RISC-V for HPC: Where we are and where we need to go

Nick Brown (n.brown@epcc.ed.ac.uk)

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:

- As the HPC community moves towards decarbonising their workloads, leveraging bespoke compute which is more suited to specific applications is more efficient
- A potential opportunity to unify the programming of CPUs and accelerators with a common ISA and ecosystem, running across both transparently
- Ubiquity via stealth, with RISC-V embedded inside other technologies such as networking and ML accelerators which themselves are adopted

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

The ExCALIBUR 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:

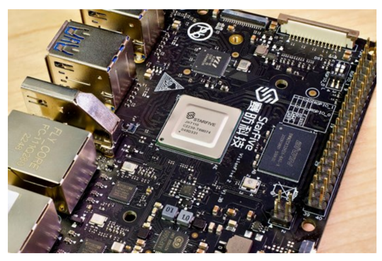
- 2 x Milk-V Pioneer (64-core SG2042)
- 1 x HiFive Unmatched (quad core U740)
- 13 x StarFive VisionFive V2 (quad core U74)
- 3 x StarFive VisionFive V1 (dual core U74)
- 4 x Allwinner D1-H (C906 CPU)
- 2 x MangoPi MQ-Pro (C906 CPU)
- 1 x Tenstorrent Greyskull e150 (RISC-V accelerator)

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.

Single core comparison: SG2042 against other RISC-V

Benchmark	SG2042	VisionFive V2	VisionFive V1	SiFive U740	All Winner D1
IS	60.6	17.84 *(29%)*	6.36 *(10%)*	9.09 *(15%)*	5.41 *(9%)*
MG	1210.05	288.65 *(24%)*	72.31 *(6%)*	90.28 *(7%)*	163.19 *(13%)*
EP	31.35	12.01 *(38%)*	7.55 *(24%)*	9.08 *(29%)*	9.23 *(29%)*
CG	205.25	43.61 *(21%)*	21.96 *(11%)*	20.09 *(10%)*	12.99 *(6%)*
FT	857.64	245.99 *(29%)*	88.35 *(10%)*	116.59 *(14%)*	DNR

Single core comparison between RISC-V technologies with performance reported in Mops/s (Higher is better) using NPB kernels running at class B. In red is the percentage performance delivered compared to the C920 core of the SG2042.

- Single-core benchmarking using NASA's NAS parallel benchmark suite (class B).
- C920 core of the SG2042 significantly out performs all other RISC-V technologies
- U74 of the VisionFive V2 performs closest to C920, delivering 21% to 38% performance of C920
- Whilst the VisionFive V1 and SiFive U740 both contain the same U74 core as the VisionFive V2, they both provide consistently lower performance (potentially partly due to lower clock)
- C906 in the All Winner D1 (the cheapest board) outperforms V1 and U740 for EP & MG

Benchmarking the 64-core SG2042 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

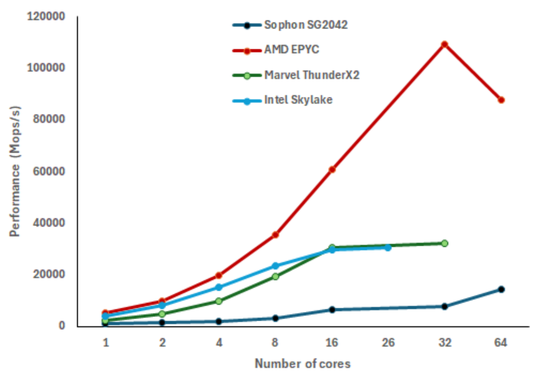
CPU	ISA	Part	Base clock	Number of cores	Vector
AMD EPYC	x86-64	EPYC 7742	2.25GHz	64	AVX2
Intel Skylake	x86-64	Xeon Platinum 8170	2.1 GHz	26	AVX512
Marvell ThunderX2	ARMv8.1	CN9980	2 GHz	32	NEON
Sophon SG2042	RV64GCV	SG2042	2 GHz	64	RVV v0.7.1

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

In a Milk-V Pioneer workstation with 128GB DDR4-3200

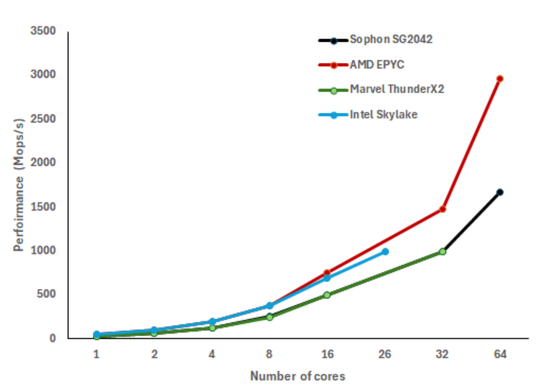
Using class C of NASA's NAS parallel benchmark suite, selected results shown here, with more detailed discussion at *https://arxiv.org/pdf/2406.12394*

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 it's 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

---

✗ It would be useful to have a prioritised list of HPC applications and libraries that need to run well on RISC-V, as well as a mechanism to ensure that work is not duplicated across groups

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

✗ We need to build evidence to clearly argue why one should adopt RISC-V in future supercomputers. This involves a risk to HPC centres, so they need to be convinced!

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