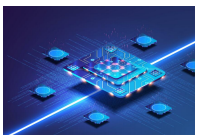
System-Level Quantum-Accelerator Integration with QAL

Ralf Ramsauer[1], Wolfgang Mauerer[1,2]

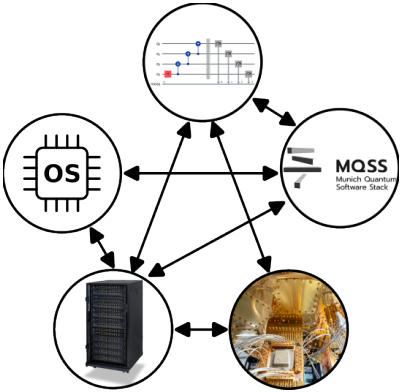
[1]Technical University of Applied Sciences Regensburg

[2]Siemens AG, Technology

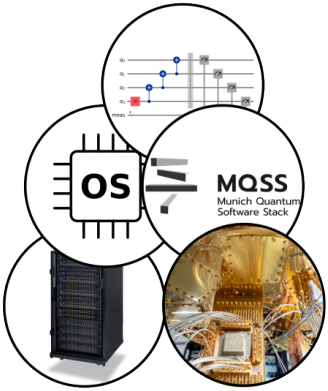
Munich Quantum Software Forum (MQSF) 2025

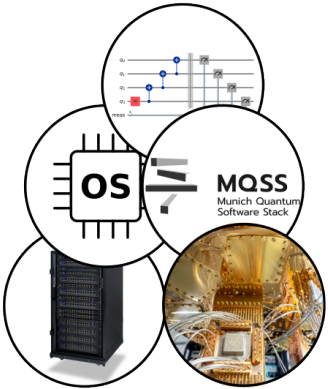


Images © MQV, Quantum Machines, FZ Jülich

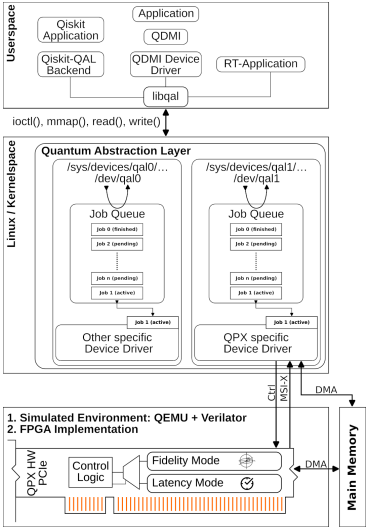


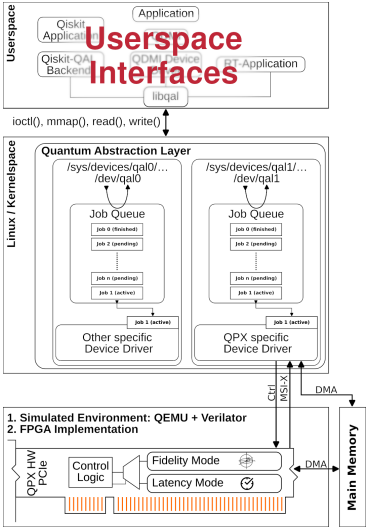
Images © MQV, Quantum Machines, FZ Jülich

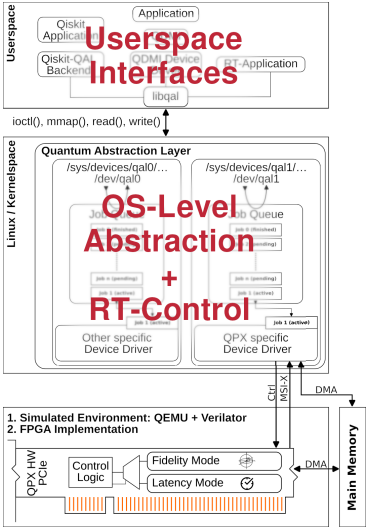


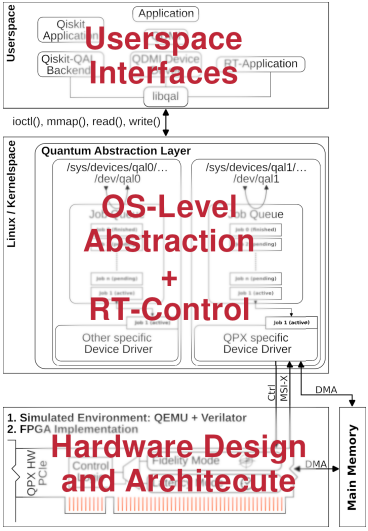


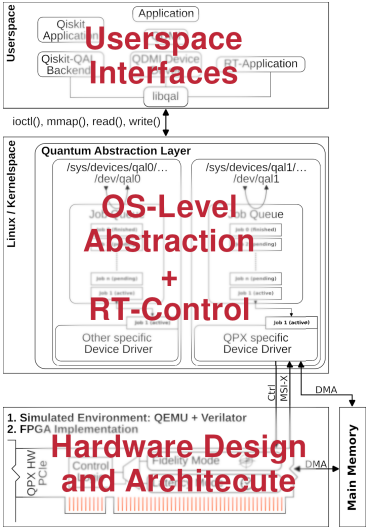
⇒ Tighter coupling between classical and quantum components

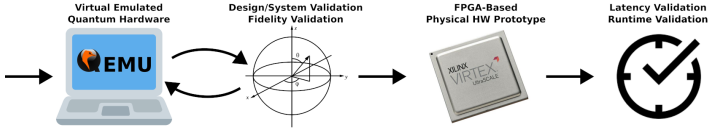

Userspace
Application
Qiskit Application
QDMI
Qiskit-QAL Backend
QDMI Device Driver
RT-Application
libqal
ioctl(), mmap(), read(), write()
Linux / Kernelspace
Quantum Abstraction Layer
/sys/devices/qal0/…
/dev/qal0
/sys/devices/qal1/…
/dev/qal1
Job Queue
Job 0 (finished)
Job 2 (pending)
Job n (pending)
Job 1 (active)
Other specific Device Driver
QPX specific Device Driver
Ctrl
MSI-X
DMA
1. Simulated Environment: QEMU + Verilator
2. FPGA Implementation
QPX HW
PCIe
Control Logic
Fidelity Mode
Latency Mode
DMA
Main Memory

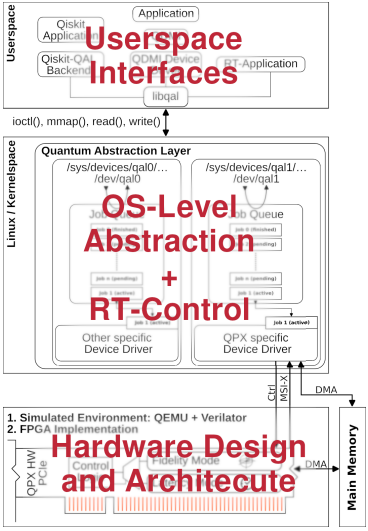

Userspace
Application
Qiskit Application
Qiskit-QAL Backend
QDMI Device
RT-Application
libqal
Userspace Interfaces
ioctl(), mmap(), read(), write()
Linux / Kernelspace
Quantum Abstraction Layer
/sys/devices/qal0/…
/dev/qal0
/sys/devices/qal1/…
/dev/qal1
Job Queue
Job 0 (finished)
Job 2 (pending)
Job n (pending)
Job 1 (active)
Other specific Device Driver
QPX specific Device Driver
Ctrl
MSI-X
DMA
1. Simulated Environment: QEMU + Verilator
2. FPGA Implementation
QPX HW PCIe
Control Logic
Fidelity Mode
Latency Mode
DMA
Main Memory


Userspace
Application
Qiskit Application
Qiskit-QAL Backend
QDMI-Device
RT-Application
libqal
Userspace Interfaces
ioctl(), mmap(), read(), write()
Linux / Kernelspace
Quantum Abstraction Layer
/sys/devices/qal0/…
/dev/qal0
/sys/devices/qal1/…
/dev/qal1
Job Queue
OS-Level Abstraction + RT-Control
Other specific Device Driver
QPX specific Device Driver
Ctrl
MSI-X
DMA
1. Simulated Environment: QEMU + Verilator
2. FPGA Implementation
QPX HW PCIe
Control Logic
Fidelity Mode
Latency Mode
DMA
Main Memory

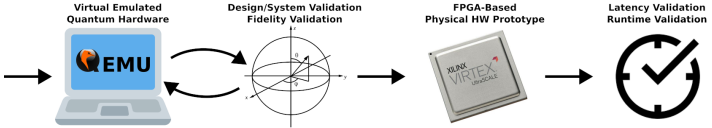

Userspace
Application
Qiskit Application
Qiskit-QAL Backend
QDMI Device
RT-Application
libqal
Userspace Interfaces
ioctl(), mmap(), read(), write()
Linux / Kernelspace
Quantum Abstraction Layer
/sys/devices/qal0/…
/dev/qal0
/sys/devices/qal1/…
/dev/qal1
Job Queue
OS-Level Abstraction + RT-Control
Other specific Device Driver
QPX specific Device Driver
Ctrl
MSI-X
DMA
1. Simulated Environment: QEMU + Verilator
2. FPGA Implementation
QPX HW
PCIe
Control
Fidelity Mode
DMA
Hardware Design and Architecute
Main Memory

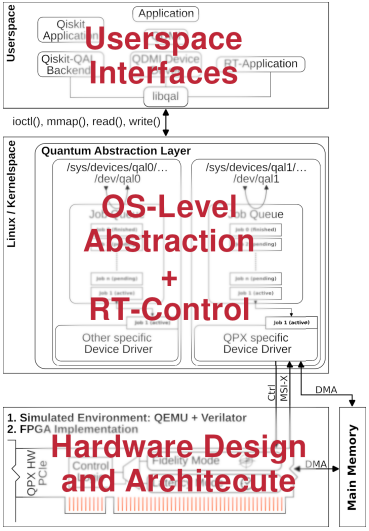

Userspace
Application
Qiskit Application
Qiskit-QAL Backend
QDMI Device
RT-Application
libqal
Userspace Interfaces
ioctl(), mmap(), read(), write()
Linux / Kernelspace
Quantum Abstraction Layer
/sys/devices/qal0/…
/dev/qal0
/sys/devices/qal1/…
/dev/qal1
Job Queue
OS-Level Abstraction + RT-Control
Other specific Device Driver
QPX specific Device Driver
Ctrl
MSI-X
DMA
1. Simulated Environment: QEMU + Verilator
2. FPGA Implementation
QPX HW
PCIe
Control
Fidelity Mode
DMA
Hardware Design and Architecute
Main Memory

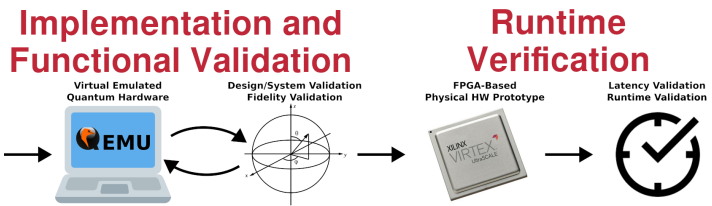

Virtual Emulated Quantum Hardware
QEMU
Design/System Validation Fidelity Validation
FPGA-Based Physical HW Prototype
XILINX
VIRTEX
ULTRASCALE
Latency Validation Runtime Validation

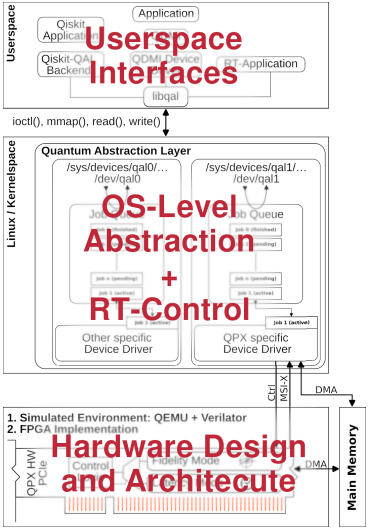


Implementation and Functional Validation




Userspace
Application
Qiskit Application
Qiskit-QAL Backend
QDMI Device
RT-Application
libqal
Userspace Interfaces
ioctl(), mmap(), read(), write()
Linux / Kernelspace
Quantum Abstraction Layer
/sys/devices/qal0/…
/dev/qal0
/sys/devices/qal1/…
/dev/qal1
Job Queue
OS-Level Abstraction + RT-Control
Other specific Device Driver
QPX specific Device Driver
Ctrl
MSI-X
DMA
1. Simulated Environment: QEMU + Verilator
2. FPGA Implementation
QPX HW
PCIe
Control
Fidelity Mode
DMA
Hardware Design and Architecute
Main Memory

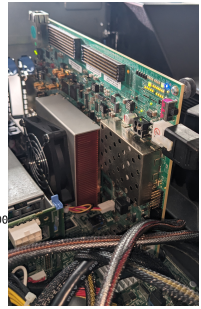

Implementation and Functional Validation
Virtual Emulated Quantum Hardware
QEMU
Design/System Validation
Fidelity Validation
Runtime Verification
FPGA-Based Physical HW Prototype
XILINX
VIRTEX
ULTRASCALE
Latency Validation
Runtime Validation




```
root@qal:~# uname -a
Linux buildroot 6.16.0-rc4 #1 SMP PREEMPT_RT riscv64 GNU/Linux

root@qal:~# dmesg | grep qal
qal_qpx_mmio 200000.qpx: Found QPX version 0.1
qal_qpx_mmio 200000.qpx: Maximum number of qubits: 100
qal_qpx_mmio 200000.qpx: Maximum shots per job: 100000
qal_qpx_mmio 200000.qpx: Registered /dev/qal0
qal_qpx_pci 0000:00:0e.0: enabling device (0000 -> 0002)
qal_qpx_pci 0000:00:0e.0: Using MSI(-X) Interrupt 20
qal_qpx_pci 0000:00:0e.0: Using MSI(-X) Interrupt 21
qal_qpx_pci 0000:00:0e.0: Found QPX version 0.1
qal_qpx_pci 0000:00:0e.0: Maximum number of qubits: 100
qal_qpx_pci 0000:00:0e.0: Maximum shots per job: 100000
qal_qpx_pci 0000:00:0e.0: Registered /dev/qal1

root@qal:~# qal_execute /dev/qal0 /share/circuits/Hadamard.qal 10000
/dev/qal0: Job 0 finished: The operation was successful
Dumping Job 0 (10000 shots)
Histogram:
  0x0000001f 4964
  0x00000000 5036
Closing /dev/qal0
```



Implementation and Functional Validation

Runtime Verification