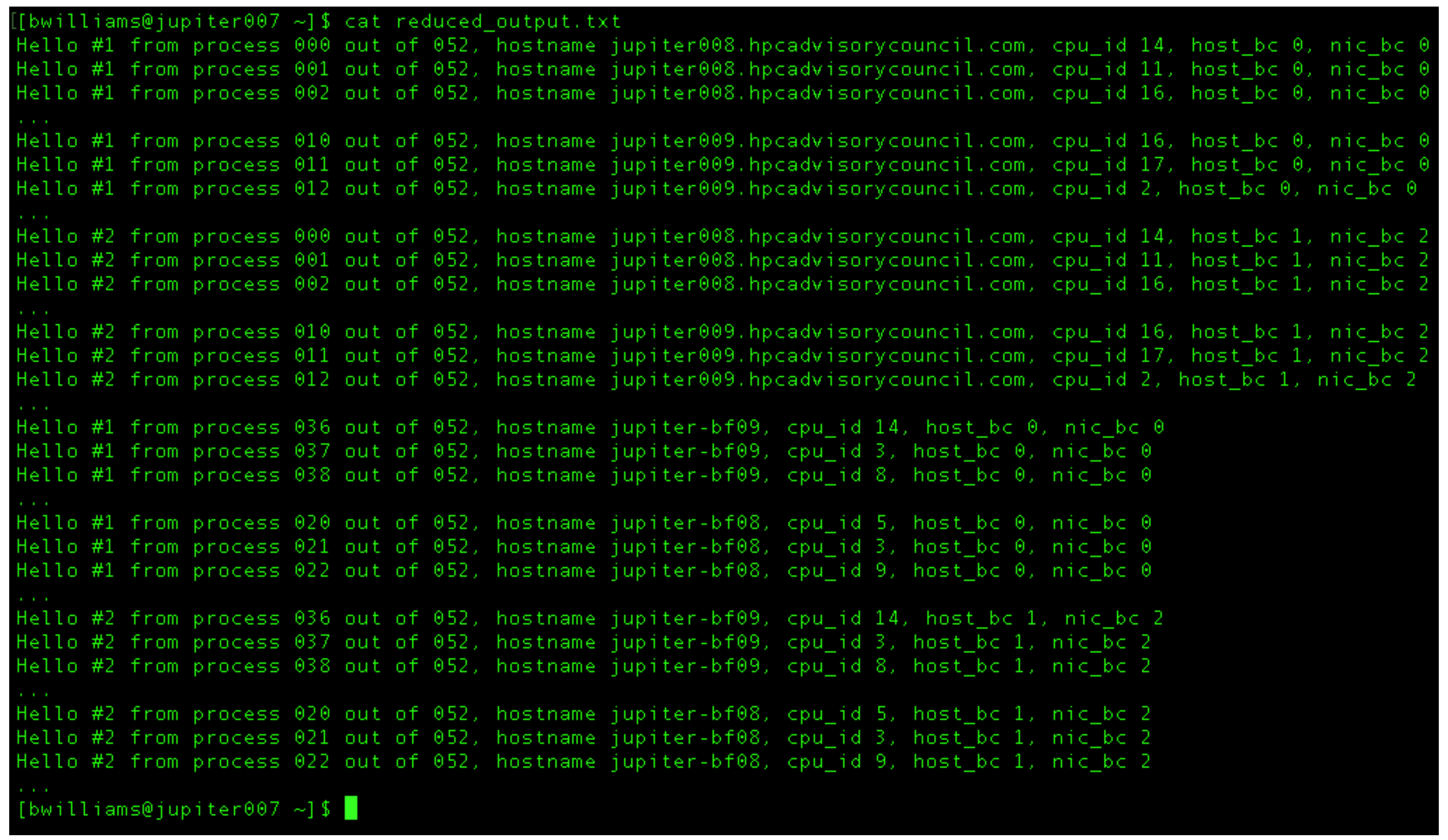
Mellanox Testbed – jupiter007 - jupiter010

- Intel Xeon E5-2680 v2 10-core processors
- 64 GB of memory
- Paired Bluefield SmartNIC with 16GB onboard memory
- CentOS 7
- OpenMPI 4.0.1 with Unified Communication X (UCX) 1.6



Experiment Program Flow

Output from our experiment shows that processor cores are properly utilized across multiple nodes and SmartNICs. Further, correct broadcast variable values demonstrate that proper inter-device communication and synchronization takes place despite buffered print statements. (Note that the output has been simplified for presentation purposes.)



```
[bwilliams@jupiter007 ~]$ cat reduced_output.txt
Hello #1 from process 000 out of 052, hostname jupiter008.hpcadvisorycouncil.com, cpu_id 14, host_bc 0, nic_bc 0
Hello #1 from process 001 out of 052, hostname jupiter008.hpcadvisorycouncil.com, cpu_id 11, host_bc 0, nic_bc 0
Hello #1 from process 002 out of 052, hostname jupiter008.hpcadvisorycouncil.com, cpu_id 16, host_bc 0, nic_bc 0
...
Hello #1 from process 010 out of 052, hostname jupiter009.hpcadvisorycouncil.com, cpu_id 16, host_bc 0, nic_bc 0
Hello #1 from process 011 out of 052, hostname jupiter009.hpcadvisorycouncil.com, cpu_id 17, host_bc 0, nic_bc 0
Hello #1 from process 012 out of 052, hostname jupiter009.hpcadvisorycouncil.com, cpu_id 2, host_bc 0, nic_bc 0
...
Hello #2 from process 000 out of 052, hostname jupiter008.hpcadvisorycouncil.com, cpu_id 14, host_bc 1, nic_bc 2
Hello #2 from process 001 out of 052, hostname jupiter008.hpcadvisorycouncil.com, cpu_id 11, host_bc 1, nic_bc 2
Hello #2 from process 002 out of 052, hostname jupiter008.hpcadvisorycouncil.com, cpu_id 16, host_bc 1, nic_bc 2
...
Hello #2 from process 010 out of 052, hostname jupiter009.hpcadvisorycouncil.com, cpu_id 16, host_bc 1, nic_bc 2
Hello #2 from process 011 out of 052, hostname jupiter009.hpcadvisorycouncil.com, cpu_id 17, host_bc 1, nic_bc 2
Hello #2 from process 012 out of 052, hostname jupiter009.hpcadvisorycouncil.com, cpu_id 2, host_bc 1, nic_bc 2
...
Hello #1 from process 036 out of 052, hostname jupiter-bf09, cpu_id 14, host_bc 0, nic_bc 0
Hello #1 from process 037 out of 052, hostname jupiter-bf09, cpu_id 3, host_bc 0, nic_bc 0
Hello #1 from process 038 out of 052, hostname jupiter-bf09, cpu_id 8, host_bc 0, nic_bc 0
...
Hello #1 from process 020 out of 052, hostname jupiter-bf08, cpu_id 5, host_bc 0, nic_bc 0
Hello #1 from process 021 out of 052, hostname jupiter-bf08, cpu_id 3, host_bc 0, nic_bc 0
Hello #1 from process 022 out of 052, hostname jupiter-bf08, cpu_id 9, host_bc 0, nic_bc 0
...
Hello #2 from process 036 out of 052, hostname jupiter-bf09, cpu_id 14, host_bc 1, nic_bc 2
Hello #2 from process 037 out of 052, hostname jupiter-bf09, cpu_id 3, host_bc 1, nic_bc 2
Hello #2 from process 038 out of 052, hostname jupiter-bf09, cpu_id 8, host_bc 1, nic_bc 2
...
Hello #2 from process 020 out of 052, hostname jupiter-bf08, cpu_id 5, host_bc 1, nic_bc 2
Hello #2 from process 021 out of 052, hostname jupiter-bf08, cpu_id 3, host_bc 1, nic_bc 2
Hello #2 from process 022 out of 052, hostname jupiter-bf08, cpu_id 9, host_bc 1, nic_bc 2
...
[bwilliams@jupiter007 ~]$
```

Output Screenshot

Acceleration Opportunities

Asynchronous Execution of Task-Parallel Code Segments
- Map host and SmartNIC cores to orthogonal tasks

Algorithm 1: Sample Execution on Host

```
for $i \leftarrow 0$ to $N$ do
    Host_Proc(Compute($i$))
    Host_Proc(Verify($i$))
end
```

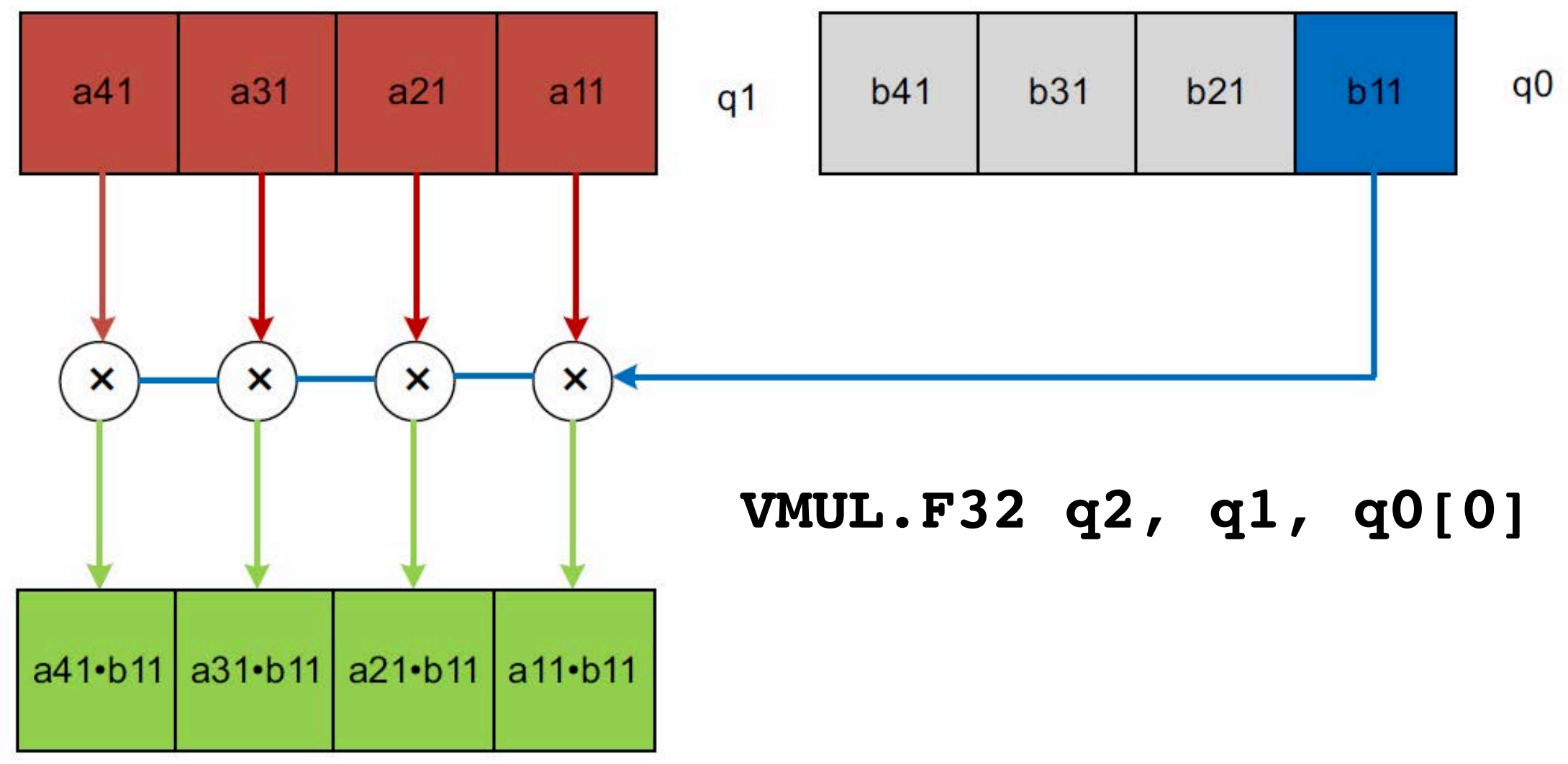
Algorithm 2: Sample Execution on Host + SmartNIC

```
for $i \leftarrow 0$ to $N + 1$ do
    if Host_Proc then
        if $i \neq (N + 1)$ then
            Host_Proc(Compute($i$))
    else if SmartNIC_Proc then
        if $i \neq 0$ then
            SmartNIC_Proc(Verify($i - 1$))
end
```

Vectorization of SIMD Operations using Neon Units
- Individual Neon unit per ARM core prevents resource contention
- Similar to acceleration using GPUs

$$\begin{bmatrix} a11 & a12 & a13 & a14 \\ a21 & a22 & a23 & a24 \\ a31 & a32 & a33 & a34 \\ a41 & a42 & a43 & a44 \end{bmatrix} \bullet \begin{bmatrix} b11 & b12 & b13 & b14 \\ b21 & b22 & b23 & b24 \\ b31 & b32 & b33 & b34 \\ b41 & b42 & b43 & b44 \end{bmatrix} = \begin{bmatrix} a11 \bullet b11 + a12 \bullet b21 + a13 \bullet b31 + a14 \bullet b41 & \cdots & \cdots & \cdots \\ a21 \bullet b11 + a22 \bullet b21 + a23 \bullet b31 + a24 \bullet b41 & \cdots & \cdots & \cdots \\ a31 \bullet b11 + a32 \bullet b21 + a33 \bullet b31 + a34 \bullet b41 & \cdots & \cdots & \cdots \\ a41 \bullet b11 + a42 \bullet b21 + a43 \bullet b31 + a44 \bullet b41 & \cdots & \cdots & \cdots \end{bmatrix}$$

Matrix Multiplication Example



`VMUL.F32 q2, q1, q0[0]`

http://infocenter.arm.com/help/topic/com.arm.doc.dui0489f/DUI0489F_arm_assembler_reference.pdf

Inter-Process Communication Calls
- Offload routines to SmartNIC cores
- Prevent blocking of host CPU cores
- Perform buffering and collective computation locally on SmartNICs

Future Work

Our preliminary work for this project indicates that acceleration using Mellanox Bluefield SmartNICs is feasible. Further, our communication experiment demonstrates that the SmartNICs are fully compatible with the already widely adopted OpenSHMEM and MPI standards. Our future work will focus on determining whether or not these SmartNICs perform effectively as accelerators, and, if so, how best to optimize and deploy code for SmartNIC acceleration.

In particular, we plan to first optimize several provided Department of Defense benchmarks using the methods discussed above. We are also interested in investigating the use of active messages to pass fully dependence-free code segments to the SmartNICs for execution. Finally, when more familiar with writing code for SmartNIC acceleration, we plan to explore developing a library that abstracts SmarNIC acceleration away from the software developer while providing optimal performance.

Acknowledgements

We would like to thank the Department of Defense for supporting this project. We are also grateful to Mellanox Technologies, Parks Fields, and John Snyder of Duke University for their collaborative efforts.