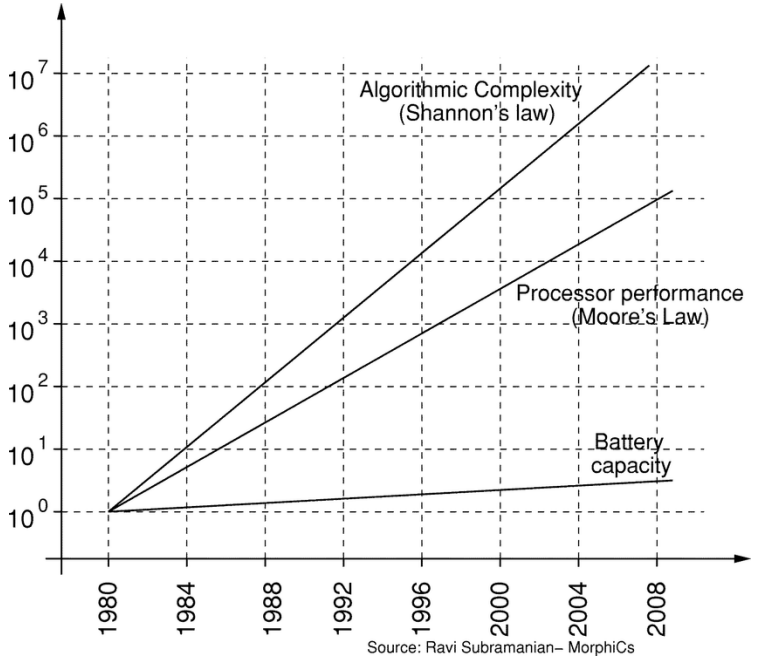
Failure Sentinels: Ubiquitous Just-in-time Intermittent Computation via hardware support for continuous, low-cost, fine-grain voltage monitoring

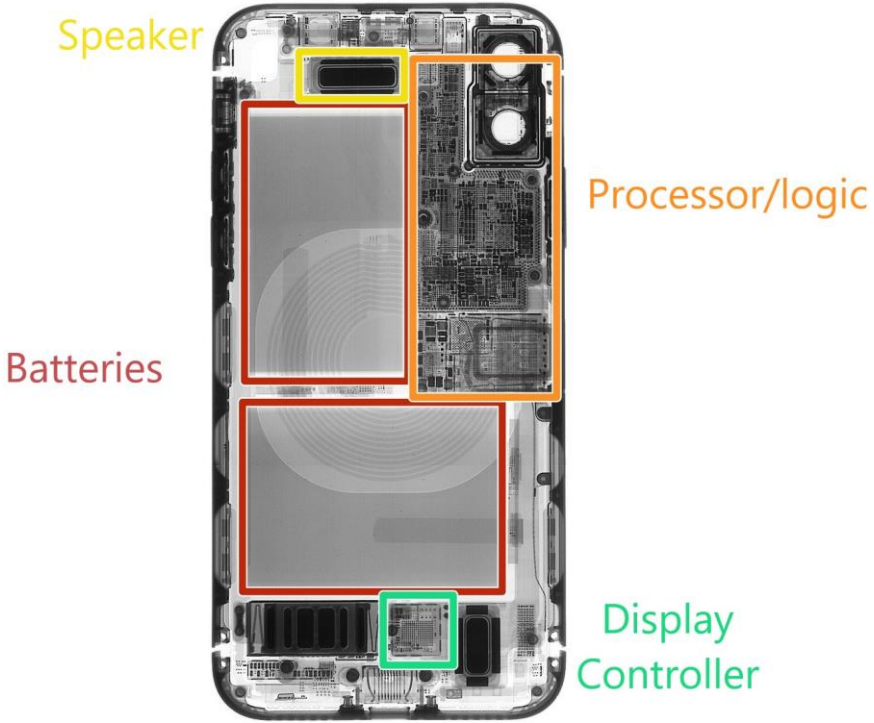
Harrison Williams
hrwill@vt.edu

Michael Moukarzel
mamoukar@vt.edu

Matthew Hicks
mdhicks2@vt.edu

Processors have outpaced batteries



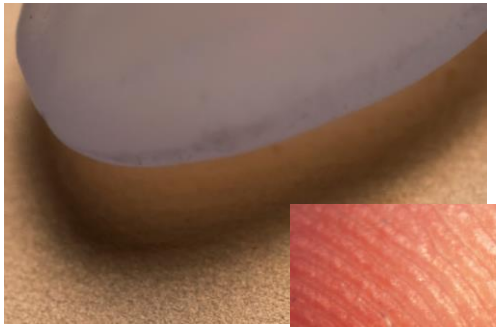


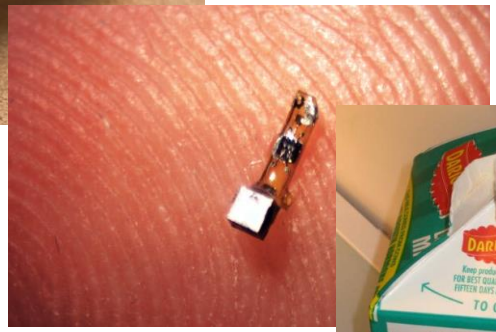
The future is batteryless!

Smaller →

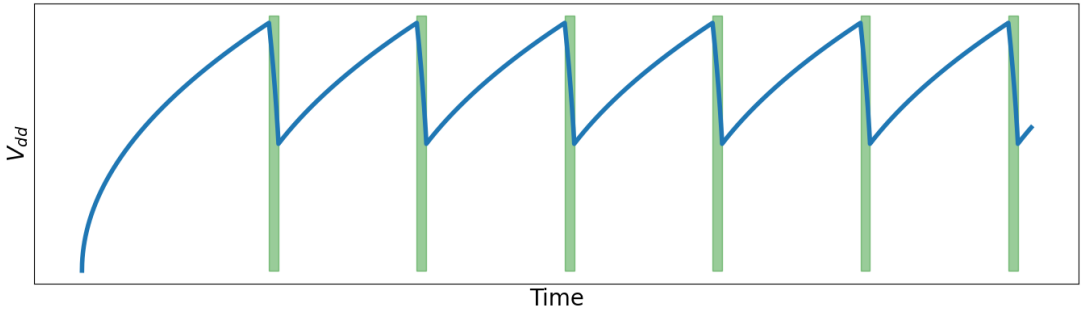
Safer →

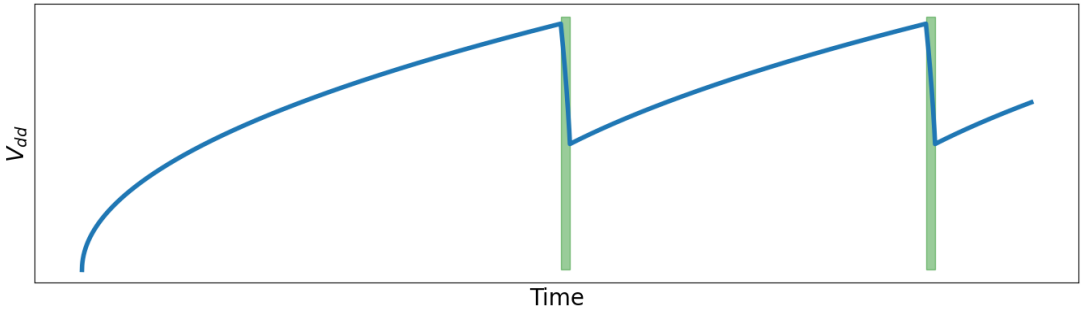
Cheaper →

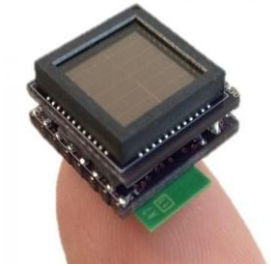




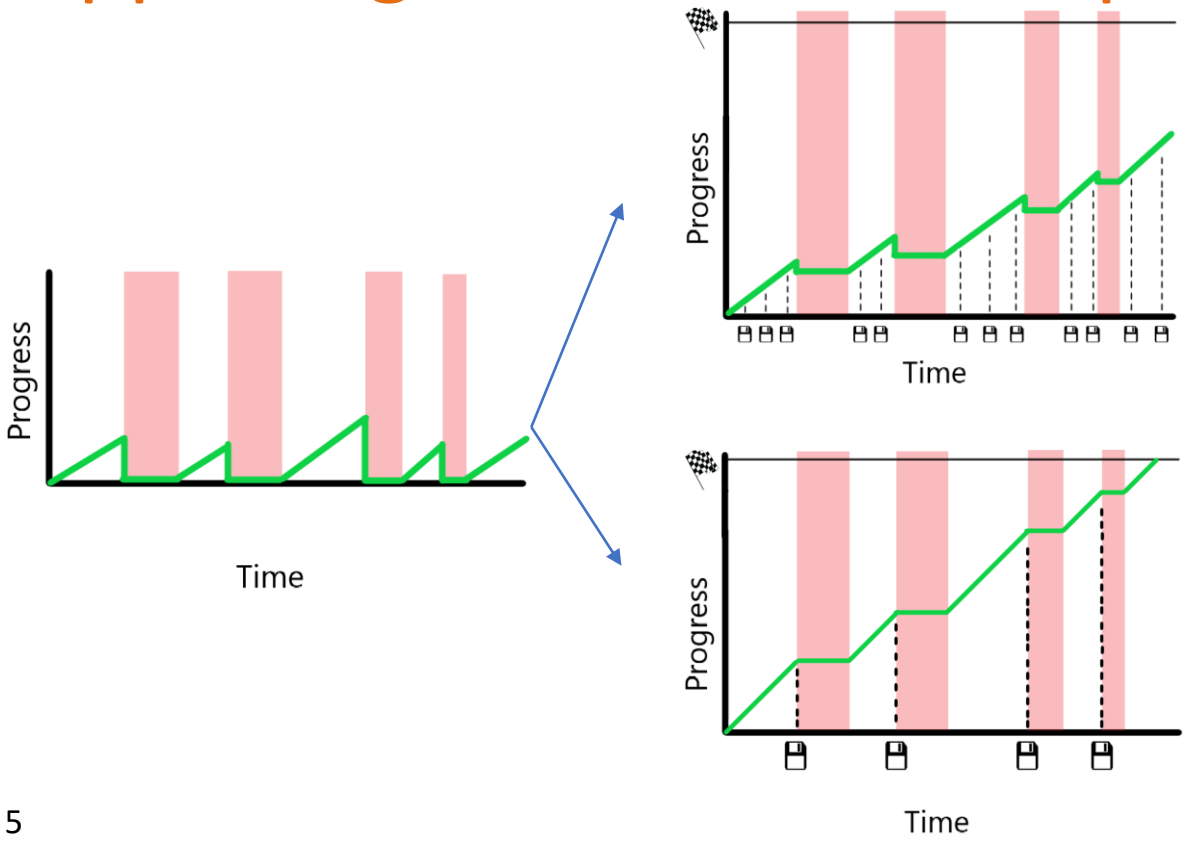
Challenge: harvested energy is unpredictable energy







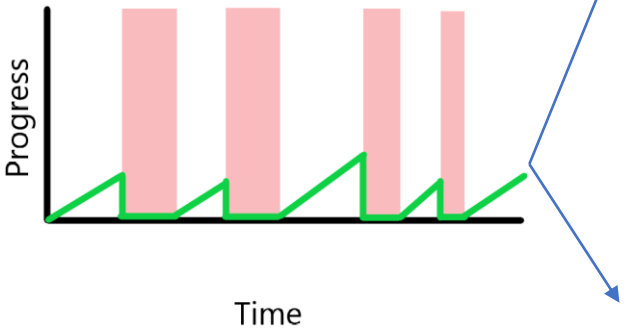
Supporting intermittent computation

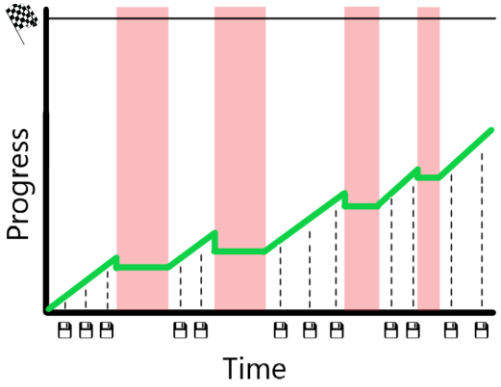


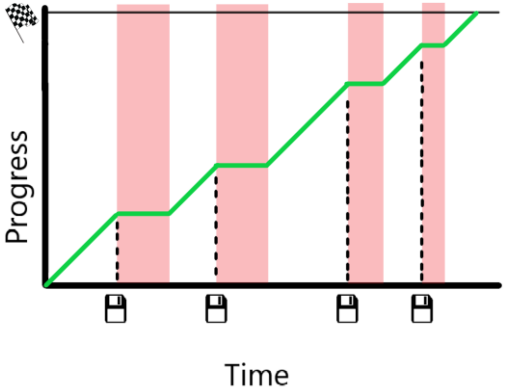
Continuously checkpoint through execution

Save state on power failure “just in time”

Supporting intermittent computation







- Software only
- High overhead (1.3-3.6x)
- Low overhead (1%*)
- **Hardware support**

Hardware support is inevitable

- Code instrumentation to reduce wasted energy
- Energy-aware task schedulers
- Energy-aware hardware accelerators

Promising intermittent systems all require **knowledge of energy availability**

What do we want in a voltage monitor?

1. Minimal power consumption
2. Sufficient performance
3. Maximal scalability

The cost of hardware support

Analog-to-Digital Converters (ADCs)

- ✚ High precision
- ✚ High sample rate
- ▬ **Massive** power consumption (>2x core power)

Power	Performance	Scalability
✗	✓ → ✗	✗

Analog Comparators

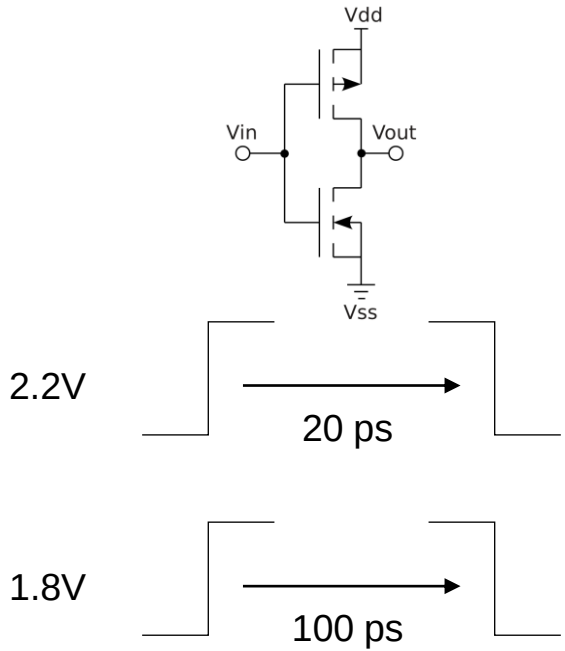
- ✚ Low response time
- ▬ Single bit precision
- ▬ High power consumption (30-80% of core power)

Power	Performance	Scalability
✗	✗	✗

CMOS gates reveal supply voltage

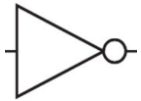
Observation: supply voltage sets digital gate delay

Idea: track gate delay to determine supply voltage

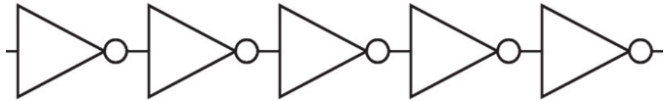


Using digital gates for voltage information

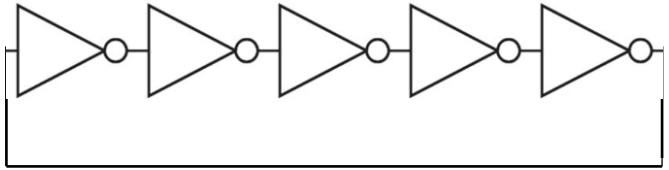
Single bit of information
(did signal propagate in time?)



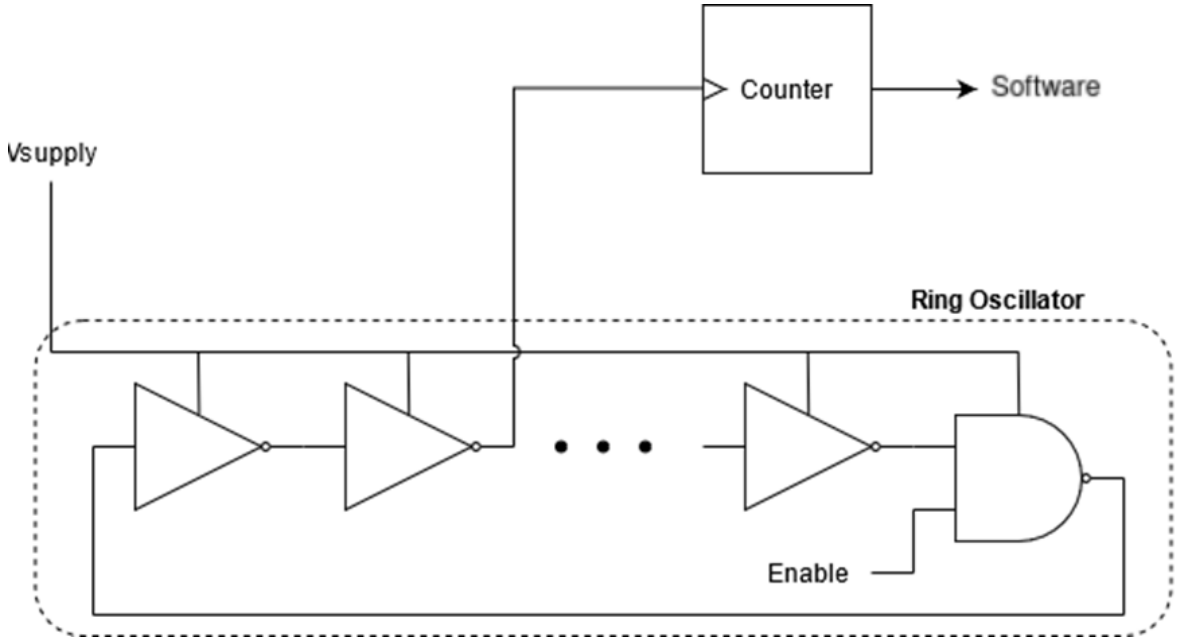
Narrow dynamic range
(how far did signal propagate?)



Wide dynamic range
(how many circumnavigations?)



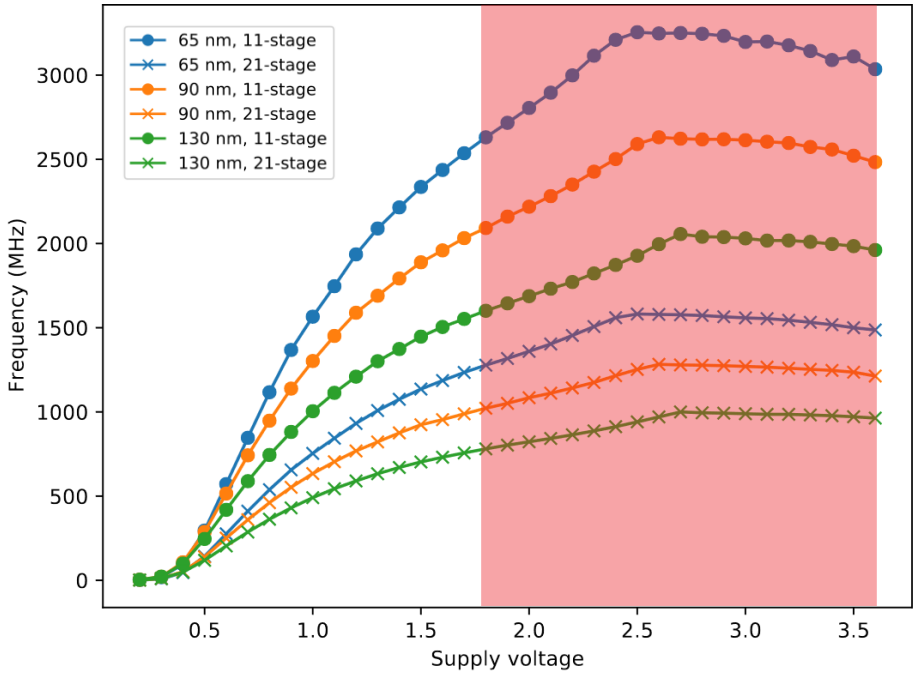
Failure Sentinels core



Mapping RO frequency to voltage

Not one-to-one

Low sensitivity

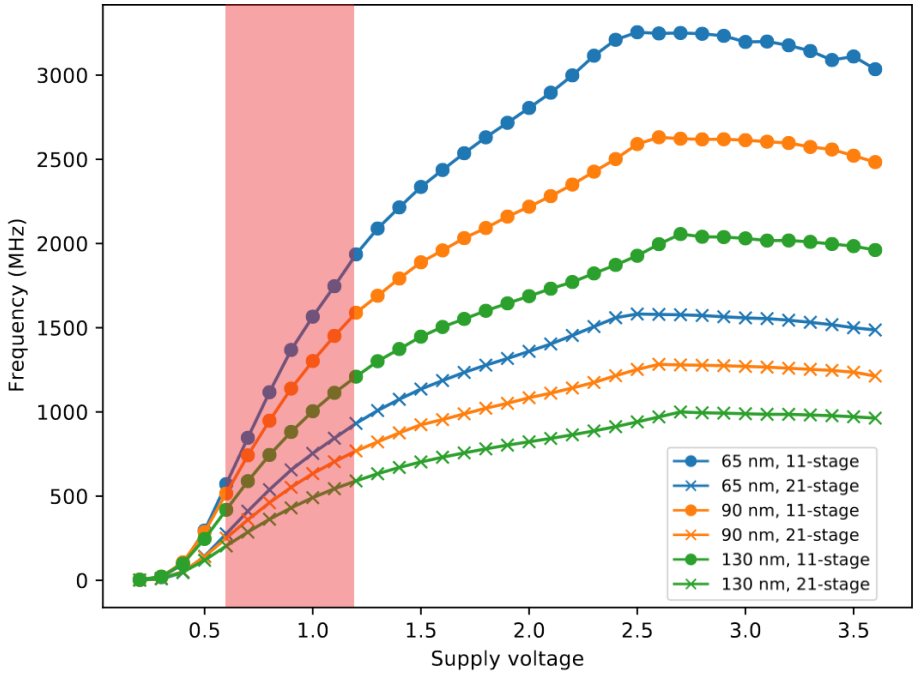


Mapping RO frequency to voltage

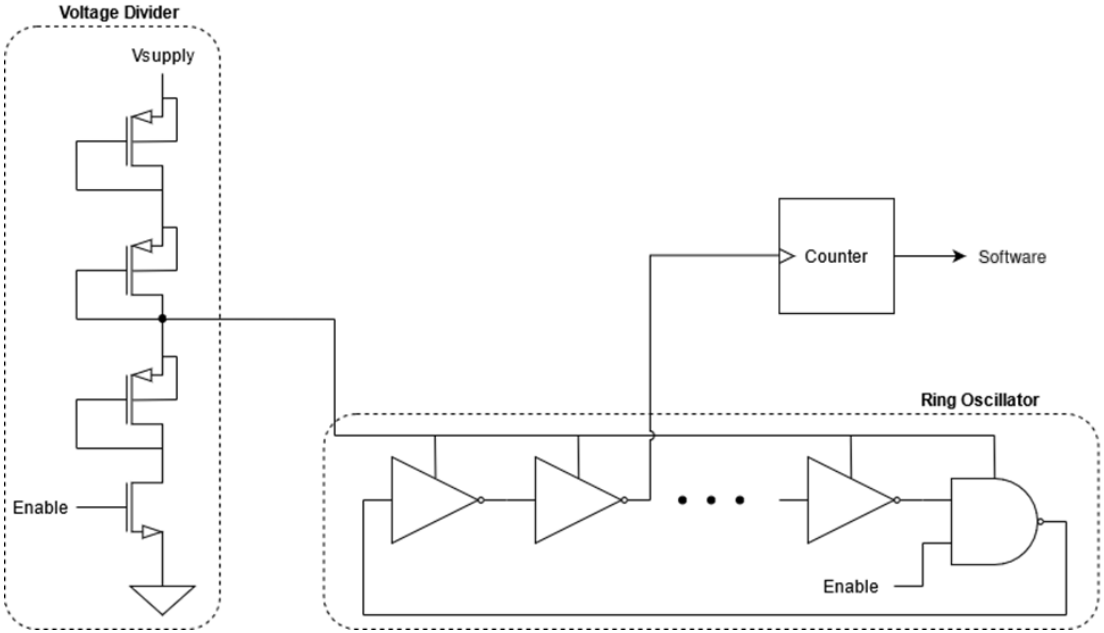
One-to-one

High sensitivity

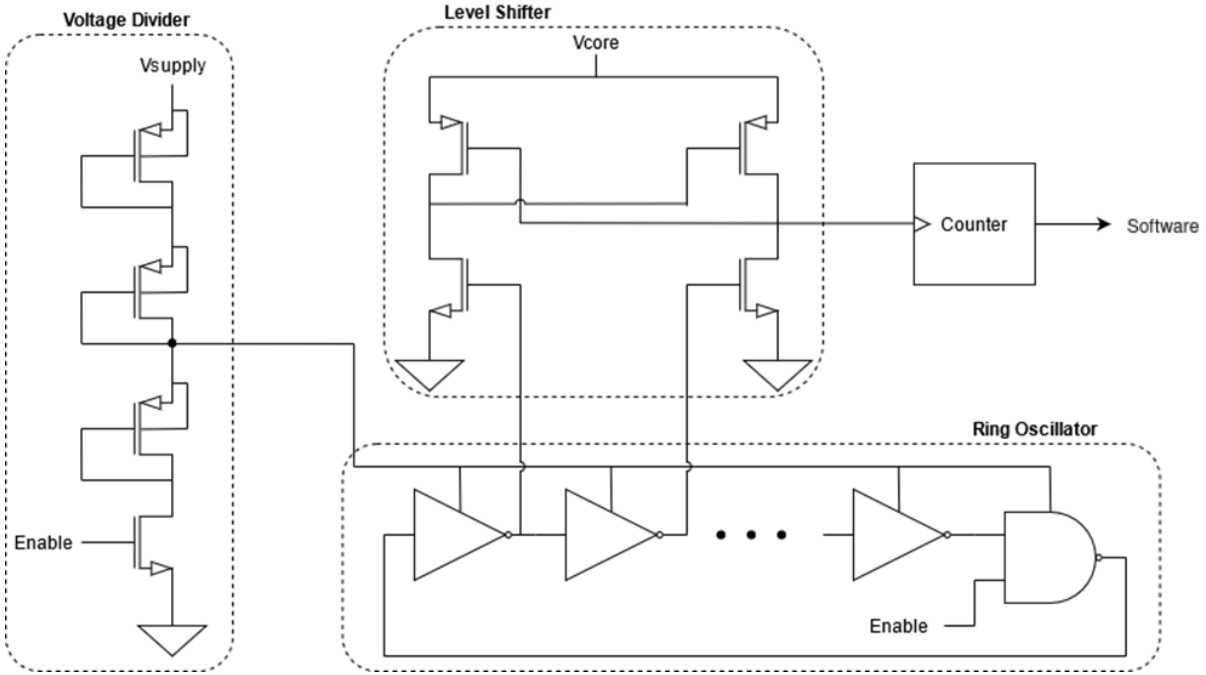
Lower power



Tuning measurement voltage



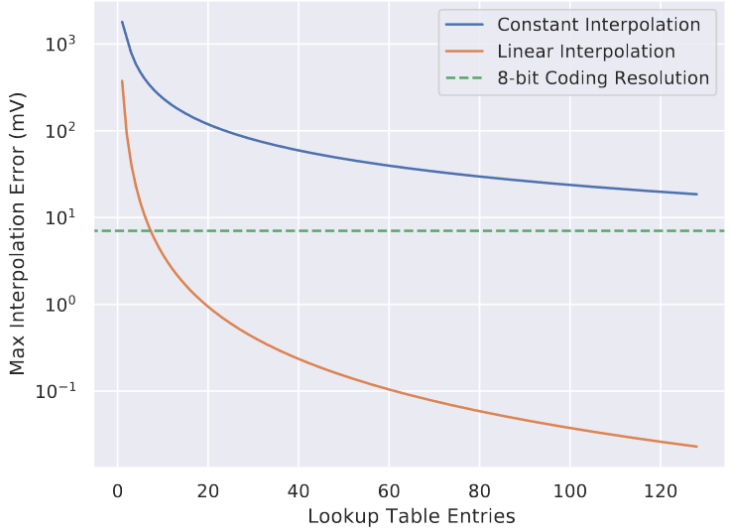
Failure Sentinels Hardware



Converting count to voltage

RO count indexes into lookup table

- Piecewise-constant: low software load
- Piecewise-linear: modest software load



Device enrollment eliminates manufacture-time error

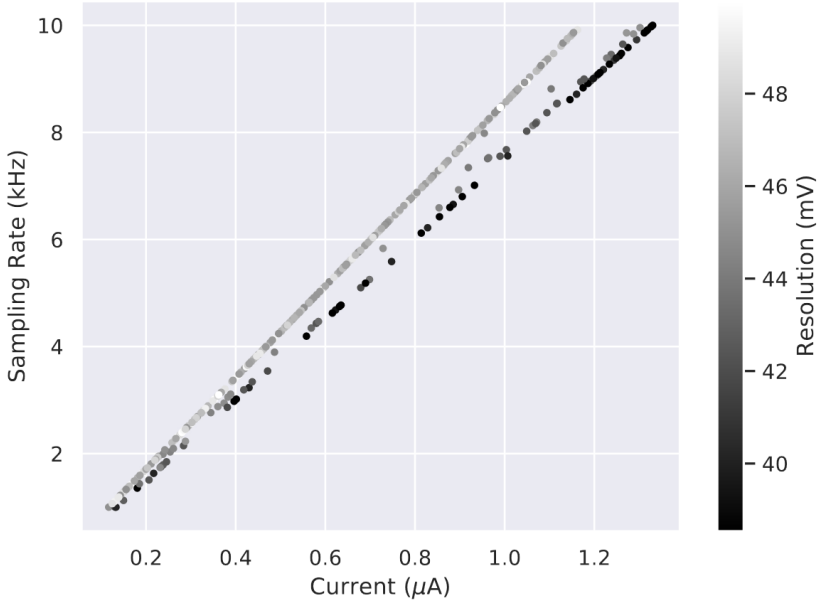
Implementation + Evaluation

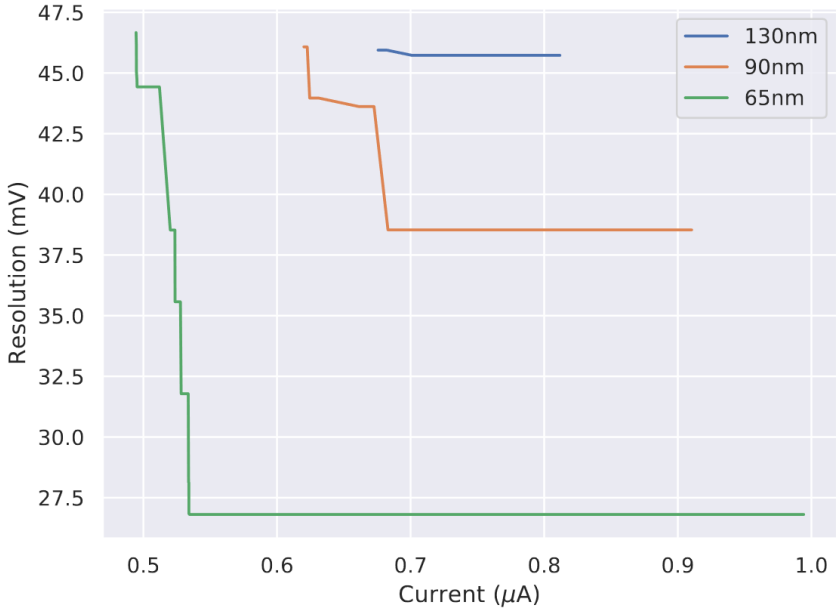
SPICE: Power consumption, performance, scalability

RISC-V (FPGA): Physical error, (partial) system integration

Energy harvester simulator: System performance improvement

Is Failure Sentinels performant/efficient?



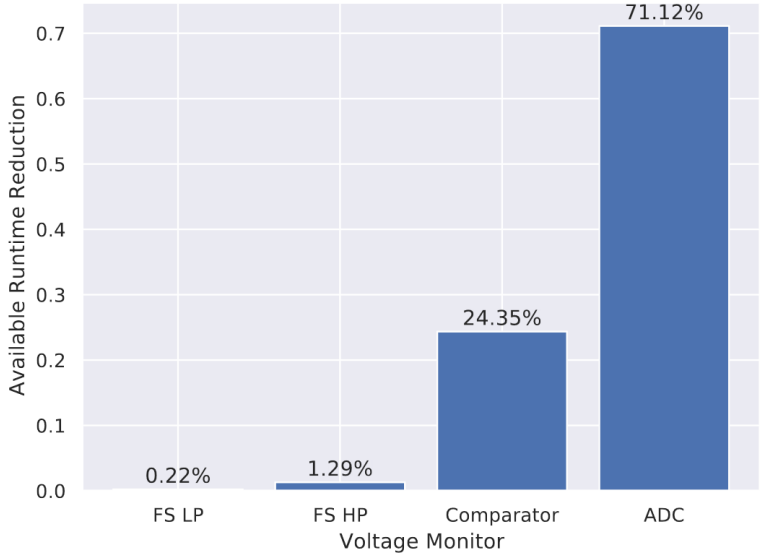


Medium resolution, F_S at < 1uA

Performance and flexibility scale with process node

Does Failure Sentinels improve systems?

Monitor	System Current (µA)	Resolution (mV)	F_S (kHz)
Ideal	112.3	Infinite	Infinite
FS (Low Power)	112.5	50	1
FS (High Performance)	113.6	38	10
Comparator	147.3	30	-
ADC	377.3	0.293	200



Low-power operation at diverse performance points

<1% runtime reduction for checkpoint systems

Failure Sentinels Summary

- Intermittent computing is **energy-driven**: systems demand energy efficiency and awareness
- But, current hardware sacrifices power for unneeded performance

Failure Sentinels makes energy a **first-class abstraction**

- Removes **99%** of energy overhead without compromising performance
- **Enables** sophisticated energy harvesting systems

https://github.com/FoRTE-Research/FailureSentinels-artifact