SEU Effects in Deep Submicron Processes

Heather Quinn

hquinn@lanl.gov

LA-UR 11-03201

Acknowledgments

This research has been supported through the DOE in the Deployable Adaptive Computing Systems project, the Sensor-Oriented Processing and Networking project, and the Joint Architecture Study; and supported through the DOD from the FPGA Mission Assurance Center.

This presentation includes the work and help of LANL, BYU, Xilinx, JPL, SWRI, SEAKR, and The Aerospace Corporation staff members.

Overview

- **First half:**
  - **A Quick Introduction to Space Physics**
    - The Earth's Magnetosphere
    - Solar Cycles, Solar Flares, and Coronal Mass Ejections
    - Cosmic Rays
    - A Few of My Favorite Particles
  - **Radiation Effects in Electronics**
    - A Historical Perspective
    - Current reliability issues with Modern Electronics
    - Radiation-Induced Failure Modes
  - Single-event effects in memory devices
- **Second half**
  - **Single-Event Upsets and Transients in FPGAs, and Microprocessors**
    - Failure modes and error rates in FPGAs
    - Mitigation methods for FPGAs
    - Failure modes in microprocessors
    - Mitigation methods for microprocessors
- **Summary and Conclusions**

Motivation

- **Radiation interactions with electronics is becoming increasingly apparent in modern systems**
  - Space-based and airplane-based onboard processing
  - High-performance and data-center computing
  - Nuclear reactors
- **Radiation induced failures can affect**
  - Data stored in “soft” data storage
  - Intermediate processing values
  - Processing
- **Left unmitigated, radiation-induced failures can cause data reliability and system availability problems**

Historical View of Radiation-Induced Faults

- **Primary concern was accumulated dose effects**
  - For space this continues to be true
- **Transient or single-event effects were less of an issue, due to small size of memories and large clock frequencies**
- **Single-event effects were an "an inside job"**
  - Alphas: radioactive bat guano, radioactive water
    - The fabrication process was altered to remove radioactive contaminates
  - Thermal neutrons-induced single-event effects from ^{10}B contamination

J. F. Ziegler. "SER – History, Trends, and Challenges: A Guide for Designing with Memory ICS", Cypress Press, 2004

Radiation-Induced Faults in Modern Electronics

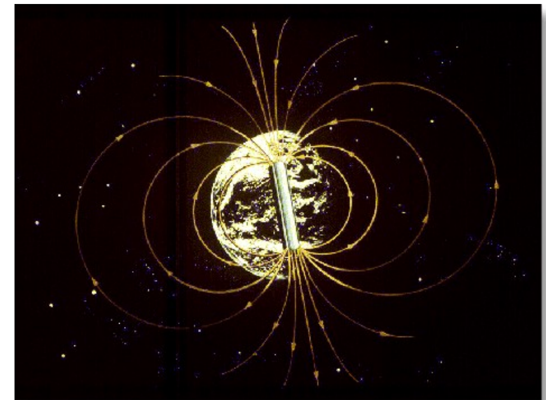
- **Physics: smaller is not better**
  - Smaller transistors are smaller targets, but easier to upset (Q_{crit})
  - Denser designs are easier to upset with multiple-bit upsets
- **System design: increasing sensitivity (cross-section)**
  - Microscopic: more complex components each generation
  - Macroscopic: larger systems each generation, memory-based devices (SRAM, DRAM, FPGAs) become larger, overall target size increase
- **System location: increasing deployment to high-radiation environments**
  - Multiprocessor and multi-FPGA systems for airborne and space applications are more common
- **Contaminants: getting worse in some cases**
  - B10 is a price point in manufacturing but can be hard to get rid of
  - Hafnium is also becoming common in parts – will have similar problems with thermal neutrons

Soft Errors and System Reliability

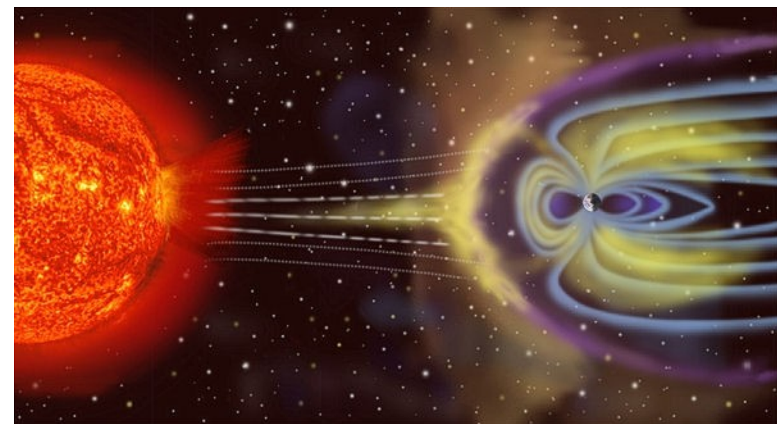
- **Soft errors are often undetected, unmitigated**
- **For airplanes, 75% of all unrepeatable system errors are caused by soft errors**
- **For large-scale, reliable systems unmitigated soft errors are disastrous:**
  - Sun Microsystems received bad press for soft error failures in their high end servers in 2000
  - Intel processors verging on radiation-hardened electronics

The Natural Radiation Environment

- **Our initial understanding of cosmic rays predates our concepts of sub-atomic particles**
  - Scientists knew charged particles were coming from the atmosphere 30 years before the neutrons were discovered
  - Original definitions were “particles that rain down from the sky but do not make me wet”
- **Galactic flux:**
  - “Debatable Origins”
  - Very energetic (1023 eV), very dense flux (100,000/m2-s)
- **Solar flux:**
  - Not energetic
  - Affected by solar winds



http://science.nasa.gov/ssl/pad/sppb/edu/magnetosphere/mag1.html

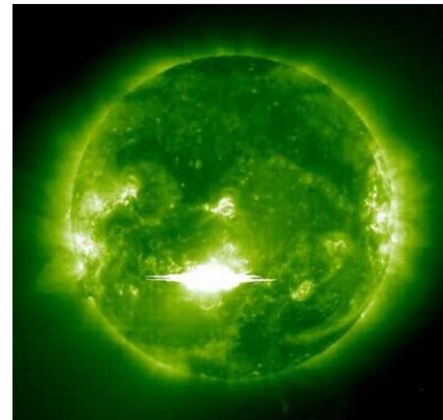


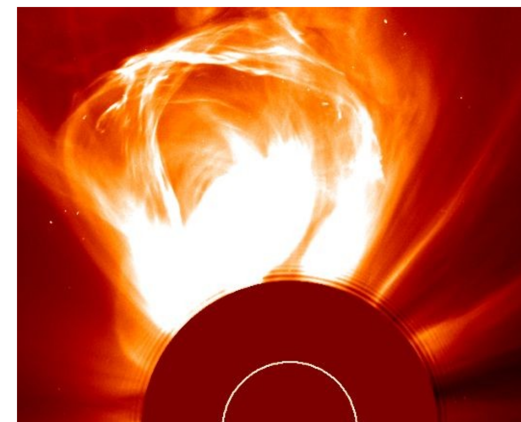
http://www.eskimo.com/~nanook/science/2007_07_01_archive.html

LA-UR 11-03201

Solar Flares and Coronal Mass Ejections

- **Coronal mass ejections (CME) release solar atmosphere**
  - Often in conjunction with solar flares, but not necessarily
  - X-Ray, gamma-rays, electrons, protons, and heavy ions released at near speed of light
- **CME/solar flares can filter to earth for a few hours after the event**
  - Auroras
  - X-Ray-induced communication problems
  - Increased soft errors
- **Every solar cycle seems to have one unusually large CME**

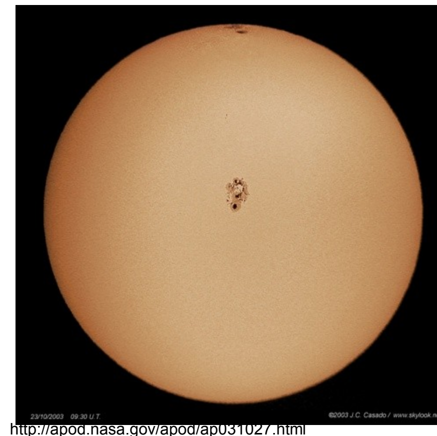

http://www.gallerita.net/2003_10_01.php


http://www.arm.ac.uk/climate/images/febcme_sohoc2_big.gif

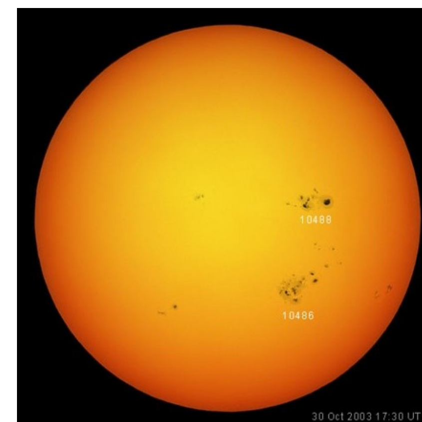
LA-UR 11-03201

The Halloween 2003 Coronal Mass Ejections

- **Three active solar spot groups (10484, 10486, 10488):**
  - All three were "remarkable in size and magnetic complexity"
  - One sun spot group (10486) was on over 13 times the size of the Earth and was the largest sun spot group observed since Nov 1990
- **17 CME ejections from mid-October to early November**
  - 12 events from 10486 alone
  - Three major events: the X17 on Oct 28, X10 on Oct 29, X28e on Nov 4
  - X28e event occurred while the GOES detector saturated and was likely an X40 event



http://apod.nasa.gov/apod/ap031027.html

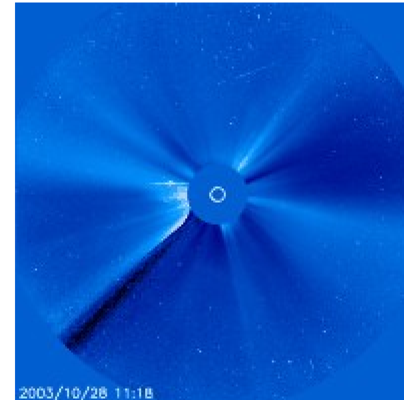


http://www.astropix.com/HTML/G_SUN/SS486488.HTM

T, Gombosi. "Comprehensive Solar-Terrestrial Environment Model for Space Weather Predictions. DoD MURI Project Report 2001-2004

Effects from The Halloween 2003 Coronal Mass Ejection

- **Auroras seen as low as CO, CA, NM, AZ**
- **Damage caused by the Halloween storms:**
  - 28 satellites (overt) damaged, 2 unrecoverably damaged
  - Diverted airplanes
  - Power failure in Sweden
- **…and two supercomputers came up in late October hoping to get onto the yearly Supercomputing list on Nov 10th**
  - At Los Alamos, the "Q" cluster had 26.1 errors a week and one unfortunate cluster topology
  - At Virginia Tech, System X architect joked they "felt like [they] had not only built the world's third fastest supercomputer, but also one of the world's best cosmic ray detectors."
    - VT processed at night while in the magnetosphere tail
    - VT replaced all of their processors within the next 6 months



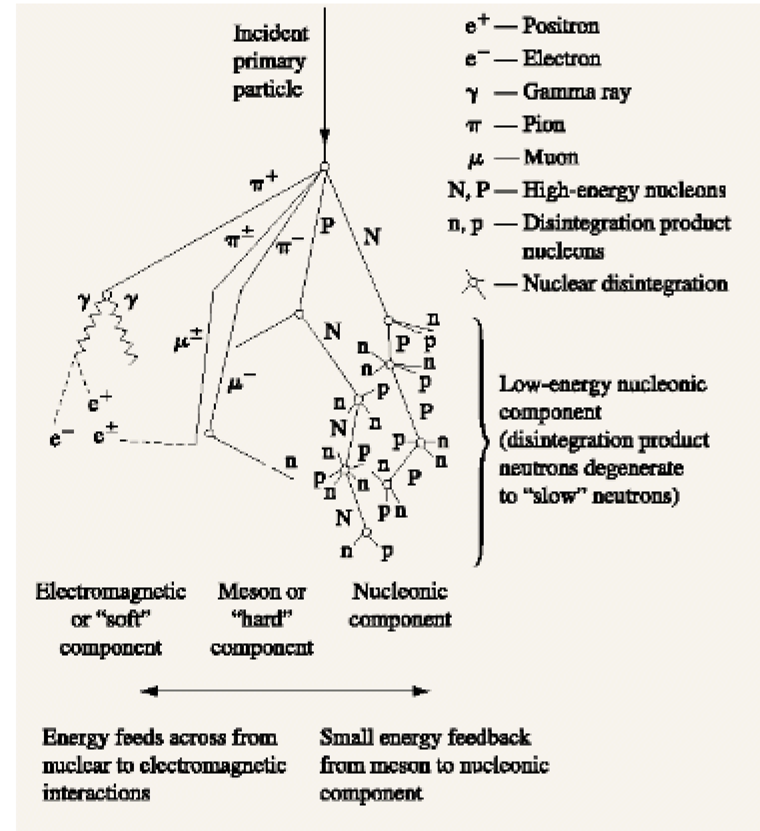
http://apod.nasa.gov/apod/ap031029.html

http://apod.nasa.gov/apod/image/0310/aurora031029b_westlake_full.jpg

T, Gombosi. "Comprehensive Solar-Terrestrial Environment Model for Space Weather Predictions. DoD MURI Project Report 2001-2004

Cosmic Rays and the Atmosphere

- **Cosmic rays that make it through the magnetosphere to the atmosphere cause a cascade of particles**
  - Neutrons, protons, pions, and muons
- **These particles can cause problems with electronics:**
  - Memory upsets
  - Transient charge changes
  - Latch-up
  - Functional interrupts



J. F. Ziegler, "Terrestrial Cosmic Ray Intesities," IBM Journal of Research and Development, Vol 42 (1), 1998

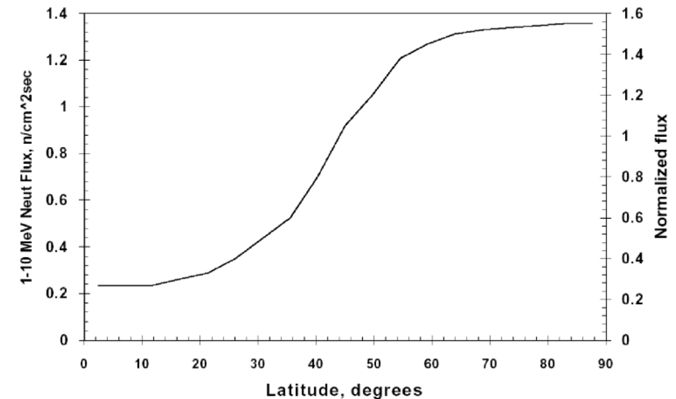
Muons and Pions

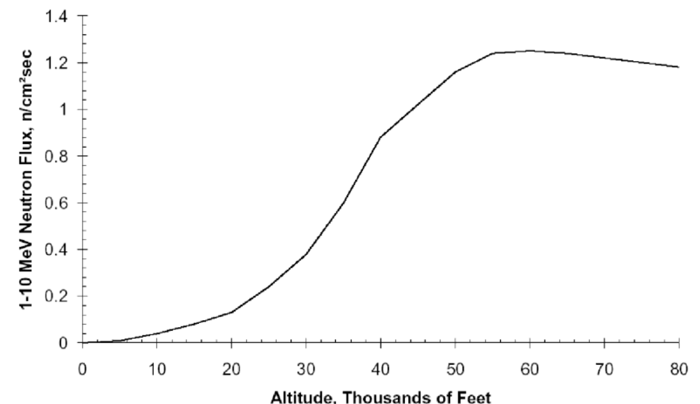
- **Pions:**
  - Unstable particles
  - Lifetime of ~26 ns
  - Very low flux at sea level (~450 pions/cm^2-year), more common at 50,000 ft
  - Rarely interact with Silicon, except for rare pion capture events
  - Cause 0.0003 fails/chip-year, considered minimum error rate possible for radiation-induced failures
- **Muons:**
  - Relativistic particles
  - Lifetime of ~2 μs
  - 100x more muons at sea level then any other sub-atomic particle
  - Very rarely interact with Silicon, except for muon capture events
  - Cause 0.0006 fails/chip-year
  - Muon-induced transient effects could be possible in the newest generations of electronics

J. F. Ziegler. "SER – History, Trends, and Challenges: A Guide for Designing with Memory ICS", Cypress Press, 2004

Neutrons

- **Lifetime of 11-12 minutes**
- **Tend to “drill through” most substances**
  - Often just loses energy as it bounces off atoms
  - A direct strike with a Silicon atom releases a heavy ion, called “nuclear recoil reaction”
  - The heavy ion causes “soft errors”
  - Protons also cause a nuclear recoil reaction and the sensitivities to both neutrons and protons often similar
- **Flux dependent on longitude, latitude, altitude, geomagnetic rigidity, solar cycles, time of day, and time of the year**
  - Radiation peaks at high altitudes and near poles
  - Some reduced affects at night or in winter months
- **Flux sensitive to surroundings**
  - Seventh transition (thermals) happens close to the electronics: either using nearby humans or building materials
  - Ship effect can increase flux by an order of magnitude
  - Can shield with water or concrete, but will need a lot of it





J. F. Ziegler, “Terrestrial Cosmic Ray Intesities,” IBM Journal of Research and Development, Vol 42 (1), 1998

Protons

- **Extremely stable outside of the nucleus – theoretically could exist for thousands of years**
- **Interacts with CMOS technology in a variety of ways:**
  - Accumulated dose effects
  - Transient effects from both direct and indirect ionization
- **While the terrestrial proton environment is very low at sea level, higher concentrations are found in space and high-altitude environments**
  - One third of the fast neutron environment in high-altitude airplane environments
  - Lower Van Allen Belt in low earth orbit has a significant trapped proton region, which can affect electronics

Direct and Indirect Ionization

- **SEEs can be caused by both direct ionization and indirect ionization**
- **Direct ionization occurs when the particle creates electron-hole pairs on its own**
- **Indirect ionization occurs when a particle hits the lattice and creates a nuclear fragment or causes a nucleus to be liberated from the lattice – nuclear recoil**
  - In this case the the ionization is caused by the nuclear fragment and not the incident particle
  - Because the particle has to hit an atom head on to cause the nuclear recoil, devices are less sensitive to particles that cause indirect ionization
- **Generally, heavy ions directly ionize and neutrons and protons indirectly ionize**

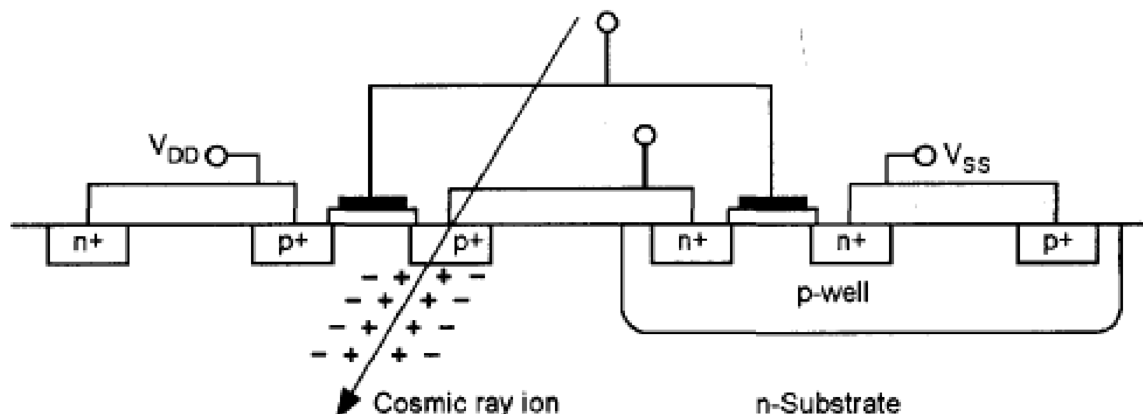
Direct Ionization from Protons

- **Researchers are now seeing direct ionization effects from protons in the 3-5 MeV range, which are very common in the space environment**
  - Currently IBM researchers have shown direct ionization from protons in both bulk Silicon and Silicon on Insulator SRAM memories in the 45-65 nm feature sizes
  - Currently looking at 45 nm processors to see if we can see direct ionization from protons in the caches
- **The new JEDEC standard indicates that (n,p) reactions in the Silicon would lead to direction ionization from protons from 3-5 MeV neutrons, which are very common in the terrestrial environment**
- **Even if there is a very low percentage of (n, p) reactions in Silicon in comparison to indirect ionization – even if the reaction rate was 0.01% the error rates would increase by 1000 times in airplanes**

Overview

- First half:
  - A Quick Introduction to Space Physics
    - The Earth's Magnetosphere
    - Solar Cycles, Solar Flares, and Coronal Mass Ejections
    - Cosmic Rays
    - A Few of My Favorite Particles
  - **Radiation Effects in Electronics**
    - A Historical Perspective
    - Current reliability issues with Modern Electronics
    - Radiation-Induced Failure Modes
  - Single-event effects in memory devices
- Second half
  - Single-Event Upsets and Transients in FPGAs, and Microprocessors
    - Failure modes and error rates in FPGAs
    - Mitigation methods for FPGAs
    - Failure modes in microprocessors
    - Mitigation methods for microprocessors
- Summary and Conclusions

SEE: the transient

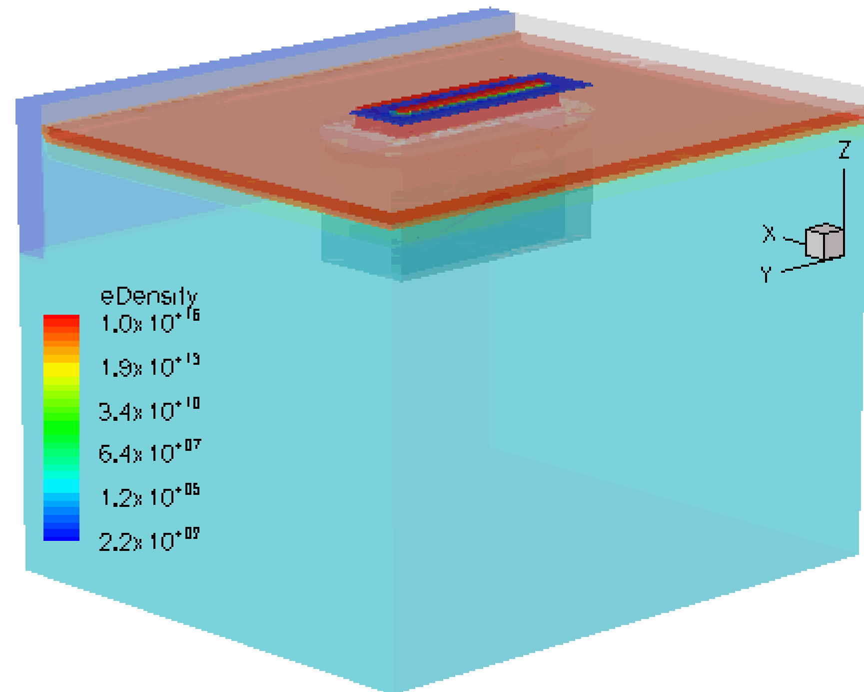


Edmonds et al, “An Introduction to Space Radiation Effects on Microelectronics” JPL-00-06

- **Particle strikes liberates e-h pairs**
- **E-h pairs cause charge generation**
- **Charge generation causes current to flow**
  - For an “on” transistor, the extra current is generally meaningless
  - For an “off” transistor, the extra current can temporarily turn the transistor on
- **Charge → current, current → voltage, voltage → signal**

SEE: the transient

- **Even though the particle is much smaller than the transistor, the charge generation cloud can be much larger than one or many transients**
  - Based on feature size
  - The LET of the particle



LA-UR 11-03201

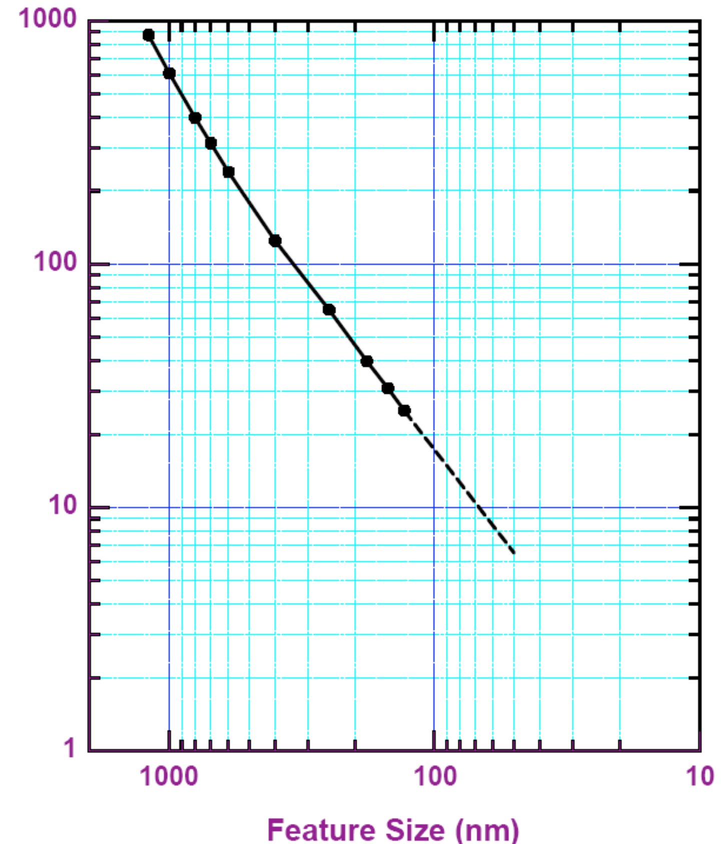
Types of SEEs

- **Transient:**
  - Single-event transient
  - Single-event upset
  - Single-event functional interrupt
- **Destructive:**
  - Single-event gate rupture
  - Single-event dielectric rupture
  - Single-event latchup
  - Single-event burnout

Single-Event Transients (Transients or SETs)

- **Radiation-induced temporary charge changes the value of gate**
  - No way to tell the difference from a real signal and a transient-affected signal
  - Transients in logic gates are a problem if latched, causes data corruption
  - Transients in the clock or reset trees can cause much larger global issues
- **Decreasing clock frequencies make it easier to latch a transient: transient pulse and clock signal are roughly the same**
- **Affects data reliability during processing**

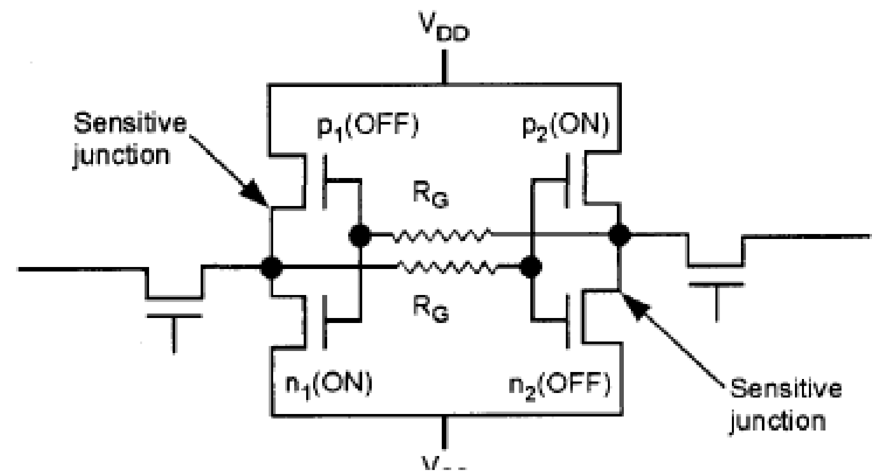
Critical Pulse Width for Unattenuated Propagation



Mavis, "Single-Event Transient Phenomena: Challenges and Solutions." MRQW, 2002.

Single-Event Upsets (Upsets or SEUs)

- **Cause bit flips in memory-based electronics**
  - Data changes from 1→0 or 0→1
  - In some parts single-bit upsets (SBUs) are as common as multiple-bit upsets (MBUs)
- **Strongly affected by feature size:**
  - Smaller feature size means smaller targets, smaller Q_{crit}, more MBUs
  - Even with a decrease in per-bit cross-section, often see an increase in per-device cross-section due to increases in system size
- **Affects data reliability during processing or data storage**



Edmonds et al, “An Introduction to Space Radiation Effects on Microelectronics” JPL-00-06

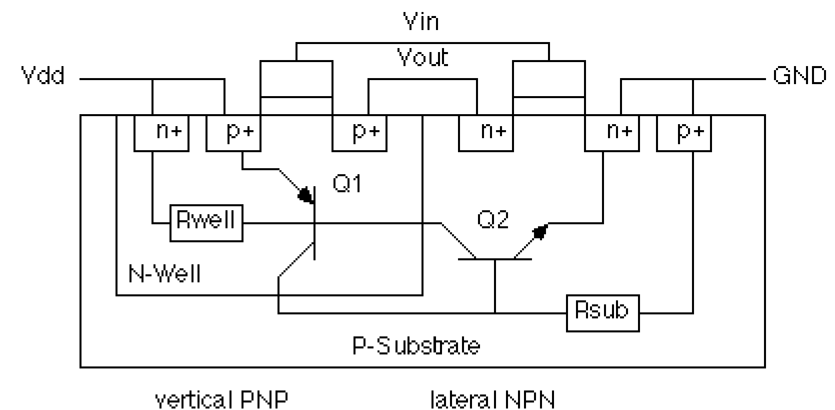
Single-Event Functional Interrupts (SEFIs)

- **An SEU or an SET to control logic can cause the device to operate incorrectly**
- **Components cannot “self-sense” it is in a SEFI mode – external circuitry must determine the SEFI has occurred an reset the device**
  - In FPGA devices, external memory scrubbers are used to both remove SEUs and monitor for SEFI behavior
  - Watchdog timers are commonly used on processors
- **SEFI modes can be very diverse:**
  - SRIO switches: lose the routing table
  - DRAM: burst errors of 1000s of corrupted words
  - FPGAs: programming data wiped clean
- **Affects system/component availability often either immediate or impending depending on SEFI mode or SEFI detection method**

LA-UR 11-03201

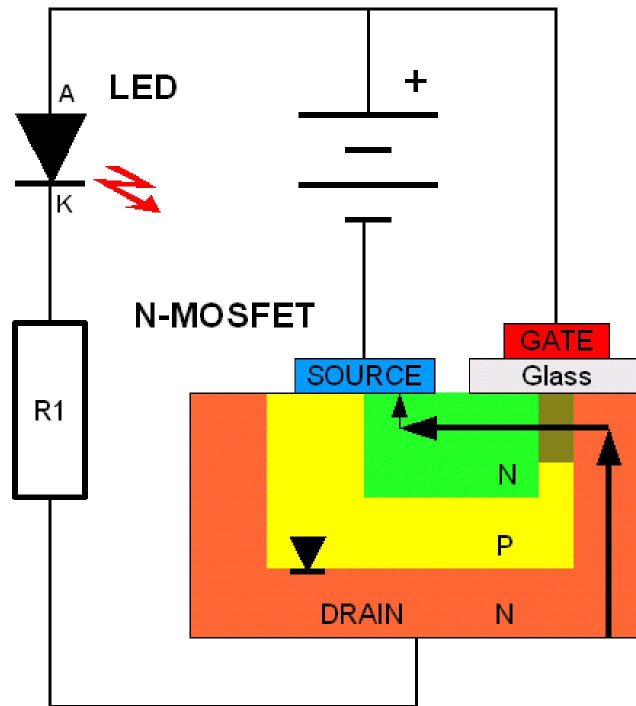
Single-Event Latch-Up (Latch-Up or SELs)

- **Traditional reliability issue with CMOS due to parasitic transistors caused by well/substrate contact**
  - Once turned on, current increases rapidly and destroys the part
  - Radiation is another avenue for turning on the parasitic transistor
- **Military/aerospace parts often have an epitaxial layer to prevent SEL, by localizing charge collection**
- **If the power can be removed immediately, then the device could survive the latchup event**
- **Affects component availability and usability, if the component is power cycled or is destroyed**



http://www.ece.drexel.edu/courses/ECE-E431/latch-up/latch-up.html

Single-Event Gate Rupture (SEGR)



- **Common only in power MOSFETs**
- **Ion-induced rupture of the gate oxide**
  - Dielectric and gate electrode material "melt and mix"
  - Ohmic short or a rectifying contact through the dielectric
- **Affects the system availability in an extreme manner**

Overview

- First half:
  - A Quick Introduction to Space Physics
    - The Earth's Magnetosphere
    - Solar Cycles, Solar Flares, and Coronal Mass Ejections
    - Cosmic Rays
    - A Few of My Favorite Particles
  - Radiation Effects in Electronics
    - A Historical Perspective
    - Current reliability issues with Modern Electronics
    - Radiation-Induced Failure Modes
  - Single-event effects in memory devices
- Second half
  - Single-Event Upsets and Transients in FPGAs, and Microprocessors
    - Failure modes and error rates in FPGAs
    - Mitigation methods for FPGAs
    - Failure modes in microprocessors
    - Mitigation methods for microprocessors
- Summary and Conclusions

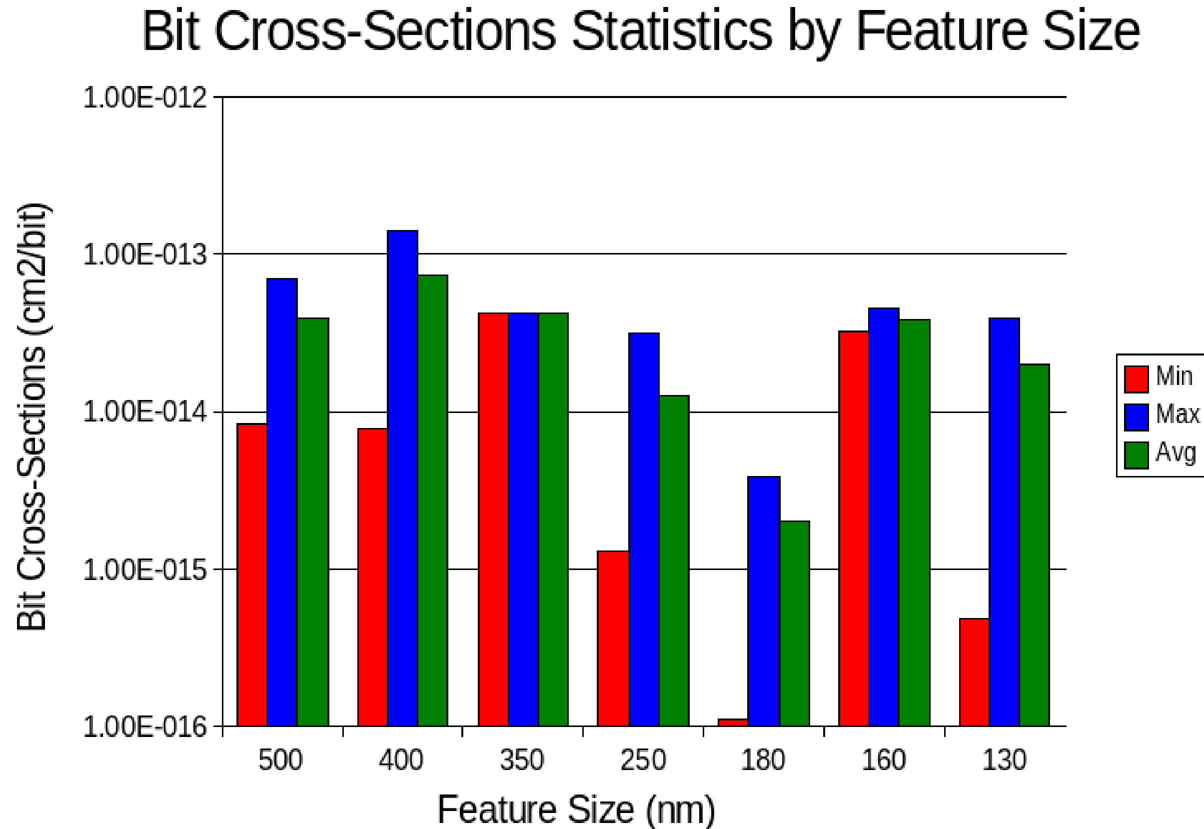
SEUs in SRAM

- **SEE mechanisms:**
  - Predominant mechanism is SEU
    - Thermal neutrons continue to be a problem
  - SEL remains somewhat commonplace
  - Micro-latching seen in some SRAM devices
- **Failure modes:**
  - Data corruption
- **Mitigation Methods**
  - Mask errors through redundancy, bit interleaving, Hamming codes
  - Repair errors through scrubbing

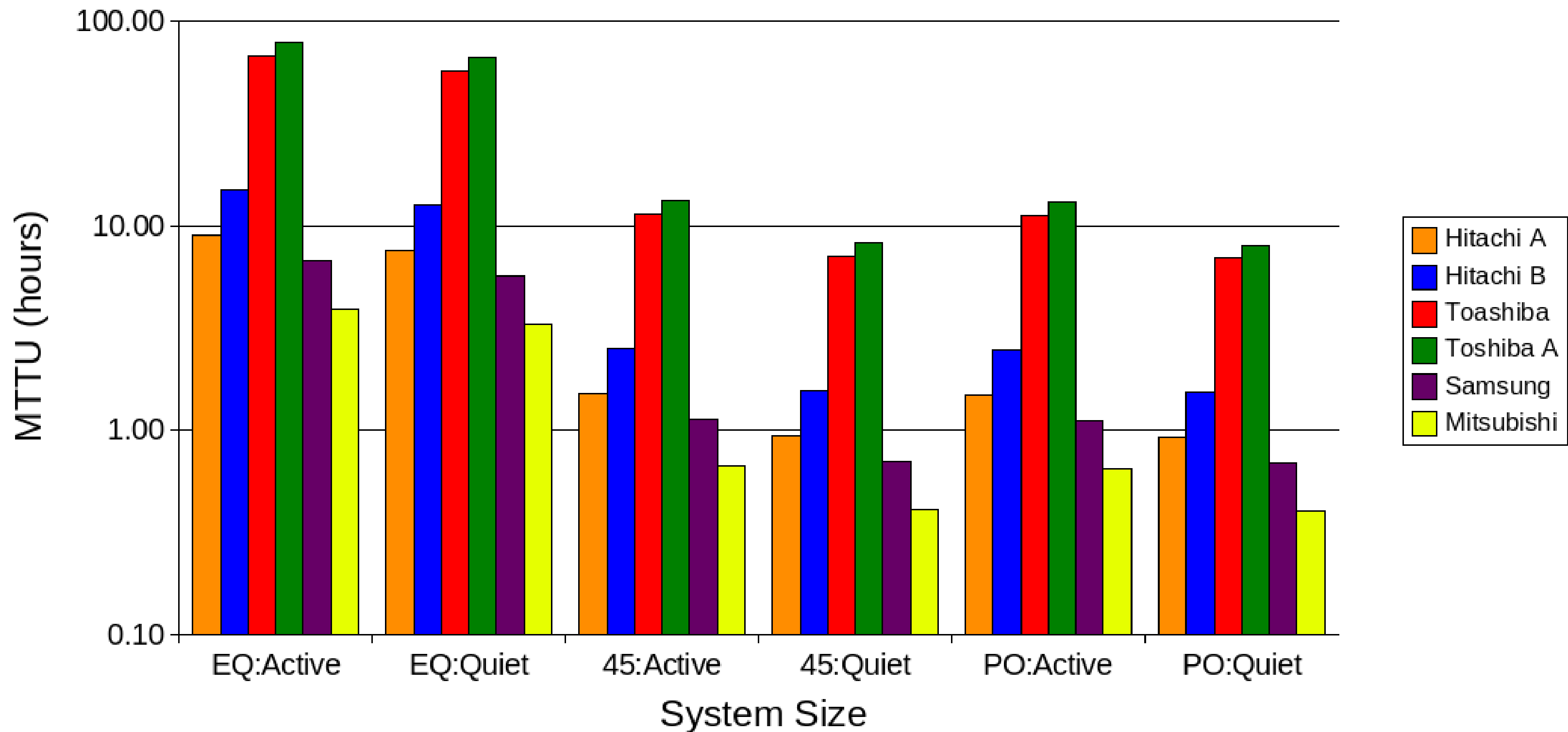
SEUs in SRAM

- **Neutron estimates were found in several NSREC papers**
  - Dyer, TNS 2004
  - Granlund, TNS 2006
  - Armani, TNS 2004
- **Increasingly difficult to find any radiation data on SRAM devices – many organizations are defaulting to the same radiation-tolerant QDRs and radiation-hardened SDR SRAMs**
- **"On-line" SRAM, such as the BlockRAM in Xilinx FPGAs, becoming large enough to do a soft store of data**
- **SDRAM playing some of the role that SRAM would use**

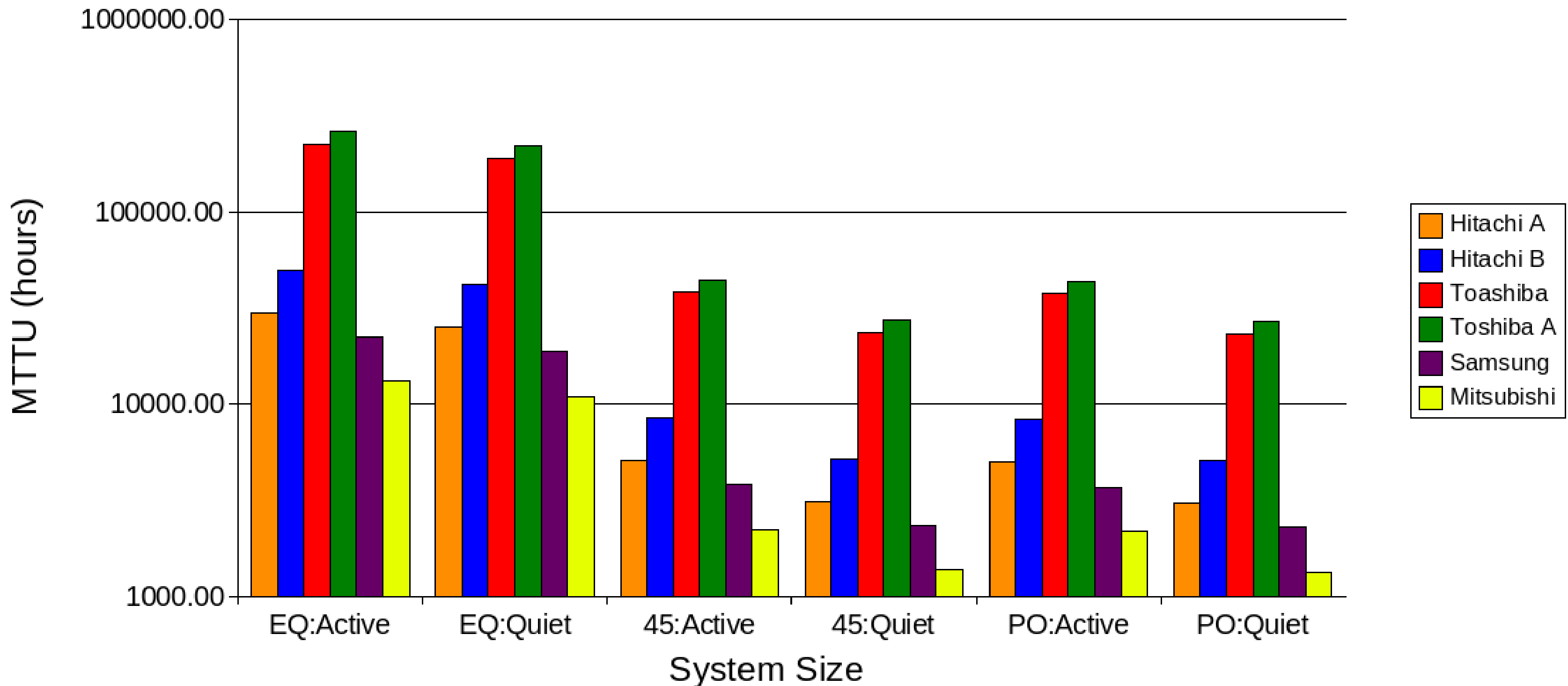
Bit Cross-Section by Feature Size



MTTU for All Neutrons: 512 Mb

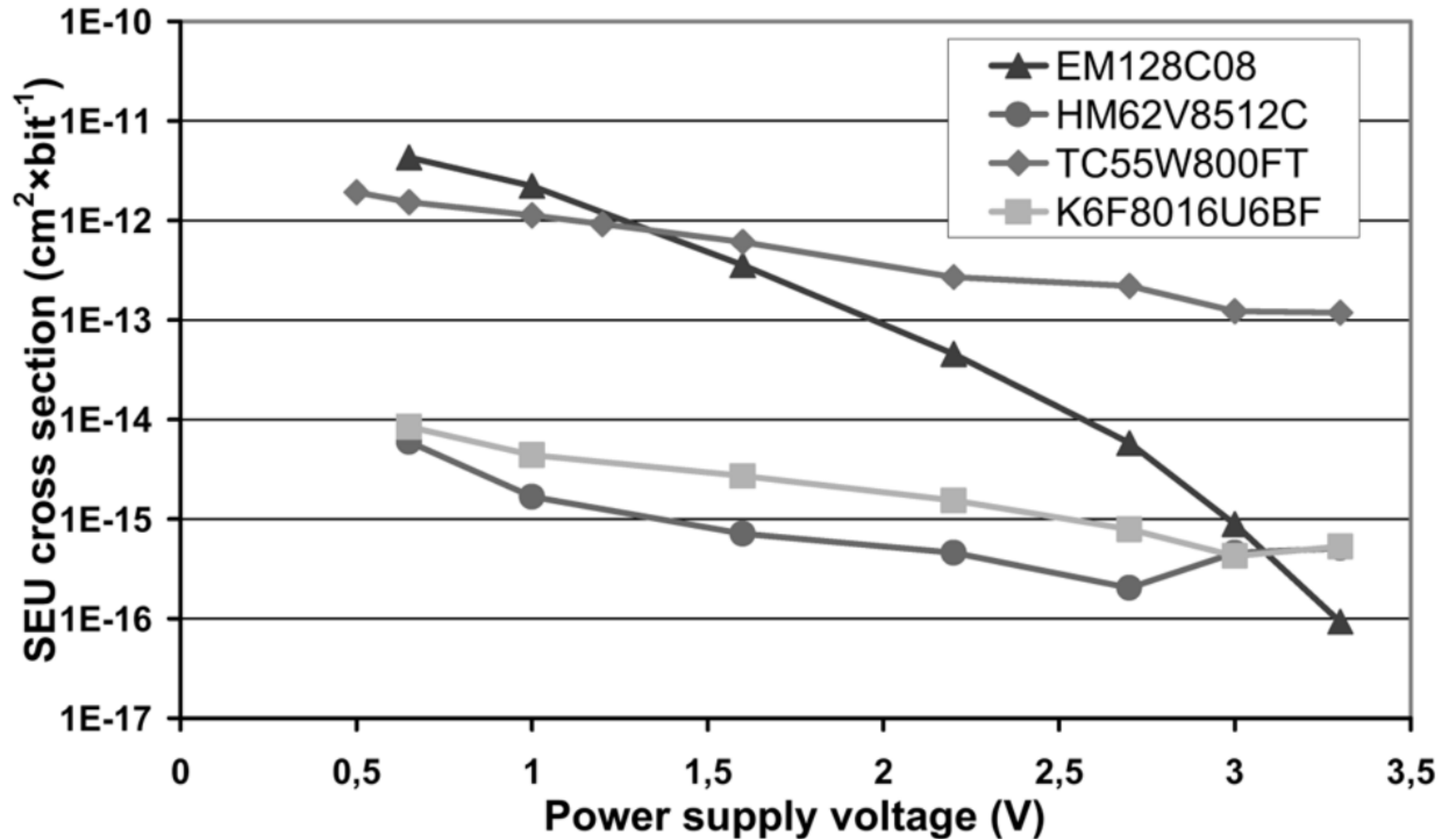


MTTU for All Neutrons: 512 Mb with ECC



Assuming 0.03% MBUs and Scrubbing

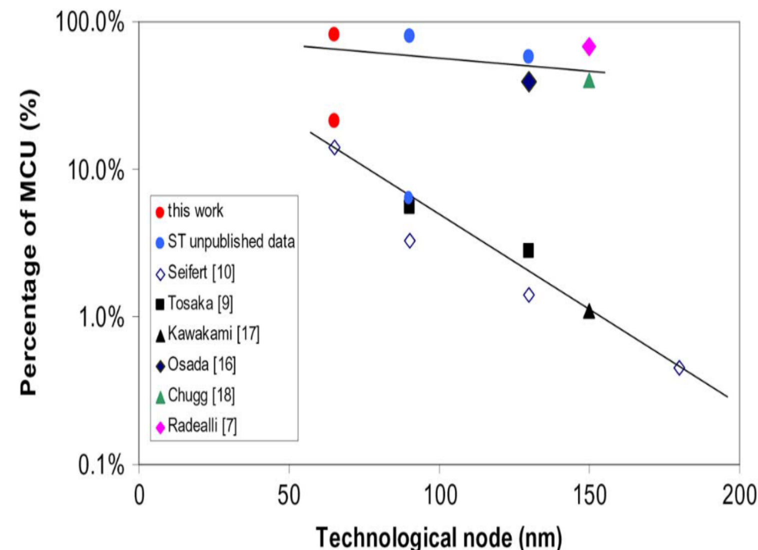
Bit Cross-Section and Voltage



Armani et al, "Low-Energy Neutron Sensitivity of Recent Generation SRAMs," Transactions on Nuclear Science, 51(5), October 2004, 2811--2816

Other Factors to Consider….

- **Part selection is important in the memory subsystem**
- **The layout of the memory is important**
  - Triple-well SRAM layouts have higher SER rates due to multiple-cell upsets (Gasiot, TNS Dec 2007)
  - Trench-in-Channel SRAM layouts have very low SER rates (Ziegler, "SER – History, Trend, and Challenges: A Guide for Designing with Memory ICs)
- **Using ECC-protected memory can help**
  - FPGAs read the memory as "raw" signals, will need to decode the data on the input side and encode on the output side
  - Typical ECC protection is single error correct and double error detect
    - Protection from all but the MBUs
  - Scrubbing can keep errors from accumulating, increasing the chance that ECC works



Gasiot et al, "Multiple Cell Upsets as the Key Contribution to the Total SER of 65 nm CMOS SRAMs and Its Dependence on Well Engineering, Transactions on Nuclear Science, 54(6), December 2007, 2468--2473

Feature Size	MBU% (educated guesses)
65 nm	12%
90 nm	5%
130 nm	2%
180 nm	0.5%

SEUs in DRAM

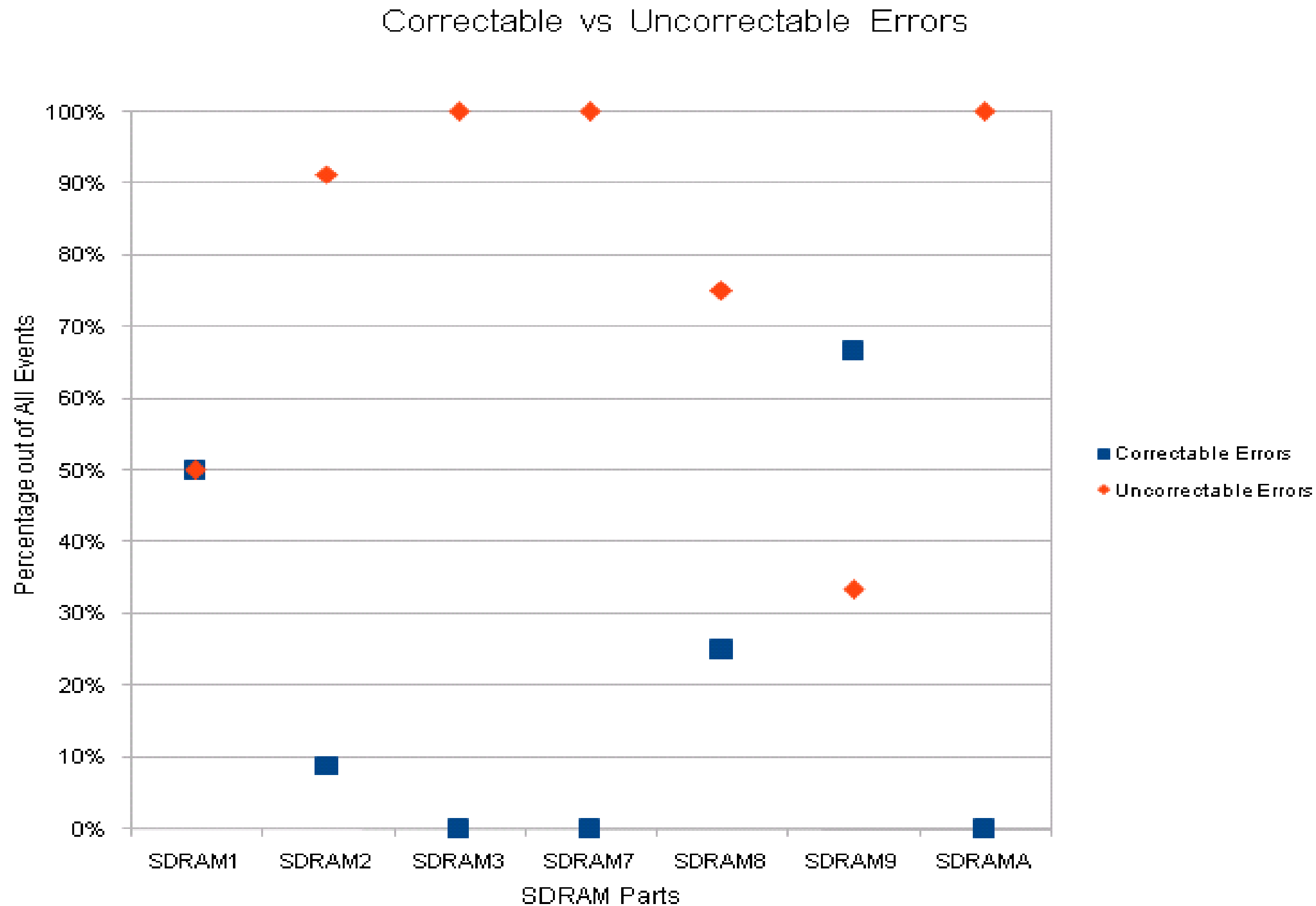
- **SEE mechanisms:**
  - Predominant mechanisms are SEUs in the memory array and SEU/SETs in the control circuitry that cause a SEFI mode of burst errors
  - SEL or high-current events an ongoing problem
  - "Stuck bits" are not uncommon – a single bit or a page will become stuck at some value for some number of seconds, which will eventually relax back to a writeable state
- **Failure modes:**
  - Data corruption
- **Mitigation Methods**
  - Mask errors through redundancy, bit interleaving, Hamming codes for bitflips
  - RAID striping might be necessary to mitigate SEFI modes
  - Repair errors through scrubbing

SDRAM SEU Bit Cross-Sections

Sample	SEU Bit Cross-Section (cm^2/bit)	SEFI Bit Cross-Section (cm^2/device)
SDRAM1	2.14×10^{-20}	4.76×10^{-12}
SDRAM2	2.15×10^{-20}	1.62×10^{-10}
SDRAM3	7.54×10^{-20}	7.71×10^{-12}
SDRAM7	7.23×10^{-20}	1.79×10^{-11}
SDRAM8	1.72×10^{-20}	6.94×10^{-11}
SDRAM9	$(0, 2.32 \times 10^{-19})$	1.26×10^{-11}
SDRAMA	4.43×10^{-20}	$(0, 2.20 \times 10^{-11})$

LA-UR 11-03201

SDRAM EDAC Failures



Overview

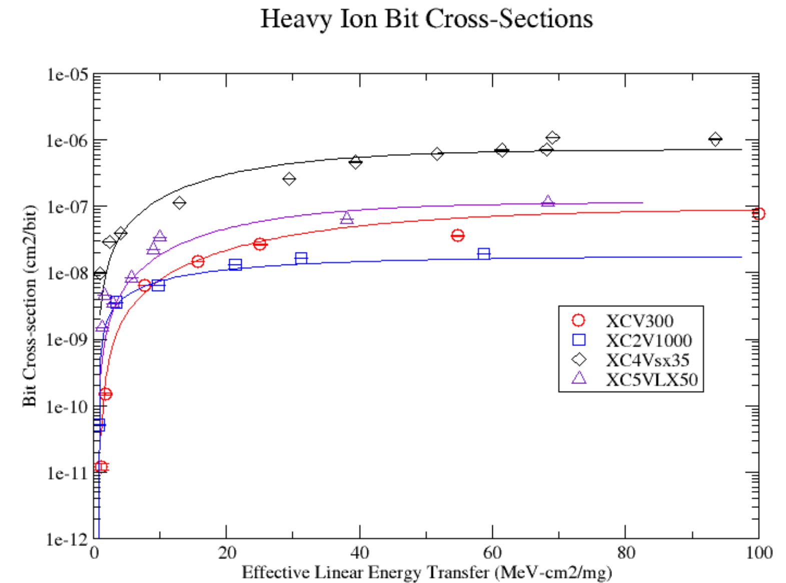
- **First half:**
  - **A Quick Introduction to Space Physics**
    - The Earth's Magnetosphere
    - Solar Cycles, Solar Flares, and Coronal Mass Ejections
    - Cosmic Rays
    - A Few of My Favorite Particles
  - **Radiation Effects in Electronics**
    - A Historical Perspective
    - Current reliability issues with Modern Electronics
    - Radiation-Induced Failure Modes
  - Single-event effects in memory devices
- **Second half**
  - **Single-Event Upsets and Transients in FPGAs, and Microprocessors**
    - Failure modes and error rates in FPGAs
    - Mitigation methods for FPGAs
    - Failure modes in microprocessors
    - Mitigation methods for microprocessors
- **Summary and Conclusions**

SEUs in Xilinx FPGAs

- **SEE mechanisms:**
  - Predominant mechanism is the SEU
  - SEU-induced SEFIs very, very rare in terrestrial-based systems
  - Xilinx has no SEL problems
  - SETs impossible to see through the SEUs
- **Failure Modes:**
  - SEUs in user memory (flip-flops, BlockRAM) can change intermediate data
  - SEUs in routing can either short or open your routing
  - SEUs in lookup tables (LUTs) can change logic values
  - SEUs in half-latches (VI) or power network (VII) can change logical constants for a region of the design
  - SEFIs in control logic can reprogram or de-program device
- **Mitigation Methods:**
  - Mask through redundancy methods
  - Repair through scrubbing

Static Test Results:
Bit Cross-Sections

Device	Energy (MeV)	σ_{bit} (cm^2/bit)
XCV1000	63.3	$1.32 \times 10^{-14} \pm 2.69 \times 10^{-17}$
XC2V1000	63.3	$2.10 \times 10^{-14} \pm 4.64 \times 10^{-17}$
XC4VLX25	63.3	$1.08 \times 10^{-14} \pm 2.71 \times 10^{-17}$
XC5VLX50	65	$1.10 \times 10^{-14} \pm 2.08 \times 10^{-16}$
XC5VLX50	200	$1.41 \times 10^{-14} \pm 1.18 \times 10^{-16}$



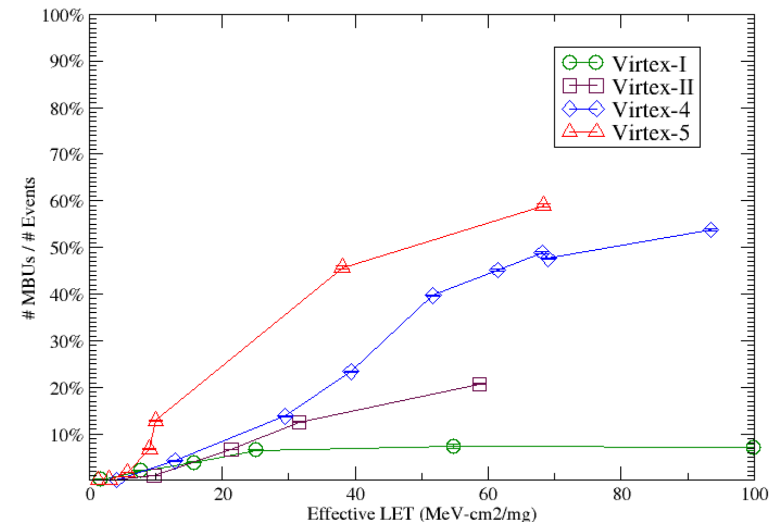
Heavy Ion Data

Static Test Results: MBU Data

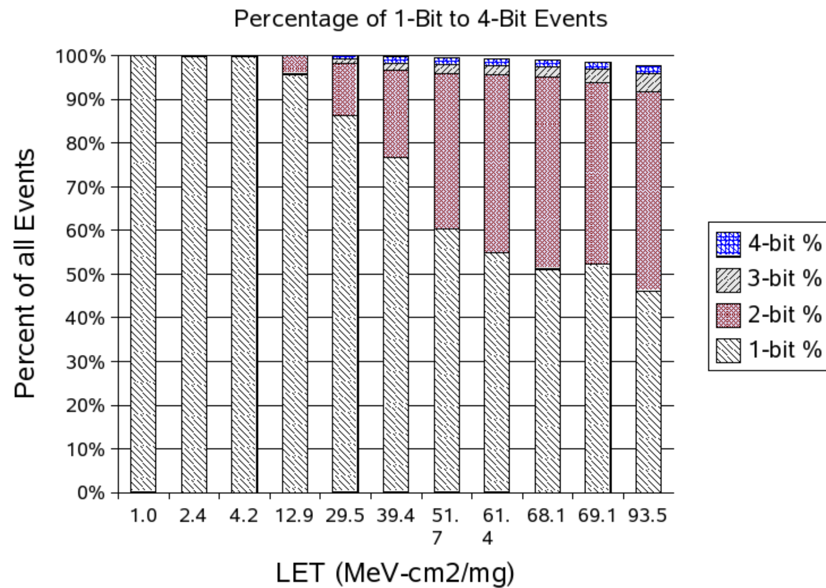
Device	Energy (MeV)	1-Bit Events	2-Bit Events	3-Bit Events	4-Bit Events
XCV1000	63.3	99.96 %	0.04%	0.00%	0.000%
XC2V1000	63.3	98.42%	1.16%	0.01%	0.001%
XC4VLX25	63.3	96.44%	2.99%	0.05%	0.005%
XC5VLX50	65	94.23%	5.43%	0.30%	0.03%
XC5VLX50	200	89.86%	8.79%	0.92%	0.43%

Proton Data

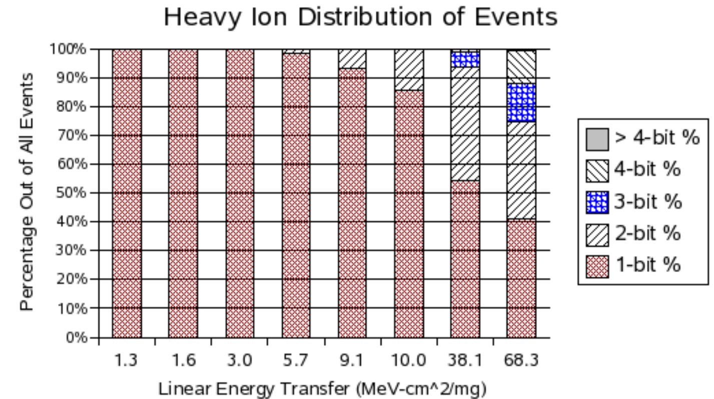
Heavy Ion Data

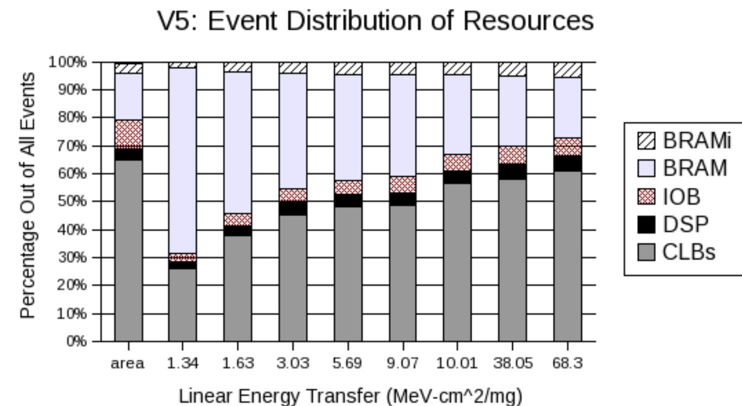


Static Test Results: Distribution of Events



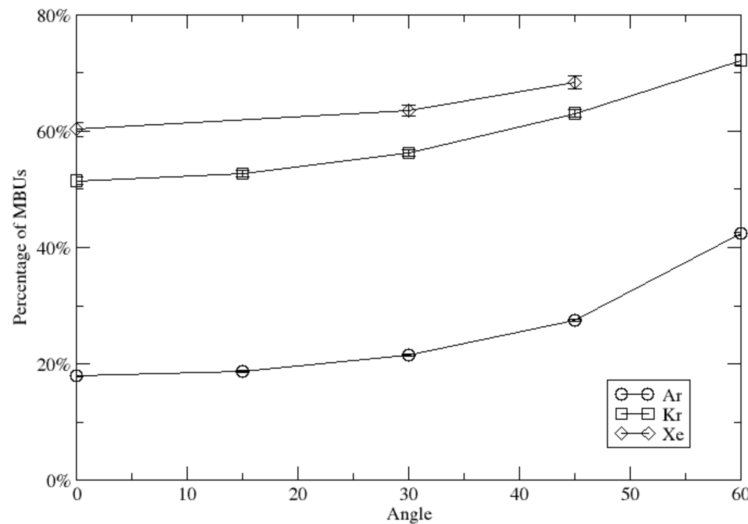
V4 Data



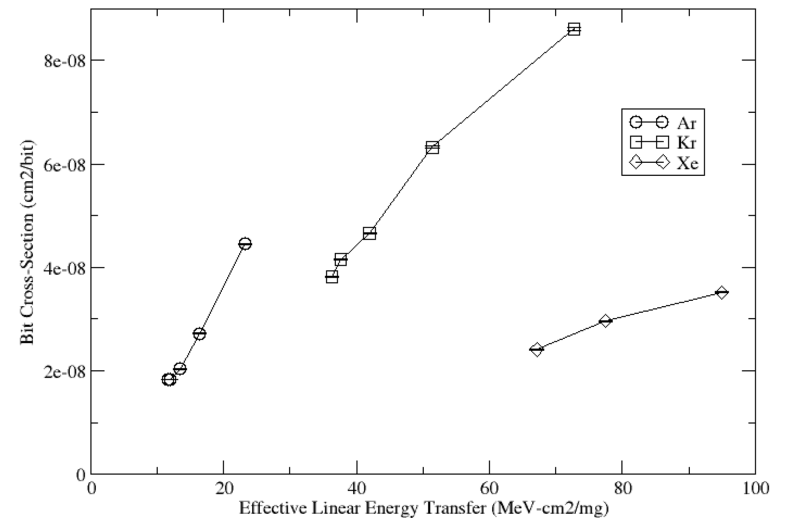


V5 Data

Static Test Results: V5 Angular Data



MBUs



Bit Cross-Sections

Xilinx SEFI Modes

- **Side Effects of SEFIs:**
  - Immediate loss of full device function
    - POR, GSIG, Scrub
    - Scrub SEFI could damage device
    - Reprogram by pulsing PROG as soon as possible
  - No impact to device function
    - SMAP/JTAG, FAR
    - Reprogram as soon as possible
  - Possible loss of full device function
    - Shutdown SEFI
    - Mitigate by scrubbing CFG_CLB column.

Virtex-4: How Often Do SEFIs Occur on Orbit?

Mean Time to SEFI for Selected Orbits in YEARS, calculated by CREME96

Orbit	Altitude (km)	Incl*	POR	GSIG	SMAP+	TOTAL
LEO	400	51.6°	896	1042	1374	356
	1200	65.0°	25	27	29	9
GPS	20200	55°	240	308	595	110
GEO	36000	0°	67	85	207	32

* Incl = Inclination
SMAP+ = SMAP & FAR SEFIs combined

Virtex-4: How do SEFIs affect Availability?

- **When SEFIs occur, the scrubber needs to detect and correct the SEFI**
- **Correction of the SEFI includes stopping the circuit and performing a complete off-line reconfiguration of the device**
  - SEFI detection << 1s
  - Complete reloading of the application from PROM/CRAM: ~ 6 minutes (almost entirely decompression time in SPARC software)
  - Complete reloading of the application from SDRAM: < 10 s
- **As a worst case scenario, this process will take 6.02 minutes and the device is inoperable the entire time**
- **As a best case scenario, this process will take between 1-11 seconds and the device is inoperable the entire time**
- **Poisson statistics complicate these problems**

Virtex-4: Availability Rate from SEFIs Assuming 6 Minute Recovery

Orbit	One SEFI	Two SEFIs	Three SEFIs	Four SEFIS
LEO (400K)	0.999999968	0.999999936	0.999999904	0.999999872
LEO (1200K)	0.999998741	0.999997482	0.999996223	0.999994964
GPS	0.999999896	0.999999792	0.999999687	0.999999583
GEO	0.999999639	0.999999278	0.999998917	0.999998556

- **Even the worst, worst case scenario (LEO 1200K with 4 SEFIs in the time frame) meets the minimum availability rate of most satellites – 5 “9s”**
- **In the best case scenario (LEO at 400KM) the maximum availability rate is 7 “9s”**

SEUs in SRAM-based FPGAs

- **As the SEFIs seem reasonable, the next problem with SRAM-based FPGAs are SEUs**
  - First, we will discuss how SEUs affect SRAM-based FPGAs
  - Second, we will discuss error rates for unmitigated circuits
  - Third, we will discuss error rates for mitigated circuits

SEUs in SRAM-based FPGAs

- **There are many different types of memory cells within the device each with their own sensitivity and consequence to radiation-induced faults, including configuration memory and user memory**
  - Given the sheer quantity of memory, the error rate on the memory is not insignificant
- **Configuration memory controls much of the device:**
  - Defining the "equation" in LUTs
  - Defining the functionality of LUTs and user FFs
  - Defining the routing for the circuit
  - Defining the logical constants for the circuit
- **Designers have a few types of user memory available:**
  - FFs for pipelining data in the circuit
  - LUTs or BRAM configured as ROMs for data lookup – designers can approximate more complicated calculations by interpolating pre-calculated values stored in ROMs
  - LUTs or BRAM configured as RAMs for storing larger chunks of inflight data

Configuration Memory: Lookup Table Vulnerabilities

- **The logic in FPGAs is predominantly implemented in lookup tables (LUTs) which translate 2:1, 3:1, and 4:1 logic as a memory table with a decoder**
- **There are some embedded user cores in the device:**
  - Inverters
  - Multipliers/DSP units
  - DCMs
  - Processors
- **Under some circumstances the CAD tools will use LUTs instead of embedded cores:**
  - Not enough embedded cores were available
  - Inversion simplified into a LUT with another equation
- **Two types of LUT vulnerabilities:**
  - LUT equation changes
  - LUT “control” or “functionality” changes

Configuration Memory: LUT Equation Vulnerabilities

- **Configuration memory bits are used to store the LUT's values**
  - The LUT takes on a slightly different equation due to the changes
  - In the figures to the right, the 4 input AND gate equation is changed into a constant 0 equation
- **Except in cases of multiple-bit upsets, a LUT with an SEU in it is still correct for 15 out of 16 input combinations**
- **Logic masking can cause output errors from one LUT to not become an output error for the circuit**
- **Only 2-5% of upsets that occur in the V-4 device occur in the LUTs**
- **At most 1.5% of the MBUs that occur in the V-4 device occur in the LUTs and mostly at very high LET heavy ions**

LUT

Inputs: F1 = 1, F2 = 1, F3 = 1, F4 = 1; O = 1

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	1

=F1*F2*F3*F

Original

LUT

Inputs: F1 = 1, F2 = 1, F3 = 1, F4 = 1; O = 0

0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

Configuration Bit SEU

O=0 (constant zero)

After Upset

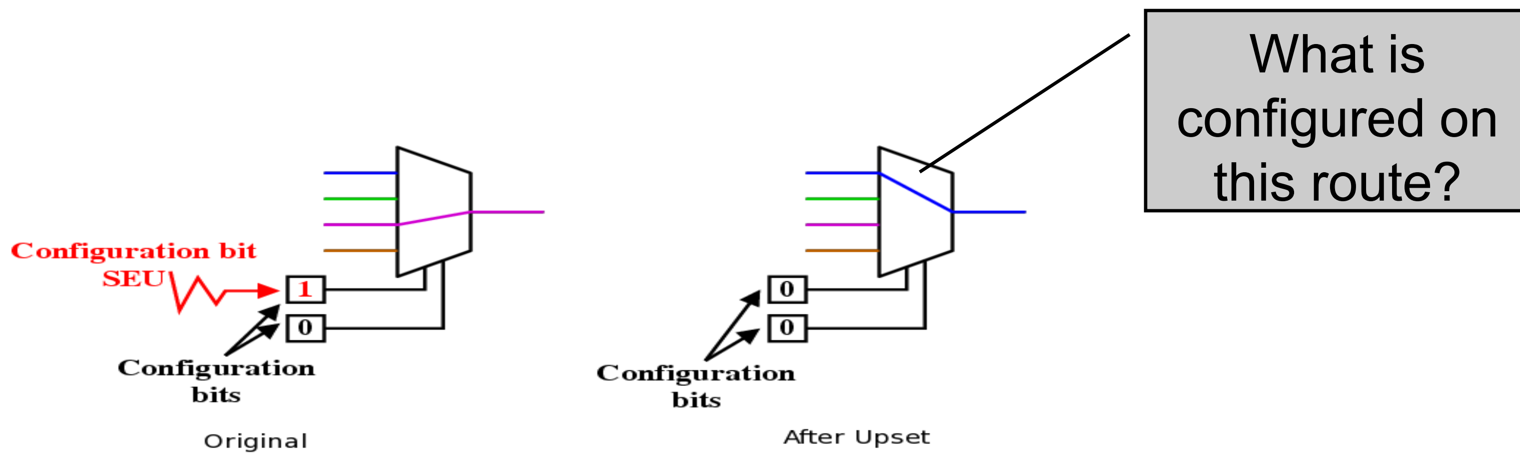
Configuration Memory: Routing Vulnerabilities

- **Routing comprises ~80% of the device**
  - Errors in routing affects routing data to LUTs, DSPs, Microprocessors, BRAM
  - Errors in routing affects routing global signal, such as clocks and resets
- **In analysis of failures in unmitigated designs 2/3rds of the "sensitive" cross-section (i.e., bits that when flipped cause noticeable errors in the output stream) is in the routing configuration memory**
  - Routing errors are not sensitive to input data
  - Corrupt routing is wrong no matter what the data is
- **Global signals are particularly vulnerable to SEUs**
  - Clock and reset trees can route to the entire device and SEUs can open or short the trees
  - Corrupting a global signal close to the input pin can affect the entire circuit
  - Corrupting a global signal near the leaves will have a more limited impact
  - Follow up research into domain crossing errors is showing that one of the vulnerabilities is that MBUs will switch the global signal routing – clocks from two domains switching, clocks and resets switching, etc.

Configuration Memory:

Mux Vulnerabilities

- **Much of the routing in the Virtex-4 is mux-based.**
  - Muxes in the routing switches and the slices determine how the data is moved from point A to point B.
  - Routes are "defined" by moving data from one mux to the next mux until it reaches it's destination
  - Muxes have a specific select line values stored in configuration memory that determines the input line on a route
- **An SEU can change the configuration memory storing the select line values, causing the route to be driven by the wrong signal**
  - Using a wire that is actively used by different logic (OMUX)
  - Using a wire that is being driven by a half latch, which imitates a stuck-at value
  - Opening or shorting a route
- **SEUs in routing are 32-50% of all upsets in heavy ion on the Virtex-4, based on energy**



Configuration Memory: Upset Rates

Mean Time to Upset for Selected Orbits in DAYS, calculated by CREME96

Orbit	Altitude (km)	Incl*	SX55	FX60	LX200
LEO	400	51.6°	0.95	1.52	2.65
	1200	65.0°	29.7	28.0	82.9
GPS	20200	55°	4.03	3.79	11.3
GEO	36000	0°	14.9	4.03	12.0

From the Virtex-4QV Static SEU Characterization Summary: http://www.xilinx.com/products/v4qv/index.htm

User Memory

- **SEUs in these memory cells are a concern and lead to a corruption of circuit state**
  - SETs in the device would be visible through a change in the user FF cross-section
  - No published data on SETs in the user FFs
- **SEUs in user memory are difficult to mitigate**
  - Mitigate the logic attached to user FFs
  - Triplicate the BRAM
- **User memory that can be written to cannot be scrubbed traditionally without corrupting the contents of the memory**
  - Use Xilinx's BRAM scrubber to scrub user memory in BRAM
  - Mitigate the logic around the other user memory so there is no need to scrub

User Memory: BRAM Upset Rates

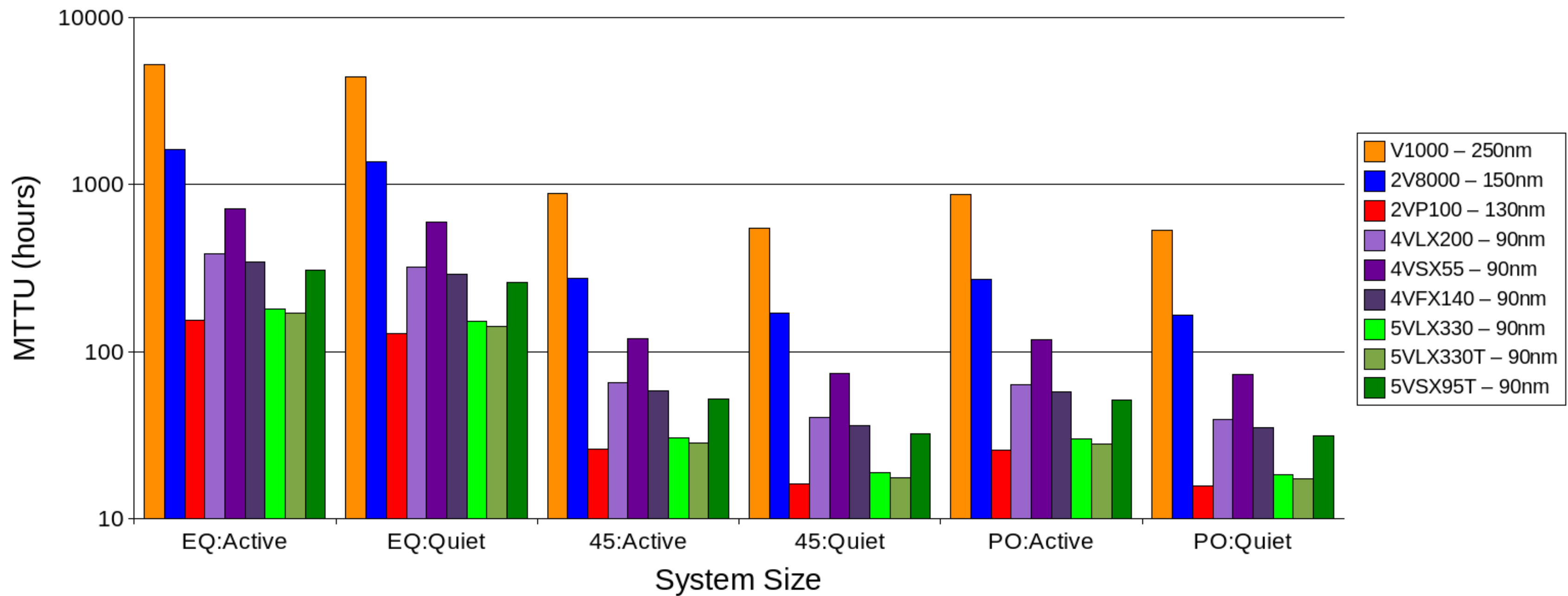
Mean Time to Upset for Selected Orbits in **DAYS**, calculated by CREME96

Orbit	Altitude (km)	Incl*	SX55	FX60	LX200
LEO	400	51.6°	0.85	0.61	0.89
	1200	65.0°	16.7	12.1	17.6
GPS	20200	55°	4.25	3.08	4.46
GEO	36000	0°	13.0	4.71	3.25

* Incl = Inclination SMAP+ = SMAP & FAR SEFIs combined

From the Virtex-4QV Static SEU Characterization Summary: http://www.xilinx.com/products/v4qv/index.htm

MTTU for Largest Devices for the Virtex-I to Virtex-5 for Airplane Environments



Errors in the Output Data Stream

- **In unmitigated circuits approximately only 1-20% of the device will cause a noticeable output error from the user circuit**
  - These numbers are based on fault injection and beam testing using random input data
  - These numbers are based on designs that do not have mitigation applied to user circuit
  - Approximately 1/3rd of errors are in the logic and 2/3rd are in routing
  - These numbers are also design-dependent, which means that testing will be necessary to determine the sensitivity of your design to output errors
- **Each circuits has its own inherent sensitivity to errors**
  - Many digital signal processing applications are very insensitive to errors
  - Circuits with a lot of feedback loops, state and where the output data is "well-tied" to the input data are more sensitive to errors
- **It takes 5 seconds for the scrubber to detect the error in the bitstream and fix it**
  - Resynchronizing the circuit could be very quick or very slow depending on the circuit

Error Rates for Unmitigated Circuits

- **The LX200 is a 51Mb device**
- **When there is more 60 errors per year, the availability rate will fall below the 5 "9"s mark**
  - Small circuits in many orbits might not need mitigation
  - Large circuits in all orbits will need some mitigation

Orbit	1% -- upsets per year	1% -- lost seconds per year	20% -- upsets per year	20% -- lost seconds per year
LEO (400K)	13.04	65.20	260.79	1303.97
LEO (1200K)	370.52	1852.59	7410.35	37051.74
GPS	57.82	289.10	1156.42	5782.09
GEO	61.33	306.67	1226.68	6133.39

Mitigating Circuits

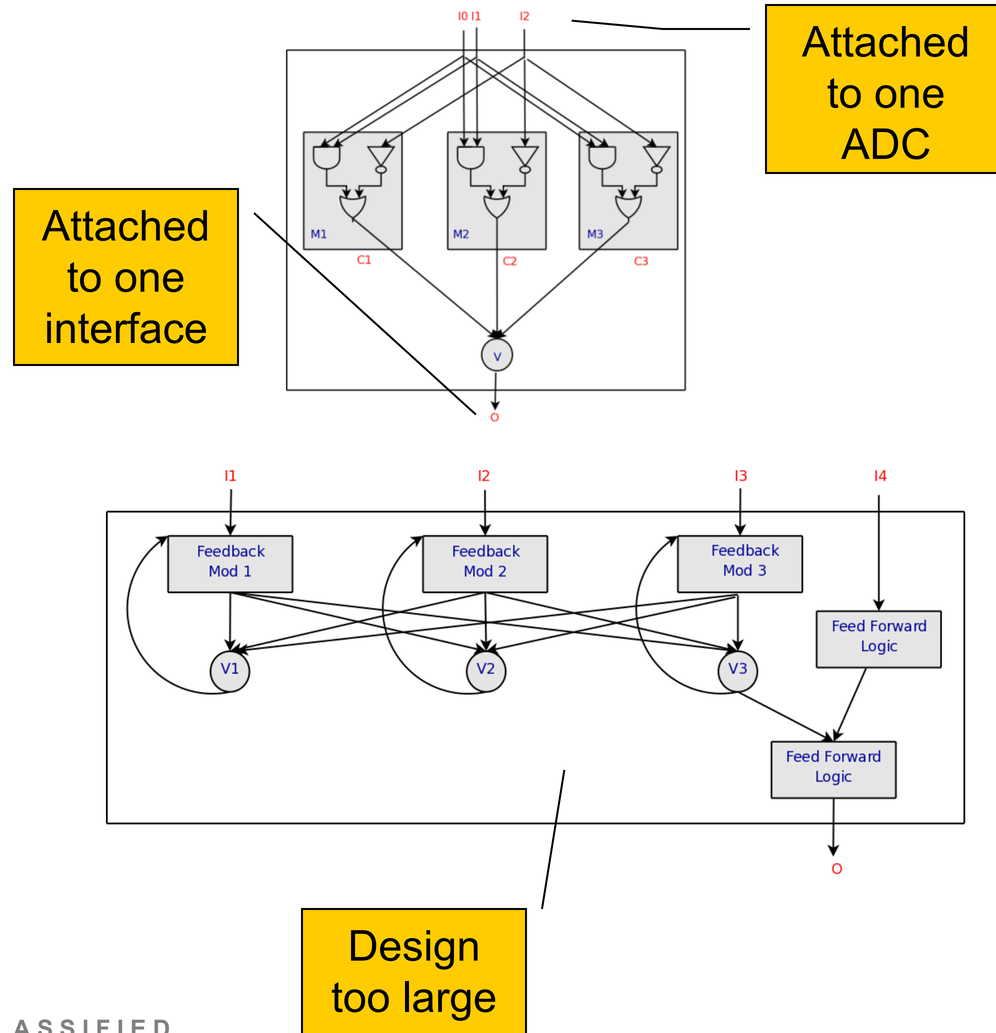
- **“Gold standard” for “perfect” triple-modular redundancy is**
  - Triplicated input and output data streams
  - Triplicated clock, resets, and other global signals
  - Triplicated logic
- **A fully TMR-protected FPGA design should mask all <u>single-bit</u> SEUs as long as there is only <u>one</u> in the system at a time.**
  - Scrubbing must ensure that all upsets are removed and the circuit resynchronized before the next upset occurs.
  - TMR is not as successful with either multiple-bit upsets or multiple independent upsets.
- **While the concept of TMR is simple, the implementation of TMR in FPGA designs is often not simple.**
  - The circuit description could vary widely from the circuit implementation.
  - A number of scenarios exist that can affect the reliability.

Correct Circuit Design for TMR-Protected FPGA Designs

- **Do not manually apply TMR in VHDL.**
- **Use one of the three mitigation tools to apply TMR.**
  - Both automatically apply TMR to post-synthesis circuit representations, called EDIF.
  - After synthesis, major circuit optimizations will not occur.
  - Less likely to lose TMR-based redundant modules
  - Feedback loops properly cut
- **Do not expect TMR to solve any existing problems in the design.**
  - If your design cannot meet timing, has design flaws or is really large without TMR, applying TMR will not fix these problems.

Design Constrained Scenarios that Affect TMR

- **Sometimes designers are unable to fully triplicate the design due to either area or pin issues.**
- **Triplicating input/output signals can be impossible due to pin constraints and can be difficult to manage due to skew.**
- **Triplicating all of the logic might not be possible due to the chosen device's size.**
  - BL-TMR can automatically apply partial TMR for this scenario through prioritized redundancy based on device size.
- **Without full triplication of logic and signals, some unprotected cross-section will exist.**



SEUs/SETs in Processors

- **SEE mechanisms:**
  - SEUs in caches which may or may not be corrected by ECC
    - Not all processors have ECC or only ECC on the L2 cache
    - Not all errors are ECC-correctable
  - SEUs in the registers and SETs in the logic are the predominant mechanisms
  - No known SEL
  - Rather exotic set of SEFI modes
- **Failure modes:**
  - Data corruption by SEUs in registers
  - Data corruption by SETs in gates
  - Unrepeatable crashes
- **Mitigation methods:**
  - Mask errors through redundancy
  - Clever uses of duplicate computation and checkpoints using multiple cores

Microprocessor Reliability

- **Microprocessor reliability in high radiation environments has two components:**
  - Static cross-section: single-event upsets (SEUs) in the memory structures
  - Dynamic cross-section: single-event transients (SETs) in the functional logic
- **SEUs in the memory structures can cause changes in intermediate processing values, changes to instruction operators/operands, or changes to cached data**
  - Standard test procedures make SEUs reasonably easy to measure
- **SETs in the functional logic can cause changes to the intermediate processing values**
  - Currently, there is no standard test procedure for SETs
  - SETs are dependent on the clock frequency and how the device is being used
  - A likely factor in silent data corruption (SDC)
- **As of right now, not a great understanding which causes more problems, although SEUs cache likely dominates many current devices**

Software Reliability

- **The fundamental question for software reliability is "how reliable is program on a particular processor"**
  - The microprocessor provides a basis for the software reliability
  - Without mitigation, software will translate many of the hardware reliability problems in to either SDC or crashes
- **One of the difficulties with software reliability is that it's hard to translate static and dynamic cross-section data for the processor into an idea of how reliable software will be:**
  - Translating a hardware failure in the cache to a software failure depends on how caches and registers are being used
  - Translating transient hardware failures into software failure, depends on the software
  - Visibility of errors dependent on the software – some software is more resistant to errors
- **As best we know, software reliability is probably a subset of the static cross-section and dynamic cross-sections**

Intel Microprocessor Estimates

- Intel publishes without numbers or units on their y-axis
- They've told NASA that they are having problems with SETs in the combinatorial logic and SEUs in the register files
- They've told us that their caches have no visible SEUs due to ECC and bit-interleaving, but they have stopped listing whether caches are ECC-protected
- It's also clear that they are addressing an issue with cosmic rays, since they have become progressively more radiation-hardened over the years
- The best number we have from them is a server quality microprocessor has a fail rate of once every 25 years, assuming that number is from sea level, that means a fail ever 123-1210 hours at 60,000'.

Recent LANSCE Test Results

- **90nm Silicon-on-Insulator (SOI) Microprocessor:**
  - Most common failure mode is microprocessor crashes (single-event functional interrupts)
  - 3.5% of tests exhibit SDC
- **90nm SOI PPC-based Microprocessor:**
  - Static testing measures sensitivity of caches and registers to SEUs
  - Dynamic testing of benchmarks do not exhibit any SDC over 48 hours of testing – caches disabled for the test
- **65nm bulk Silicon, multi-core digital signal processor (DSP):**
  - DSP fails every 1-5 minutes at LANSCE due to a large SEFI cross-section from the JTAG/ICEPICK
  - SDC affects nearly all tests

DSP Cache SEUs

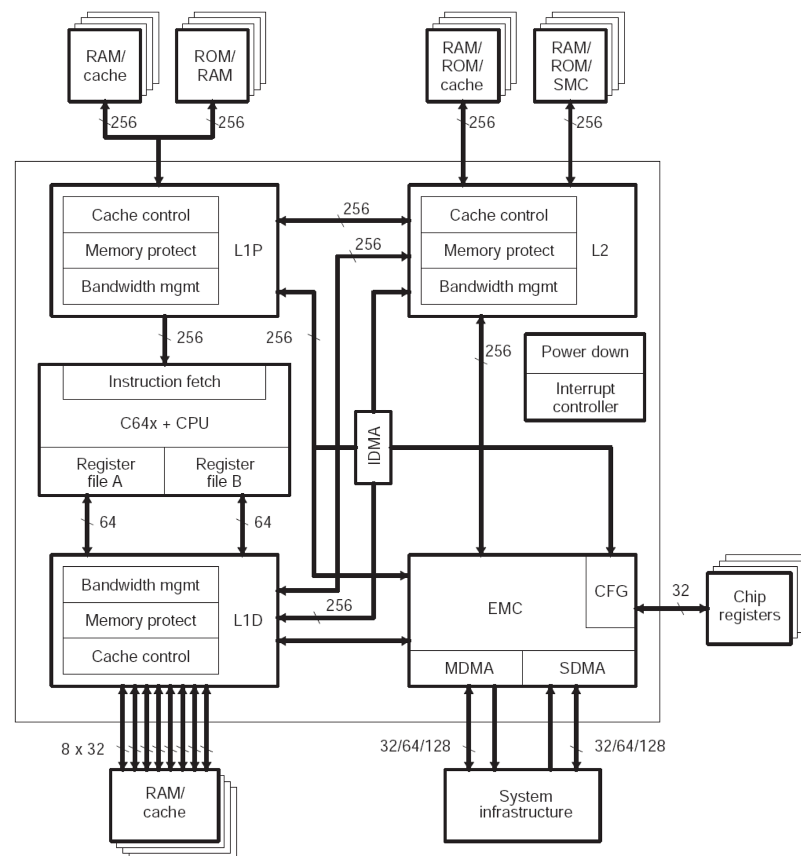
- **Based on sheer size, SEUs in the cache dominate the non-SEFI reliability problems**
- **SEUs in the cache cause a number of different failures with different signatures**
- **SDC**
  - Evidence of data corruption for variables stored in caches
  - No evidence of corruption of operands, corruption of operators, or SETs – errors are repeatable
- **Fatal SEUs**
  - Subroutines locations corrupted in the cache cause the program to crash
  - Global constants corrupted in the cache cause the program to stop executing

How Role Does Cache Utilization Play in Software Reliability

- Recently, a lot of research points to the caches being one of the first order effects affecting software reliability
  - Sandia's work on soft and hard FPGA processors shows that using FT caches allows software to execute longer in radiation environments
  - BYU's work on soft FPGA processors shows that the processor's memory is 60% of the footprint and can be fragile in radiation environments
- Study hypothesis: How does cache utilization play a role in software reliability?
  - How much cache is being used?
  - How long data stays "resident" in the cache?
  - Can we pose the question of "how reliable is a software code" as "how is the cache utilized?"
  - Can we improve software reliability by mitigating cached data variables?
- Currently studying the TI DSP C6474

TI C6474 Tri-core DSP

- We tested the device in a TMDXEVMC6474 development board
- Tri-core, fixed-point DSP
- Focused on testing the Megamodules in each core
- Parity-protected L1 cache
- ECC-protected L2 cache: 72Mb total



Static Test Results

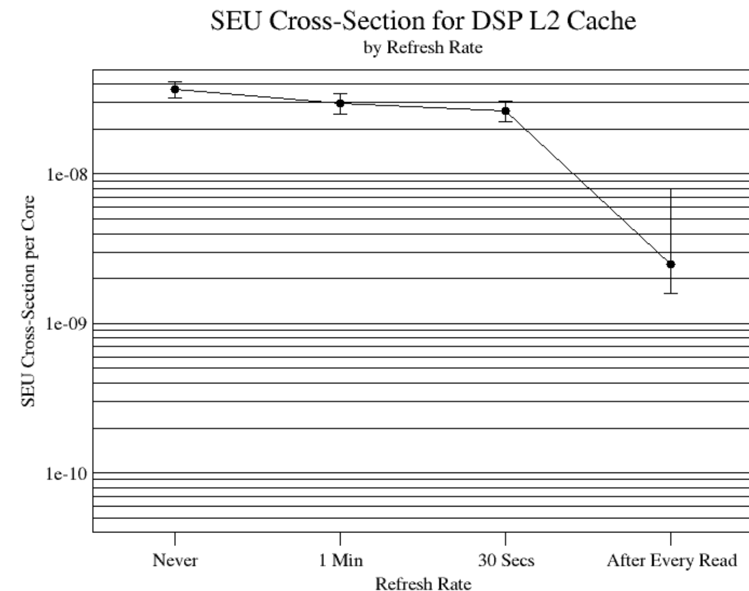
	Cross-Sections
SEU bit-cross-section	7.30×10^{-16} cm^2/bit (4.45×10^{-16} cm^2/bit, 1.12×10^{-15} cm^2/bit)
SEU device cross-section	1.65×10^{-7} cm^2/device (1.01×10^{-7} cm^2/device, 2.54×10^{-7} cm^2/device)
SEFI cross-section	4.13×10^{-10} cm^2/device (1.76×10^{-10} cm^2/device , 8.16×10^{-10} cm^2/device)

Unmitigated Software Test Results

- **Not all SEUs will create SDC, crashes, or other types of errors**
  - Device utilization, logical masking, and compensating failures lower the error rate
  - SEUs can be categorized into ones that create observable errors by affecting calculations and ones that do not
- **The length of time the data is in the cache is important**
  - For data that is read once, the SEU would need to occur in between writing and reading – any SEUs after reading would not be observed and likely overwritten
  - Global values or constants are more likely to have observable errors because the values are read repeatedly without refreshing
- **The amount of data needed for a calculation is important**
  - The more data that a calculation uses, the more likely SDC will affect the calculation

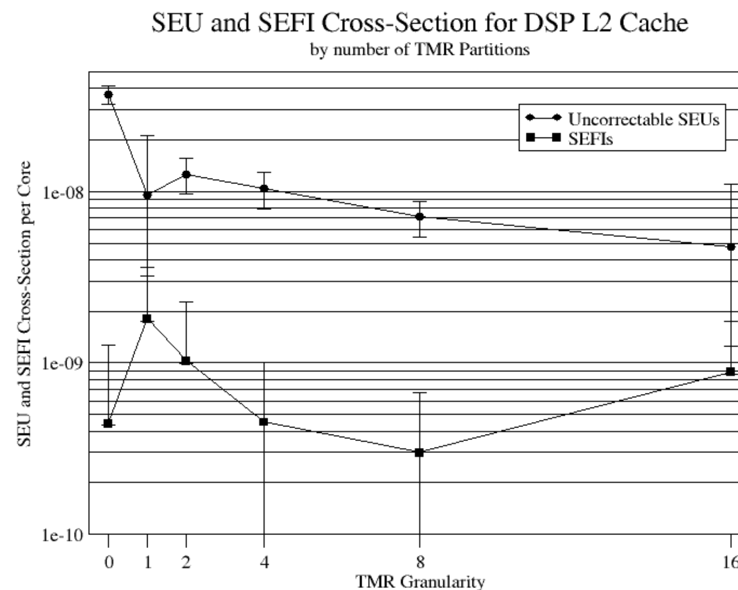
Unmitigated Software Test Results

- By studying the amount of time data remains resident in the L2 cache, we can understand the difference in the reliability of long-term and short-term resident data variables
- Some data will be read many times and some data will be read only once
- These results show that there is nearly 15 times decrease in noticeable errors from data that is read frequently to data that is read once
- This result indicates that selective TMR approaches will be more useful for data that is written once and read many times, such as global constants.



TMR Granularity and Data Dependence Experiments

- As the granularity increased and the amount of data in each calculation decreased, that the calculations were nearly eight times more reliable than the unmitigated error rate.
- This result shows us that in calculations that use a lot of data might need more mitigation than calculations that use less data.
- Two of the tests did not have any SEFIs (4 and 8)

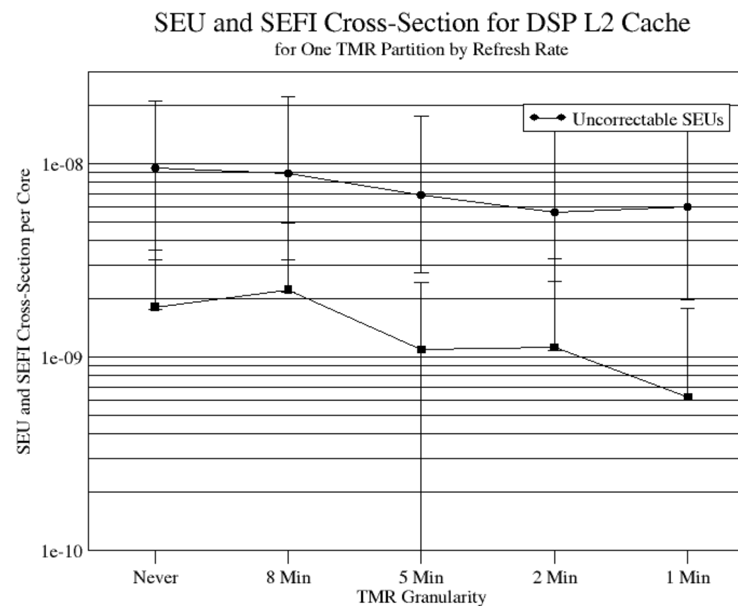


Mitigated Software Test Results

- **While the SEU bit cross-sections are quite small, the SEFI cross-sections are 400 times smaller**
  - For many calculations dual module redundancy (DMR) would not be strong enough
  - Triple-modular redundancy (TMR) would provide masking, which can be useful for higher error rates
    - DMR fails at 2x the rate of the unmitigated code and must be reset after each error
    - TMR fails at 3x the rate of the unmitigated code and can mask at least 1 error
- **The TMR granularity is important**
  - The more data that is used, the more likely the calculation fails
  - Fine-grained granularity can tolerate more errors
- **The software structure is important**
  - The reliability of recursive codes will be dependent on the iteration – the more iterations, the more likely a failure could be accumulated

Time and Time Granularity Tests

- We used the largest granularity of TMR from the previous test and varied the number of times the data is the refreshed
- Results were consistent with our earlier results – 50% decrease in SDC based on the number of times the data is read
- System crashes decreased for many tests



Mitigated CRC Code

- We explored a number of different mitigation methodologies
- The weak point is that there is a data value that is returned as a value and used as an input value in the next iteration
  - Attempts at “voting down” to one return value did not work
  - Most effective methodology involved returning triplicated values and inputting triplicated values
    - Allowed each module of TMR to compute results independently
    - Each TMR module could fail independently
- Code that loops or recurses with dependencies possibly has a lower reliability, due to the reliability of the dependencies
- Code that loops or recurses has “persistent state” as shown in FPGA circuits with feedback

Analysis

- **Reliability of calculations is proportional to the amount of data used in the calculation**
- **Reliability of data is proportional to size and length of time resident in cache**
- **TMR is not necessary for all calculations, but can be selectively applied to the largest data variables or the most data-dependent calculations**
- **Multiple-bit upsets were widely spaced (64 addresses apart) – another possible failure mode for small data structures**
- **Still trying to understand the decrease in the crashes**
  - How much of previous crashes is caused by SEU corruption of the program – could the SEFIs be tied into program crashes?

Conclusions

- Radiation can cause errors to be introduced into airborne systems
- SEUs can cause a number of operational issues in memories, FPGAs, and processors
  - Recent trends have shown that direct ionization upsets from protons is possible
  - DRAM errors can include both singular SEUs or SEFIs that cause a burst of SEUs
  - MBUs are also increasing as feature size decreases
- Mitigation for SEUs in FPGAs and processors is possible

LA-UR 11-03201