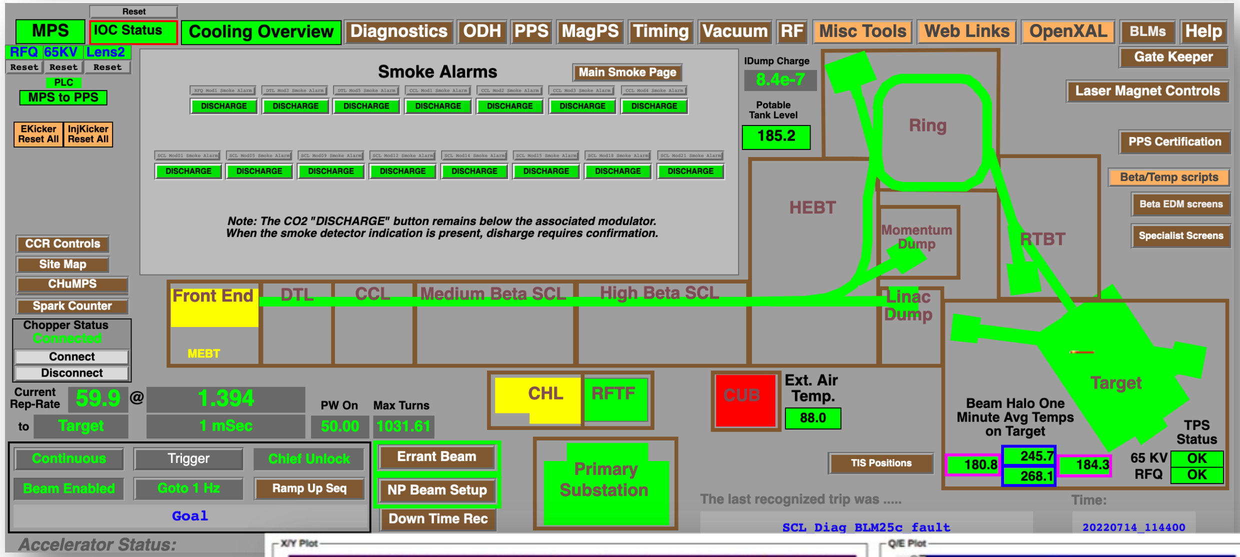
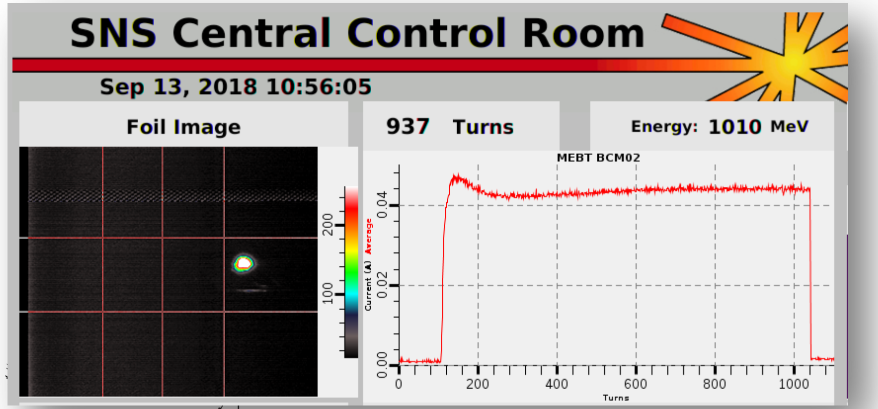
Control System Thoughts

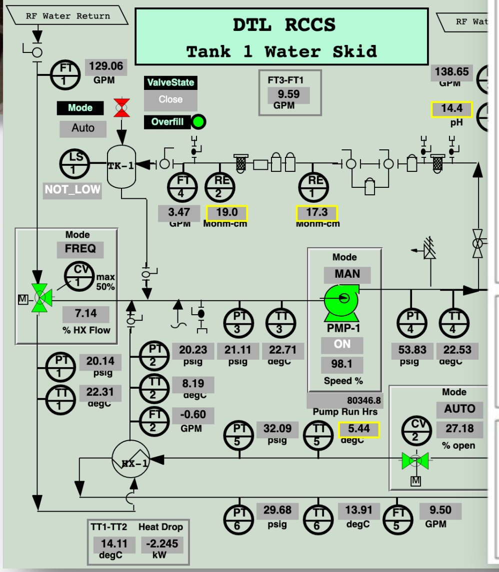
Kay Kasemir

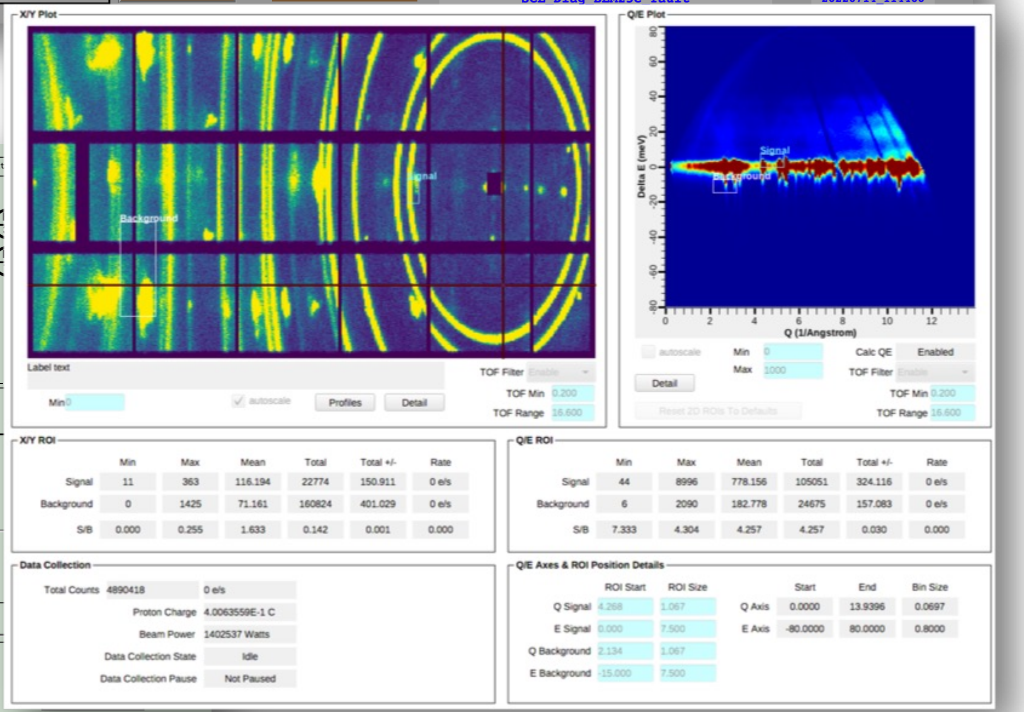
July 2026

Different Views of the Control System: What you see



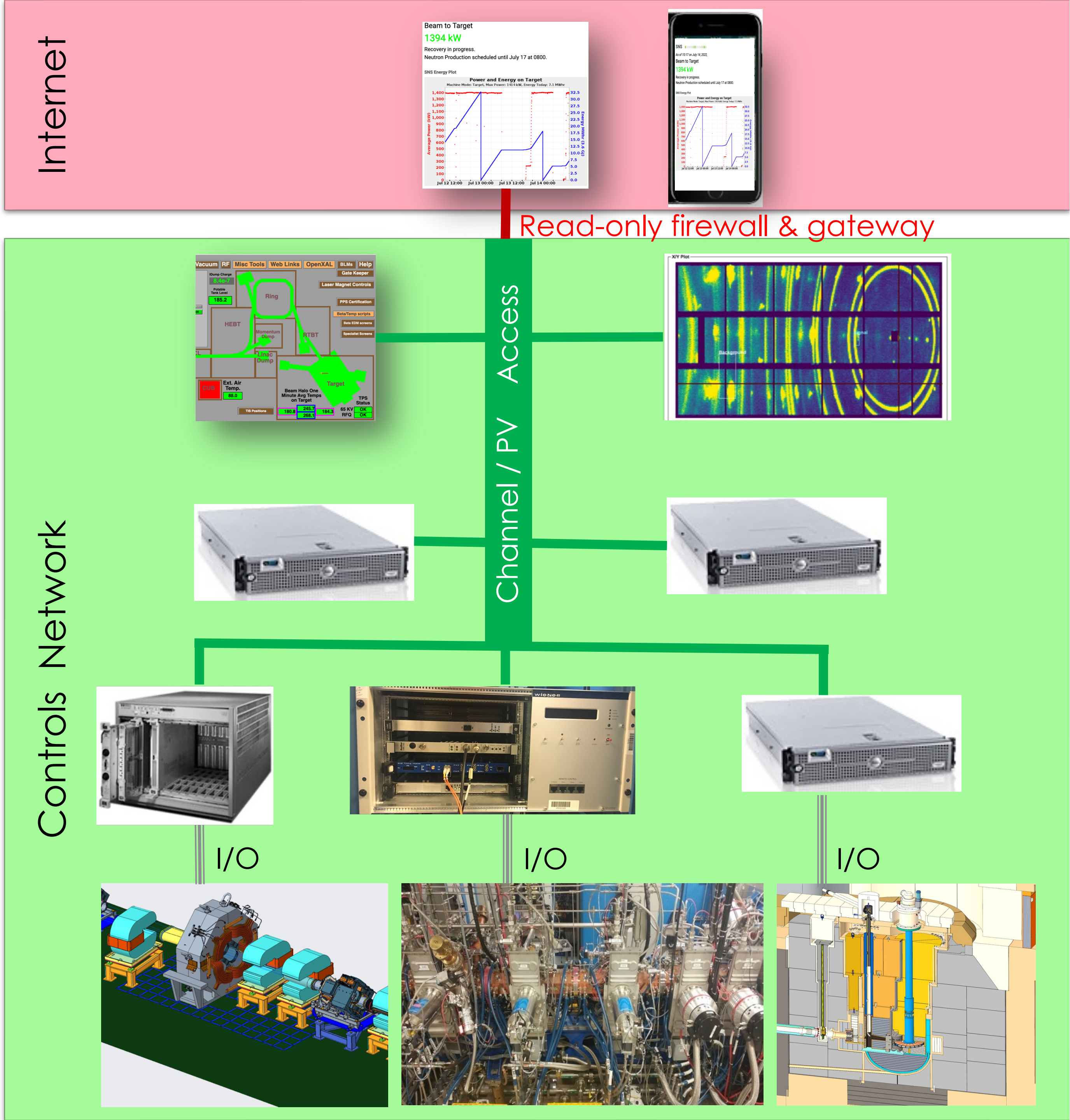






Different Views of the Control System:

Hierarchy of components



Oak Ridge National Laboratory | Spallation Neutron Source

Different Views of the Control System: Managerial

- Controls are roughly 5-12 % of the project cost
  - Scope:
    Do you buy the Power Supplies or just control them?
  - Important but outside of this:
    Work Breakdown Structure, risk registry, cable database, design reviews, building a team with necessary expertise (hire, matrix, contract?), …

- Most of controls cost is labor
- Most of that labor is spent on handling new devices unique to this project
  - EPICS provides the SCADA tools to support a large number of commonly used devices. You simply "configure" them.
  - EPICS allows adding support for unique, site-specific devices. This will require some "programming".

Integrated Control Systems

Integrate different types of devices

- Industrial controls
- Laboratory & scientific devices
- Custom hardware

Industrial Controls

- COTS sensors and controllers for temperature, flow, pressure, voltage, current, motor speed, position, ..
- For vacuum, cooling, PPS, …
- Rugged, designed to be panel/rack/rail-mounted
- Typically monitored every ~ second

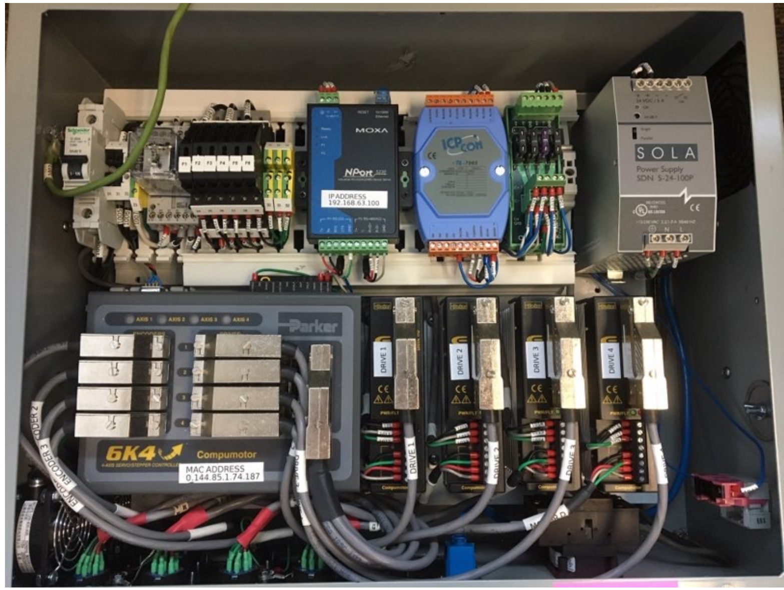
✓ Use reputable brands with track record

✓ Limit the number of variants

✓ Standardize on serial or ethernet interfaces

✓ .. with open, documented protocol

\- Avoid one-off gadgets that only come with Win95 software



Industrial Controls: PLCs

- COTS option for handling a lot of basic I/O
  - Low-voltage analog inputs and outputs
  - Binary inputs and outputs, low-voltage, 120V relays
  - Temperature and flow sensors
  - Motor controllers
- Can perform control loops and interlocks so IOC only monitors the result
- Typically monitored every ~ second

In several industrial settings, PLCs connected to Panel Views are the control system.

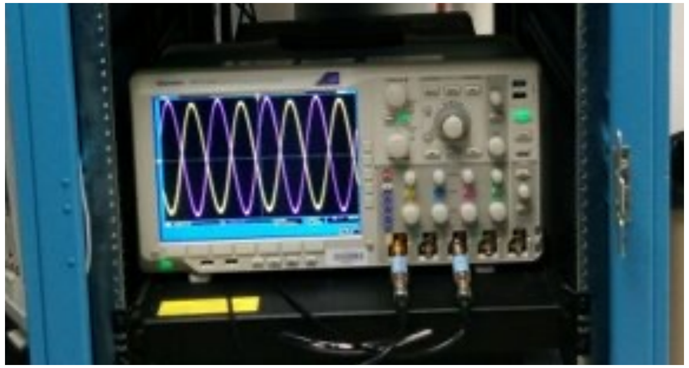
To us, PLCs are intelligent devices but nevertheless just another type of device to integrate into the larger ICS.

✓ Use established brands with existing EPICS support

✓ Agree on
  - Division-of-labor between IOC and PLC
  - Suggested arrangement of "tags" for optimized communication with IOC

Commercial Laboratory & Scientific Devices

- COTS waveform generators, frequency/trigger/delay generators, intelligent power supplies, temperature controllers, coordinated motor controllers, digital storage oscilloscopes, spectrum analyzers, gas analyzers, ..
- For LLRF, timing system, neutron detectors, cryostats and other sample environment, ..
- Highly accurate, more options than industrial controls
- Internally operating at kHz, MHz or GHz rates, with status monitored every ~ second via control system



Often meant for benchtop laboratory usage.
Harder to rack-mount, less robust, less reliable 24/7/365.

✓ Use reputable brands with track record

HP → Agilent → Keysight → …

✓ Limit the number of variants

✓ Standardize on serial or ethernet interfaces

✓ .. with open, documented protocol

\- Avoid one-off gadgets that only come with Win95 software

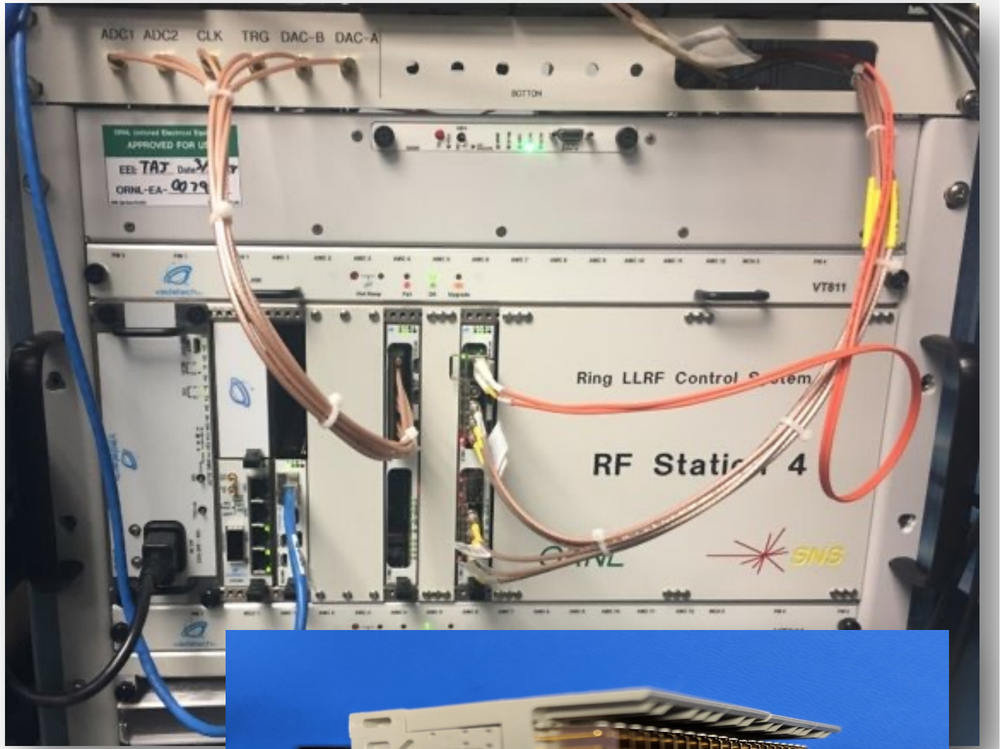
Custom aka High-Performance Hardware

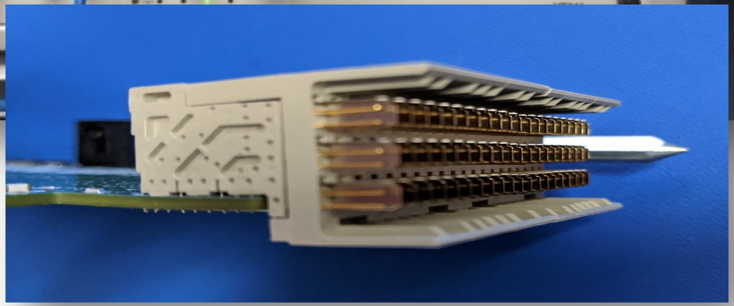
Timing system, LLRF, MPS, beam diagnostics, neutron detectors have unique requirements not met by COTS devices

- Exact timing down to nanoseconds
- Control loops and interlocks that can only be implemented in FPGA hardware
- Monitor status every ~second,
  but "stream" for example 10 million neutron events per sec.

Need to design, develop, implement,
then maintain hardware, FPGA firmware, software

- For ~100 LLRF systems, not 1.2 billion iPhones
- No bulk order discount for us!





Electronic components tend to be obsolete as soon as first devices are installed

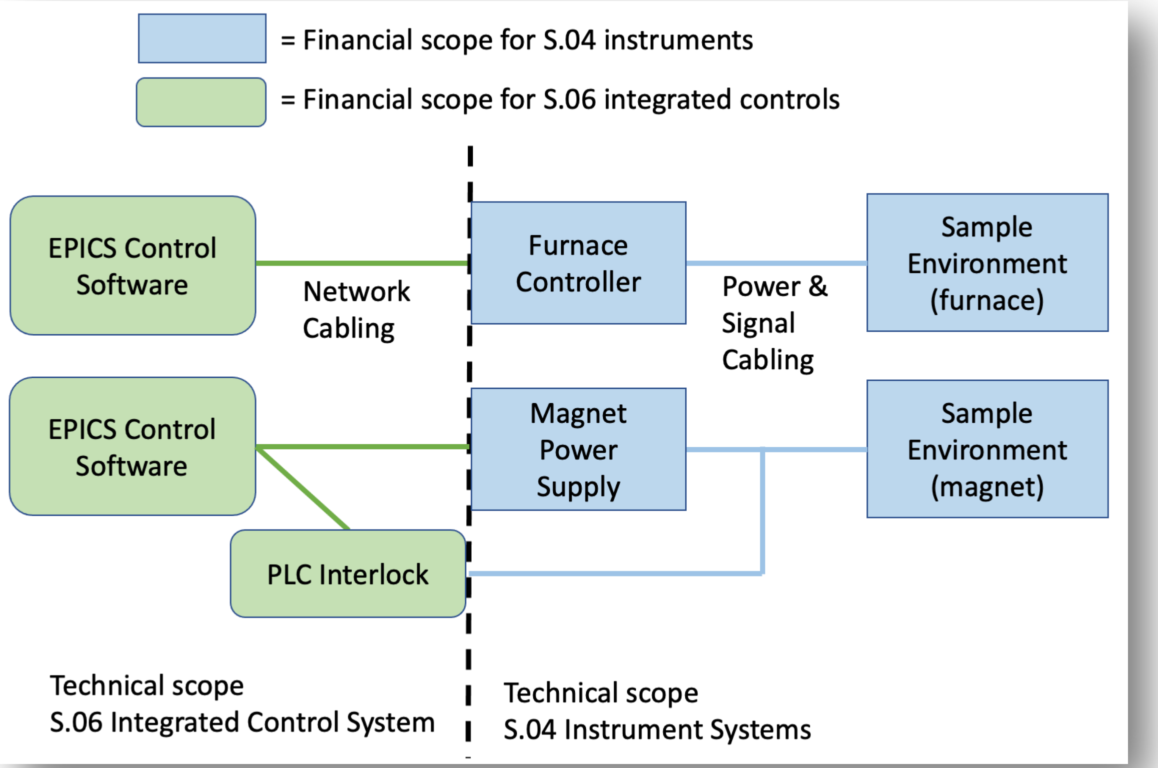
✓ Standardize on a few architectures

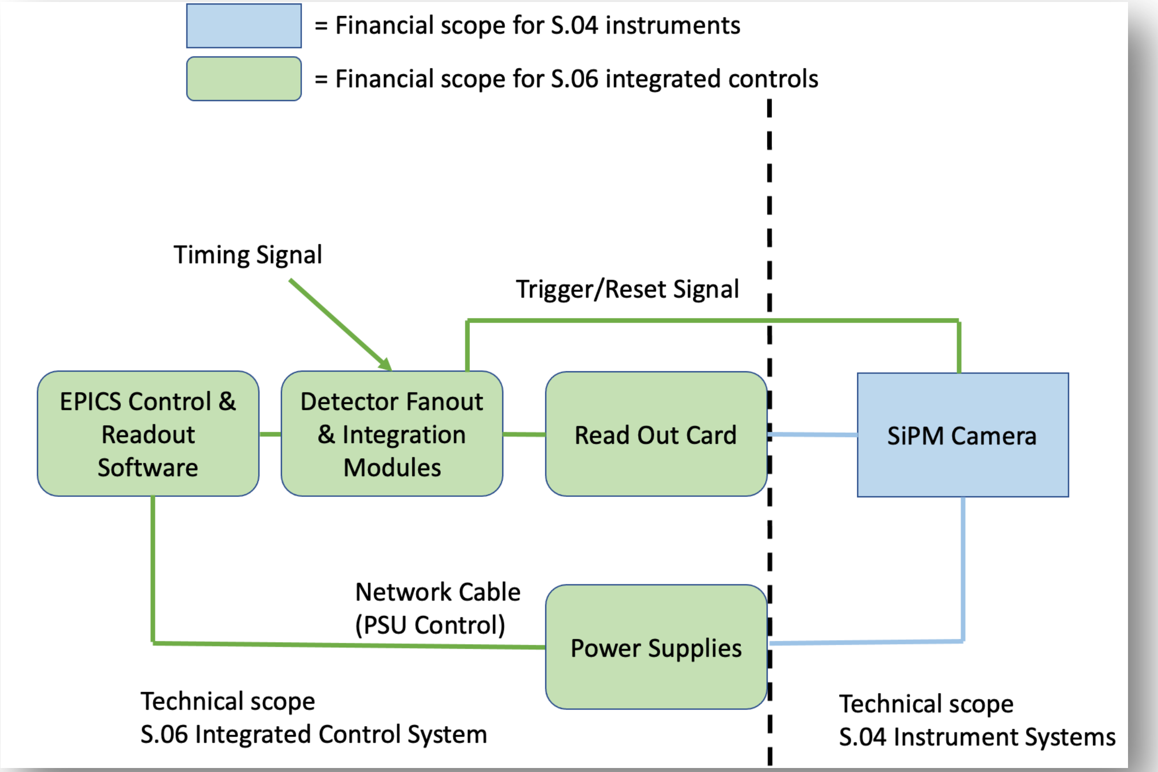
SNS started with VME, now focusing on μTCA

✓ Limit the number of FPGA variants ('Xilinx') and firmware languages ('VHDL')

✓ Re-use designs of timing receiver in LLRF, MPS, Beam Power Limiting System, …

Interfaces



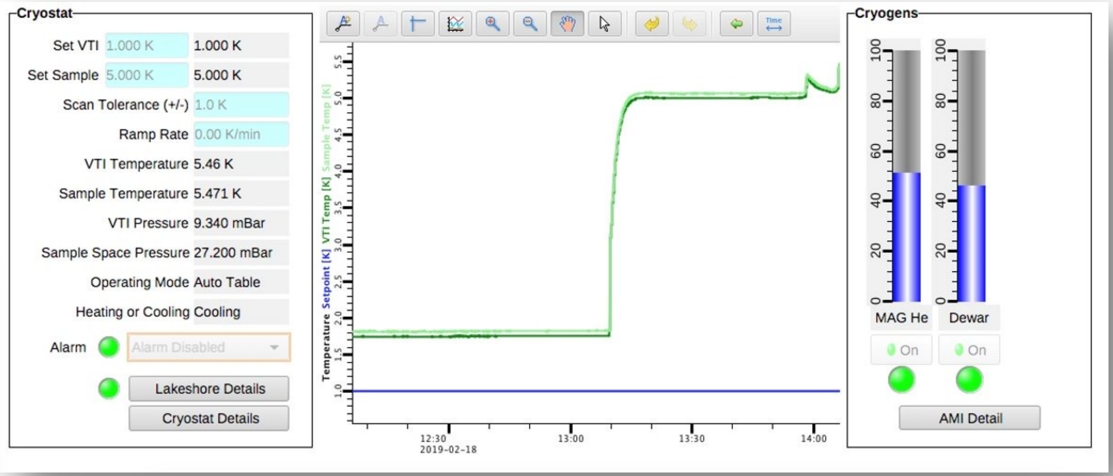


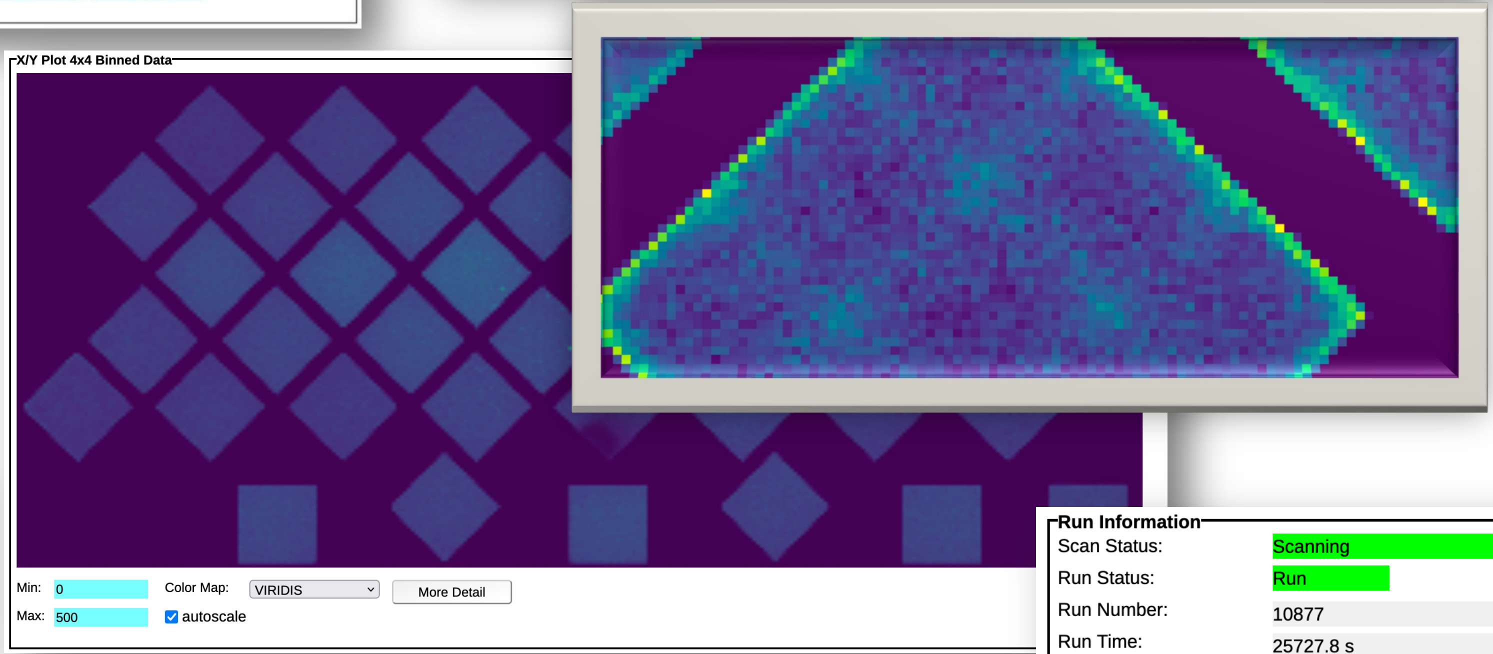
- Who provides the sensor, motor, controller, … ?
  - Does Power Supply go with device or with controls?
  - Who runs the cables?
- Case-by-case agreement

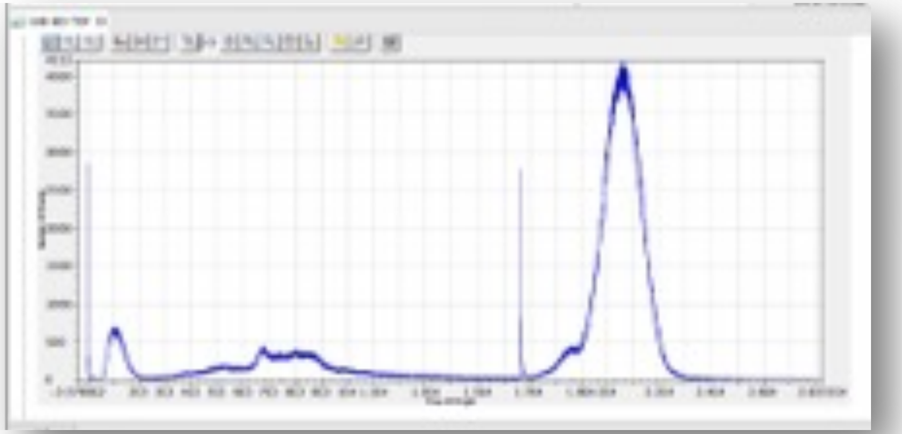
Matt Pearson, STS interface sheets

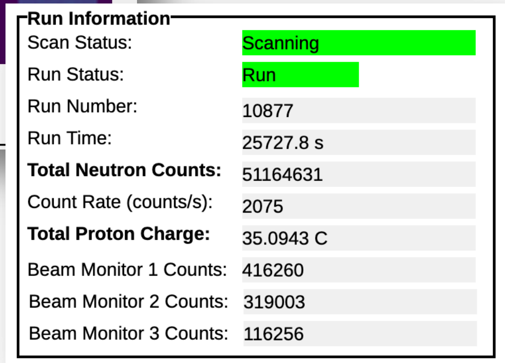
Integration for an Experiment

- Industrial & Lab devices
  - Motors
  - Cryostat
- Custom Hardware
  - Detector readouts, ‘raw’ position and d-space, q-space, …
  - Data acquisition status









This involves a lot of Software…

.. but GE is still trying to figure out how to be successful as a software company…

Software is different!

Since forever:

1. Collect Requirements

   Single person resting place,
   Additional storage,
   Huge!

2. Design

3. Build, build,…

4. Done!

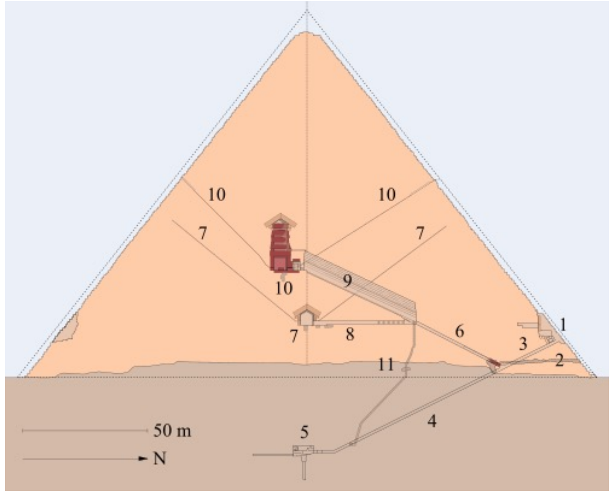
Software:

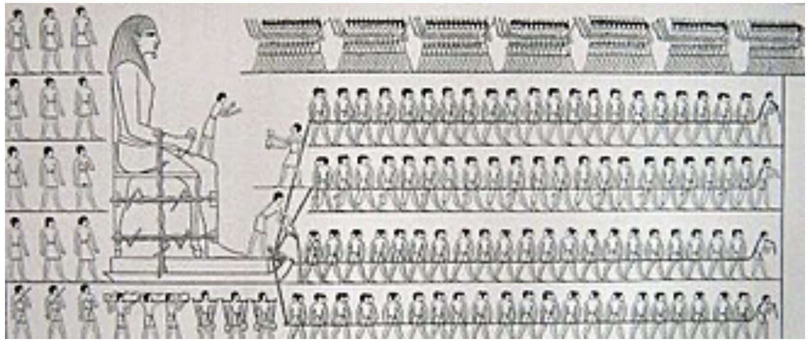
1. Limited Requirements
   Make the gadget that we're about to build work

2. Design, implement, rethink, implement, …

3. Deploy

4. New requirements → 1.



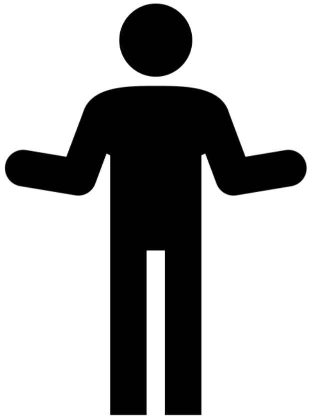


Involve ICS in the design process, avoid "build and then add software"

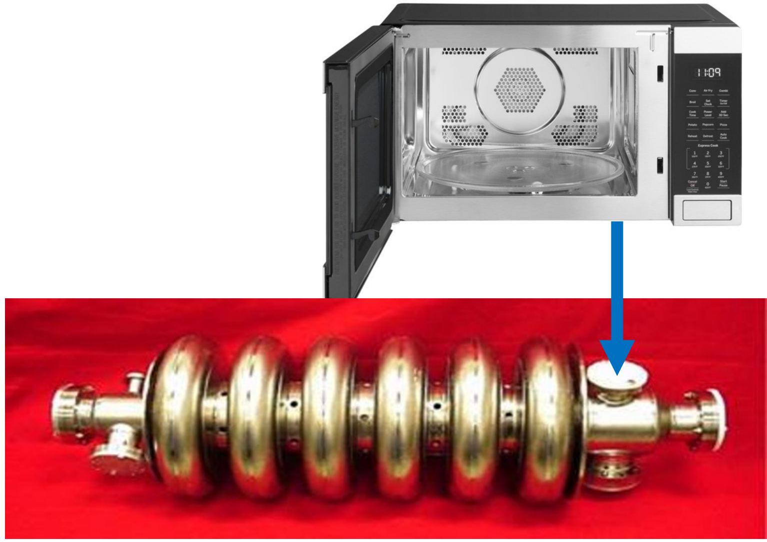
https://en.wikipedia.org/wiki/Egyptian_pyramids

Requirements are hard

- Controls: “Do the Beam Position Monitors need to be correlated?”
- Physics: “No”
  - Thinking: No need for exact correlation. 35 µs will be good enough
- Controls: “OK”
  - Thinking: No need for timing system. Let’s read them at 1 Hz



Control System is expected to be flexible and fill gaps in System Engineering



- Early SNS requirements example:
  1) Turn cavity RF on @ 60Hz to accelerate beam.
  2) Any cavity RF off? → Stop beam!

Implemented in LLRF and MPS hardware/firmware to be fast and reliable

.. then physics & ops decided accelerator much easier to tune up if we "coast" beam through idle cavities, but don't turn cavities off for long because that changes heat loads

→ 1) Turn RF off just for beam pulses
2) In spite of RF being off, allow beam!!

Add "beam blanking" to software and firmware:
Run RF @ 60 Hz except for on-demand beam pulses,
suppress the RF-off-error, …

Won't we soon have AI create the control system?

- AI is likely to become "a highly capable assistant"
- AI can create software, given the proper Prompt

Prompt = Requirements

- Requirements are still hard to get…

How can Control System Software ..

- ❑ Support many types of devices?
- ❑ Be well tested by many people?
- ❑ Be flexible?
- ❑ Plan for longevity?

- ✓ Don’t do it all by yourself. Share software with others
- ✓ Use open-source, so (time permitting) you can add/fix anything
- ✓ Distributed design allows independent addition/removal/update of components
- ✓ Apply saved time to the collaboration

EPICS

Open-Source Toolkit

- ✓ Not tied to any vendor
- ✓ We can see everything and if necessary, fix or adjust

Distributed

- ✓ Independently add/remove/upgrade components

Collaboration

- ✓ Started ~30 years ago at Argonne and Los Alamos National Labs
  For software, that is outstanding longevity
- ✓ Used by many accelerator and astronomy sites world-wide
  "Has anybody interfaced the XYZ gadget?"
  - "Yes, here you go" or
  "Well, the older version.
  Maybe you can start with that?"
  "We have a problem" – "We saw that, too,
  and fixed it by …"

Control Systems Summary

- Control system is about **Integration**
  - Industrial, laboratory/science and custom devices
- Collaboration and Standardization
  - Improve longevity