ELT-SCALE REAL-TIME CONTROL ON A XEON PHI

David Jenkins
OUTLINE

- The challenges involved in ELT-scale adaptive optics real-time control
- The Knights Landing Xeon Phi standalone processor
- Initial investigation and hardware benchmarking
- Preliminary base-line results for a simple adaptive optics SCAO loop
- Optimisation of DARC for the Xeon Phi and initial results
- Conclusions + Future considerations
The problem size for an adaptive optics (AO) RTC scales with the fourth power of telescope diameter for a simple SCAO system.

More complex systems such as MCAO and LTAO have even greater computational requirements.

A large matrix vector multiplication (MVM) is required for many reconstruction techniques and is one of the most computationally demanding operations.
WHAT'S THE PROBLEM?

- An RTC not only needs to run at a high frame rate but the frame times need to consistent, there needs to be low jitter.

- Using multiple processing nodes and/or accelerator cards can introduce increased latency and jitter due to the offload of data and instructions.

- FPGAs can be very deterministic and provide very consistent frame times, however they can have high start up costs, longer development times, and are not as immediately flexible as other options.

- GPUs provide very high computational throughput for highly parallelisable tasks, however they can introduce accelerator offload latency and use proprietary drivers and software.
A standalone socketed host processor or PCI accelerator card with 64+ processing cores and 16GB of High Bandwidth memory

512bit wide vector registers for 16 simultaneous single precision FP operations

Socketed host is the target of this investigation to eliminate accelerator off-load latency and allow the use of standard programming tools and languages

The Knights Landing (KNL) runs standard Linux with the option of an optimised kernel provided by Intel for full functionality
THE XEON PHI: KNIGHTS LANDING

- Is a single Xeon Phi host processor capable of meeting the requirements for ELT scale SCAO?

- Then for systems with more than one wavefront sensor a single Xeon Phi per wavefront sensor should provide the necessary computational performance.

- The Xeon Phi has the option of integrated OmniPath fabric interconnect which can help to reduce the latency of distributed workloads.
Knights Landing Overview

Chip: 36 Tiles interconnected by 2D Mesh
Tile: 2 Cores + 2 VPU/core + 1 MB L2

Memory: MCDRAM: 16 GB on-package; High BW DDR4: 6 channels @ 2400 up to 384GB
IO: 36 lanes PCIe Gen3, 4 lanes of DMI for chipset
Node: 1-Socket only
Fabric: Omni-Path on-package (not shown)

Vector Peak Perf: 3+TF DP and 6+TF SP Flops
Scalar Perf: ~3x over Knights Corner
Streams Triad (GB/s): MCDRAM: 400+; DDR: 90+

Source: Intel. All products, computer systems, dates and figures specified are preliminary based on current expectations, and are subject to change without notice. KNL data are preliminary based on current expectations and are subject to change without notice. Binary Compatible with Intel Xeon processors using HpcSand. Evaluated using HPCMark and STREAM. Bandwidth numbers are based on STREAM-like memory access pattern with MCDRAM loads to fast memory. Results have been estimated based on internal Intel analysis and are provided for informational purposes only. Any reference to system performance is for comparison purposes only and does not reflect actual performance.
# Knight Landing 7210

## Choose Your Optimization Point

<table>
<thead>
<tr>
<th></th>
<th>Integrated</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>7210</strong></td>
<td></td>
<td><strong>64</strong></td>
<td><strong>1.3</strong></td>
<td><strong>16GB</strong></td>
<td><strong>38GB</strong></td>
<td><strong>215W</strong></td>
<td><strong>$2438</strong></td>
</tr>
<tr>
<td><strong>Best Value</strong></td>
<td></td>
<td></td>
<td></td>
<td><strong>6.4</strong></td>
<td><strong>2133</strong></td>
<td><strong>96GB</strong></td>
<td></td>
</tr>
</tbody>
</table>

1. Available beginning in September.
3. Pricing shown is for parts without integrated fabric. Add additional $278 for integrated fabric versions of these parts. Integrated fabric parts available in October.
# Knight Landing Options

## Choose Your Optimization Point

<table>
<thead>
<tr>
<th></th>
<th>Integrated</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CORES</td>
<td>GHZ</td>
<td>MEMORY</td>
<td>FABRIC</td>
<td>DDR4</td>
<td>POWER</td>
<td>RECOMMENDED</td>
</tr>
<tr>
<td>7290</td>
<td>72</td>
<td>1.5</td>
<td>16GB</td>
<td>Yes</td>
<td>384GB</td>
<td>245W</td>
<td>CUSTOMER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>7.2 GT/s</td>
<td></td>
<td>2400 MHz</td>
<td></td>
<td>PRICING</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$6254</td>
</tr>
<tr>
<td></td>
<td><strong>Best Performance/Node</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7250</td>
<td>68</td>
<td>1.4</td>
<td>16GB</td>
<td>Yes</td>
<td>384GB</td>
<td>215W</td>
<td>$4876</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>7.2 GT/s</td>
<td></td>
<td>2400 MHz</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Best Performance/Watt</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7230</td>
<td>64</td>
<td>1.3</td>
<td>16GB</td>
<td>Yes</td>
<td>384GB</td>
<td>215W</td>
<td>$3710</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>7.2 GT/s</td>
<td></td>
<td>2400 MHz</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Best Memory Bandwidth/Core</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7210</td>
<td>64</td>
<td>1.3</td>
<td>16GB</td>
<td>Yes</td>
<td>384GB</td>
<td>215W</td>
<td>$2438</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>6.4 GT/s</td>
<td></td>
<td>2133 MHz</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Best Value</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Available beginning in September
2. Plus 15W for integrated fabric
3. Pricing shown is for parts without integrated fabric. Add additional $278 for integrated fabric versions of these parts. Integrated fabric parts available in October.
A tool for measuring the memory bandwidth of a system
Provided as part of Intel’s MICPERF benchmarking utility

- Results for the available 7210:
  - Memory performance: 442.14 GB/s (approx 500 GB/s for top end Xeon Phi)

- Compare to the quoted maximum performance:
  - Memory maximum: 400+ GB/s

- NVidia Tesla P100 quoted maximum:
  - Theoretical memory maximum: up to 732 GB/s
  - Roughly 580 GB/s achievable (estimate)
**Matrix Inversion**

- Important for the calculation of control matrices for most reconstruction techniques.

- MOAO will have particularly high requirements due to the number of WFSs and DMs involved in the reconstruction.

- LU ('lower upper') decomposition using the Intel Math Kernel Library, MKL
  - $M^T \times (M \times M^T)^{-1}$

- MCAO: for 60,000 x 10,000: $3.1s + 2s + 3.1s = 8.2s$
MATRIX INVERSION & MATRIX MULTIPLICATION

LU decomposition

Matrix Multiplication

\[ M^T \times (M \times M^T)^{-1} \]

\[ M \times M^T \]

(size: 2N×N)

20,000×10,000
MICADO SAMI module for first light SCAO correction, European-ELT

MICADO will eventually use the MAORY MCAO system

E-ELT SCAO
- 80x80 or 74x74 WFS sub-apertures with 8x8 pixels, Shack-Hartman
- ~5000 actuator DM :- E-ELT high-order adaptive mirror (M4)

Potential requirements
- 500Hz framerate
- <100μs RMS jitter
MVM BENCHMARK

- Simple matrix vector multiplication (MVM) operation using the Intel MKL cblas_sgemv function

- The MVM is the most computationally demanding aspect of least-squares reconstruction

- It's also a major aspect of other reconstruction techniques such as LQG

- Profiling how the MVM performance scales on the KNL with problem size and available processing threads
MVM BENCHMARK

Multi-threading:

Problem size scaling:

80x80:
~ 5k x 10k matrix

Mean for 80x80 sub-apertures
60 threads:
frame-time: 0.66 +/- 0.0078 ms
~ 8 µs rms jitter
~ 1.53 kHz frame-rate
SIMPLE RTC LOOP

- Performance profiling of a very simple least-squares SCAO like RTC loop on the Xeon Phi, not pipelined

- Pixel calibration, centre of gravity slope calculation, MVM reconstruction and application of a constant gain to DM commands

- Using OpenMP for multi-threading and MKL for the MVM, timing is done in software using an OpenMP function call

- All written in the standard C programming language with pragma directives to enable OpenMP functionality
**SIMPLE RTC LOOP**

<table>
<thead>
<tr>
<th>(80x80 subaps) Pixels</th>
<th>Mean Frame Time (ms)</th>
<th>Mean Frame Rate (kHz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2x2</td>
<td>0.716</td>
<td>1.40</td>
</tr>
<tr>
<td>4x4</td>
<td>0.763</td>
<td>1.31</td>
</tr>
<tr>
<td>6x6</td>
<td>0.808</td>
<td>1.24</td>
</tr>
<tr>
<td>8x8</td>
<td>0.817</td>
<td>1.22</td>
</tr>
<tr>
<td>10x10</td>
<td>0.843</td>
<td>1.19</td>
</tr>
</tbody>
</table>

Problem size:
SIMPLE RTC LOOP

Currently no real-time kernel. Should help with peaks (outliers)

80x80, 6x6:

Mean frame time: 0.80 +/- 0.027 ms
⇒ 27 μs rms jitter

Mean frame rate: 1.26 +/- 0.021 kHz
NEXT: DARC ON THE XEON PHI

- Optimisation of the Durham Adaptive optics Real-time Controller (DARC) for the Xeon Phi KNL

- DARC is written in C and so is immediately able to be compiled and run on the Xeon Phi

- However some optimisation is required to achieve peak performance and these optimisation can be carried over to other standard CPU based systems

- Currently very early preliminary stage with only minor changes carried out
RESULTS: LATENCY AND JITTER

74x74, 8x8:
Mean frame time: 1.45 ± 0.032 ms
⇒ 33 μs rms jitter
Mean frame rate: 692 ± 15 Hz
# RESULTS: DARC SUBAPS AND PIXELS

<table>
<thead>
<tr>
<th>Sub-apertures</th>
<th>Pixels</th>
<th>Mean frame-time (μs)</th>
<th>Mean frame-rate (Hz)</th>
</tr>
</thead>
<tbody>
<tr>
<td>80x80</td>
<td>6x6</td>
<td>1550 ± 39</td>
<td>646 ± 16</td>
</tr>
<tr>
<td></td>
<td>8x8</td>
<td>1650 ± 36</td>
<td>607 ± 13</td>
</tr>
<tr>
<td>74x74</td>
<td>6x6</td>
<td>1350 ± 34</td>
<td>743 ± 18</td>
</tr>
<tr>
<td></td>
<td>8x8</td>
<td>1450 ± 33</td>
<td>692 ± 15</td>
</tr>
</tbody>
</table>

74x74, 8x8 percentiles:
- 99% 1520 μs 660 Hz
- 99.9% 1550 μs 640 Hz
- 99.99% 1610 μs 620 Hz
MAIN AREAS OF OPTIMISATION

- Optimise the current multi-threading capabilities of DARC for many more threads

- DARC multi-threading uses POSIX threads (Pthreads), a low level execution model independent from any one language

- As the KNL is a socketed CPU running Linux Pthreads are already supported

- Optimise array operations for vectorisation

- Set-up the BIOS, kernel, and OS for optimal low-latency performance
CONCLUSIONS

- KNL can compute an ELT scale MVM at a rate greater than 1.5 kHz
- A baseline simple RTC loop can perform at around 1.26 kHz
- DARC, full RTC in simulation mode, currently running >600 Hz
- Still some more work needs to be done to achieve optimal performance
- Jitter is really quite good already <40 microseconds for DARC
- Could meet the potential specification for ELT scale SCAO of ~500Hz with <100microsec jitter
- Shows promise also for multiple WFS multiple Xeon Phi MOAO systems
More testing and optimisation to be done

Real-time kernel

Memory and cache

More vectorisation

Pixel calibration can be done as pixels arrive rather than on a sub-aperture basis

Other algorithms, LQG, correlation, match filtering

Real 10G camera data

Investigate multiple Xeon Phis for more complex systems

Full MAORY demonstration

Future generations? Knights Hill ~2X performance
- Thank-you for listening