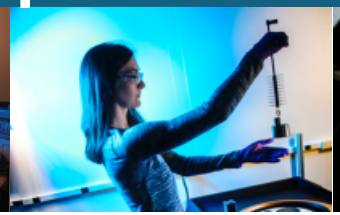
SAND2022-13928C

Sandia National Laboratories

Dishwashers in Space, formal verification of Satellites to benefit Nuclear Weapons?



PRESENTED BY

Noah Evans, 8741

U.S. DEPARTMENT OF ENERGY | NNSA National Nuclear Security Administration

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Our project: Deep Specifications of Sandia Systems of interest

- Essential idea: write specifications and hardware together in formal language, proving:
  A. That the specification is complete (i.e. not underspecified)
  B. That the implementation obeys the specification for all possible executions of the hardware artifact.
- Can think of this approach as 100% unit testing.
- This approach was previously impractical (10-20 person years for an OS/Hardware system) for everyone without billions of dollars to spend (i.e. Intel).
- New advances in “Proof Engineering” (Software engineering but for mathematics) makes it possible to modularly and reusably write proofs for large systems.
- Most people at the conference are likely familiar with this kind of work.
- We’re looking at applying these techniques to Sandia Systems of Interest.

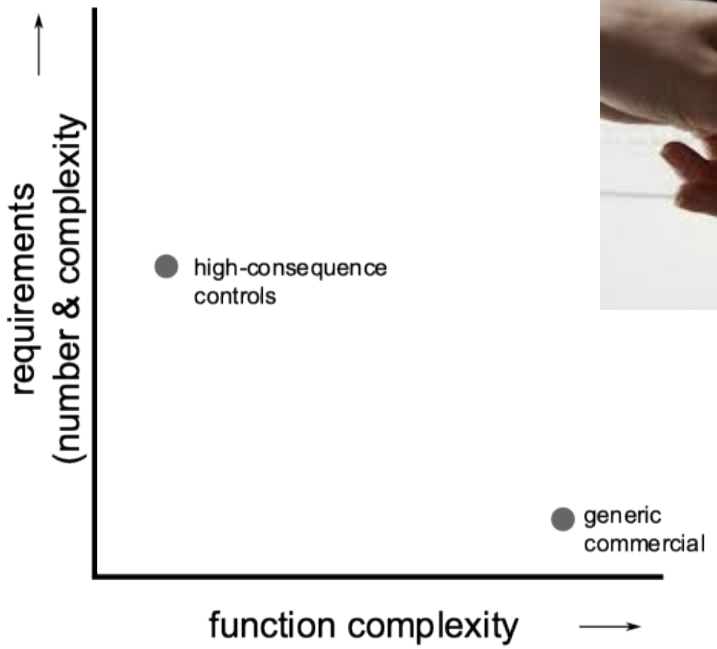
deep spec

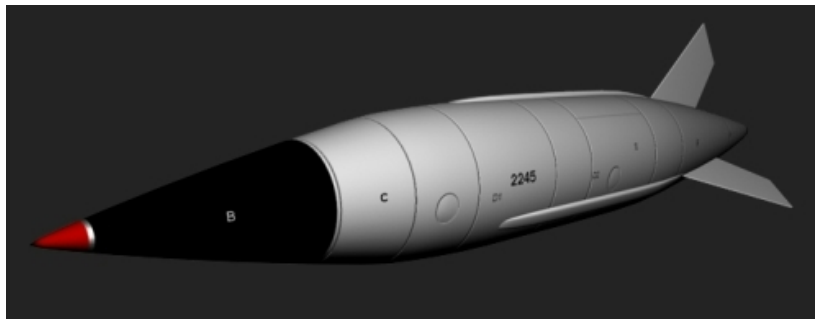
Requirements for Sandia Systems of Interest

Our control systems are mostly low complexity, relatively easy to analyze, like a dishwasher.

But, they often have a large number of complex, high-consequence safety, security, and reliability requirements.

Low complexity + high consequence + complex requirements = ideal for a formal approach to design and/or verification.





Heavily resource constrained: Back to the 80s future

This system is simple, dumb, and resource constrained

We build from scratch

Our own fab, our own processor, our own peripherals

Processor:

- 5-10 Mhz (can go 10-50 Mhz, for higher requirements, or Khz for low power)
- X Mbytes of Ram
- 100k total storage for bootimages
- No MMU

We write custom firmware to drive this currently.

Firmware is hard real time and hard to verify

The hypothetical firmware for this device is a simple event loop which counts cycles and makes sure that certain events fire at particular multiples of the clock frequency to meet real time deadlines.

Very close to an old Nintendo Entertainment system where games were implemented by using an event loop and cycle counting was used to blank the screen and communicate.

This leads to code which is classic ”spaghetti code”. No clear separation of concerns makes it very to mod[illegible]

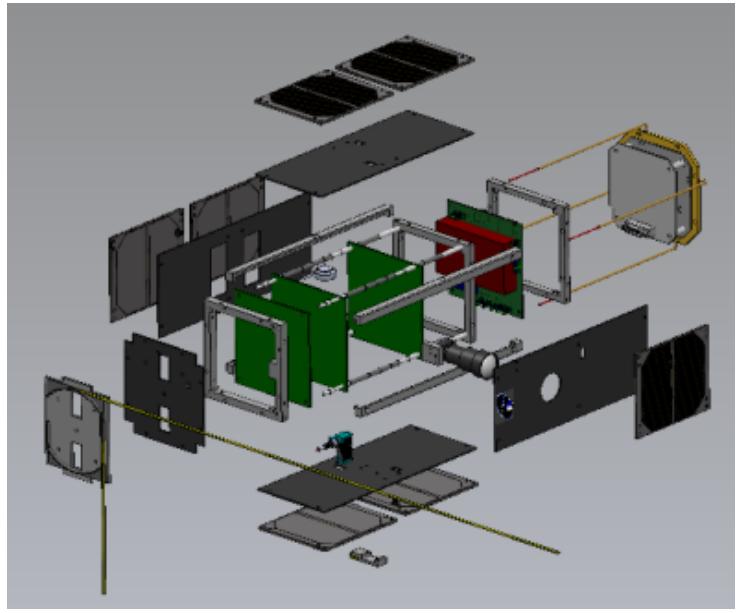
=

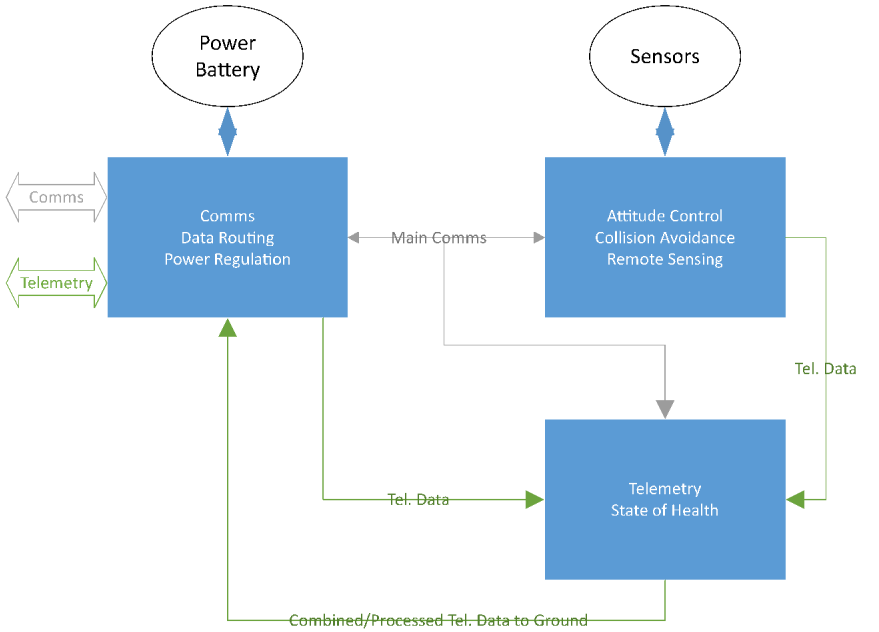
=

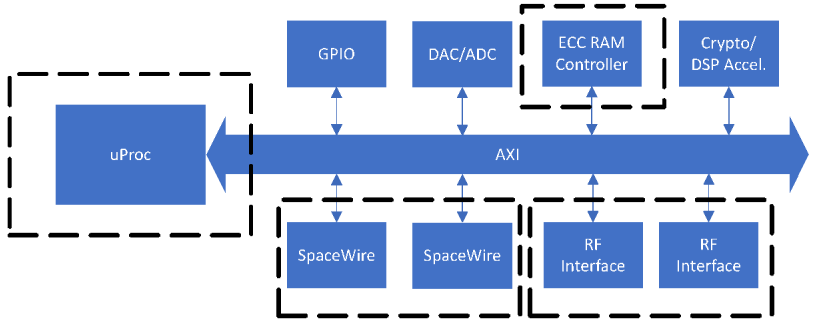
Proxy Architecture: a CubeSat analogous to Sandia Systems

Observation: Cubesats provide an unclassified analog to Sandia's typical sensitive systems of interest.

- Low complexity: 3 microcontroller class systems communicating over a bus
- High consequence: Any deviations from the specification mean you burn up in the atmosphere.
- Complex requirements: RF Comms, Collision Avoidance, Telemetry

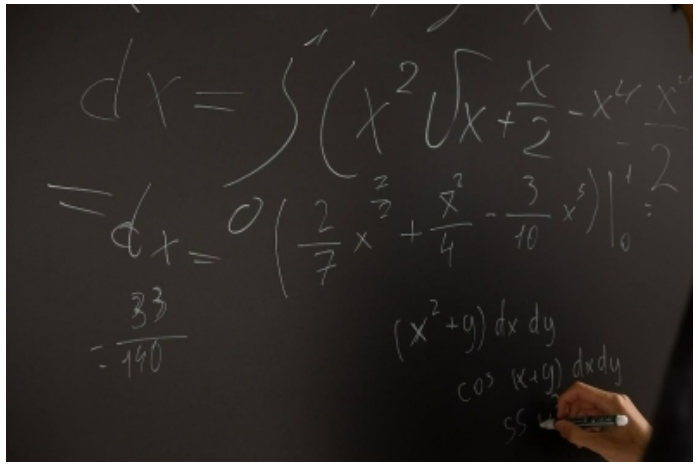






Goal: Reasoning about systems to enable tech transfer

- **Fundamental goal:** incorporate outside verification methodologies to verify the safety and security of Sandia Mission systems
  - **Probablistic**: how much do we know about the system?
    - Explore combined set of systems engineering artifacts, tests to go with the firmware
  - **Correct by construction**: systems engineering artifacts are progressive “refinements” of requirements in rigorous mathematical language.
    - Incorporate techniques to specify and run systems with provable relationships and behaviors
- Both need extensive cyber requirements.



Current Proxy Foundation: UPSat

- **UPSat**: Greek open source satellite out of the University of Patras in Greece.
  - Fully open source, full cad designs and firmware
  - Uses ARM STM32 microcontrollers and Texas Instruments Transceivers for Communication
- Close enough to our needs for now
- Will gradually develop it into a system the is fully representative (but still functional and useful as a satellite!)
- In collaboration with University of Patras spinoff Librespace.

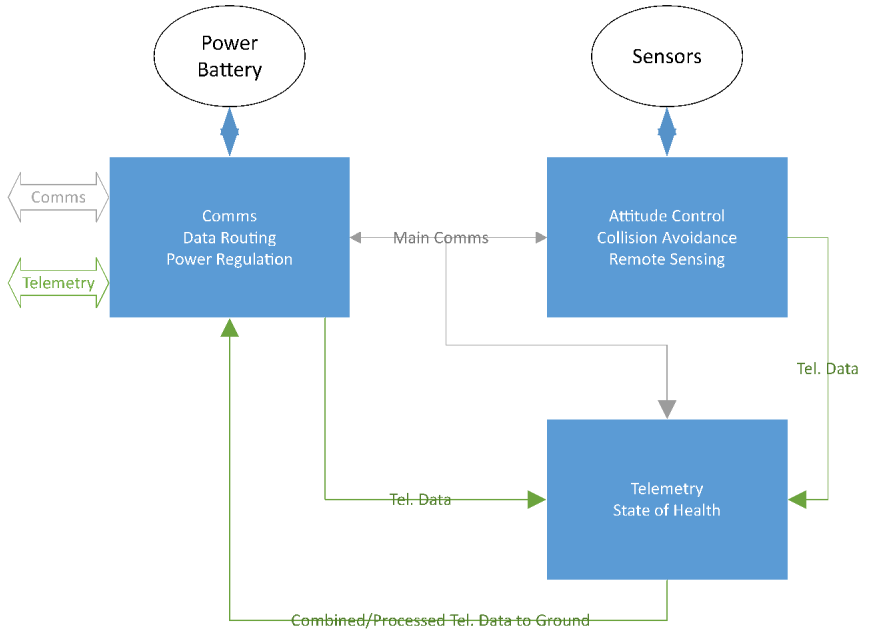
Sandia Future Proxy Design

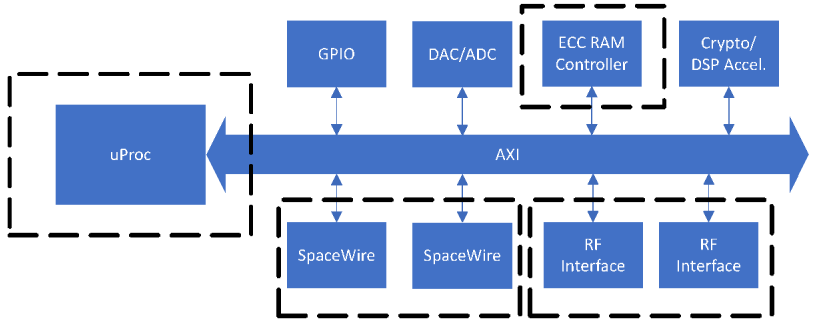
Move to a bus based, RISC-V architecture that is closer to Sandia's needs.

Same basic architecture as the upsat, but moving to industry standard

More complicated to verify but closer to the sorts of real systems we expect in the future.

Open Standards also allows us to use third party verification techniques and IP (e.g. AXI verification suites) without having to do all the bespoke verification in house.





Challenge: Relevant Cyber Requirements.

- Currently building a corpus of Cyber Requirements that are relevant to both Sandia ND and Space systems.
- Have the basic system designed, however, no systems engineering artifacts.
- we are currently developing a set of “best practices” Systems Engineering IP.
- **Goal:** a set of cyber requirements and systems designs that represent both Sandia’s and others needs.
- Currently leveraging Lincoln Labs and Galois IP from other formal methods projects (not available in this venue, but please contact them!).
- Would welcome any contributions from the community.

Example Requirement: Solar Panel Deployment

- Fundamentally destructive, failure to deploy correctly can lead to the destruction of the satellite
- A natural target for attackers.
- Verifying the functional correctness and cyber resilience of this behavior is imperative in the design of robust satellites.
- First requirement we are exploring

Conclusions

Sandia is developing a CubeSAT architecture for experimenting with verification and hardening of mission systems.

This work is in progress, but would benefit from community assistance.

- In particular we need use cases and cyber requirements to test assurance methods and verification methodologies.
- In particular requirements that are meaningful in an unclassified setting, but which can act as proxies for sensitive requirements
- Allow us to engage the broader verified systems community.