Oxidizing cryogenic sample storage in ultra-high vacuum

Scientific Computing in Rust 2026

Reto Trappitsch

EPFL

July, 08 2026

blog.galactic-forensics.space - @galactic-forensics.space

So you want to store frozen samples…

Scientific need

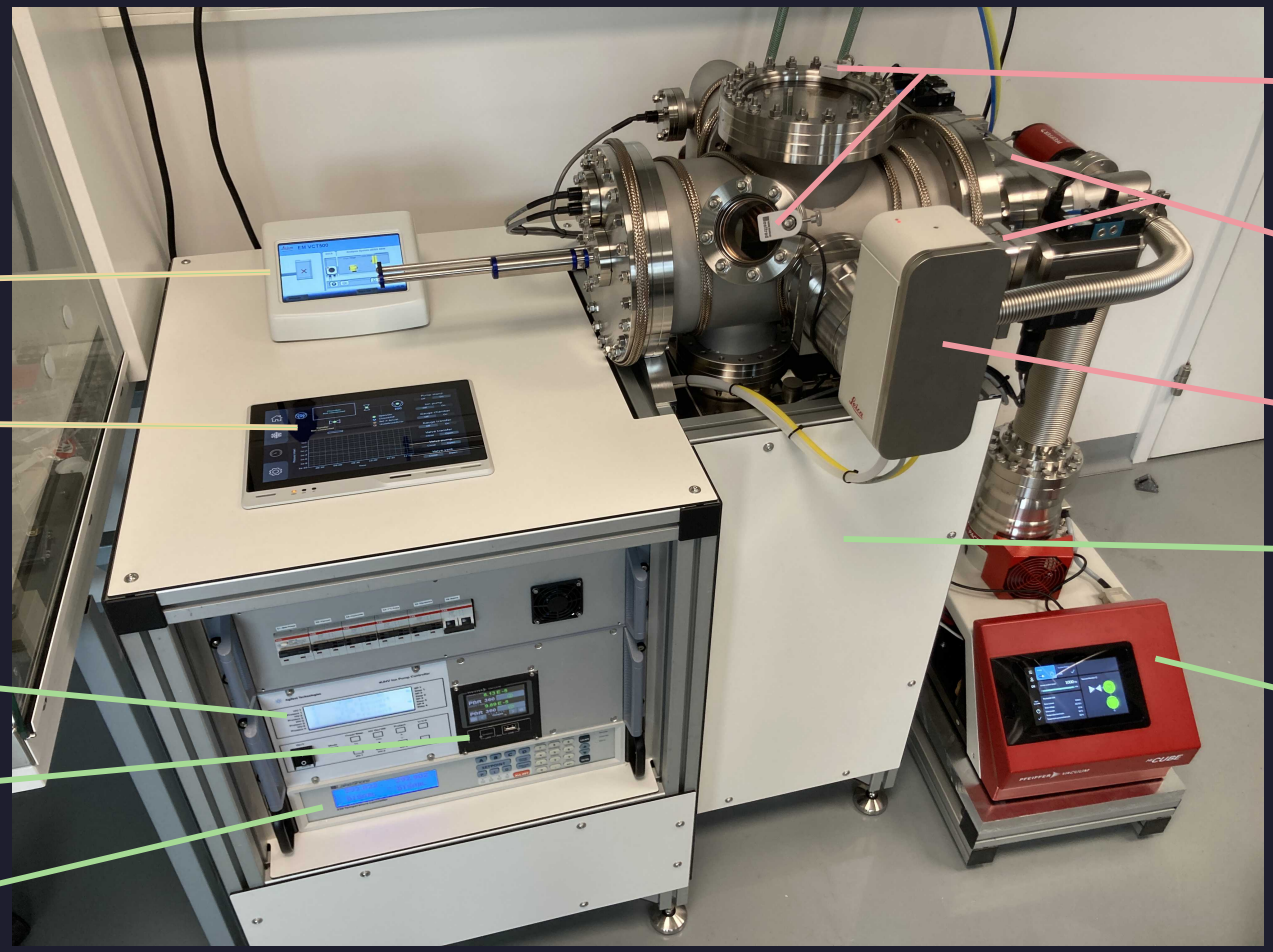
- Analysis of biological tissues in mass spectrometers
- Requirements:
  - Frozen or dehydrated
  - Ultra-high vacuum compatible
- High-pressure freezing leads to vitrification

  → Everything stays in place!

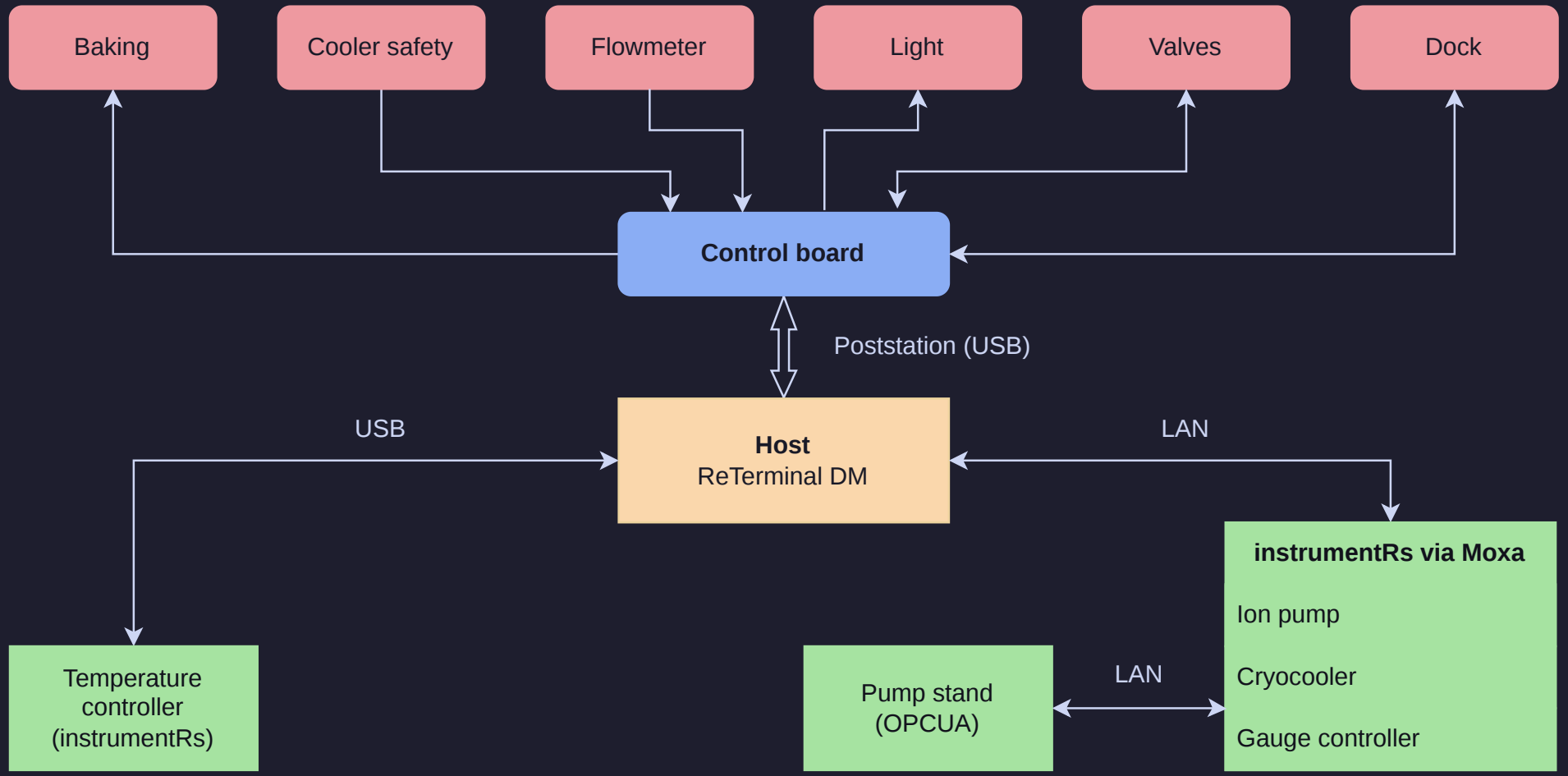
Storage of these valuable samples before analysis?

Quite a few components are needed!

Quite a few components are needed!

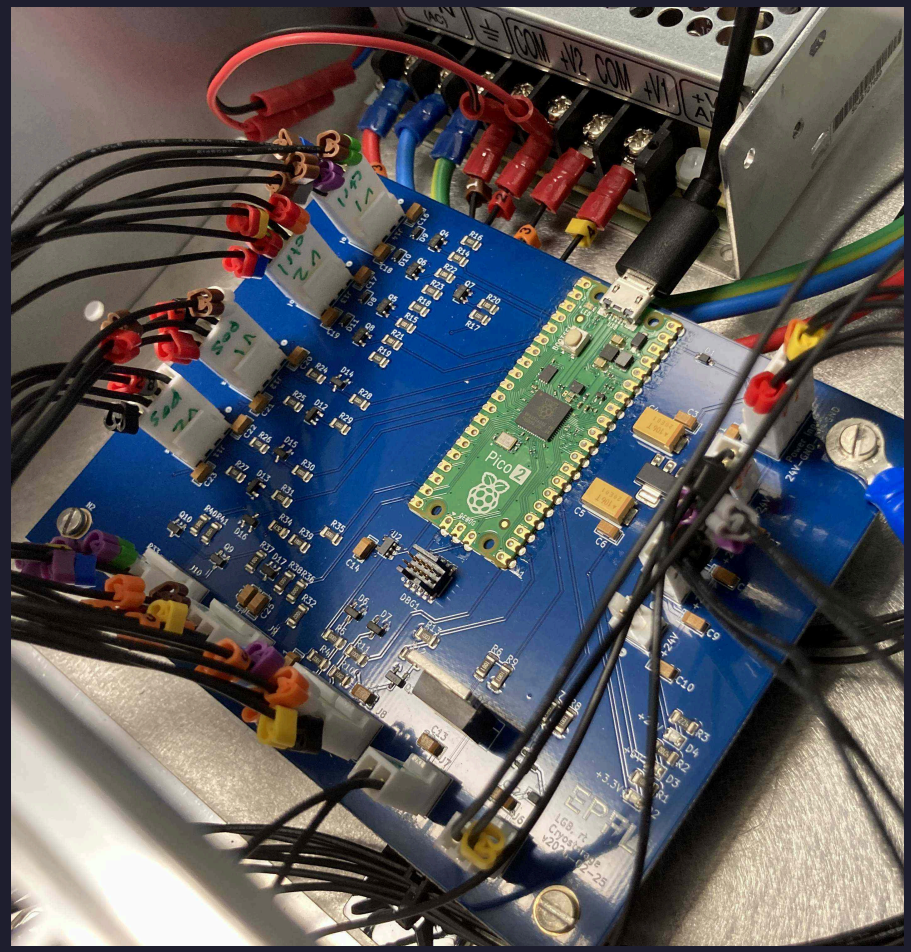


Overview of all instruments and controls



Home-built electronics via RaspberryPi Pico2

- Control lights, valves, baking
- Read flowmeter, valve state
- Hardware safety:
  - Cooler hardware interlocked
  - Transfer valve closing disallowed when dock is connected
- Rust-based firmware:
  - Async executor `embassy`
  - USB communication with host computer via `postcard-rpc`



Instrument communication: instrumentRs

- “Serial”: RS-232, RS-485
  - Connected to host via Moxa server
  - `serialport` crate
- OPCUA: Pump stand
  - Event based communication
  - `opcua` crate
- Standardize instrument control using `instrumentRs`
- All “serial” instruments are regularly polled for their status

`instrumentRs` example

```
let port = "/dev/ttyACM0"
let interface = new_serialport(port)
  .open()
  .unwrap();

let mut inst =
  Lakeshore336::new(interface);
let temp_a = inst
  .channel(Input::InA)
  .get_temperature();

println!("Ch A: {}", temp_a.unwrap());
```

Host software and UI

- UI based on Slint
  - Ideal for touch interface
  - Rust-inspired DSL to write UI
- Slint/Tokio collaboration for runtime
  - Listen for user interactions via UI
  - Poll instruments regularly
  - Listen to control board broadcast
- Control board communication via Poststation


Sample temperature
294 K
Chamber
1.22E-7 mbar
Transfer
4.85E-8 mbar
Start Sample Transfer
Vacuum Chamber
Vent
Pump
A1
empty
B1
empty
C1
empty
D1
empty
A2
empty
B2
empty
C2
empty
D2
empty

Sample temperature
294 K

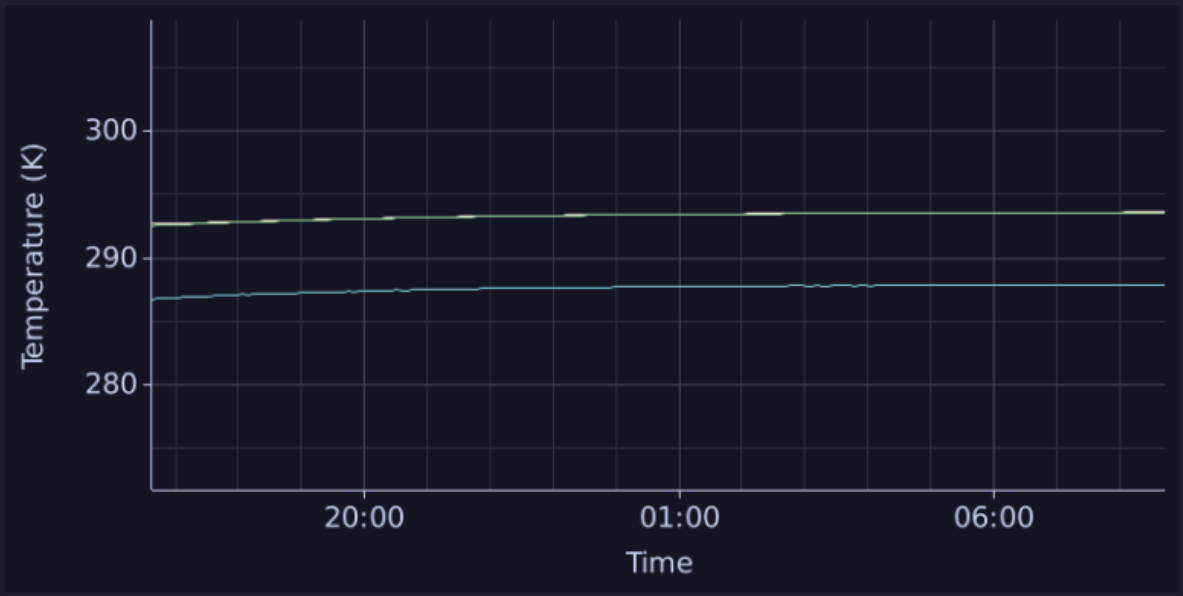
Bridge temperature
294 K

Cooler temperature
288 K

Cryocooler

Off | On

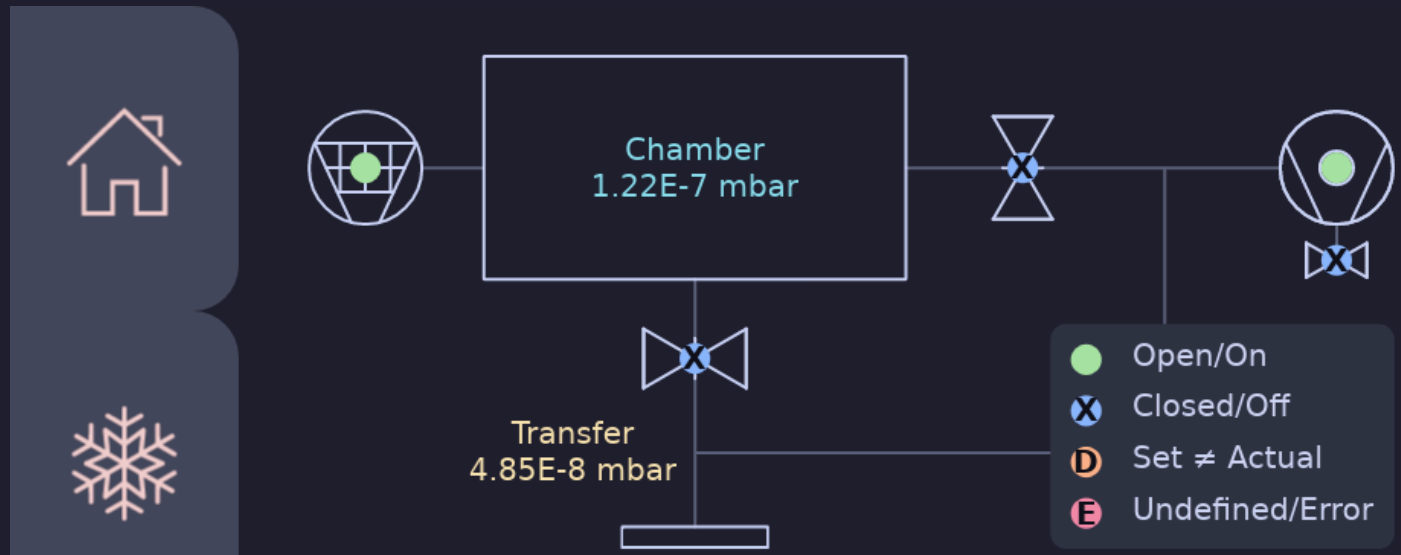
Water flow

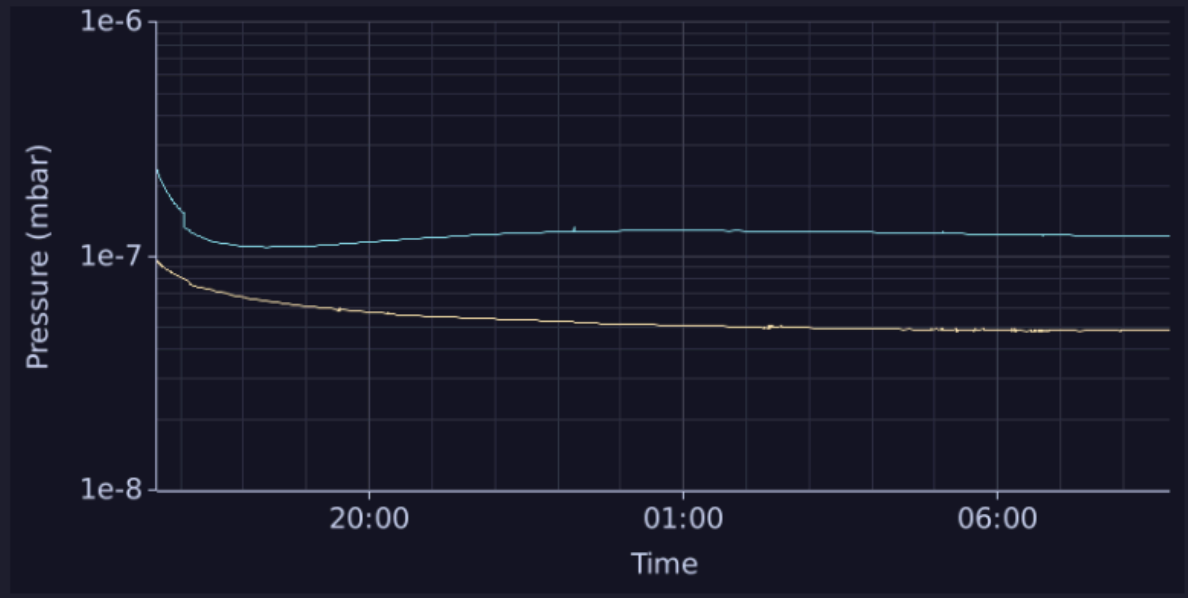


Target temperature
120 K

Cooler power
0 W

Set target temperature





Pump stand
Off | On

Ion pump
Off | On

Gauge chamber
Off | On

Gauge transfer
Off | On

Valve transfer
Close | Open

Valve pump
Close | Open

Valve vent
Close | Open

Ready for VCT?

No | Yes

Test alarm

Admin mode

Exit program

Log messages:

Baking

00:00:00

+ d | - h | + h | -10 min | +10 min | Clear

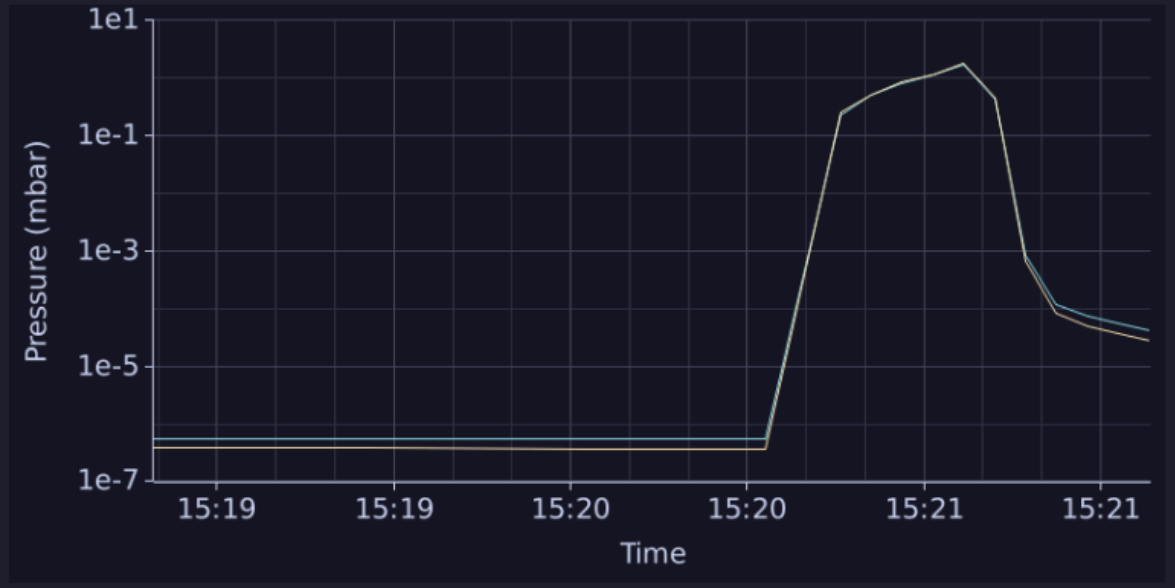
Stop | Start

Pumping the chamber

Pumping the chamber and transfer system. This process can take up to around 3h.

Waiting for chamber pressure to drop below 1e-5 mbar.

Remaining time:
00:39:24



Cancel

Rust is wonderful for lab integrations!

- Full stack in Rust: Firmware, communications, host software, and UI
- The type system allows development without connected hardware

instrumentRs v0.2.0

- Overhaul how instrument driver can be written
- Idea gathering repo: github.com/trappitsch/instrumentRs2
- blog.galactic-forensics.space
- Feedback is very welcome!

More information

- Code, Docs: github.com/trappitsch/cryostorage

Thanks to Rust our samples stay safe, cool, and do not oxidize!