EXCALIBUR Hardware and Enabling Software RISC-V Testbed

Citation for published version:
Brown, N 2023, 'EXCALIBUR Hardware and Enabling Software RISC-V Testbed', ISC High Performance 2023, Hamburg, Germany, 21/05/23 - 25/05/23.

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

General rights
Copyright for the publications made accessible via the Edinburgh Research Explorer is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The University of Edinburgh has made every reasonable effort to ensure that Edinburgh Research Explorer content complies with UK legislation. If you believe that the public display of this file breaches copyright please contact openaccess@ed.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.
EXCALIBUR Hardware and Enabling Software RISC-V Testbed

Providing a complete ecosystem where users can experiment with RISC-V for their HPC and ML codes

What is RISC-V?

Instruction Set Architectures (ISAs) are typically proprietary, limiting the number of implementations (e.g. x86) and/or requiring licences and restrictions (e.g. ARM). By contrast, RISC-V is an open ISA developed by the community where anybody is able to take the specification and then provide a CPU implementation of this. Not only does this encourage a collaborative effort in developing a solid and mature ISA, it also results in a large number of RISC-V CPU implementations and a rich software ecosystem.

Testbed project timeline

The purpose of the testbed is to provide access to RISC-V hardware for HPC & ML code developers to experiment with

- **April 2022**: Project starts, website and documentation available
- **June 2022**: Enabling software development starts
- **July 2022**: Early access testbed available
- **August 2022**: Coupling soft cores with accelerators begins
- **January 2023**: First soft cores made available to users
- **September 2023**: First courses around using the testbed and hardware design
- **December 2023**: Final results of benchmarking and enabling software
- **March 2024**: User case studies published and project completes

The testbed will be available for use at least until 2025

Physical RISC-V CPUs

We provide access to physical RISC-V CPUs which represent a variety of different technologies and capabilities. These all run Linux and are provided as compute nodes, where compilation is undertaken on the login node. A variety of common HPC libraries are provided including FFTW, PETSc, MPI etc.

The testbed currently contains the following type of nodes:

- HiFive Unmatched (quad core U740)
- StarFive VisionFive V2 (quad core U74)
- Allwinner D1-H (C906 CPU)
- Lichée RV Dock (C906 CPU)
- MangoPi MQ-Pro (C906 CPU)
- And more types of RISC-V node added as they become available!

Physical boards enable easy access to RISC-V, however RISC-V is moving very quickly and so can be somewhat behind the cutting-edge state of the art

Access to cutting edge soft core RISC-V designs

Soft cores provide a software description of a CPU which can then be used to program an FPGA. This enables us to provide a catalogue of many different types and configurations of RISC-V CPU at larger core counts

For example, the image on the right illustrates a single-core NeoRV32 (the central large block) with other blocks providing infrastructure support. This enables the CPU core to access memory, GPIO, UART and interact with the host machine

- We provide numerous pre-built soft cores in a catalogue which can be loaded by users. Additional configurations of these can be provided as required
- All soft cores run on a state of the art AlphaData P101 which provides the Versal FPGA
- We have developed Launchpad, which provides seamless interaction with the soft cores

Enabling software development

An important aspect of the project is to also enhance the software ecosystem for RISC-V. To this end we have been porting libraries and developing new tools

An example: Supporting 1.0 vectorisation on 0.7 hardware:

- **Problem**: Physical RISC-V cores tend to support version 0.7 of vectorisation, whereas version 1.0 has been released and the only version supported by up-to-date/current/upstream compilers
- **Solution**: We have developed a tool that manipulates the generated assembly code, to backport executables so that they comply with the 0.7 vectorisation standard, https://github.com/RISCVtestbed/rvv-rollback

Exploratory benchmarking

We are also undertaking benchmarking to understand the relative performance of the RISC-V cores and options. When vectorisation is enabled, we provide a variety of benchmarks to compare against D1 scalar

RISC-V is not just for future HPC CPUs, it also provides an extension interface, enabling coupling with accelerators

- RISC-V Core undertakes management and executes non-accelerated portions of code
- The accelerator might or might not be driven by the RISC-V ISA (e.g. V extension)
- This specialisation is a benefit for HPC, coupling CPUs with bespoke accelerators

Funded by ExCALIBUR H&ES

The ExCALIBUR program addresses the challenges and opportunities offered by computing at the exascale

https://excalibur.ac.uk/

The Hardware and Enabling Software (H&ES) component aims to provide novel hardware in the form of testbeds. Our testbed enables HPC developers to experiment with RISC-V.