|  |  |
| --- | --- |
|  |  |

The Ministry of Education and Science of the Russian Federation

Lobachevsky State University of Nizhni Novgorod

Computing Mathematics and Cybernetics faculty

The competitiveness enhancement program   
of the Lobachevsky State University of Nizhni Novgorod   
among the world's research and education centers

Strategic initiative  
“Achieving leading positions in the field of supercomputer technology   
and high-performance computing”

**PROGRAMMING AND OPTIMIZATION   
FOR INTEL XEON PHI**

*Lecture 1. Intel Xeon Phi architecture*

Nizhni Novgorod

2014

# Objectives

The objective of this lecture is investigation of architecture and programming model of Intel Xeon Phi coprocessor. We consider the basic architecture blocks and features of the coprocessor: core, vector data processing block, high performance bi-directional ring bus, fully coherent L2 cache and the principles of component interaction. Our main focus is on elements most significantly influencing performance and understanding ways to optimize programs for Intel Xeon Phi.

# Abstract

This lecture considers architecture and programming model of Intel Xeon Phi. We introduce terminology (host, host OS, Micro Operating System, Intel Manycore Platform Software Stack). We describe coprocessor architecture, distinguish the main features of its components. The principle of core pipeline operation is formulated. We consider memory hierarchy and the corresponding performance aspects. A brief overview of instruction code is given. The lecture supposes basic knowledge of computer architecture.

# BRIEF OVERVIEW

We begin with the basic terminology to be used throughout the course: host, host OS, Micro Operating System, Intel Manycore Platform Software Stack.

Then we consider the key elements of Intel Xeon Phi architecture:

* The coprocessor includes 61 cores, connected with high performance ring bus. There are 8 controllers serving 16 GDDR5 channels providing 5.5 GT/s (billions transfers per second, with 64 byte width yielding 352 GB/s peak throughput). A separate component implements client PCI Express logic.
* Each core is fully functional and supports instruction fetching and decoding for 4 instruction streams. To increase memory efficiency coprocessor implements a distributed cache tag directory that allows using more efficient protocol to support cache coherence. Memory controllers provide theoretical peak throughput of 352 GB/s.
* A core performs instruction fetching and decoding of 4 hardware threads. It supports 32- and 64-bit code compatible with Intel64 architecture. A core includes 2 pipelines (U-pipe and V-pipe) and is capable of executing 2 instructions per cycle. V-pipe is not capable of executing all kinds of instructions, the possibility of parallel execution of instructions in U-pipe and V-pipe is defined by a set of rules. Out-of-order execution is not supported, as well as MMX, SSE, and AVX vector instruction sets. A core includes 32 Kb 8-channel multi-associative instruction and data caches (L1 I-Cache and L1 D-Cache).
* 512-bit vector processor unit (VPU) includes extended math unit (EMU) and is capable of sending one vector instruction per cycle to execution (thus handling 16 single precision floating point numbers, or 16 32-bit integers, or 8 double precision floating point numbers). For fused multiply-add (FMA) operations this yields 32 operations on floating point numbers per cycle. The vector processor unit contains 32 512-bit registers (zmm0-zmm31) and additionally provides execution of fill-in and permutation of vector register data, computation of powers of 2 (2^x), binary logarithm (log2x), reciprocate (1/x), and reciprocate square root (1/sqrt(x)) computation for single precision numbers. One of the operands may be read from RAM along with a type conversion when necessary.
* The Core-Ring Interface (CRI/L2) provides connection of a core to the high performance coprocessor interconnect – a ring bus, and includes 512 Kb 8-channel multi-associative L2 cache, advanced programmable interrupt controller (APIC) and Tag Directory.
* The memory controller (GBOX, GDDR MC) include three main components: ring bus interface (FBOX), request planner (MBOX) and interface to GDDR devices. Each memory controller includes two independent memory access channels. All memory controllers of the coprocessor are independent.
* SBOX component implements client PCI Express logic including Direct Memory Access mechanism (DMA) and limited capabilities of power management.
* Bi-directional ring bus provides data transfer between coprocessor components.

Intel Xeon Phi contains 61 cores and executes own operating system with one core dedicated for OS code execution, interruption handling, etc. Thus, in performance calculations it is supposed that 60 cores out of 61 are used for computations.

Theoretical peak performance of Intel Xeon Phi coprocessors with 60 cores and 1.1 GHz core frequency is calculated as follows:

• 16 (vector width) \* 2 flops(FMA) \* 1.1 (GHz) \* 60 (number of cores) = 2112 GFLOPS – for single precision floating point numbers and

• 8 (vector width) \* 2 flops (FMA) \* 1.1 (GHz) \* 60 (number of cores) = 1056 GFLOPS – for double precision floating point numbers.

2 flops per cycle is possible due to fused multiply-add (FMA) instruction.

Then we describe core pipeline of Intel Xeon Phi. The pipeline consists of 7 stages: Pre Thread Picker Function, Thread Picker Function, Decode Prefixes, Instruction Decode, Microcode Control, Execution, Write Back. All stages but the last one (WB) support speculative execution. Each core can execute instructions of 4 threads allowing to reduce losses from memory access latency, execution of vector instructions, and others. The lecture describes the essence of operations performed at each stage and relation between stages.

The next segment of the lecture investigates memory hierarchy with a focus on performance aspects. Each core of an Intel Xeon Phi coprocessor has its own L1 and L2 caches, all cores collectively use coprocessor RAM. L1 and L2 caches are inclusive, that is, all data stored in L1 are also in L2. Both caches employ pseudo-LRU replacement policy. We describe technical parameters of L1 and L2 caches, consider principles of its operation and features. Then Virtual Address Space is considered. Intel Xeon Phi runs special OS based on Linux kernel (kernel.org) with minimal extensions for compatibility. The coprocessor supports 32-bit physical addresses in 32-bit mode, 36-bit addresses while using Physical Address Extension (PAE) in 32-bit mode, 40-bit physical addresses in 64-bit mode. The coprocessor operating system supports 64-bit mode only. Processes are given linear virtual address space and can use 64-bit addresses. VAS is supported via standard x86\_64 architecture – page addressing with 4 levels of page tables. We review organization of translation look-aside buffer at Xeon Phi and give its main characteristics. Finally, we describe working with RAM.

The last segment of the lecture is devoted to instruction set of Intel Xeon Phi. Intel Xeon Phi coprocessor is compatible to Intel64 architecture except MMX, AVX, and SSE vector extensions, it supports original vector instruction set. The main features of the new instruction set are:

* It is mostly oriented on High performance computing (HPC). There are operations with 32-bit integer and floating point numbers, various type conversions frequently used in HPC applications, arithmetic operations with 64-bit floating point numbers and Boolean operations with 64-bit integers.
* 32 512-bit vector registers assist hiding data access latency. Each vector can be handled as 16 32-bit or 8 64-bit integer or floating point values.
* It makes use of ternary instructions with explicitly given two inputs and output. In FMA instruction the output is also the third input.
* There are special mask registers that can execute vectorized conditional operations and vectorize loops with conditional statements.
* Special scatter/gather instructions pack non-unit stride data into vector registers allowing vectorization of algorithms with complex data structures.

# FOR STUDENTS

Intel Xeon Phi architecture is described in [4] (Chapter 8, Coprocessor Architecture).

# References

1. Reinders J. An Overview of Programming for Intel Xeon processors and Intel Xeon Phi coprocessors. [http://software.intel.com/en-us/blogs/2012/11/14/an-overview-of-programming-for-intel-xeon-processors-and-intel-xeon-phi].
2. Loc Q Nguyen et al. Intel Xeon Phi Coprocessor Developer's Quick Start Guide. [http://software.intel.com/en-us/articles/intel-xeon-phi-coprocessor-developers-quick-start-guide].
3. Pseudo-LRU. [http://en.wikipedia.org/wiki/Pseudo-LRU]. J. Dongarra et al. Templates for the solution of linear systems: building blocks for iterative methods. SIAM, 1994.
4. Jeffers J., Reinders J. Intel Xeon Phi Coprocessor High Performance Programming. Morgan Kaufmann, 2013.

# Individual work

1. Distinguish main architecture features of Intel Xeon Phi compared to multicore CPUs.
2. Describe operating of core pipeline of Xeon Phi.
3. Formulate principles of memory organization of Xeon Phi.
4. Find out main differences in vector extensions of Xeon Phi from AVX instruction set.
5. Name features of Xeon Phi architecture that are most performance influencing.