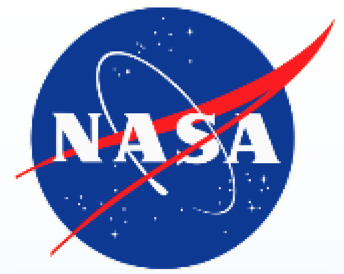
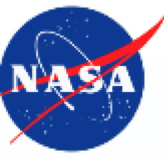


Focusing on NAND Flash SEFI Susceptibilities

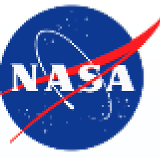
Ted Wilcox
NASA Goddard Space Flight Center

Acronyms



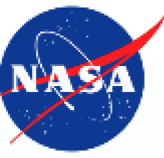
- EDAC – Error Detection and Correction
- GSFC – Goddard Space Flight Center
- LET – Linear Energy Transfer
- MBU – Multiple Bit Upset
- MFTF – Mean Fluence To Failure
- MTTF – Mean Time to Failure
- NAND – Not And (Flash)
- SBU – Single Bit Upset
- SEB – Single Event Burnout
- SEE – Single Event Effects
- SEFI – Single Event Functional Interrupt
- SEGR – Single Event Gate Rupture
- SEL – Single Event Latchup
- SET – Single Event Transient
- SEU – Single Event Upset

Background

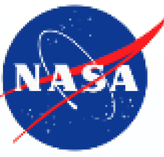


- This derives from a 2022 SEEMAPLD presentation “Risk-Driven and Mitigation-Focused SEFI Testing of NAND Flash Devices”
- That talk was directed to an audience of testers, radiation engineers, and stakeholders
- This version adds context for NEPP’s involvement and the benefits NEPP provides to radiation hardness assurance efforts

Origins of Renewed Focus on SEFI



- An operational mission was experiencing a higher rate of SEFI (single-event functional interrupts) than anticipated, and consequences were not well-understood by all parties
- Subsequent work to understand the problem and implement mitigation was challenged by a lack of detailed test data
- It is not NEPP's purpose to resolve on-orbit anomalies, but NEPP had existing infrastructure and expertise available to the Project to turn around a test in short order
- Following an immediate application-focused test, NEPP was able to follow-up with a broader test campaign to:
  - Develop better SEE test methods for NAND flash
  - Demonstrate the need to test based on risk and mitigation needs
  - Provide data to other projects using the same components

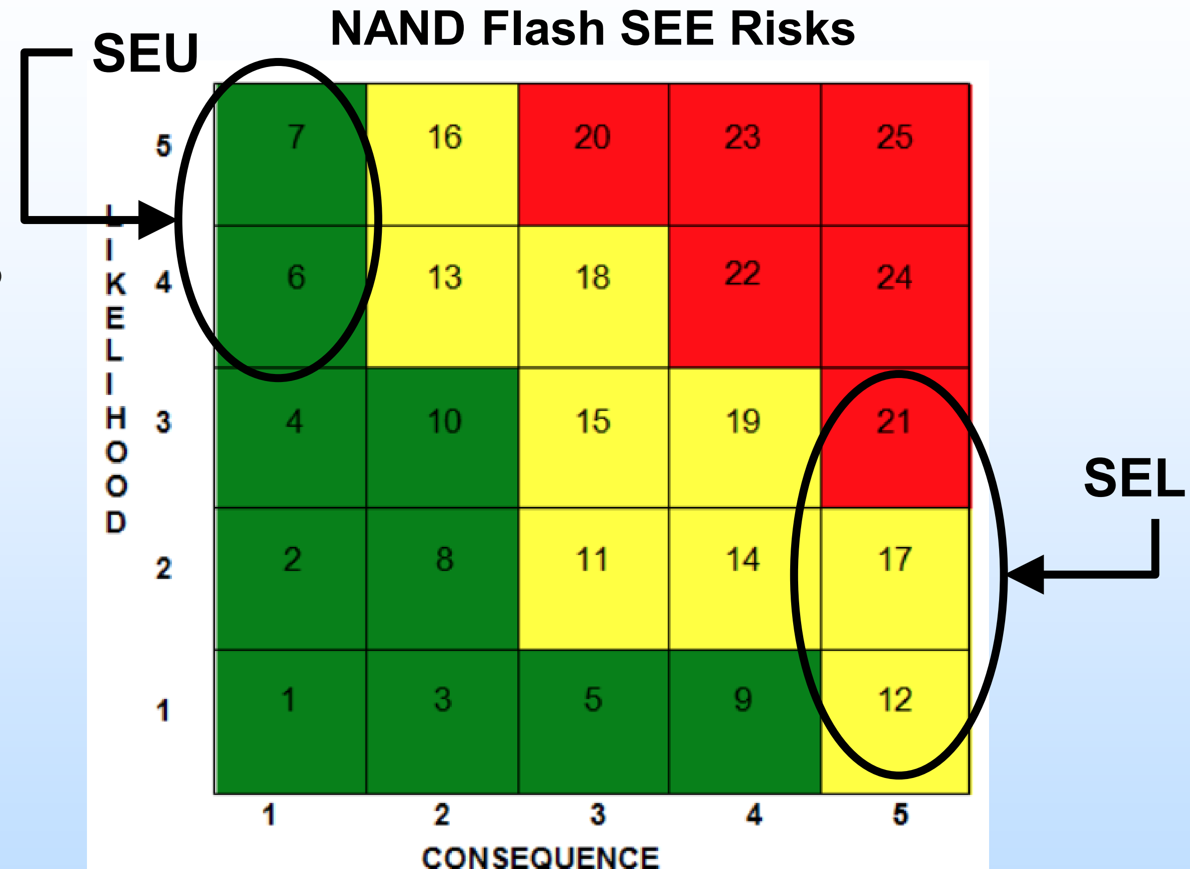


Original Title:

"RISK-DRIVEN AND MITIGATION-FOCUSED SEFI TESTING OF NAND FLASH DEVICES"

Risk-Driven Testing

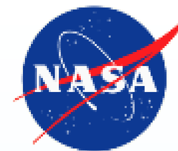
- Consider a common 16Gb NAND flash device for space flight
- Why are we performing SEE testing?
  - Often, to identify and characterize a risk
- We know SEUs, we can explain SEUs, and we can test for SEUs
- But are they an appropriate focus of resources?
  - Depends on risk tolerance – and likelihood vs criticality



Standard GSFC 5x5 risk matrix scorecard

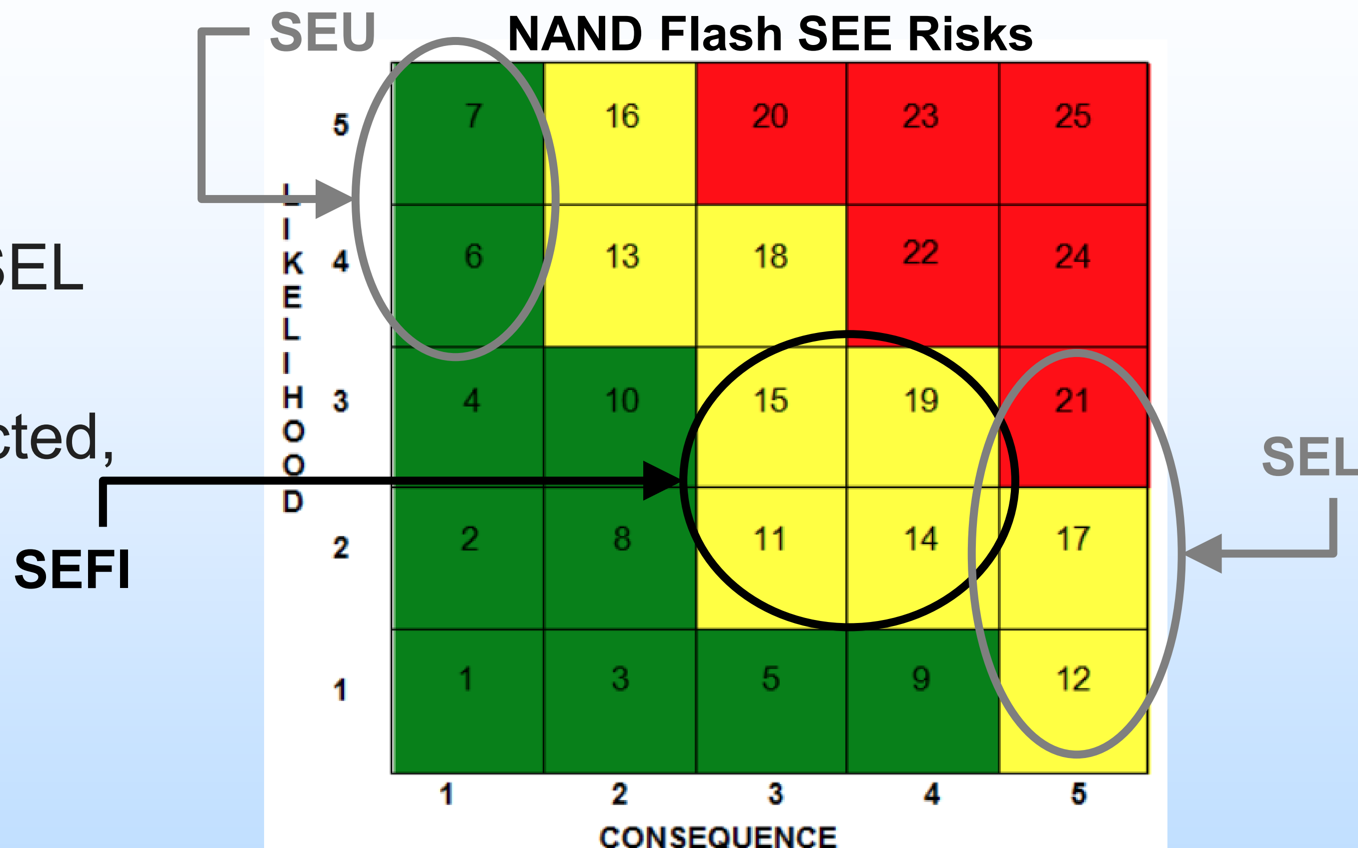
- **SEU are expected and will be handled by EDAC**
- **SEL are feared and will be avoided (or accepted as risk)**

Risk-Driven Testing



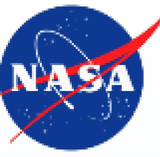
- Where does a SEFI fall on this chart?
- SEFI can be as disruptive as SEL and occur in virtually all flash
- SEFI are occasionally unexpected, often under-tested, and rarely avoidable!

Completely application-dependent, of course!



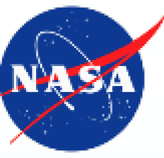
Standard GSFC 5x5 risk matrix scorecard

Mitigation-Focused



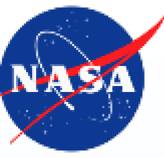
- In addition to evaluating the right risks, our motivation for testing is also to inform robust mitigation
    - An SET rate is relatively meaningless for designers if we can't provide enough information for them to develop an effective filter circuit
    - SEU -- we gather information on SBU/MBU, pattern dependence, etc to guide EDAC design.
    - But what about the SEFI? Why is it an afterthought that falls out of SEU or SEL testing? How can we explicitly test for SEFI and collect useful data?

SEFIs in NAND FLASH



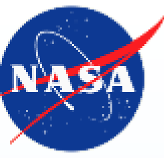
- SEFIs in NAND flash are relatively well-known – but not universally!
- Generally, breaks down into two variations
  - Localized effects preventing a block, page, or row/column from being successfully read/erased/programmed without clearing the SEFI state.
  - Device-wide effects, like a “hang-up” that result in unresponsiveness
  - In both cases, written data is usually preserved intact.
- Criticality of SEFIs varies based on the operational state
  - If a block SEFI occurs immediately before erasing, it will be immediately detected and the block can be skipped and/or the entire device reset.
  - If a block SEFIs after erasure, we won’t notice until programming, and may end up writing into an empty pipe (only noticing at readback!). Data is lost.
  - If a block SEFI occurs during readback, and is detectable, data can be recovered.
  - A distinction must exist between data-lost and data-preserved SEFIs.

The Trouble With SEFI Data



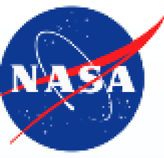
- There is a lot of insufficient SEFI data available
- Much of it is the result of generic tests, or tests focused on other phenomena, but without thought to applicability of the SEFI data
- Many times, a SEFI is just an unexpected result from a test
  - “hey what happened to all the data?”
  - “well, I don’t think that was a latch-up exactly…”
- SEFI data that explores idle vs dynamic modes is not sufficiently granular to apply to a real-world application
  - “I alternated reading, erasing, and programming, but now what?”
- How can a radiation effects (or systems) engineer apply a single cross-section vs LET curve effectively for such a complex, heterogenous error classification?

Better SEFI Data



- Radiation effects engineers warn the difficulties of testing complex parts and evaluating the full state space
- NAND flash is a complex device (relative to some), but with the simplicity of a small number of operational modes
  - Generally, an application will ERASE, PROGRAM, and READ, interspersed with periods of IDLE standby.
- If we could at least obtain data for each mode, we can tailor error rates to a given application’s duty cycle
- If we could determine when errors are detectable and their consequences, we can tailor mitigation to minimize data loss.

Testing for Better SEFI Data

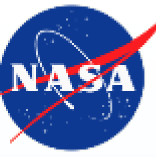


- Can't just ERASE continuously in beam – need programmed cells to make the erase meaningful. Can't program a page over and over.
- Cycling through Erase -> Program -> Read takes time
  - It obscures both the source of the error and the appearance of the error
  - We're creating errors and detecting errors all at once, taking seconds or minutes to cycle through a region of memory
  - Testing a sub-set of the memory doesn't really help
- Let's block the beam during most of the operation
  - E.g., erase with beam on, then program and readback with beam blocked
  - Rinse and repeat; now we know that SEFI beginning during an erase cause specific effects when programmed or readback
  - Can automate this sufficiently that many independent events (with recovery as needed) occur in reasonable amount of time

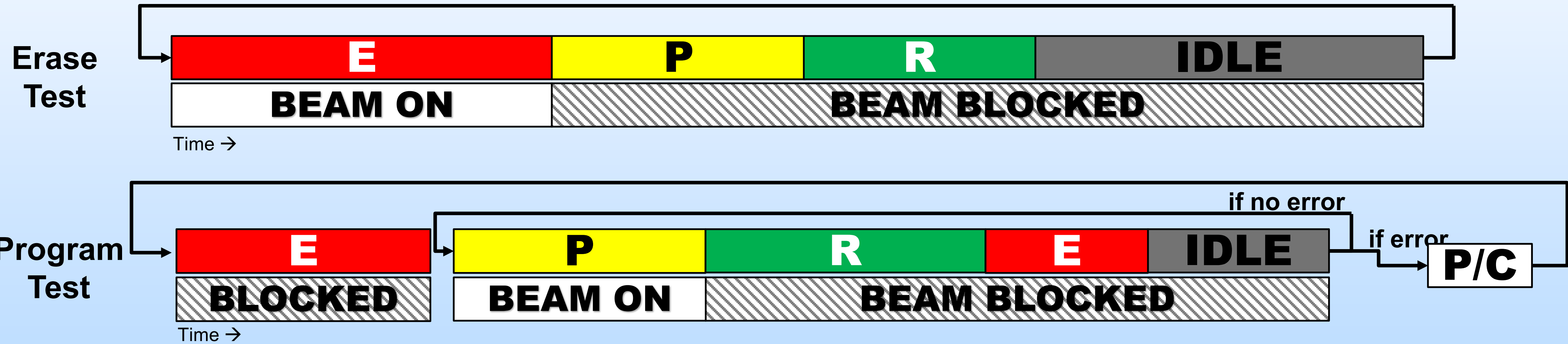
Demonstrating with Heavy-Ion Testing

- A common part: Micron MT29F16G08ABABA
  - 16Gb planar NAND flash; 34nm process
  - 4096 blocks per device
  - Commercially obsolete; used in space
- Tested at LBNL with 16 MeV/amu tune
- Fast Thorlabs optical shutter isolates op modes:
  - *ERASE*
  - *PROGRAM*
  - *READ*
  - *IDLE*
- Immediate (~10ms) beam block when SEFI detected, measures accurate fluence-to-failure

Operations During Heavy Ion Test



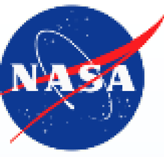
- Complete data collection requires all modes in their natural order (erase-program-read) regardless of which is irradiated.



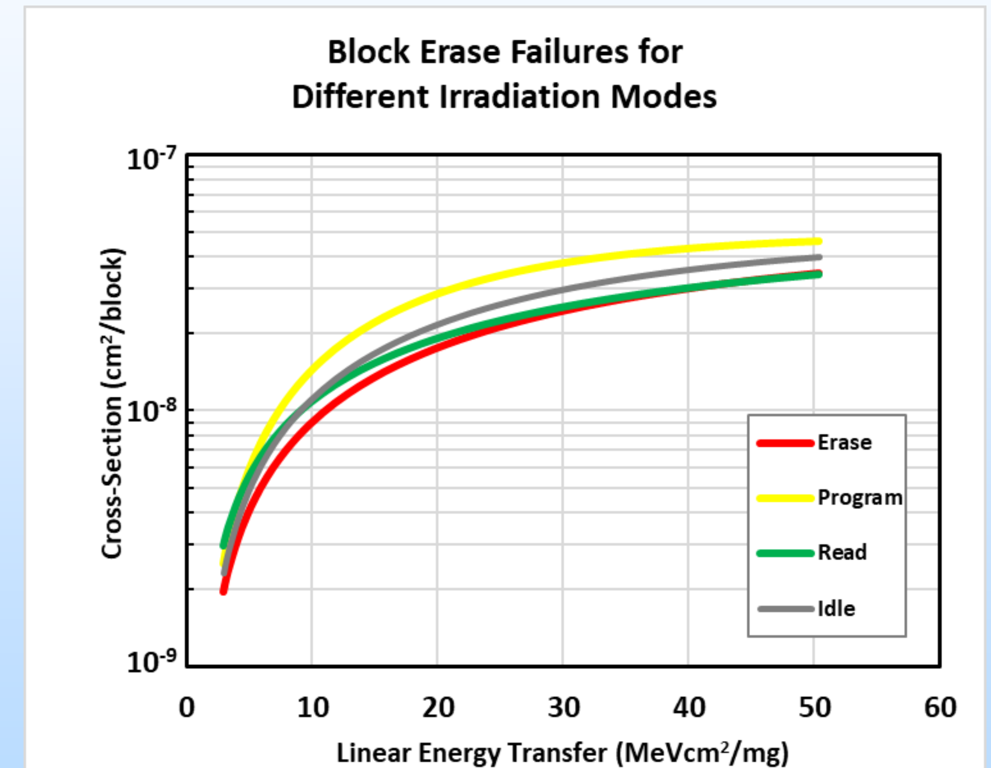
And so forth….

(Operational duty cycles not to scale)

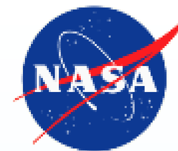
SEFI Data – Irradiating While Erasing



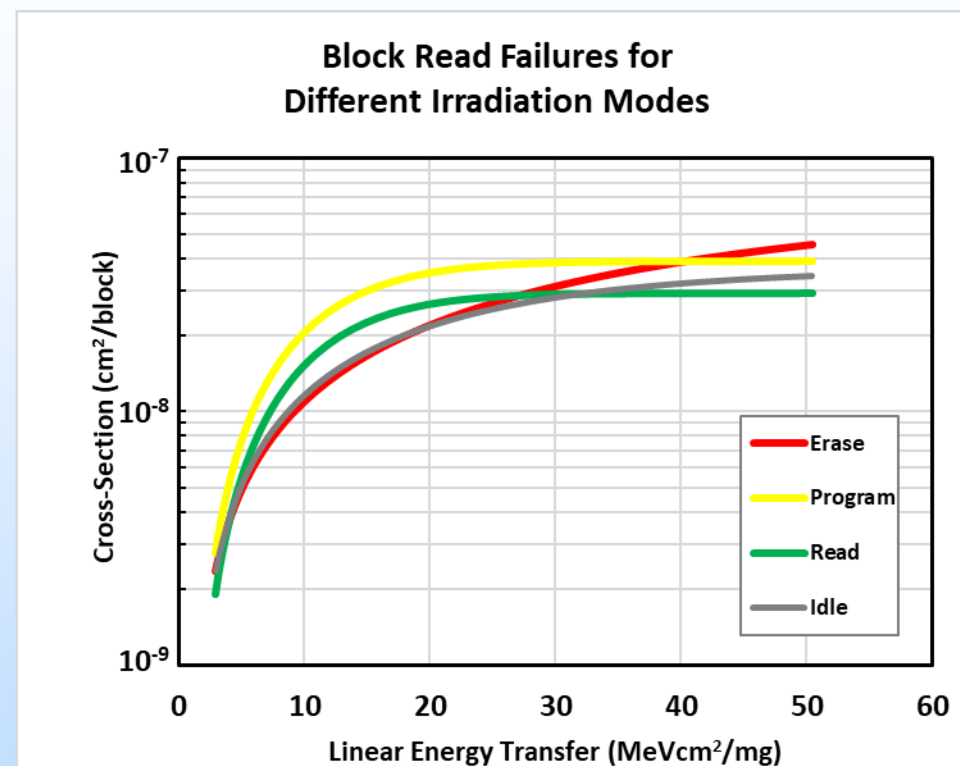
- Let's look at block erase failures (one error mode) for multiple irradiation conditions
  - Single error signature, but irradiated while erasing, programming, or reading
  - Count the number of erase operations that fail (bad status bit OR timeout)
- Results aren't drastically different
- Consequence of these errors? Minimal, as they are detectable immediately during a block erase.
- Smart system architecture will simply skip these as "bad blocks"



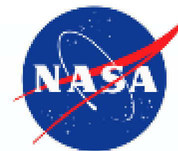
SEFI Data – Irradiating While Reading



- Similar results for block read failures
  - Identified by implausibly large errors
  - If the particle strike occurred during erase or programming – data is probably lost.
  - If it occurred after programming – data is probably there, parts needs reset
- With this approach, we can begin to separate those two cases by selectively irradiating different operations to separate the "data loss" SEFIs from the "data preserved" SEFIs.
- We inform mitigation strategies to (e.g.) power cycle the flash device at the right time to minimize data loss SEFIs.



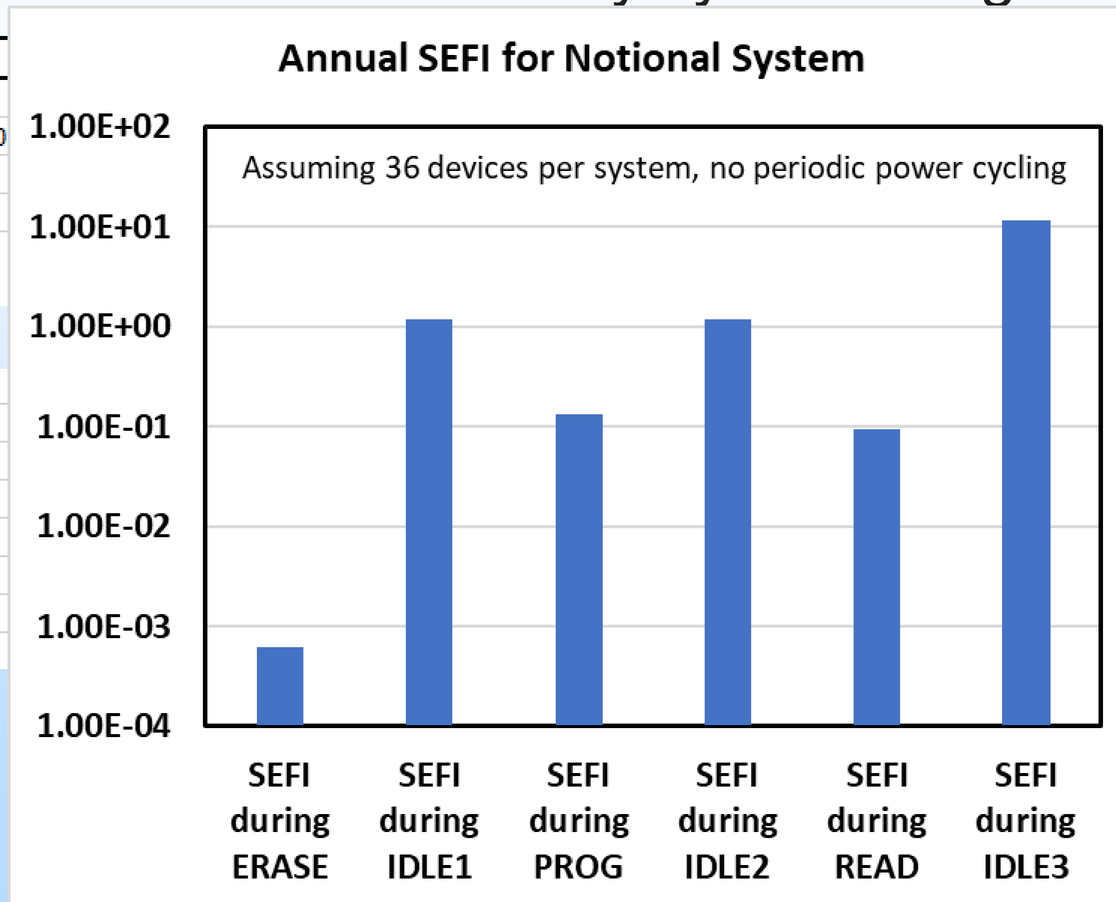
Applicability to Mitigation



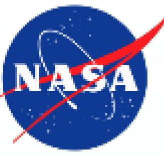
- Use preferred tool to model flash duty cycles and generate rates:

Operation type:	P/C		P/C	IDLE3	totals	Total(year)
Time for each (s):		3		36000	43590.7	0.001382
Power Cycle (0/1)?	0		0			
Sat CS		2		0.00014		
LET0.25		3		7.91		
Block SEFIs per op day (from FoM GEO)		3		1.08E-03		

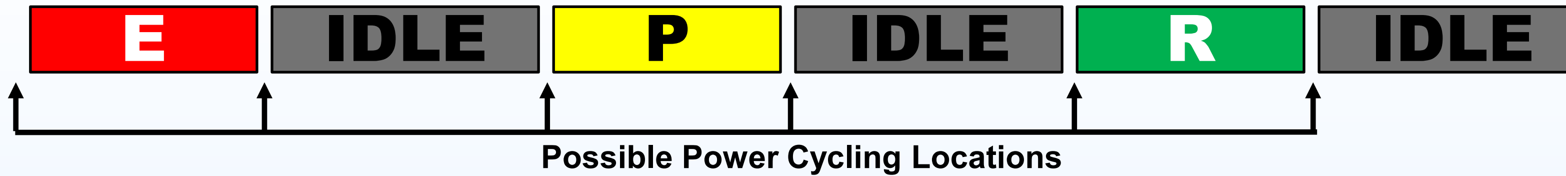
FAILED BLOCK PROGRAMS						
due to SEFI during erase						
due to SEFI during IDLE1						
due to SEFI during PROG						
due to SEFI during IDLE2						
due to SEFI during READ		6				
due to SEFI during IDLE3				4.51E-04		
total failed programs (blocks)					**5.47E-04**	3.96E-01

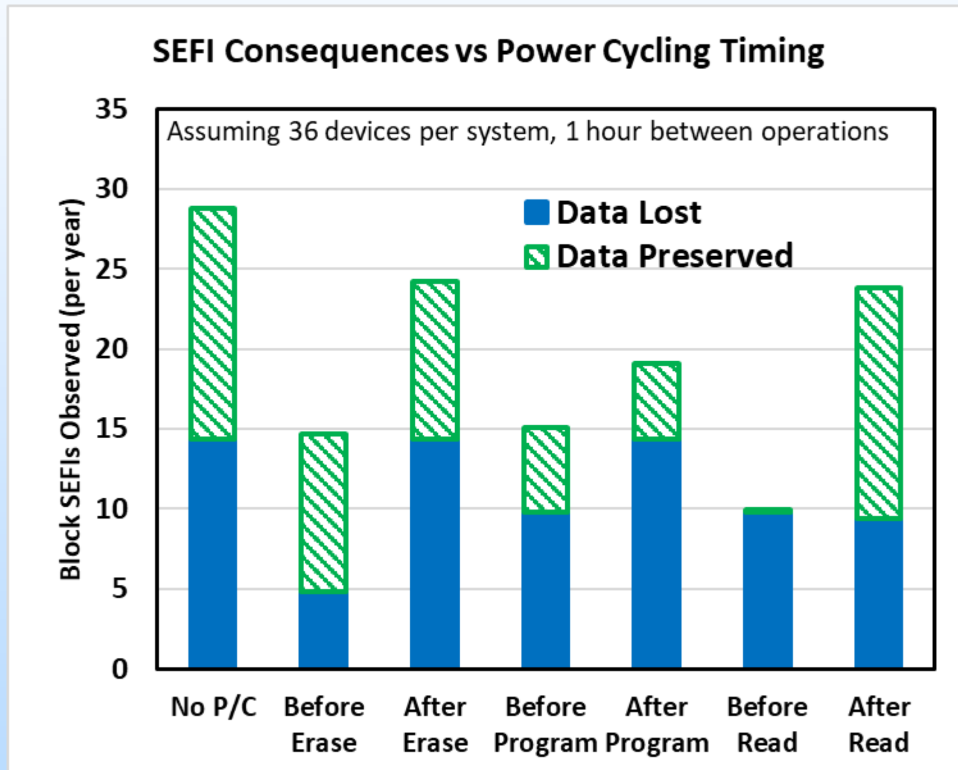


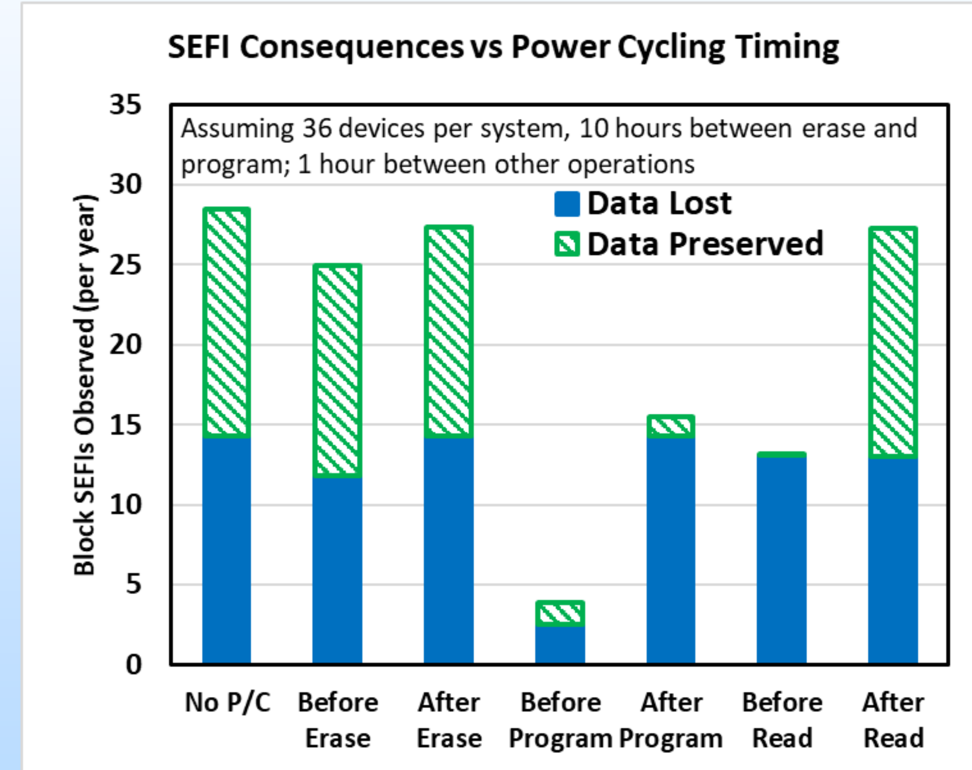
The Application Matters for Mitigation



Notional Application:



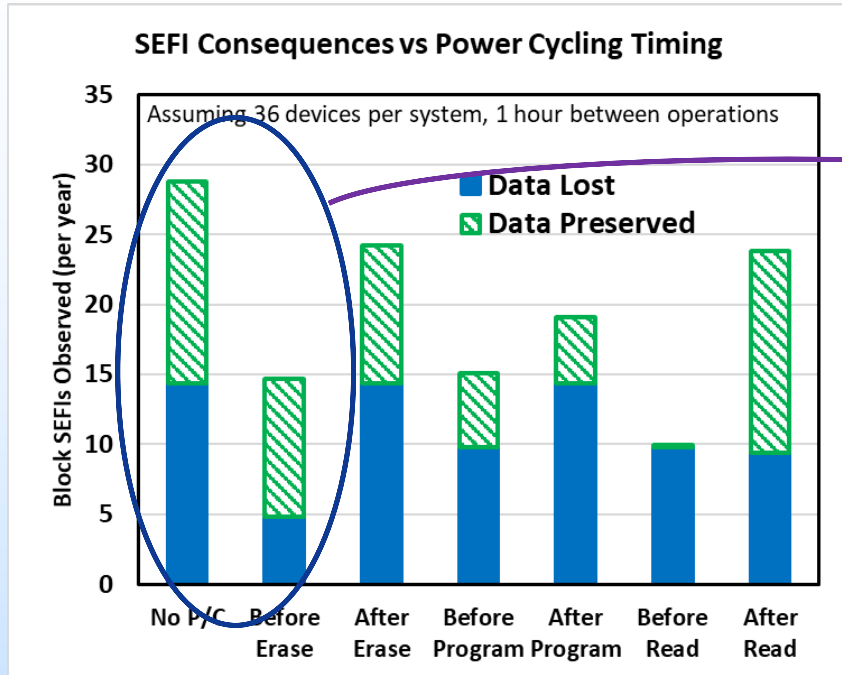




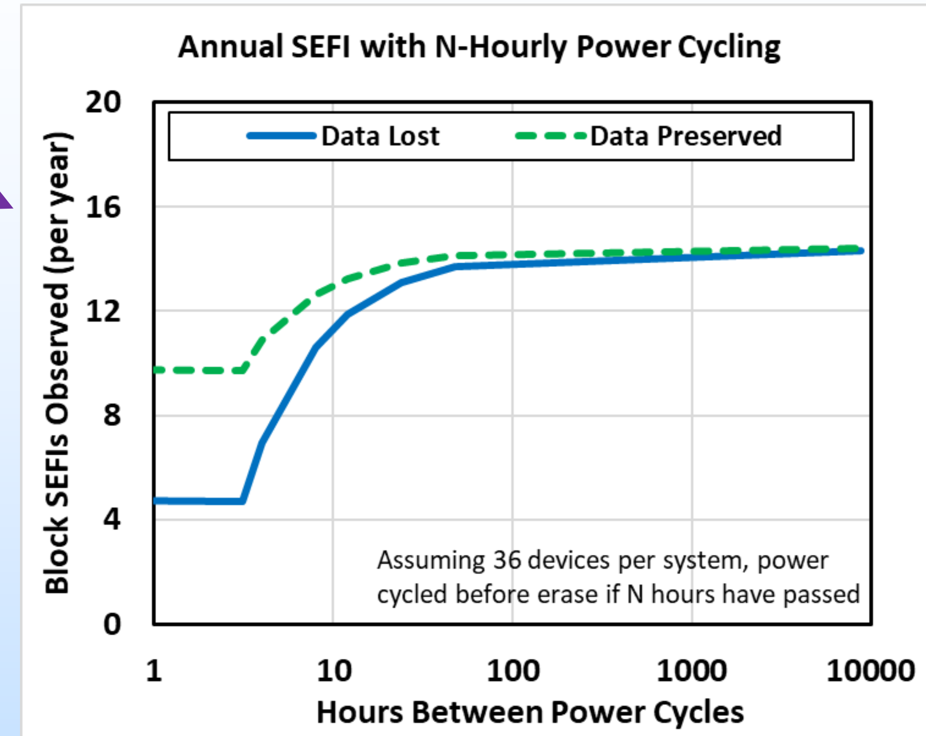
Minimizing data loss requires consideration of individual SEFI consequences

A different application duty cycle changes the answer completely…

Mitigation by N-Hourly Power Cycling

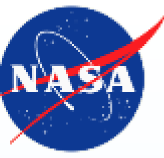


What happens between *never* power cycling and *always* power cycling?



- Power cycling daily, weekly, once-per-orbit is common suggestion → won't eliminate SEFIs
- If the device is powered, SEFIs will happen.
- Critical systems still have to mitigate (redundancy, monitoring, etc)

Future Work for NEPP



- Provide more guidance for NAND flash SEE testing
  - Update for relevance to 2022 parts
  - Ensure applicability to systems-level RHA
  - Encourage mindset of application MTBF and availability rather than simple characterization of the part on its own
- Continue to demonstrate effective testing
  - Emphasize risks that matter most for systems in 2023+
  - Develop the right data to mitigate appropriately