Photonic Many-Core Architecture Study

Nadya T. Bliss[1], Krste Asanovic[2], Keren Bergman[3], Luca Carloni[3], Jeremy Kepner[1], Vladimir Stojanovic[4]
[1]{nt, kepner}@ll.mit.edu
MIT Lincoln Laboratory, Lexington, MA 02420
[2]krste@eecs.berkeley.edu
University of California at Berkeley, Berkeley, CA 94720
[3]{bergman@ee, luca@cs}.columbia.edu
Columbia University, New York, NY 10027
[4]vlada@mit.edu
MIT Research Laboratory of Electronics, Cambridge, MA 02139

Abstract

Several recent device technology developments have been fundamentally changing the microprocessor architecture design space. These developments include photonic interconnects, feature size reduction, 3D fabrication, and aggressive energy management. These technologies create a large parameter space of possible future architectures. The focus of this talk and research effort is to demonstrate a set of efficient architecture parameters that yield good performance for DoD relevant applications. To achieve this goal, two key challenges must be addressed. First, an expressive logical abstraction of the hardware needs to be developed. This abstraction or machine model can then be parameterized given hardware requirements and capabilities. Second, detailed modeling at the application level has to be performed. The modeling engine must account for mapping representative application kernels onto the parameterized architecture. Additionally, the complexity of the mapping yielding the best performance allows for programmability assessment of the architecture. This talk presents solutions to the two challenges and preliminary recommendations of the architecture study.

Introduction

Over the last few years, photonic on- and off-chip interconnect feasibility has been demonstrated. This technology can provide high-bandwidth, low-power communication fabric [1]. Additionally, a new DARPA program is developing a hardware architecture utilizing this technology [2].

The photonic interconnect technology along with other recent advancements create a large parameter space of possible architecture specifications. In order to evaluate this parameter space, it is necessary to narrow down the set of possible hardware configurations to a particular system view. The system view used for the research effort discussed here is illustrated in Figure 1.

The system design was chosen based on DoD application requirements for memory, processing, and power. The architecture under study is limited to 500 Watts, which makes it suitable for medium-sized unmanned aerial vehicles (UAVs). The number of processing cores greater than 64 was selected in order to exercise both communication scaling and programmability. The memory requirement of at least 128GB was derived based on representative and near term expected sensor data sizes.
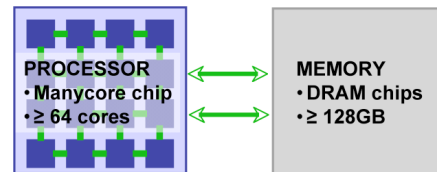

Figure 1: System level view. The architecture study assumes a manycore chip processor with greater than 64 cores, off-chip memory of at least 128 GB, and a power limit of 500 Watts.

While the system level architecture constrains the design space, large number of both system and device parameters have to be analyzed and modeled. Additionally, a detailed modeling capability at the application level is necessary to quantify the performance benefits of optical over electrical interconnects. Both of these requirements motivate the need for an abstraction between hardware and logical views of the architecture. The next section reviews the Kuck diagram representation of hardware architectures along with a few possible Kuck diagrams of the board design illustrated in Figure 1. The following section discusses the modeling engine and presents a sample result.

Logical Abstraction: Kuck Diagram

The Kuck diagram notation [3] provides a clear way of describing a hardware architecture along with the memory and communication hierarchy. Figure 2 is an example of a 2-level hierarchy.
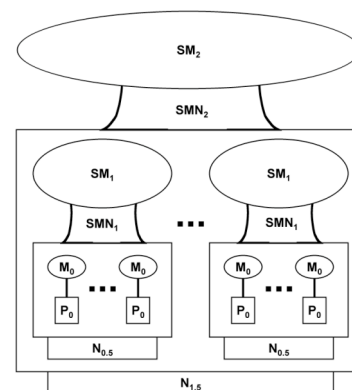

Figure 2: Kuck Diagram. 2-level hierarchy [3].

Let us walk through the diagram in detail. Processing cores are indicated by the letter P. The subscript 0 indicates that the processors are at the 0th level of the hierarchy. There is an implicit superscript that ranges from 0 to the number of processing cores in a hierarchy level. The letter M stands for memory with the S signifying shared memory. In Figure 2, M_0 describes the local memory of each processor (such as cache or local store), while SM_1 describes shared memory between processors. Similarly, N stands for network and SMN for shared memory network. Subscripts that end in .5 indicate that the memory access has to occur indirectly, via message passing or a similar approach. For example, a processor P_0 would have to go over $N_{0.5}$ to access another processor's local memory, M_0.

Kuck diagrams of two possible options of memory organization for the reference architecture design are illustrated in Figure 3.
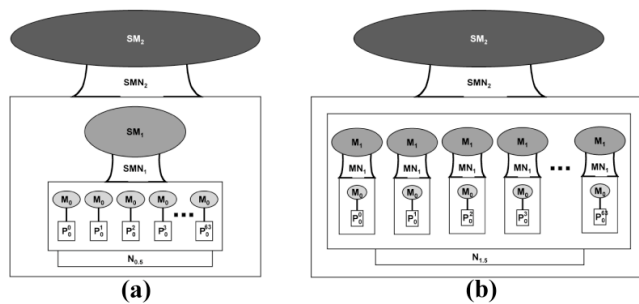
Figure 3: Kuck diagrams of the reference architecture. (a) represents a uniform memory access to on-chip memory, while (b) represents non-uniform memory access.

Modeling and Mapping

The Kuck diagram abstraction defines an interface between hardware parameters and logical architecture description necessary for application modeling. Once a family of Kuck diagrams is defined, such as the family of UMA and NUMA on-chip hierarchies, the MIT Lincoln Laboratory mapping and modeling engine [4,5] is used for mapping and simulation of application kernels running on the architecture. The Kuck diagram parameters can trivially be translated into machine model parameters used by the mapping technology. The mapping engine takes as an input an application specification and performs runtime code analysis. The result of the code analysis is a dependency graph. Given an underlying machine model derived from the Kuck diagram and a dependency graph, a set of maps is generated. The maps define how the application code is distributed on the architecture. Finally, a simulation capability computes a performance estimate.

In addition to assessing throughput and latency, the output of the mapping and modeling can be used to reason about programmability of the underlying hardware. If the mapping found is complex, then that indicates low programmability.

The parameterized architecture is evaluated against the HPEC challenge benchmarks [6] and an image processing application. A sample result of a simulation of the image

processing application is shown in Figure 4. The performance surface indicates that there is an advantage to using photonic interconnects for both on- and off-chip communication.
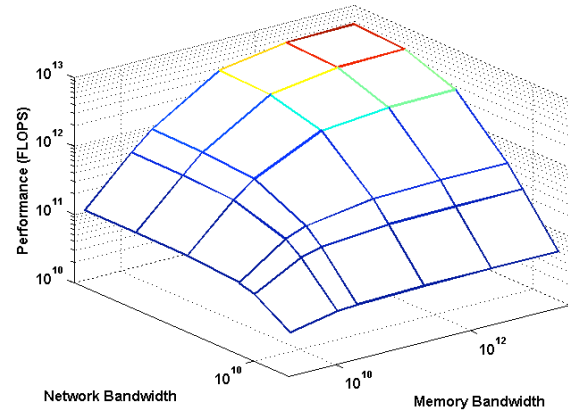
Figure 4: Sample performance result. The result indicates that best performance is achieved with both on-chip (network) and off-chip (memory) photonic interconnects.

Summary

This talk will present preliminary results of an architecture study focusing on performance of photonic networks for both on-chip and off-chip communication. The parameterized architecture is evaluated against HPEC challenge kernels. Memory hierarchy families evaluated will be inclusive of both NUMA and UMA access patterns. Based on throughput, latency, and programmability, an instantiated reference system-level architecture will be presented.

References

[1] K. Bergman and L. Carloni, "On-Chip Photonic Communications for High-Perforamnce Multi-Core Processors," *HPEC 2007 Workshop*, Lexington, MA, September 2007.
[2] J. Markoff, "Replacing Wire with Laser, Sun Tries to Speed Up Data," *New York Times*, March 24, 2008.
[3] D. Kuck, *High Performance Computing*. Oxford University Press, New York, New York, 1996.
[4] N. T. Bliss, J. Dahlstrom, D. Jennings, S. Mohindra, "Automatic Mapping of the HPEC Challenge Benchmarks," *HPEC Workshop 2006*, Lexington, MA, September 2006.
[5] N.T. Bliss, S. Mohindra, V. Aggarwal, U.M. O'Reilly, "Analysis and Mapping of Sparse Computations," *HPEC 2007 Workshop*, Lexington, MA, September 2007.
[6] R. Haney, T. Meuse, J. Kepner, J. Lebak, "The HPEC Challenge Benchmark Suite*," HPEC 2005 Workshop*, Lexington, MA, September 2005.