Definitions

Manycore Chip:
A chip having many small CPUs, typically statically scheduled and 2-way superscalar or scalar.

Manycore Vector Chip:
A manycore chip in which each core has a wide vector FP unit.

These notes will describe Intel's Manycore Vector Chip, the Xeon Phi.

Currently these notes not complete.

References

Description of the Larrabee project.

Seiler, et al, "Larrabee: A many-core x86 architecture for visual computing," *ACM Transactions on Graphics*, vol. 27, no. 3, article 18, August 2008. (Linked to `http://www.ece.lsu.edu/koppel/gp/gpu-descriptions.html`.)

Some details are outdated, but does a good job explaining rationale for features. Described in the context of graphics, an abandoned target application.

Intel-Written Material on Phi.

Rahman, "Intel Xeon Phi Coprocessor Architecture and Tools," APRESS OPEN, 2013. (See Chapter 3 and 4.) `http://www.ece.lsu.edu/koppel/gp/refs/rahman-phi-book.pdf`.

Intel, "Intel Xeon Phi Coprocessor System Software Developers Guide", November 2012.

Intel, "Intel Xeon Phi Coprocessor Instruction Set Architecture Reference Manual," September 2012.

Intel's Manycore GPU.

Larrabee

Initially a project to develop a GPU.

Plans to develop a GPU product dropped but design targeted at GPGPU use.

Knights Corner

Code name for Intel's planned manycore GPGPU product.

Like Larrabee but not intended for graphics use.

Knights Ferry

Name of manycore GPGPU Intel development package, including chip.

Specs match Larrabee closely.

Xeon Phi

Name of the product that came out of the Larrabee project.

Overview

Chip consists of multiple cores.

The Xeon Phi has ≈ 57 cores.

A Phi core is synonymous with a CUDA multiprocessor, **not** a CUDA core.

Phi 3120P clocked at 1.1 GHz ...
... slow compared to contemporary CPUs ...
... but comparable to GPUs.

Cores are complete enough to host an operating system ...
... perhaps simplifying code porting.

Each Phi Core Has

- A 2-way superscalar, statically scheduled (in order) Intel 64 processor.
- A 16-lane vector unit.
- Four thread contexts.
- A 32 kiB L1 data cache.
- A 32 kiB L1 instruction cache.
- Share of coherent L2 cache: 512 kiB.

Address Spaces and Registers

Address Space

A 64-bit global address space.

No scratchpad, read-only, or other specialized spaces.

Registers (per Context)

One set of Intel 64 registers.

32 × 512-bit vector registers: `zmm0-zmm31`.

8 × 16-bit vector mask registers, `k0-k7`.

Storage Per Core

Low-Latency Storage (About 1-cycle access.)

32 kiB L1 data cache.

8 kiB vector registers ($4 \times 32 \times 64$).

Medium Latency Storage (From 11-80-cycle access.)

512 kiB L2 cache.

Cache Details

L1 Data Cache

Per Core: 32 kiB, 8-way.

64-byte line size.

Access to scattered 4-byte values wastes $\frac{15}{16}$ of cache.

One cycle load-to-use latency (integer instructions).

L2 Unified Cache

Unified (holds data and instructions).

Per Core: 512 kiB, 8-way, 64 B line.

Latency 11-80 cycles.

Inclusive: Data in L1I and L1D must also be in L2.

The line for address A consumes space . . .
. . . in the L2 cache of each core accessing A.

L1 and L2 are **Coherent**

Coherent Cache:
A cache in which the order of stores to a particular location is the same to all processors.

In addition to coherence, parallel systems should implement a **memory model**.

Memory models not discussed here.

Implementation of Coherence

Ensure that line being written is only in one cache.

Use **tag directory** and **coherence messages** to implement this.

Coherence Benefits

Makes it easy to share data.

Coherence is a standard feature on CPU memory systems.

Interconnection

Cores interconnected by a ring network.

Ring network used for messages to maintain coherence, among other purposes.

Width is 512 bits, two rings, one in each direction.

Instruction Issue

Two instructions per cycle.

At most one of these can be a vector instruction.

Scalar Pipeline Stages:

```
PPF PF D0 D1 D2 E WB
```

Vector Pipeline Stages:

```
PPF PF D0 D1 D2 E VC1 VC2 V1 V2 V3 V4 WB
```

Scalar Pipeline Stages

Briefly:

PPF: Fetch to prefetch buffer.

PF: Fetch from prefetch buffer.

D0: thread picker, insn rotate, decode of some insn

D1: Decode, Sunit reg file read

D2: Microcode control, address gen, dcache lookup, reg file read.

E: Integer ALU execution.

WB: Writeback

Important Instruction Set Features.

Memory Instructions

Prefetch

Cannot rely on massive multithreading to hide latency.

Cache Placement Hints

Addresses for vector load can come from a vector register.

Instruction Operand Features

Swizzling

Source operand conversion.

Can convert memory source of an arithmetic insn to 32-bit int and floats.