

Why RISC-V for HPC?

Change is a risk for supercomputer operators, therefore the key question we need to be able to answer is “*what are the killer reasons why HPC centres should adopt RISC-V in their systems?*”

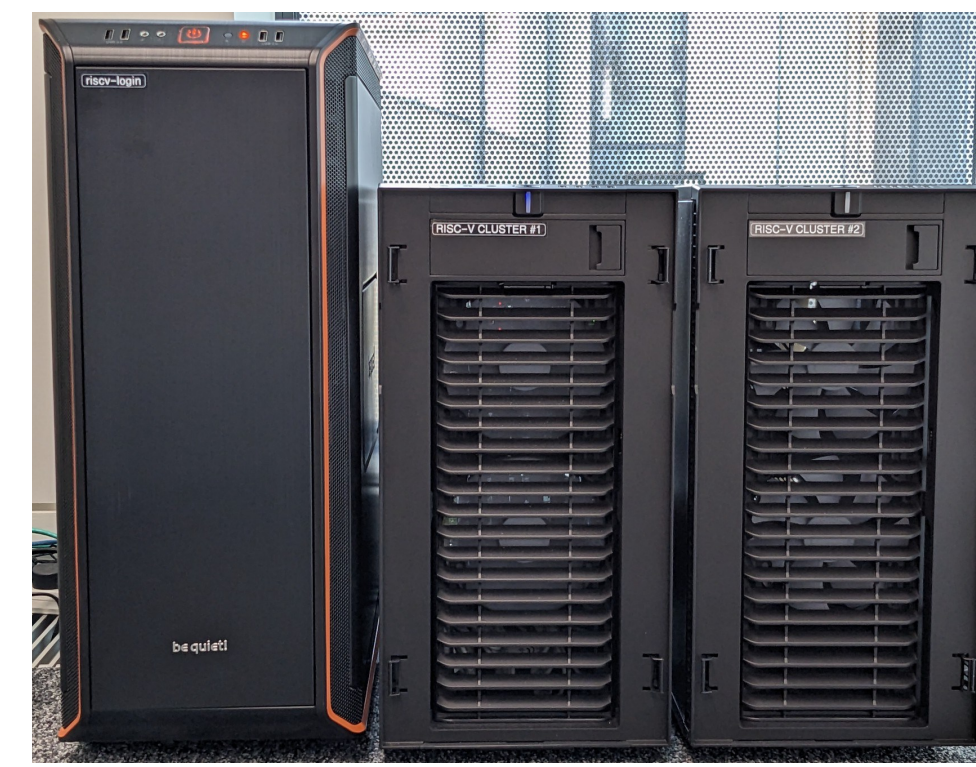


Some potential answers:

- **Openness** where standard but specialised solutions can be built by anyone, ultimately meaning hardware can be driven by HPC market demand.
- **Community driven** the wide range of perspectives involved in RISC-V means that many people are shaping the standard and can get involved in the evolution of RISC-V.
- **Modularity** means that it is possible to leverage the standard in a flexible manner, resulting in solutions that can deliver improved performance and energy efficiency.

The first widespread adoption of RISC-V in HPC might be driven by accelerators, and potentially those that have been designed for AI/ML workloads but are also beneficial more widely for HPC

Our RISC-V HPC testbed



- The purpose of the testbed is to provide free access to RISC-V for HPC application developers and users so they can experiment with the technology for their codes
- Set up as any other HPC system, with a login node, shared file system, using a batch queue system to run on the compute nodes and software organised via the module environment



The testbed currently provides the following:

- 3 x Milk-V Pioneer (64-core SG2042)
- 1 x HiFive Unmatched (quad core U740)
- 13 x StarFive VisionFive V2 (quad core U74)
- 3 x StarFive VisionFive V1 (dual core U74)
- 4 x Allwinner D1-H (C906 CPU)
- 2 x MangoPi MQ-Pro (C906 CPU)
- 2 x BananaPi (Spacemit K1/M1)
- 2 x Milk-V Jupiter (Spacemit K1/M1)
- 1 x Tenstorrent Greyskull e150 (RISC-V accelerator)
- 2 x Tenstorrent Wormhole n300 (RISC-V accelerator)

Watch this space: More RISC-V hardware is added as it becomes available!

Availability of high performance RISC-V hardware

When we first stood up the testbed, there was a shortage of RISC-V hardware. RISC-V solutions built around SoCs then became available such as the VisionFive V2 (quad core, 8GB DDR)



- But this was still a long way away from HPC!



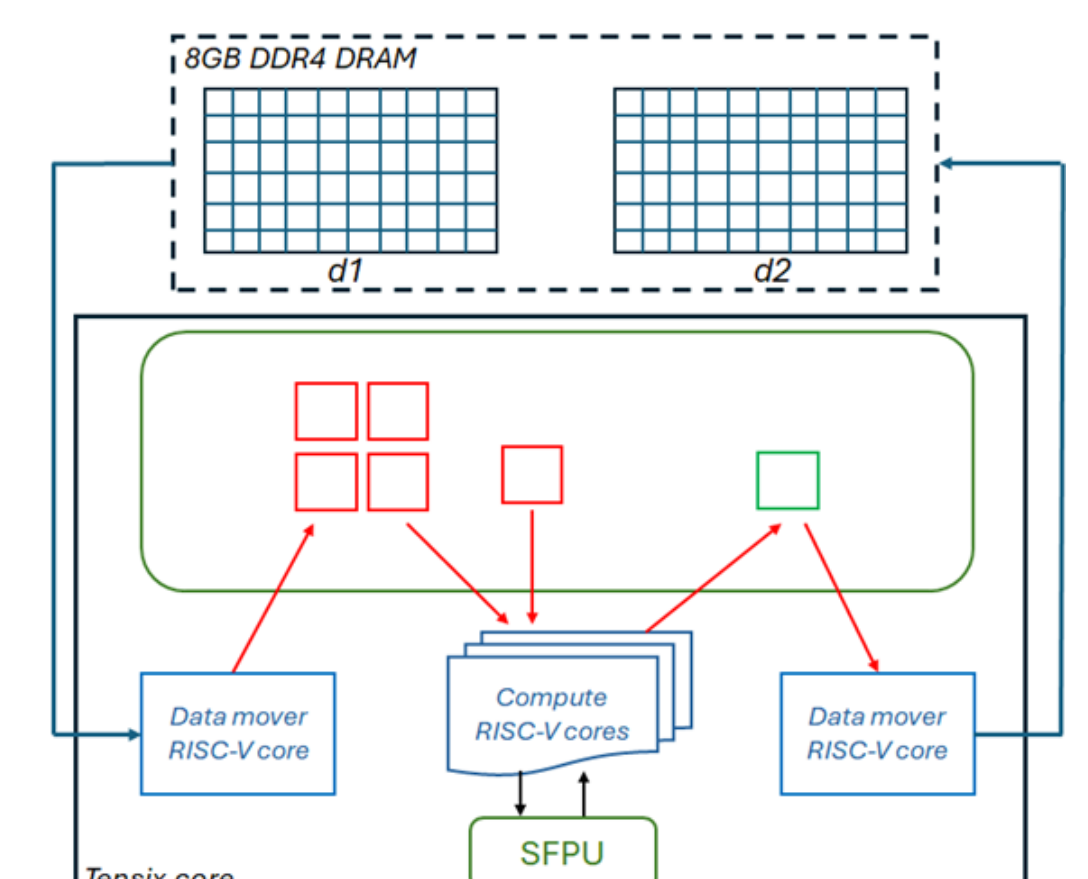
An important development was in late 2023 when the commodity 64-core SG2042 by Sophon became available. The large core count, focus on higher performance workloads, and ability to use more memory means this feels like a much more serious proposition for HPC codes.

RISC-V accelerator exploration

The Tenstorrent family of accelerators is built around Tensix cores, the Grayskull has 108 Tensix cores:

- Each Tensix cores has 5 RISC-V “baby” cores, driving a wide matrix and vector unit
- 1MB(+) of SRAM within each Tensix unit

An example of leveraging RISC-V for a bespoke hardware solution tuned to a specific class of problem



Type	Total cores	Cores in Y	Cores in X	Performance (GPI/s)	Energy (Joules)
CPU	1	-	-	1.41	1657
CPU	24	-	-	21.61	588
e150	1	1	1	1.06	2094
e150	2	1	2	2.48	893
e150	4	1	4	2.92	744
e150	8	4	4	7.99	276
e150	32	8	4	9.20	240
e150	64	8	8	12.96	170
e150	72	8	9	17.26	128
e150	108	12	9	22.06	110
e150 x 2	216	24	9	44.12	102
e150 x 4	432	48	9	86.75	108

Performance and energy usage comparison for a problem size of 1024 by 9216 (9.4 million) BF16 elements, over 5000 iterations

We explored a stencil code (Jacobi iteration solving Laplace’s equation for diffusion in 2D) on the Grayskull

- Some challenges, ultimately comparative performance to a 24-core Xeon Platinum but 5 times less energy.

More details at <https://arxiv.org/pdf/2409.18835>

Benchmarking a multi-socket SG2042 system against other architectures that are used in HPC

This is in ARCHER2, an HPE

Cray EX and the UK national supercomputer with 256GB

DDR4-3200 per node

In a local workstation with 192GB DDR4-2666 per node

CPU	ISA	Part	Base clock	Number of cores	Vector
AMD EPYC	x86-64	EPYC 7742	2.25GHz	64	AVX2
Intel Skylake	x86-64	Xeon Platinum 8170	2.1 GHz	26	AVX512
Marvell ThunderX2	ARMv8.1	CN9980	2 GHz	32	NEON
Sophon SG2042	RV64GCV	SG2042	2 GHz	64	RVV v0.7.1

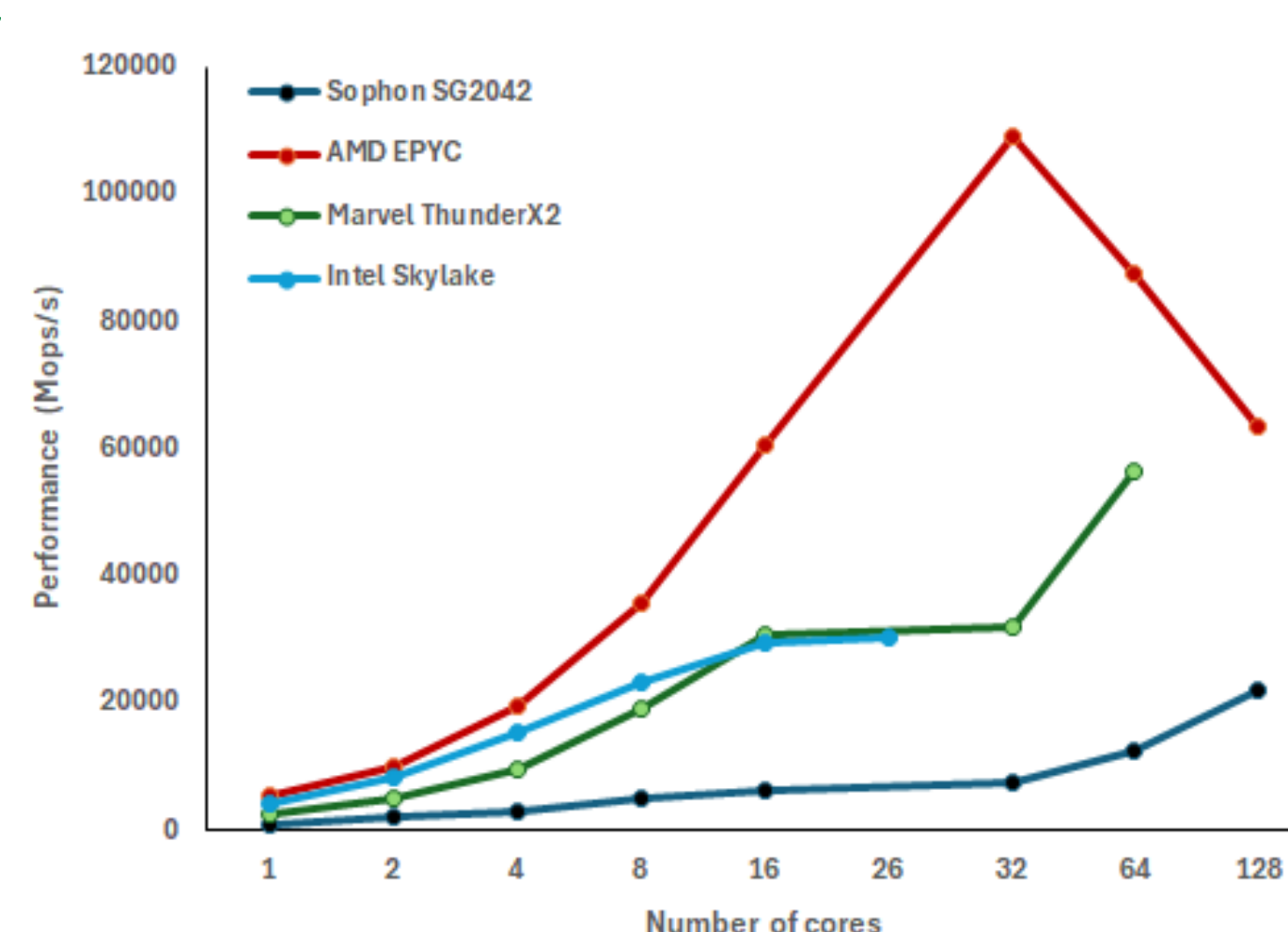
In Fulham, an HPE machine with 128GB DDR4-2666 per node

Thanks to E4 computer engineering for access to this multi-socket SG2042 system

In E4 Computer Engineering’s test system (thanks for access!)

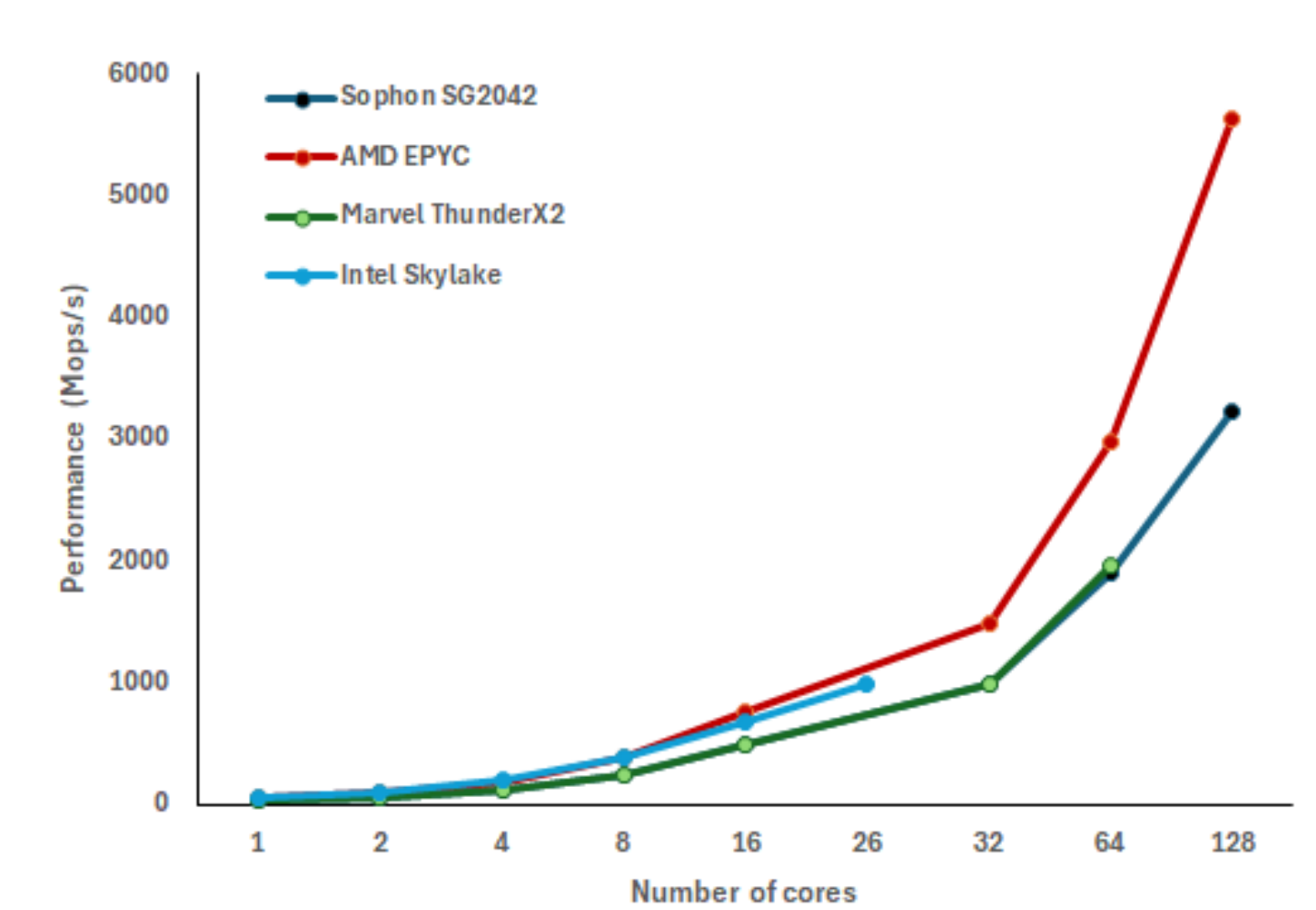
Using class C of NASA’s NAS parallel benchmark suite, selected results shown here, with more detailed discussion at <https://arxiv.org/pdf/2502.10320>

MultiGrid (MG) benchmark



The MG benchmark is memory (bandwidth) bound and it can be seen that the SG2042 performs lowest compared to the other CPUs. This is also a similar situation with the IS benchmark (which contains indirect, random, memory accesses) and suggests that the SG2042 is limited by its memory subsystem for codes that are memory bandwidth or latency bound.

Embarrassingly Parallel (EP) benchmark



The EP benchmark is designed to test compute performance, and here the SG2042 performs, core for core, very similar to the Marvell ThunderX2. The large core count on the SG2042 compared to the Intel Skylake and Marvell ThunderX2 means that it ultimately outperforms these at 64 cores. This demonstrates that the compute performance of the SG2042 is competitive against other CPUs.

What are we doing right & what is missing for HPC?



The RISC-V software ecosystem has matured rapidly, and we have found that the vast majority of our user’s HPC codes build out of the box



A large majority of HPC tools and libraries are ported to RISC-V, although often not optimised for instance not being able to leverage RVV



There is a growing interest in RISC-V by the HPC community, and awareness of RISC-V as a *hot topic* has improved substantially in the past few years



We need to support the Lustre parallel filesystem and enhance maturity around high performance networking to enable scaling out.

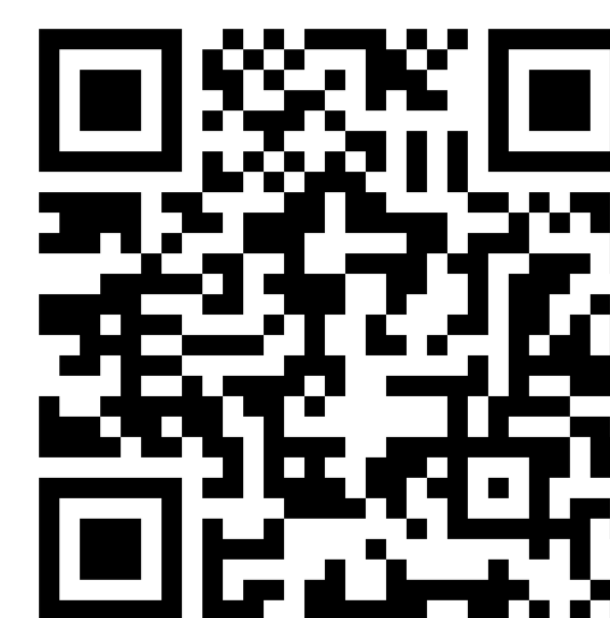


Support by mature profiling tools for RISC-V, along with a rich set of hardware events, is crucial to help HPC developers optimise their code for the technology.



We need to ensure that activities such as the matrix extension(s) suit the needs of HPC, the HPC SIG can provide a range of use-cases here.

Free access to our RISC-V testbed



Using our RISC-V testbed is free, and we invite people who are wanting to explore porting and optimising codes on RISC-V to sign up via the website

<https://riscv.epcc.ed.ac.uk>

Conclusions

The advances in the RISC-V ecosystem have been phenomenal since we began our HPC testbed two years ago, and with a range of new hardware anticipated for 2024 it looks highly likely that this pace will continue to accelerate. It is important that RISC-V and the HPC community continue to work together, identify the key software building blocks required and are able to make a strong case for the role of RISC-V in HPC

