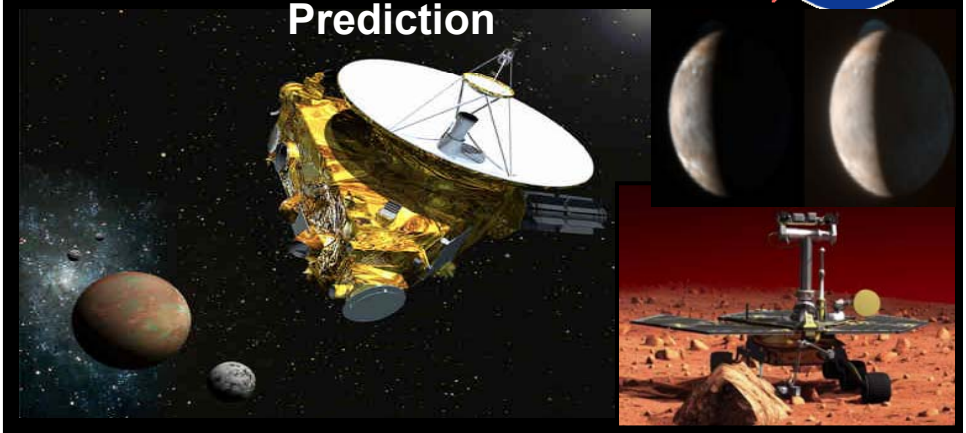
An Overview of the NASA Goddard Methodology for FPGA Radiation Testing and Soft Error Rate Prediction



Melanie Berg, MEI Technologies in support of NASA/GSFC,
Kenneth A. LaBel, and Jonathan A. Pellish, NASA/GSFC

Presented by Melanie D. Berg at the Microelectronics Reliability and Qualification Workshop (MRQW), December 11-13, 2012, Los Angeles, CA and published on nepp.nasa.gov.

Acknowledgements

- *Some of this work has been sponsored by the NASA Electronic Parts and Packaging (NEPP) Program and the Defense Threat Reduction Agency (DTRA)*
- *Thanks is given to the NASA Goddard Radiation Effects and Analysis Group (REAG) for their technical assistance and support. REAG is led by Kenneth LaBel and Jonathan Pellish*

Contact Information:

Melanie Berg: NASA Goddard REAG FPGA Principal Investigator:

Melanie.D.Berg@NASA.GOV

Presented by Melanie D. Berg at the Microelectronics Reliability and Qualification Workshop (MRQW), December 11-13, 2012, Los Angeles, CA and published on nepp.nasa.gov.

Presented by Melanie D. Berg at the Microelectronics Reliability and Qualification Workshop (MRQW), December 11-13, 2012, Los Angeles, CA and published on nepp.nasa.gov.

Background: Single Event Effects (SEEs) and Common Terminology

- Single Event Latch Up (SEL): Device latches in high current state
- Single Event Burnout (SEB): Device draws high current and burns out
- Single Event Gate Rupture: (SEGR): Gate destroyed typically in power MOSFETs
- Single Event Transient (SET): current spike due to ionization. Dissipates through bulk
- Single Event Upset (SEU): transient is caught by a memory element. Causes an incorrect state. SETs are categorized under SEUs
- Single Event Functional Interrupt (SEFI) - upset disrupts function

Background: SEUs and Field Programmable Gate Arrays (FPGAs)

- Ionizing particles cause upsets (SEUs) in FPGAs
- Each FPGA type has different SEU error signatures:
  - Temporary glitch (transient)
  - Change of state (in correct state machine transitions)
  - Global upsets: Loss of clock or unexpected reset
  - Configuration corruption (not all FPGAs)
- When creating a design targeted for a specific FPGA it is important to take into account:
  - Soft Error Rates (SERs)
  - Types of upsets (error signatures)
- Based on SEU characterization, the designers will decide:
  - What type of mitigation is necessary (or none at all)
  - Can the FPGA and its intended design implementation meet project requirements

Overview:

- This presentation focuses on how to characterize SEU responses of FPGA designs using accelerated test an analysis
- The FPGA is the device under test (DUT)
- Topics covered:
  - What to look for regarding the basic elements of an FPGA prior to testing
  - FPGA configuration test and analysis
  - Considerations regarding design test structure selection

SEU testing of FPGA designs requires comprehending the basic elements within FPGA devices and the basic building blocks of digital designs

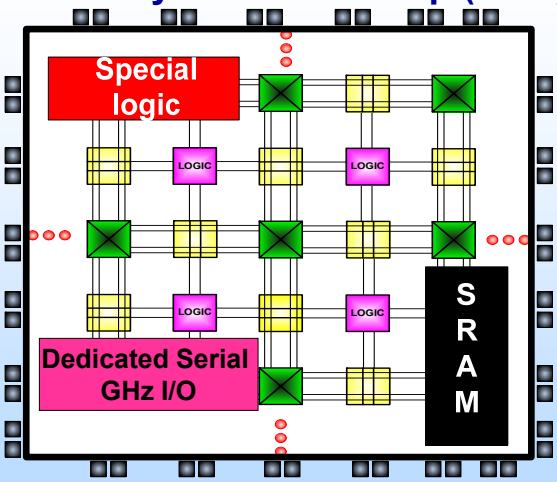
Configuration

Functional Digital Logic

Global Routes

Hidden (specialized) device circuitry

Field Programmable Gate Array (FPGA) FABRIC – System On A Chip (SOC)

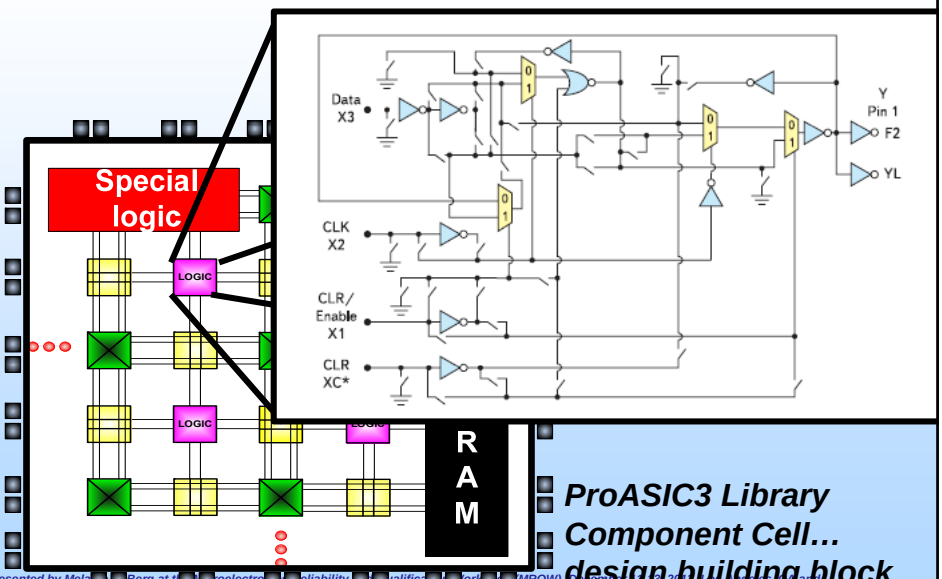


User creates a design by configuring pre-existing logic blocks and routes.

Presented by Melanie D. Berg at the Microelectronics Reliability and Qualification Workshop (MRQW), December 11-13, 2012, Los Angeles, CA and published on nepp.nasa.gov.

7

A Closer Look at an FPGA Logic Cell: Microsemi ProASIC3

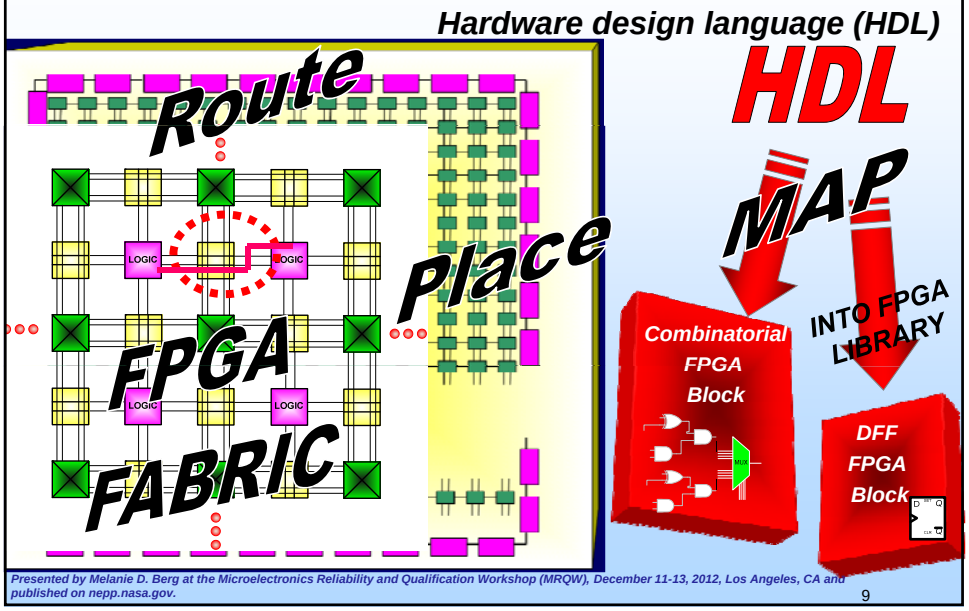


ProASIC3 Library Component Cell… design building block

Presented by Melanie D. Berg at the Microelectronics Reliability and Qualification Workshop (MRQW), December 11-13, 2012, Los Angeles, CA and published on nepp.nasa.gov.

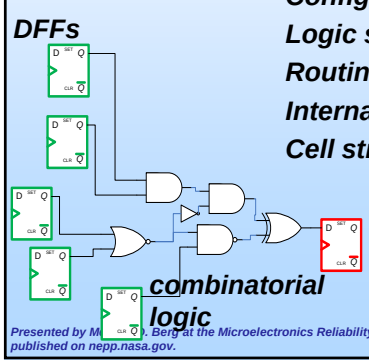
8

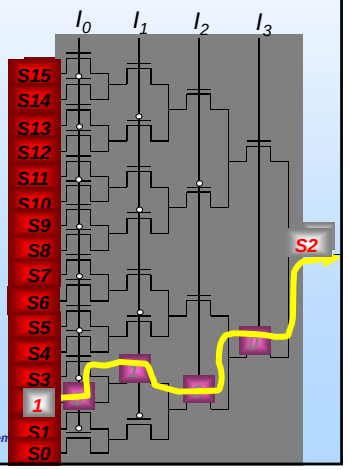
FPGA Building Blocks: How Gates and Routes Are Utilized in FPGA Fabrics



Evaluate the DUT-FPGA Fabric Prior to Testing:

- This evaluation involves understanding the FPGA's elements and how designs are mapped into its elements.
- From this information, specific radiation tests and test structures can be developed to target the DUT's various components.
- Data sheet information:
  - *Configuration type*
  - *Logic switching speed*
  - *Routing*
  - *Internal mitigation*
  - *Cell structures*





The Categorization of FPGA Fabric Structures and Their Potential SEUs as defined by NASA Goddard REAG:

Cross section: σ_{SEU} ***= #upsets/(#particles /cm²)***

$$P(fs)_{error} \propto P_{Configuration} + P(fs)_{functionalLogic} + P_{SEFI}$$

Design σ_{SEU} — ***Configuration*** σ_{SEU} — ***Functional logic*** σ_{SEU} — ***SEFI*** σ_{SEU}

Sequential and Combinatorial logic (CL) in data path

Global Routes and Hidden Logic

SEU Testing is required in order to characterize the σ_{SEU} ***for each of FPGA categories***

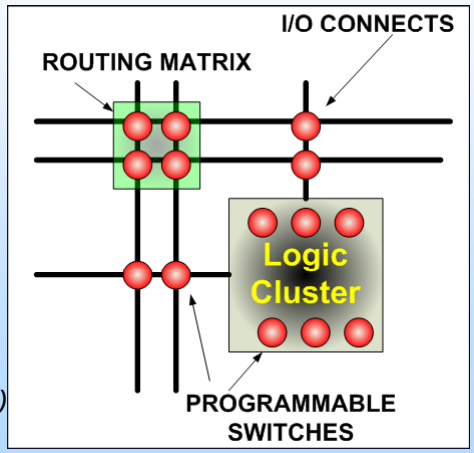
Presented by Melanie D. Berg at the Microelectronics Reliability and Qualification Workshop (MRQW), December 11-13, 2012, Los Angeles, CA and published on nepp.nasa.gov.

11

FPGA Configuration

HDL ***FPGA MAPPING*** **Configuration**

- ***Configuration Defines:*** *Arrangement of pre-existing logic via programmable switches*
  - *Functionality (logic cluster)*
  - *Connectivity (routes)*
- ***Programming Switch Types:***
  - ***Antifuse:*** *One time Programmable (OTP)*
  - ***SRAM:*** *Reprogrammable (RP)*
  - ***Flash:*** *Reprogrammable (RP)*



Presented by Melanie D. Berg at the Microelectronics Reliability and Qualification Workshop (MRQW), December 11-13, 2012, Los Angeles, CA and published on nepp.nasa.gov.

SEU Configuration Testing

- Configuration is a static element
- Once the design is completed, the configuration does not change.
- Antifuse testing:
  - Based on error signatures. Is there a permanent fault?
- Flash Testing
  - Configure the device, irradiate, then read-back configuration.
  - Unfortunately, current flash devices will not allow a full read-back but supply configuration validation software. Pass-fail.
- SRAM testing:
  - Configure the device, irradiate, then read-back configuration.
  - User has full visibility of configuration after read-back

Configuration SEU Test Results and the REAG FPGA SEU Model

$$P(fs)_{error} \propto P_{Configurat\ \ ion} + P(fs)_{functional\ \ Logic} + P_{SEFI}$$

Configuration	REAG Model
Antifuse	$P(fs)_{error} \propto P(fs)_{functionalLogic} + P_{SEFI}$
SRAM **(non-mitigated)**	$P(fs)_{error} \propto P_{Configuration}$
Flash	$P(fs)_{error} \propto P(fs)_{functionalLogic} + P_{SEFI}$
Hardened SRAM	$P(fs)_{error} \propto \cancel{P_{Configuration}} + P(fs)_{functionalLogic} + P_{SEFI}$

Evaluating Functional Logic Data Path Soft Error Rates

- The goal is to approximate the SER of a design
- Unfortunately, the goal does not make much sense regarding a complex system.
  - Some portions of a system are significantly more susceptible than others. SER will depend on what portion of the system is operating and what the environment is when it is operating.
  - Example: Due to data flow and speed of operation, an RS232 interface will have a different susceptibility than an Arithmetic Logic Unit (ALU)
  - Accelerated testing can disguise the differences
- It's best to partition the system and determine susceptibility per partition.

Test Structures that allow for Error Differentiation and Visibility

- Visibility: Which component caused the system error?
- Component error differentiation is essential:
  - Allows for understanding what portions of the system are under-mitigated
  - Assists in determining rates for various portions of the system using accelerated test data
- As test structure complexity grows upset signatures become convoluted

What Design is Best for Single Event Testing?

Simple Architecture

No/minimal functional Masking

Easy to base-line across FPGAs

increases state space coverage

Data may not map into real design

Differentiation of errors in simpler

Complex Architecture

Functional Masking

Reduction in state space coverage

Data may be a better fit for real designs

Error differentiation becomes more complex

Actual flight Architecture

Usually not available at test time

Can be very expensive to test

Will not cover a significant amount of state space while testing

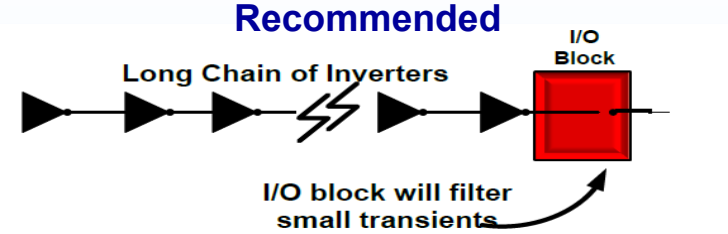
SEFIs will usually require a system reset for each error event

Rate calculations are misleading

Caution When Selecting FPGA Test Structure Designs for Accelerated Testing

- **An element's error response can be different when the element is isolated versus when the element is connected in a system**
- **If not taken into account, this can lead to inaccurate design level SEU susceptibility characterization**
- **Example:**
  - **Studying the response of a combinatorial logic gate may not reflect the susceptibility of the gate when placed with other circuits in a design**
  - **System topology will most likely reduce the error cross section**
    - **Capacitive loading**
    - **SET propagation**
      - **Routing**
      - **Logic masking**

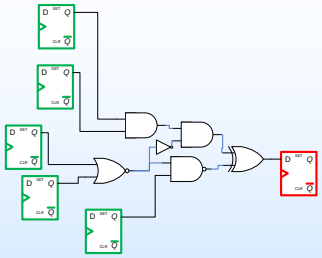
Implementing Traditional Test Structures In FPGA Devices. Long Inverter Chains Are Not Recommended



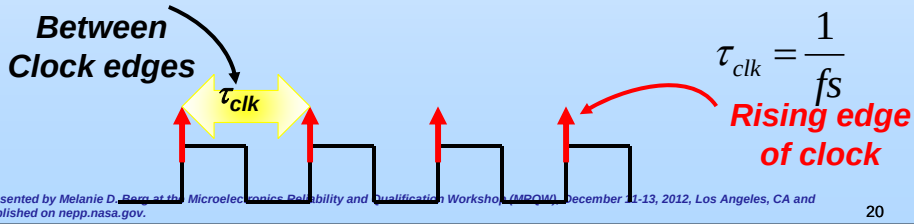
- Goal is to calculate susceptibility of combinatorial logic gates. Issues:
  - Assumes cascaded combinatorial logic has linear SEU effects – However this is not true (capacitive effects such as attenuation)
  - Does not take into account inverters are not inverters in FPGA devices. The formation of an inverter requires additional circuitry
  - Does not take into account complex routing due to the length of the chain
  - Inverter chains in FPGA devices have a significant amount of noise. Most FPGAs are made to implement synchronous designs

Most Designs Follow Synchronous Design Methodology... So Should Test structures

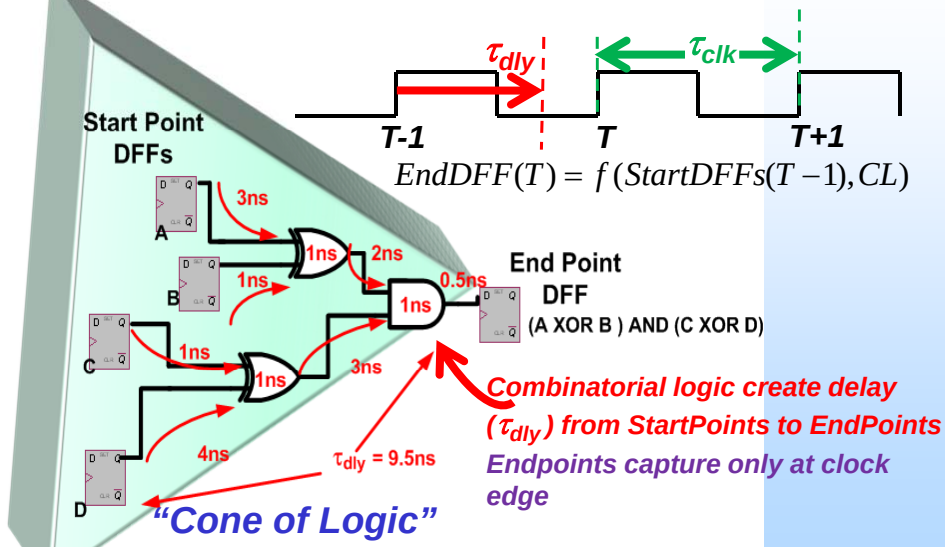
- Designs are comprised of:
  - Combinatorial Logic (CL)
  - Edge Triggered Flip-Flops (DFFs)
  - Clocks and resets
- All DFFs are connected to a clock
- Clock period: τ_{clk}; Clock frequency: f_s



- *Combinatorial Logic: Compute between clock edges*
- *DFFs: Hold (or sample) at the rising edge of a clock*



Synchronous System Data Paths: StartPoint DFFs → EndPoint DFFs



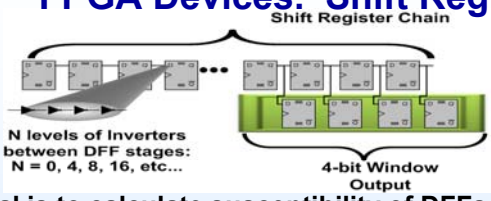
$$EndDFF(T) = f(StartDFFs(T-1), CL)$$

Combinatorial logic create delay (τ_{dly}) from StartPoints to EndPoints

Endpoints capture only at clock edge

Every DFF has a cone of logic

Implementing Traditional Test Structures In FPGA Devices. Shift Register Chains



- **Goal is to calculate susceptibility of DFFs and CL gates.**
- **Benefits**
  - **Simple test structure that has no logic masking**
  - **Use of combinatorial logic and DFFs helps to study trends. What happens as the amount of combinatorial logic is increased? Does frequency matter?**
  - **Can easily traverse the entire state space several times over during accelerated testing**
- **Disadvantages - Does not represent a complex design well:**
  - **No logic masking**
  - **Linear path of combinatorial logic**
  - **Capacitive loading is minimized**

Increasing Test Structure Complexity: Shift Registers versus ?

- There are benefits to increasing design complexity. However, limitations must be taken into account:
  - State space traversal during testing
  - Amount of logic masking
  - Visibility of upsets
- NASA REAG uses counters and digital signal processing units (e.g. multipliers and accumulators) as test structures
- Interface (I/O) management can be difficult for high-speed circuits or designs with a large number of I/O.
  - Built-in-Self-Test (BIST) can be a solution
  - Caution: BIST circuits have limited visibility –
    - Error differentiation can become extremely difficult
    - Determining if the test is operating correctly can become difficult

Conclusion

- This presentation covered a small portion of SEU characterization for FPGA designs:
  - Configuration testing and
  - test structure selection
- Appropriate test structure selection is key to accurate SEU characterization
  - Simple test structure error responses may not represent complex designs. Hence mapping of radiation data may not be accurate
  - Complex test structures will limit visibility of errors and state space traversal
  - It is best to study a variety of test structures and analyze radiation data trends:
  - Amount of combinatorial logic
  - Frequency
  - Input data pattern