# Sinara: An Open Hardware Ecosystem for Quantum Physics

D. Allcock[1], C. Ballance[2], S. Bourdeauducq[3], J. Britton[4], M. Gaska[5], T. Harty[2], J. Jarosinski[5], R. Jördens[3,6], P. Kulik[5], D. Nadlinger[2], K. Pozniak[5], T. Przywozki[5], D. Slichter[6], M. Sowinski[5], W. Zhang[2], G. Kasprowicz[5]

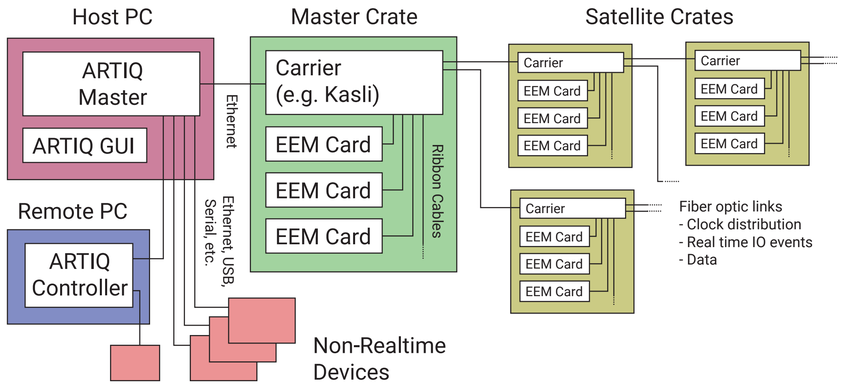
[1]University of Oregon, [2]University of Oxford, [3]M-Labs Ltd., [4]Army Research Laboratory,
[5]Warsaw University of Technology, [6]QUARTIQ GmbH [7]National Institute of Standards and Technology

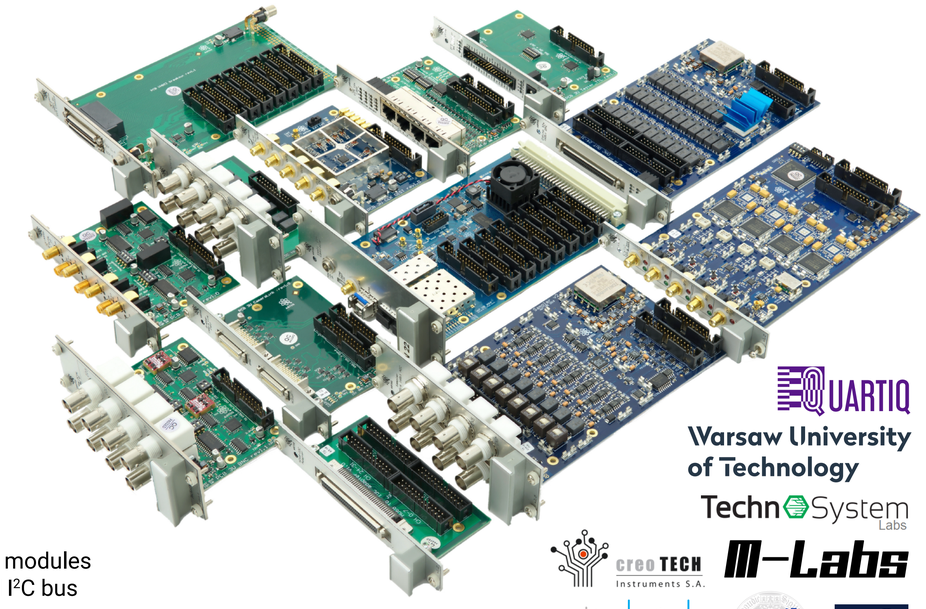
open source hardware

**https://sinara-hw.github.io**

## Abstract

Sinara is a modular, open-source measurement and control hardware ecosystem dedicated to quantum applications that require deterministic high-resolution timing. It is based on industrial standards and consists of over 50 scalable card-based modules built to perform a variety of analog (dc to microwave), and digital input and output tasks with precision timing. The hardware is controlled and managed by the ARTIQ open-source software platform, which provides nanosecond timing resolution and sub-microsecond latency via a high-level programming language.



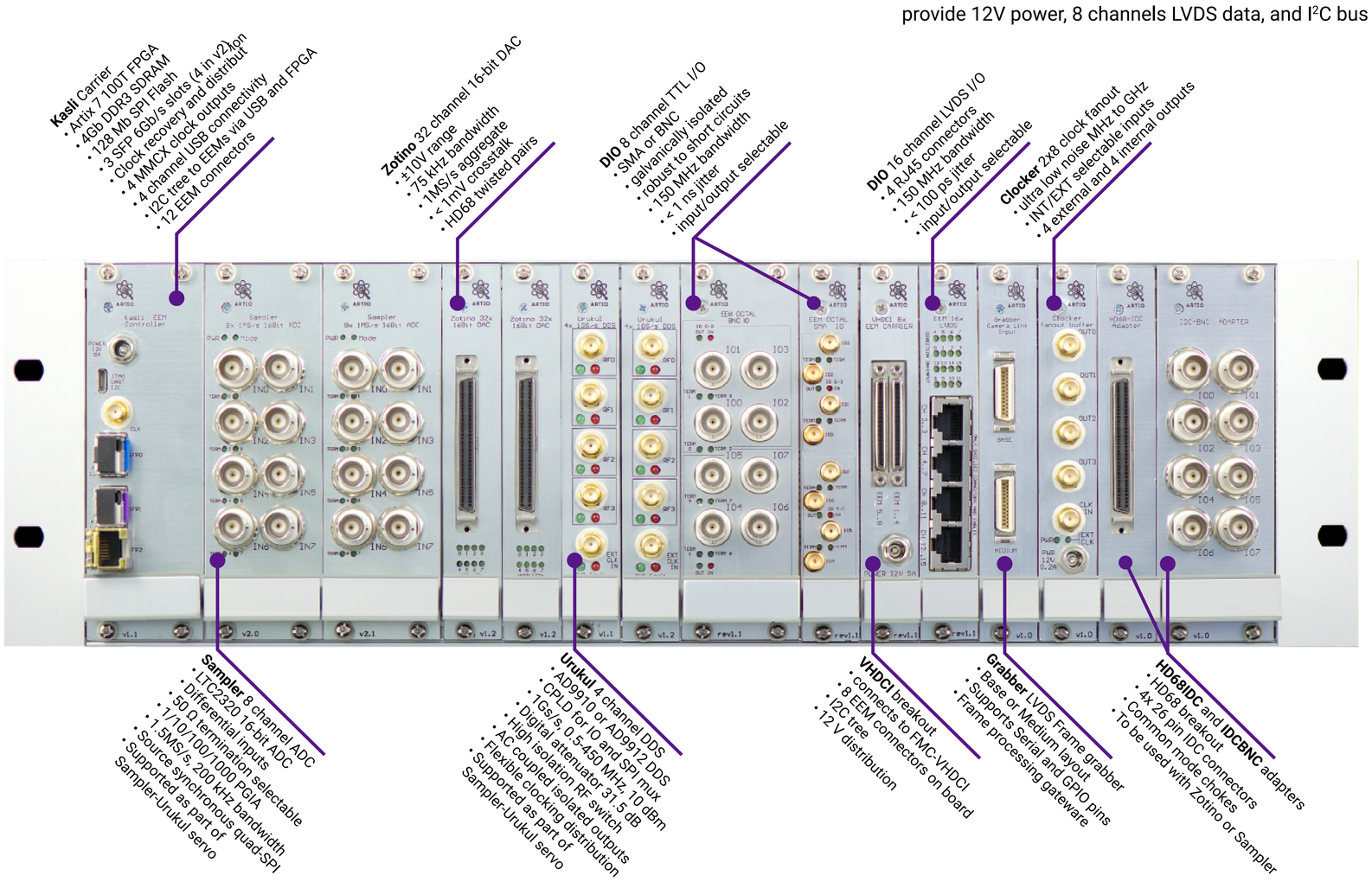



QUARTIQ
Warsaw University of Technology
TechnoSystem Labs
creotech Instruments S.A.
M-Labs
opticlock
UNI FREIBURG
UNIVERSITY OF OXFORD
MIT
PTB Physikalisch-Technische Bundesanstalt Braunschweig und Berlin
NIST
UNIVERSITY OF MARYLAND
Duke UNIVERSITY
UNIVERSITY OF OREGON
Leibniz Universität Hannover
ARL
JOINT QUANTUM INSTITUTE

## Production Hardware

Over 100 crates deployed in labs worldwide

**Eurocard Extension Module (EEM) standard:**
- 4HP or 8 HP 100x160mm Eurocards
- Ribbon cables between carriers and extension modules provide 12V power, 8 channels LVDS data, and I²C bus




## Prototype Hardware

Prototype modules have been produced in small quantities and tested but require further revision before being released to production.

**Stabilizer** CPU-based servo
- Dual 16-bit 2MS/s ADC with PGA
- Dual 16-bit 2MS/s DAC
- Cortex M7 CPU @400 MHz
- Ethernet (+PoE) and EEM interfaces
- Capable of PID with 500 kHz bandwidth

**Pounder** 2-ch PDH lock generator
- Add-on card for Stabilizer
- 4-ch 500MS/s DDS + phase detectors

**Phaser** 2-ch arbitrary waveform generator
- 4-ch 1.25 GS/s 16-bit DAC
- 2-ch IQ mixer upconversion to 0.3-4.8 GHz
- 2-ch of 5 MS/s ADC
- Artix XC7A100T FPGA

**Mirny** 4-ch programmable synthesizer
- 30 MHz to 6 GHz (12 GHz with planned Mezzanine board)
- CPLD for IO and SPI mux

**Fastino** 32-ch 16-bit DAC
- ±10V range
- 2 MS/s per channel, 1 GS/s aggregate
- High speed version of Zotino

**Sayma** 8-ch arbitrary waveform generator
- 8-ch 1.25 GS/s DACs
- 8-ch 5 MS/s ADCs
- Kintex UltraScale KU040 FPGA
- uTCA form factor

**Booster** 8-ch power amplifier
- 40-500 MHz
- P1dB 36 dBm, 40 dB gain
- >35% total power efficiency
- Protection, monitoring
- Remote control

**DIO_MCX** 16-ch I/O
- MCX connectors
- 3.3V, 50R capable outputs
- Switchable direction & 50R term.

**Thermostat** 2-ch temperature controller
- Thermistor input
- 8W TEC/heater drive
- Cortex M4 CPU
- Ethernet interface (+PoE)

**Humpback** EEM carrier for single board computers
- Compatible with Nucleo 144, Beaglebone Black, Orange Pi Zero, ESP32, Wiznet WIZ550web
- Provides power, mounting, and connectivity

**Banker** Versatile 128-ch GPIO
- ICE40 FPGA
- 3.3V or 5V with 50R drive capability
- DIN-rail breakout boards for D-sub, BNC, SMA, & screw term.

**Metlino** uTCA Carrier
- Kintex UltraScale KU040 FPGA
- uTCA form factor

## FAQ

**How do I find out more about the hardware?**
- The hardware designs are open source and on our GitHub: https://github.com/sinara-hw
- Documentation uses the GitHub wiki: https://github.com/sinara-hw/meta/wiki
- Issue tracking is done using the GitHub issue tracker on each repository

**Where do I obtain hardware?**
- Two commercial vendors (Creotech and TechnoSystem) can supply tested modules
- Two systems integrators (M-Labs and QUARTIQ) can provide assembled and tested crates as well as access to gateware/software build system
- As the hardware is open, you are free to produce your own boards

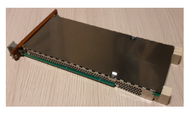
**Who develops Sinara?**
- Sinara is a collaboration between electrical engineering academics, commercial software and hardware suppliers, and the AMO research groups who form the end users
- G. Kasprowicz at Warsaw Technical University is the hardware lead

**Can physics groups without electrical engineers or FPGA developers contribute?**
- Yes! Use production boards, submit bug reports, and make suggestions based on your lab experience
- Much of the documentation is user-contributed and there are many gaps that need filling
- Fund new gateware/software development to add new features and help bring prototype boards to production

**I have a use case not covered by current or planned hardware:**
- Open a new request for comments at: https://github.com/sinara-hw/meta/issues and we'll work with you

## Infrastructure Improvements

**CPCIS**
- Forthcoming upgrade for EEM ecosystem
- Switch from ribbon cables to CompactPCI backplane
- Passive adapter to convert existing boards
- Compatible with CERN Distributed I/O Tier (DI/OT)



**uTCA**
- For boards that need more IO, power, and cooling than is available in CPCIS

## Planned Hardware

**Kasli SoC** Controller
- Zynq 7030 FPGA (dual ARM core)
- Kasli form factor

**Shuttler** 16-ch fast DAC
- 125 MS/s
- FMC form factor

**Fast Servo** FPGA servo
- Stabilizer form factor

**Driver** Diode laser current source
- 250mA/2V or 1.5A/3V

**Zapper** 8-ch piezo driver
- 16-bit DAC
- 100, 150, or 200V range

**Aux PSU** 3-ch power supply
- Switch-selectable output voltages

**Line Trigger** 50/60Hz line trigger
- To sync your experiments

**RFSoC**
- 8 channel 12-bit ADC, 4GS/s
- 8 channel 14bit DAC, 6GS/s
- uTCA form factor
- Will allow Sinara to be used for superconducting and spin qubits

## ARTIQ control framework

**https://m-labs.hk/artiq**

- **A**dvanced **R**eal-**T**ime **I**nfrastructure for **Q**uantum physics, integrated software/gateware system that controls atomic physics experiments
- High performance -- nanosecond resolution, hundreds of ns latency
- Expressive -- describe algorithms with a few lines of code written in a subset of Python
- Portable -- treat hardware (FPGA boards) as commodity, make software quickly and reliably deployable
- Modular -- separate components as much as possible: managing/scheduling experiments, driving distributed devices, analyzing/displaying/archiving results
- Open -- developed with and for research groups worldwide as open source software (LGPLv3)

ARTIQ