https://ntrs.nasa.gov/search.jsp?R=20140005409 2018-02-25T21:34:46+00:00Z

# Single Event Testing on Complex Devices: Test Like You Fly versus Test-Specific Design Structures

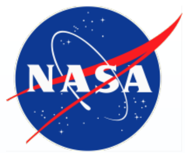
Melanie Berg, AS&D Inc. in support of NASA/GSFC

Melanie.D.Berg@NASA.gov

Kenneth Label: NASA/GSFC

*Deliverable to NASA Electronic Parts and Packaging (NEPP) Program to be published on nepp.nasa.gov originally presented by Melanie Berg at the Government Microcircuits Applications & Critical Technologies Conference (GOMAC) Tech, Charleston, SC, March 31 to April 4, 2014.*

# Acronyms




- Block random access memory (BRAM)
- Combinatorial logic (CL)
- Device under test (DUT)
- Edge-triggered flip-flops (DFFs)
- Field programmable gate array (FPGA)
- Input – output (I/O)
- Linear energy transfer (LET)
- Low cost digital tester (LCDT)
- Probability of logic masking ($P_{logic}$)
- Radiation Effects and Analysis Group (REAG)
- Single event transient (SET)
- Single event upset (SEU)
- Single event upset cross-section ($\sigma_{SEU}$)
- Space Environment Information System (SPENVIS)
- Static random access memory (SRAM)

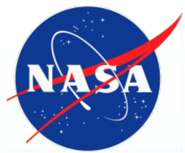
# Abstract

- We present a mechanism for evaluating complex digital systems targeted for harsh radiation environments such as space.
- Focus is limited to analyzing the single event upset (SEU) susceptibility of designs implemented inside Field Programmable Gate Array (FPGA) devices.
- Tradeoffs are provided between application-specific versus test-specific test structures.

# Goals of Single Event Upset Testing (1) Error Rates

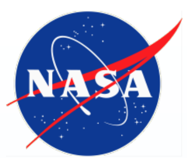
- Calculating SEU-error-rates based off of SEU cross sections ($\sigma_{SEU}$s).
  - $\sigma_{SEU}$s are calculated by counting the number of DUT malfunctions given the number of particles the DUT is exposed to per LET.
  - SEU-error-rates are calculated by curve fitting $\sigma_{SEU}$s and inputting this information into tools such as SPENVIS.

# Goals of Single Event Upset Testing (2) Error Responses




- In complex designs, error responses will vary:
  - Depends on what circuitry is hit:
    - Clock or reset global routes.
    - Dormant versus highly-active circuits.
  - Depends on which state the design is operating in when the SEU occurs:
    - Various states can invoke different error responses.
  - Depends at what portion of the clock-cycle the SEU occurs:
    - Some upsets may not get caught because of when in the clock-cycle the upset occurs.
    - If the upset temporarily disturbs the data input of a DFF close to a clock edge, metastabililty can occur.

# Difference between Test Structure and Application-Specific Design



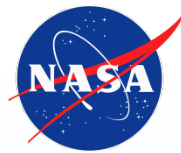
- Test structure is a design implemented in a DUT that is created specifically for SEU testing
- Application-specific design is circuitry implemented in a DUT that is either the final design targeted for space or a subset of the final design.

# Test Structures versus Final Designs

- **Although error rates and error responses are design dependent, useful information can be extrapolated from test structures versus the final design.**
- **Why use test structures versus final designs?**
  - **By the time the final design is complete, it is usually too late to perform radiation testing on it.**
  - **Can be too difficult to apply input stimuli to a final design.**
  - **Can be too difficult to monitor DUT responses.**

**The following slides give more insight into the benefits of using test structures versus full designs during radiation testing**

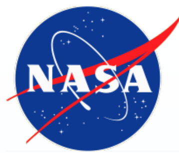
# Best Practice for Radiation Testing: Logic Replication for Statistics




| Best-Practice for DUT Test Structure Development | How Application-Specific Test Structures Violate Best-Practice Considerations |
|---|---|
| Test structures should contain a large number of replicated logic in order to increase statistics: e.g., shift-registers with thousands of stages. | • Statistics are poor because usually there is not a significant amount of replication.<br>• In addition, trends for specific elements are not able to be clearly identified/established. |

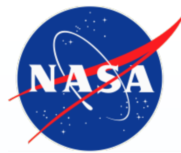
**Example of replicating circuits for statistical purposes: DUT containing hundreds of counters versus 1 counter**

# Best Practice for Radiation Testing: State Space Traversal




| Best-Practice for DUT Test Structure Development | How Application-Specific Test Structures Violate Best-Practice Considerations |
|---|---|
| A test structure's state space should be traversable such that it can be covered within one radiation test run. | The state space of a complex design cannot be traversed within one radiation test run. |
| | Hence, a significant amount of circuitry and system states are not tested. |
| | The result is SEU data that are uncharacteristic of the design. |

# Best Practice for Radiation Testing: Logic Masking

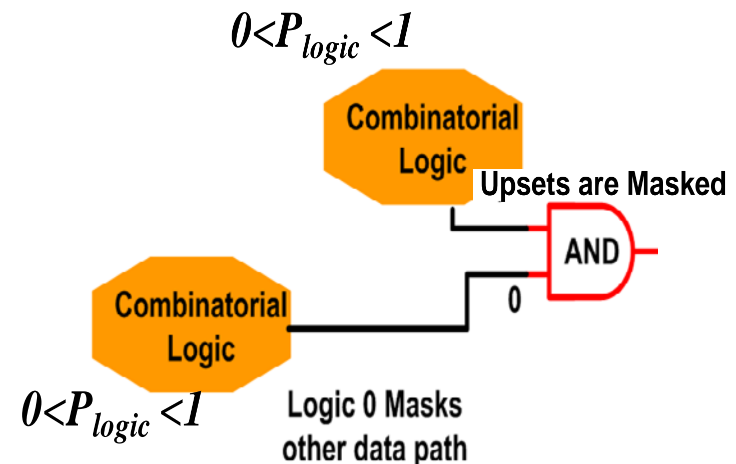



| Best-Practice for DUT Test Structure Development | How Application-Specific Test Structures Violate Best-Practice Considerations |
|---|---|
| Logic masking should be minimized or controllable. | Application-specific test structures contain a significantly higher number of masked data paths than test structures. |

***$P_{logic}$ is the probability that an upset will be masked from being captured by the system.***

***$P_{logic} = 0$ : path is 100% masked***

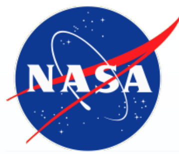
***$P_{logic} = 1$ : path has no masking***



# Best Practice for Radiation Testing: Avoiding Unrealistic SEU Accumulation

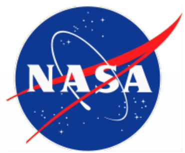
| Best-Practice characteristics of a DUT design | How Application-Specific Test Structures Violate Best-Practice Considerations |
|---|---|
| Avoid unrealistic SEU accumulation from accelerated testing: | Application-specific test structures take up most of the DUT's area. There are a lot of co-dependencies between logic.<br><br>Hence, it is difficult to control SEU accumulation in an accelerated test environment. |
| • Scrubbing (correcting) SEUs. Mostly performed on internal memories structures. | |
| • Flush through test structures; e.g., shift-registers. | |
| • Small number of gates per sub-test structure; e.g., testing hundreds of counters. | |

# Best Practice for Radiation Testing: Increasing Visibility



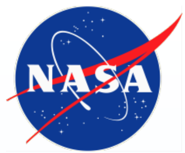

| Best-Practice characteristics of a DUT design | How Application-Specific Test Structures Violate Best-Practice Considerations |
|---|---|
| All (or a significant percentage of) potential upsets should be observable during testing. | A significant number of upsets in a complex design are generally not observable during radiation testing. |
| Test structures can easily be designed to enhance observable nodes; e.g., shift-registers and counters. | This is true mostly because of logic masking, limitations in state space traversal, limitations in I/O count, or time of upset propagation to observable nodes. |

# Testing Application Specific Designs



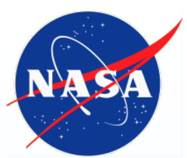

- The benefit of testing application specific designs is the ability to observe error responses specific to the application.
- However, the user must be aware of the following:
  - Unrealistic SEU accumulation in an accelerated environment.
  - Limited visibility due to masking and fractional state space traversal.
  - Poor statistics due to the variance in design circuits.
- $\sigma_{SEU}$s will most likely have a large variance if circuits are not able to be isolated and controlled.

# Case Study



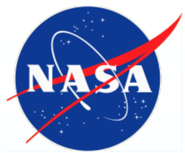

- **DUT is a Xilinx V5QV – radiation hardened FPGA**
- **Application Specific Test Structure is an embedded microprocessor (Micro-blaze™).**
- **Goal is to determine error rates for using an embedded Micro-blaze™ processor in the Xilinx V5QV with and without cache.**
  - **Question: Does using cache in embedded memory increase the $\sigma_{SEU}$s such that the Micro-blaze™ will not meet project requirements?**

# Suggestions on How to Test the Application Specific Design



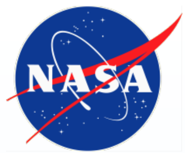

- Because the goal is to study caching SEU effects, use a test design that contains cache and one that does not.
- Test basic structures such as shift-registers and counters to get an underlying understanding of device SEU characteristics.
- Basic test-structure analysis characterizes:
  - Sequential memory elements (DFFs)
  - Combinatorial logic (CL)
  - Global routes
- Increase visibility of the Micro-blaze™ during testing

# Increasing Visibility



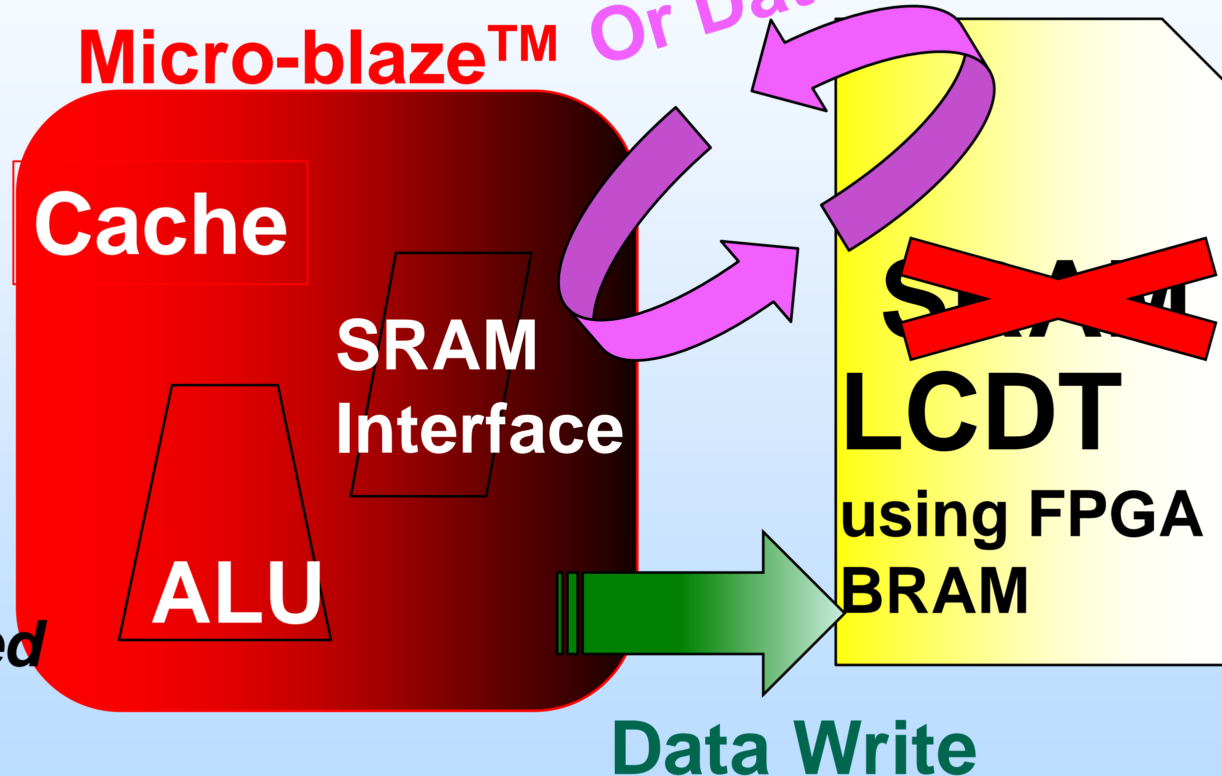

- When testing application-specific designs, there are areas where visibility will be limited and cannot be increased.
  - This is why we also test basic test structures, such as shift registers and counters.
  - From test data, we extrapolate $\sigma_{SEU}$ information to fit the application specific design.
  - Data extrapolation has been performed for this case study, but is beyond the scope of this presentation.
- Because visibility is limited, it is important to have the ability to differentiate between upsets. Performed by:
  - Supporting test structures,
  - Supporting test equipment,
  - Data post processing, or
  - Understanding the internal structures of the DUT

# Processor and SRAM Communication



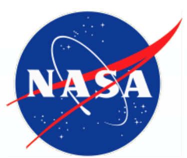
SRAM: Static random access memory
BRAM: Block random access memory

- **Processors talk to memory**

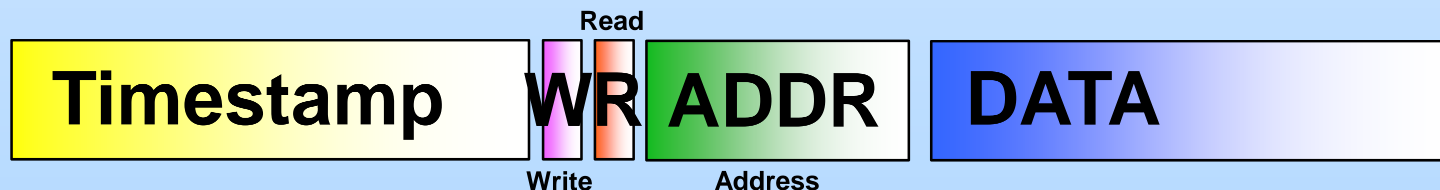



- ***Most processor radiation tests detect errors by erroneous SRAM memory writes***
- ***Visibility is significantly limited***
- ***We increase visibility by replacing external SRAM with the REAG low-cost digital Tester (LCDT)***

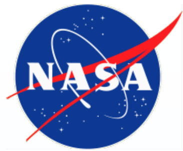
# More on Increasing Visibility with Microprocessor Testing (1)




- As previously stated, the embedded SRAM in the tester (BRAM) takes the place of normal memory accesses.
- In addition, each memory access is time stamped and logged in an alternate bank of BRAM. Only the last 512 accesses are kept.
- After each test run, the time-stamped logs are output to the user.



# More on Increasing Visibility with Microprocessor Testing (2)




DUT: device under test

**DUT**

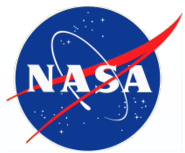
- Halted
- Error
- Trace Instruction
- Trace Valid Instruction
- Trace Exception Taken
- Trace Exception Kind
- Trace Register Write
- Trace Register Address
- Trace Data cache Request
- Trace Data cache Hit
- Trace Data cache Ready
- Trace Data cache Read
- Trace Instruction cache Request
- Trace Instruction cache Hit

**TESTER**

**Watchdogs**

**Send watchdog errors to host PC**

# Summary of Case Study Test Enhancements

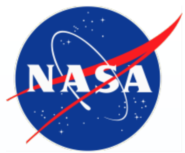



- Visibility was increased by isolating memory accesses as follows:
  - Moving the instruction and data storage to the LCDT for traffic observation.
  - Performing tests with and without cache to determine the influence cache has on upsets.
- Differentiating global upsets from the normal data set:
  - Helped to understand which upsets are prominent.
  - Gave insight on how the use of cache will affect $\sigma_{SEU}$s.
- Monitoring internal Micro-blaze$^{TM}$ signals
  - $\sigma_{SEU}$s are not reliant on detecting erroneous memory read and writes anymore.  Data are too limited and uninformative with sole reliance on memory reads and writes.
  - Can now determine when a processor crashes and how.

# Comparing Micro-blazeTM $\sigma_{SEU}$s and Global Clock $\sigma_{SEU}$s

**SEU Cross Sections:**
**Cache vs. No Cache with Global Routes**

# Summary




- We presented a framework for evaluating complex digital systems targeted for harsh radiation environments, such as space.
- Tradeoffs are provided between application-specific versus test-specific test structures.
- If performing accelerated radiation testing on an application specific design:
  - Understand limitations in data,
  - Be prepared for complex data de-convolution,
  - Pay attention to global structures, and
  - Use basic-test structures to obtain an underlying understanding of DUT SEU behavior.