river lane

At Riverlane we have been building Deltaflow.OS® and are very excited to share a first version with you!

We have a grand vision: an operating system that makes quantum software portable across qubit technologies; that is scalable to millions of qubits; and, that teases the highest possible performance out of every qubit – even for applications like error correction that need fast feedback loops.

While this vision will take a few years to be realised, we want to help engineers and scientists with their ambitious and quantum computing projects today. We want to help you get the best out of the qubits that you have built, to be able to prototype quickly and bug free, and to collaborate easily in multi-disciplinary settings.

That's why we have built Deltaflow.OS®. We see the quantum computer as a distributed control system with a CPU, FPGAs, analogue units, and qubits working under tight latency constraints. The Deltaflow® language lets the user define a graph of nodes representing the different elements of compute and the wires in between. The Deltaflow® runtime takes care of mapping a Deltaflow® programme onto real hardware in the lab. A Deltaflow® programme is portable across qubit technologies via a hardware abstraction layer, a minimal set of instructions through which a Deltaflow® programme controls analogue hardware.

Deltaflow.OS® naturally enables (formal) verification and debugging, which reduces down time and leads to faster R&D cycles in the lab. Deltaflow® enables physicists and engineers to focus on building better qubits and gates in the lab without having to worry about FPGAs.

Deltaflow-on-ARTIQ

Releasing Deltaflow-on-ARTIQ is the first step towards achieving our grand vision. Our MVP benefits you in the lab today, accelerating your research by helping you avoid bugs and enabling collaboration between experimentalists and theorists. We have connected the Deltaflow® language with ARTIQ, a control system widely used in the trapped-ion community. Here's what we've built:

- An emulator of the ARTIQ control system that lets users run experiments offline
- The Deltaflow® language with CPU nodes based on Python and FPGA nodes based on Migen
- A prototype of the hardware abstraction layer

We've released Deltaflow-on-ARTIQ to the community for free under an MIT + Commons Clause license. If you want to help us improve Deltaflow-on-ARTIQ, we are happy to accept pull requests. For commercial use, please get in touch via team@riverlane.com.

Continue reading ›

river lane

How you can use Deltaflow-on-ARTIQ

Collaborate with external partners:

If a theorist wants to try out a new algorithm or application on your qubits, they can write it in Deltaflow®, debug it offline with the help of our emulator and then run it in the lab. Have a look at our Jupyter Notebooks showing a simple Rabi Oscillation demo and VQE.

Everyone gets a lab:

With our emulator you can avoid using experimental hardware to debug gateware.

Go beyond the circuit model:

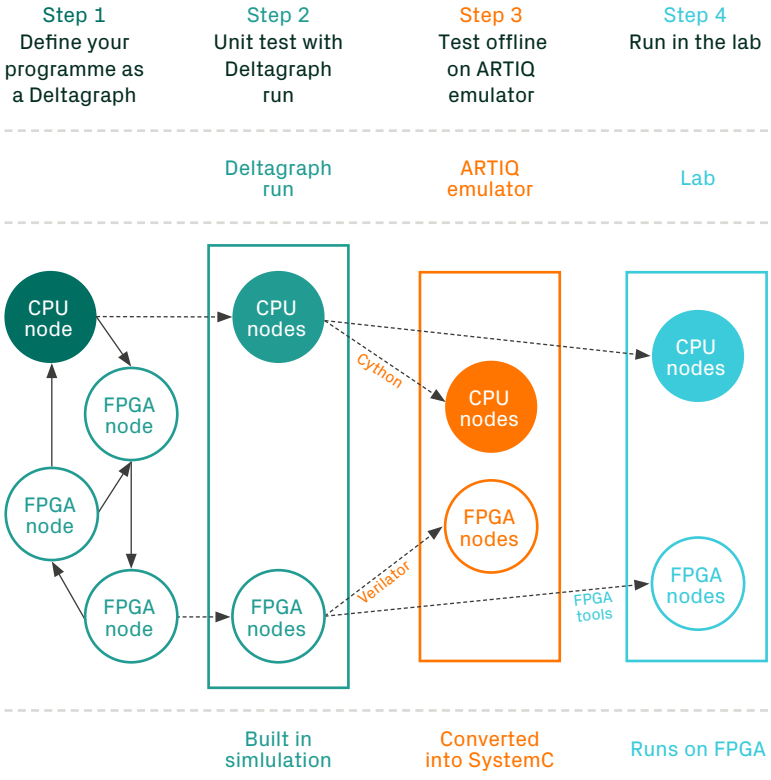
With Deltaflow®, you can implement applications that benefit from local control, such as adaptive algorithms or randomised benchmarking. Read our perspective to learn more.

Distributed experimental control

Deltaflow® can be used at the lowest level of the control stack. You can build nodes to handle communication with analogue electronics and data capture.

Deltaflow® integrates with SystemC:

Model your Deltaflow® control system as a single extra module.



We want to hear from you!

This is the beta version of Deltaflow-on-ARTIQ and we want to improve the operating system to make it as useful as possible for the community. Which additional features would you like to see? Please help us and take two minutes to complete our survey.