Andrei Poenaru
Tom Deakin
Simon McIntosh-Smith

University of Bristol
HPC Research Group

Si Hammond
Andrew Younge

Sandia National Laboratories

An Evaluation of the A64FX Architecture for HPC Applications

University of BRISTOL

UKRI Engineering and Physical Sciences Research Council

GW4

Introduction to the GW4 Isambard 2 supercomputer

- **Isambard 2** is a £4.1M EPSRC project, run by a consortium of the GW4 Alliance, the Met Office, HPE/Cray, Fujitsu and Arm, to deliver a Tier-2 HPC service to researchers across the UK and around the world
- Funded in late 2019, Isambard 2 builds on Isambard 1's achievements as the world's first Arm64-based production supercomputer
- **Isambard 1** has been a huge success, proving for the first time that Arm works for supercomputing in production environments

Isambard 2 production system

- **21,504** ARMv8 cores (336n x 2s x 32c)
  - Marvell ThunderX2 32 core @ 2.5 GHz
- Cray XC50 ‘Scout’ with Aries interconnect
- Cray HPC optimised software stack
  - Compilers, math libraries, CrayPAT, …
  - Also comes with all the open source software toolchains: GNU, Clang/LLVM etc.
- Multi-Architecture Comparison System
- HPE Apollo 80 A64FX system
- Hosted for the Consortium by the Met Office in Exeter

University of BRISTOL

UKRI Engineering and Physical Sciences Research Council

GW4

Isambard 2's A64FX system

- HPE Apollo 80 system with A64FX CPUs from Fujitsu
  - 72 nodes connected with 100 Gbps IB
    - 3,456 cores, 72 TB/s memory bandwidth, 202 TFLOP/s 64-bit
  - Comes with a Cray software stack
    - CCE, GNU, Arm, Fujitsu Compilers

University of BRISTOL

UKRI Engineering and Physical Sciences Research Council

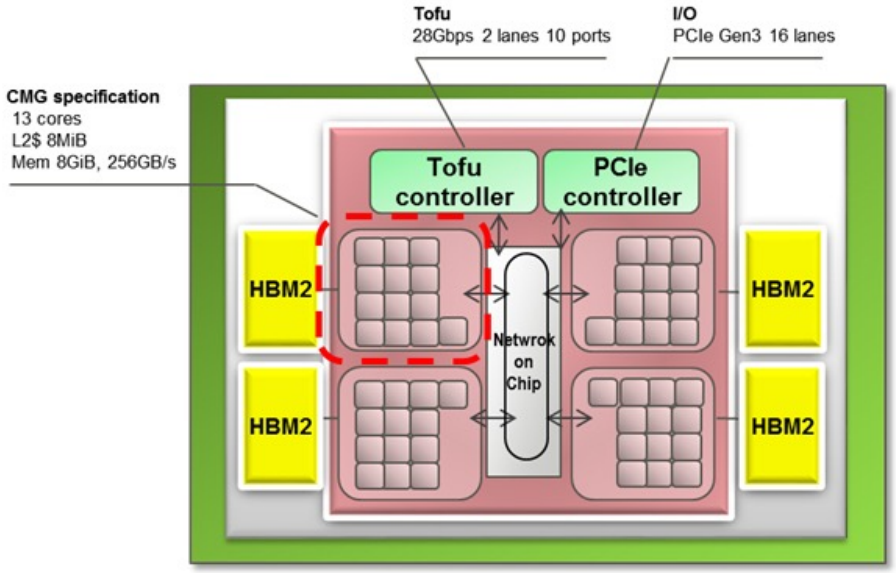
Fujitsu's A64FX

http://www.fujitsu.com/global/about/resources/news/press-releases/2018/0822-02.html

https://github.com/fujitsu/A64FX

- 48 cores, 1.8 – 2.2GHz
  - 4 CMGs
- >2.7 TFLOP/s double precision
- 2x 512-bit vector pipelines per core
  - ARMv8.3-A + SVE
- 1 TByte/s main memory bandwidth
  - 4 stacks of HBM2
- ~170 Watts
- High speed interconnect
- 8.7B transistors, 7 nm

FUJITSU



University of BRISTOL

UKRI Engineering and Physical Sciences Research Council

GW4

A new generation of Arm-based processors

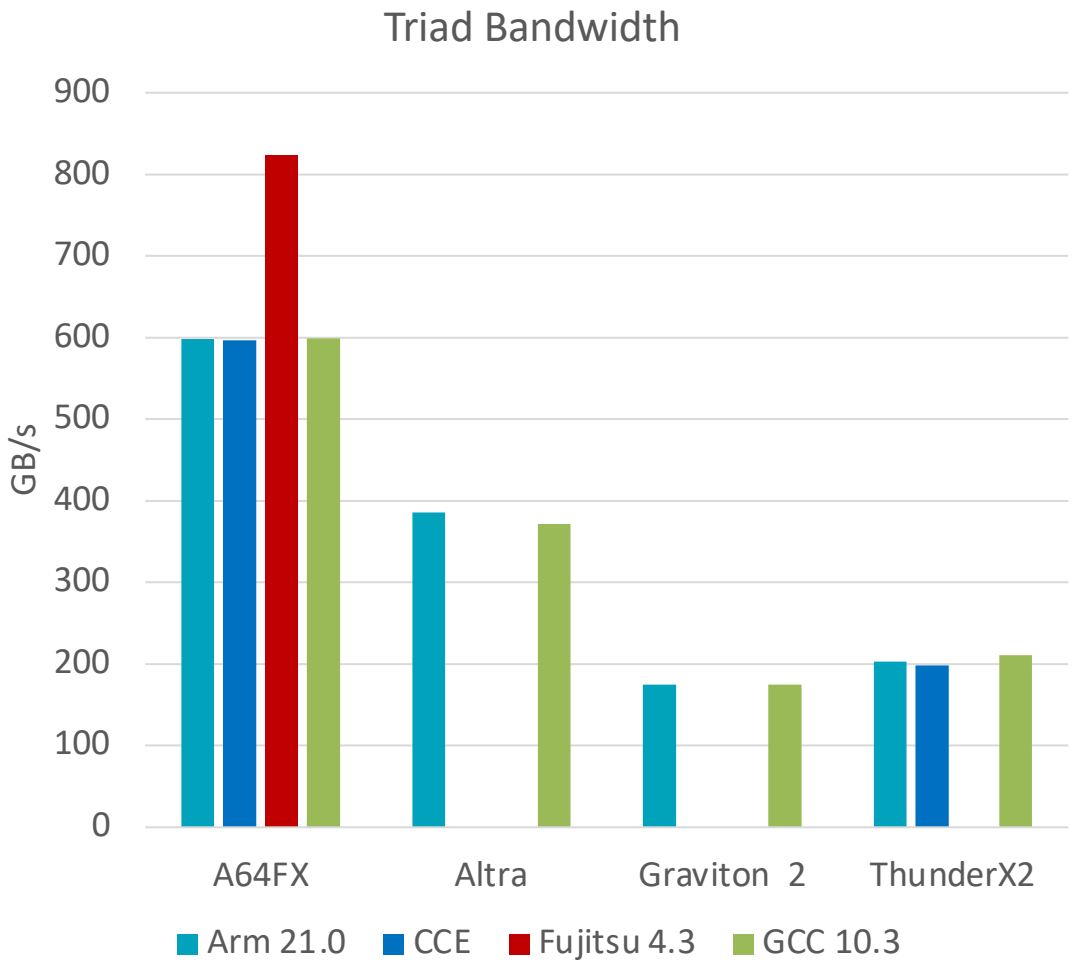
Platform	Cores	Clock Speed	Peak FLOP/s (d.p.)	Peak memory BW
Fujitsu A64FX	1 x 48	1.8 GHz	2.8 TFLOP/s	1,024 GB/s
AWS Graviton 2	1 x 64	2.5 GHz	1.3 TFLOP/s	205 GB/s
Ampere Altra	2 x 80	3.0 GHz	3.8 TFLOP/s	410 GB/s
Marvell ThunderX2	2 x 32	2.5 GHz	1.3 TFLOP/s	320 GB/s

No special configuration required on any platform

- Arm and GNU compilers support all
- Cray supports A64FX† and TX2
- Fujitsu supports A64FX
  - Trad and Clang mode

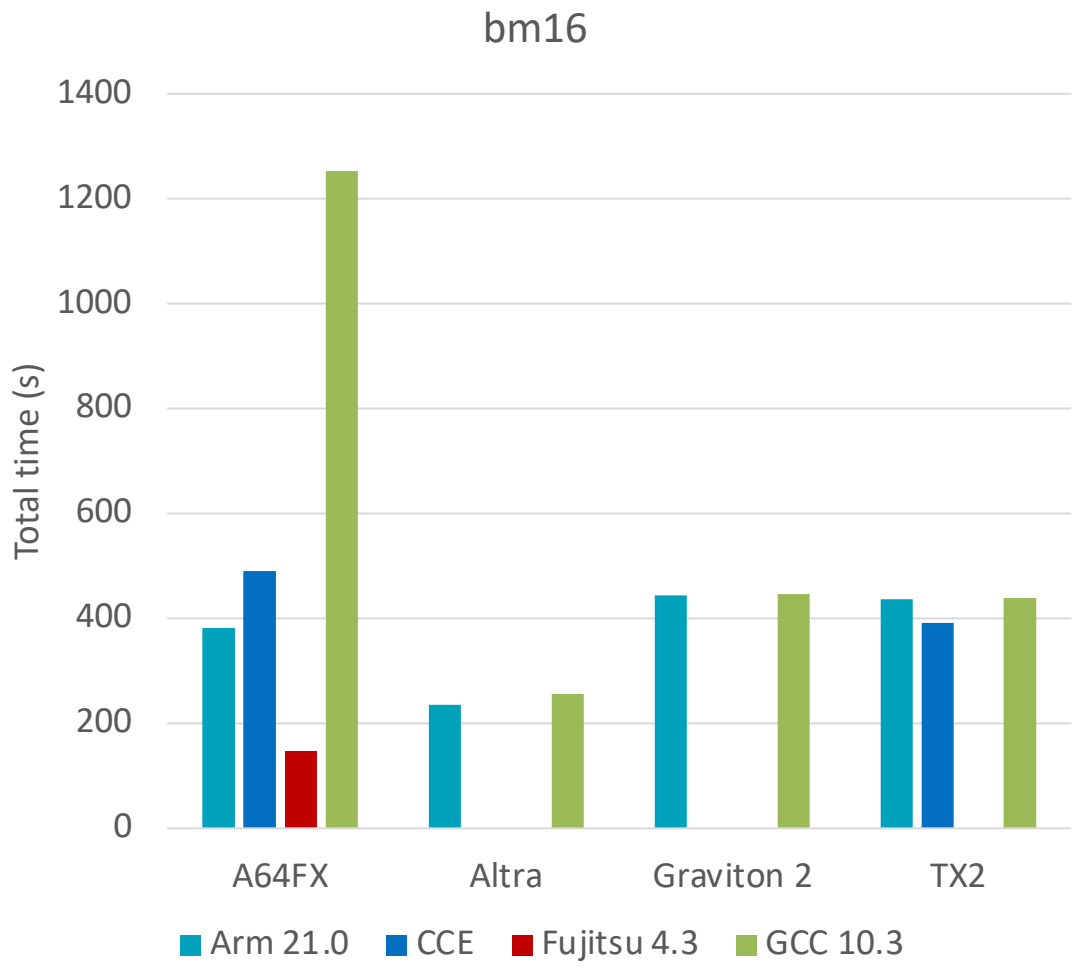
† SVE support in CCE is still beta

BabelStream



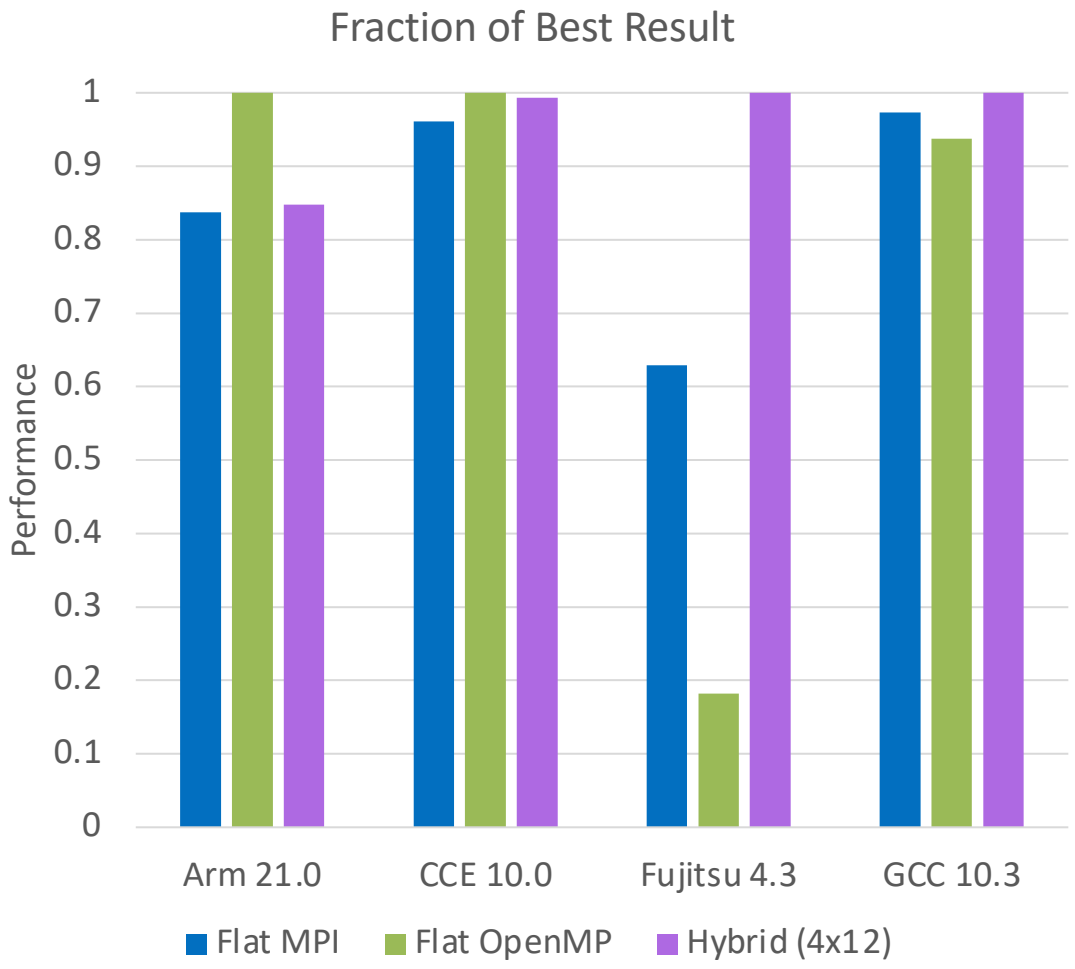
- A high fraction of the HBM2 peak bandwidth is achievable in practice
- In order to avoid reads into cache before streaming writes, FCC uses zero-fill instructions
  - Compiler flags can be used to tune prefetching if desired
- `XOS_MMM_L_PAGING_POLICY=demand:demand:demand`

CloverLeaf



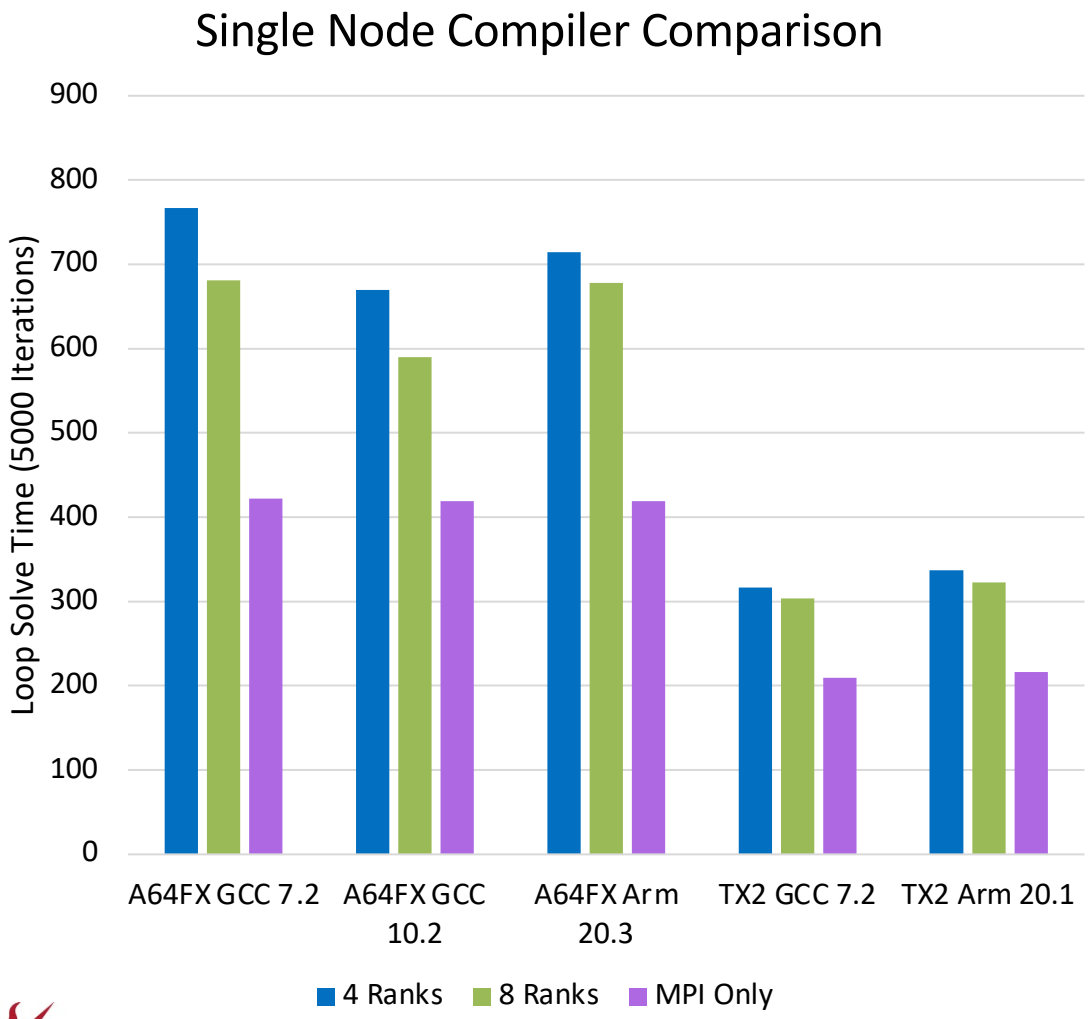
- CloverLeaf is generally bandwidth-bound, but there are also divisions and trigonometry
  - With the Fujitsu Compiler, the benefit of the HBM2 is clear
  - Arm compiler optimises division (`-fiterative-reciprocal`)
  - GCC produces bad SVE vector code
- CloverLeaf is hybrid MPI+OMP
  - Results here are the best in each case

CloverLeaf MPI–OpenMP placement



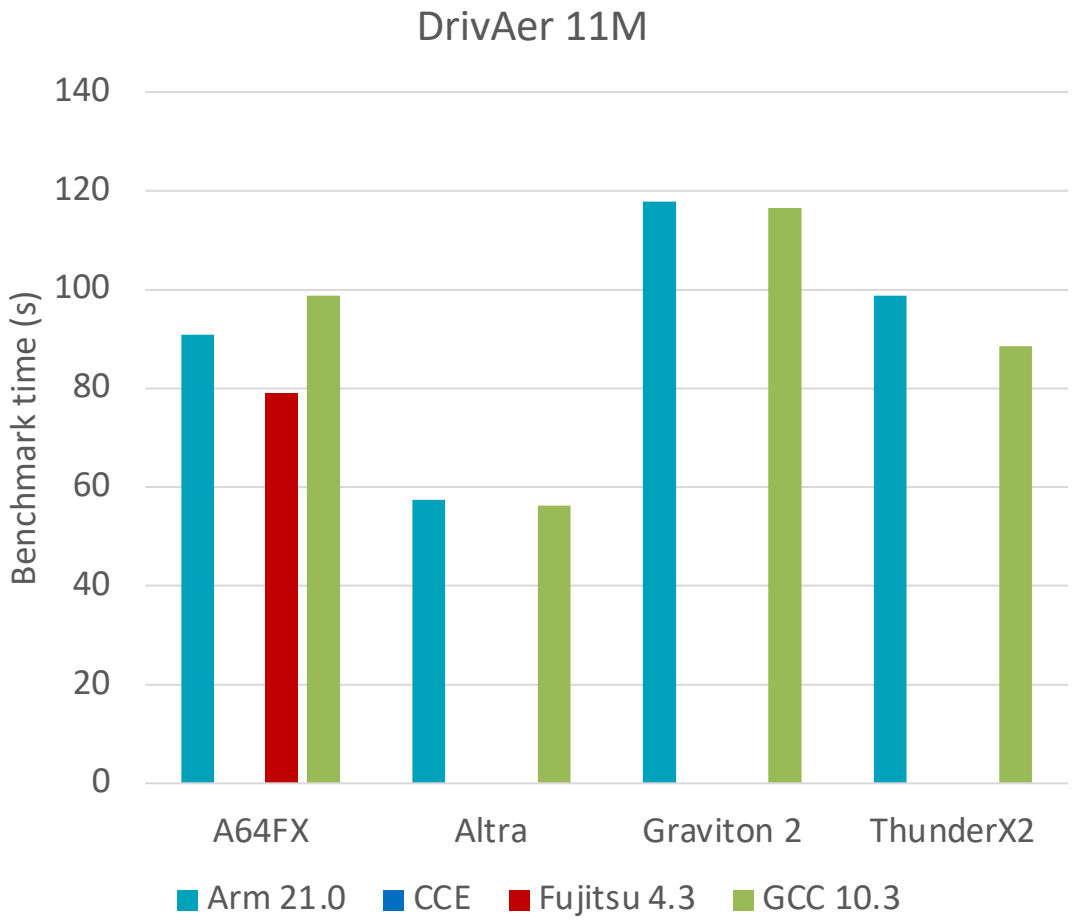
- On A64FX, hybrid MPI is generally the best choice
  - One rank per NUMA node
  - Not all MPI/OpenMP implementations place and bind ranks/threads the same way!
- Flat MPI and hybrid are close on the other platforms
- Flat OpenMP is slightly slower on multi-socket systems (Altra, TX2)

SPARTA



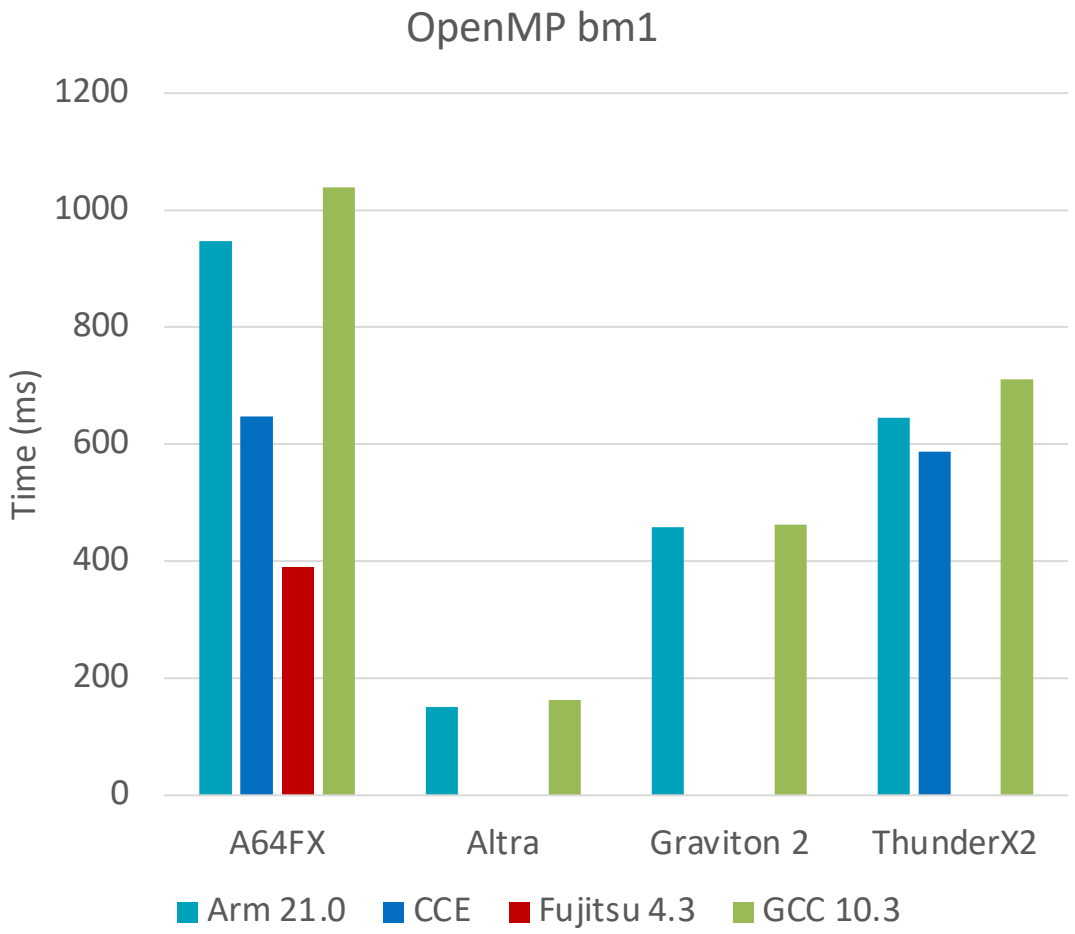
- SPARTA is an open-source direct-simulation Monte-Carlo (DSMC) application
  - Scaled out to the largest NNSA/ASC computing platforms including BG/Q, Sierra and Trinity platforms
- Written using MPI and Kokkos
  - Parallelization/threading of kernels is still being optimized
  - Kokkos provides best mapping of flags
  - GCC 7.2 does not have A64FX or SVE-512 flags
  - GCC 10.2/Arm 20.3 have SVE code generators
- Utilized in particle collision benchmarks at 10M particles per node
- Results use a single node a decompose 48 cores using N rank and 48/N OpenMP threads
- Fujitsu C++ compiler produces segmentation faults in the code in the MPI operations (no results)
- MPI everywhere results are very similar between compilers. When using OpenMP threads, GCC 10.2 provides the fastest performance.

OpenFOAM



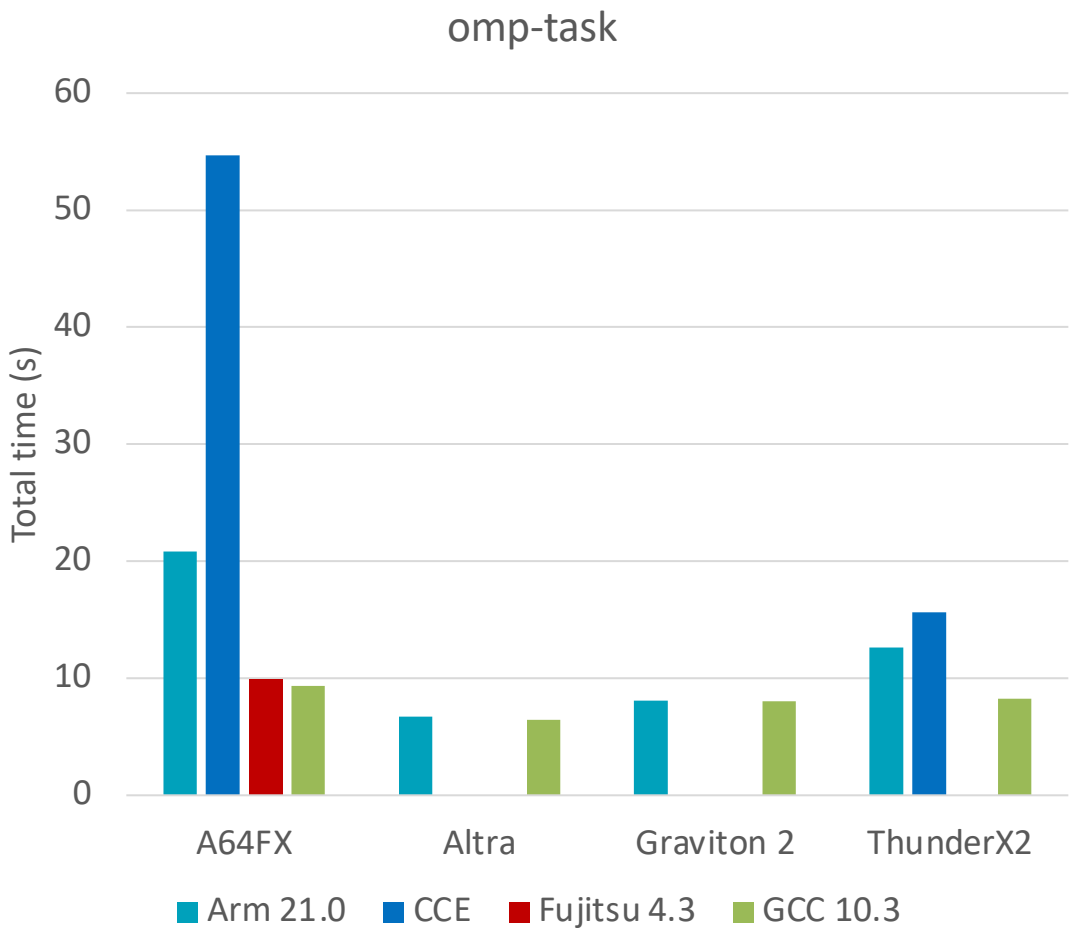
- OpenFOAM v2006, DrivAer test case, `simpleFoam` solver
- Benchmark time: last – first
- CCE: compilation errors
- In general very memory-bandwidth-bound, but compute performance does matter
  - Vectorisation is low
- A64FX result faster than TX2, but shows memory bandwidth is not every thing
- Rome and Altra are the fastest
  - Both have large caches…

miniBUDE: Molecular Docking



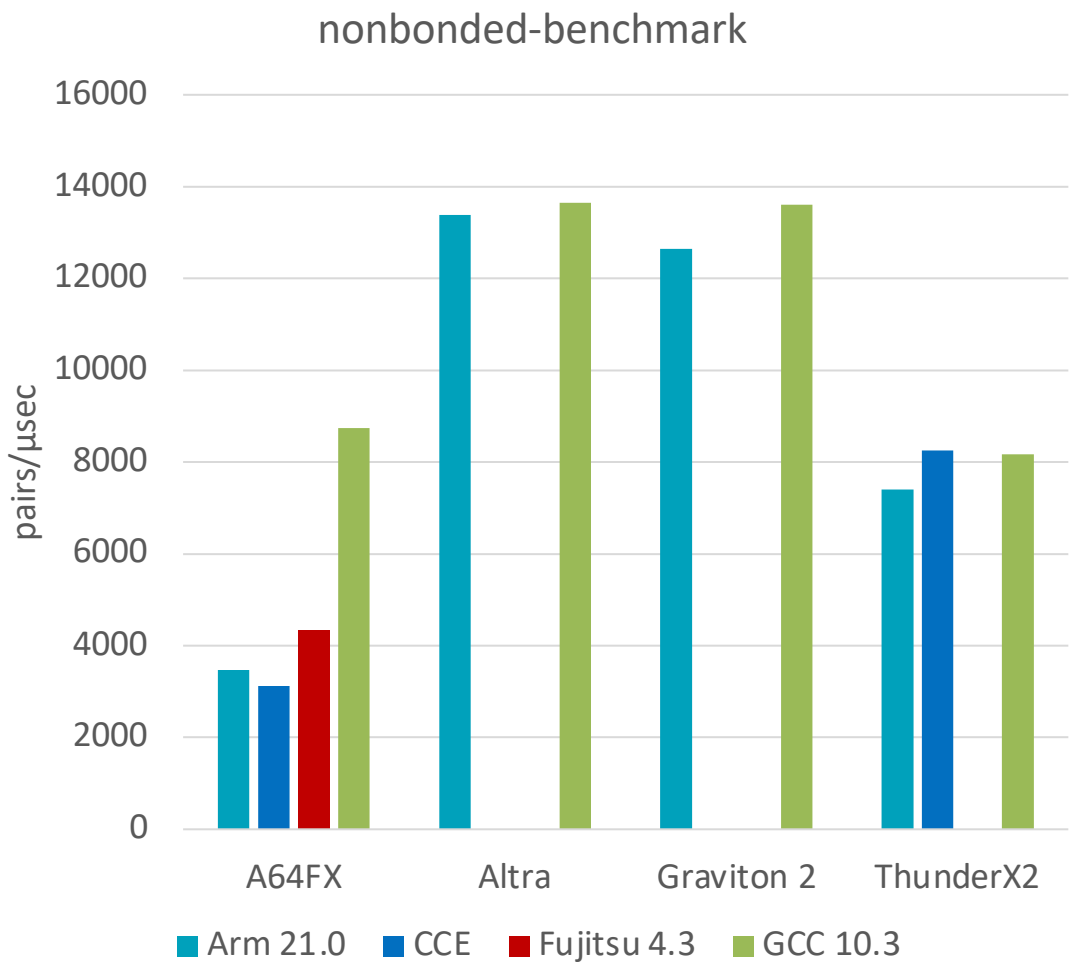
- miniBUDE is heavily compute-bound [1]
  - Achieves ~60% of peak FLOP/s
- On A64FX it benefits greatly from compiler unrolling, interleaving, and software pipelining
- On Graviton 2 and TX2, short vectors are the main limitation
- Altra performance is competitive with Cascade Lake and Rome

MiniFMM: Fast Multipole Method Mini-App



- Computation naturally suited for task-based parallelism
- None of the compilers are able to achieve good vectorisation performance, with either SVE or NEON
- CCE-SVE is based on the older Cray C++ frontend and is less efficient with tasks than the Clang-based frontend
- In contrast to BUDE, more threads do not *help* here
  - No performance gains over ~60 threads
  - On Altra, using 160 threads is much slower

GROMACS



- 2021.1, integrated PME benchmark
  - Limited to 64 threads
- Only GNU supported for SVE
  - With the other compilers, need to use NEON version on A64FX
- Can use different FFT libraries
  - For this benchmark, we found little differences between FFTW and ArmPL
  - Arm results use ArmPL, GNU use FFTW

Experience with Workloads on Isambard 2

- Everything works out-of-the-box!
  - The same experience we had with ThunderX2 in Isambard 1
  - No specific programming model or language needed
- Compiler support and libraries are already available
  - Cray, Arm, Fujitsu support A64FX
  - GCC support in 10.3 and 11 (released last week!)
- Optimised libraries and higher-level frameworks are continuously being improved
- More full-scale benchmarks soon, e.g. NAMD, OpenSBLI, UM

Working with the Fujitsu A64FX

- No difference from working with other general-purpose CPUs
- 4 CMGs (NUMA nodes)
  - Core binding is particularly important
  - Some applications benefit from running 4 MPI ranks/node
- Out-of-Order architecture benefits from software pipelining and optimised instruction scheduling
  - Use a compiler with a good cost model
- There is a configurable “sector cache”…

It's easy to apply for time on Isambard

- Please contact the Isambard PI, Prof. Simon McIntosh-Smith simonm at cs.bris.ac.uk, who will help you determine if Isambard will work for you. If it will, applying for an account is quick and easy.

- Small amounts of pump-priming time are available for free, to try porting, optimizing for Arm etc.

- Larger amounts of time for real science runs can be applied for via the regular EPSRC "Access to HPC" calls, or via some CCPs.

Summary

- A64FX already looks very promising, beating cutting-edge dual-socket nodes in many tests
- Easy to use – in most cases running unmodified flat MPI, or hybrid MPI+OpenMP
- Performance is similar to GPUs, but with a significantly lower barrier to entry in ease of use
- Isambard 2 makes most of the major technologies available in one place, enabling rigorous comparative benchmarking

Thank you

- https://uob-hpc.github.io
- https://github.com/UoB-HPC/benchmarks
- https://github.com/UoB-HPC/performance-portability
- https://gw4-isambard.github.io/docs/index.html

[1] Andrei Poenaru, Wei-Chen Lin and Simon McIntosh-Smith. 'A Performance Analysis of Modern Parallel Programming Models Using a Compute-Bound Application'. In: 36th International Conference, ISC High Performance 2021. Frankfurt, Germany, 2021. In press.