Radiation Effects and Mitigation Strategies for modern FPGAs

10th annual workshop for LHC and Future experiments

Los Alamos National Laboratory, USA
Introduction

- FPGA benefits in instrumentation design
  - High density logic
  - User configurable
    - SRAM and antifuse technologies popular
- Reliability issues in radiation environments
  - Latchup
  - Single event upsets (SEUs)
  - Multiple bit upsets (MBUs)
Introduction

• Fault mitigation strategies
  – Scrubbing SRAM devices (Xilinx specific)
    • Periodic readback and verification
    • Some limits on readback
      – RAM contention
      – Half latch constant generation
  – Fault tolerant design techniques
    • Triple module redundancy (TMR)
      – Entire design vs. persistent logic
      – Effectiveness in the face of MBUs difficult to quantify
FPGA Architecture (Xilinx Vertex)

- **SRAM based devices**
  - RAM bits control configuration
    - Logic definition
    - Signal routing

- **Xilinx Vertex family**
  - Configurable logic blocks (CLB)
    - Split into two slices
      - Look-up tables (LUT)s define logic
      - Flip flops and carry generation
  - Routing matrix
    - Pass transistor and buffered connections between CLBs
    - Generous supply of global and local interconnect
FPGA Architecture (Xilinx Vertex)

- **Vertex family (continued)**
  - Block RAM
    - 4K bit blocks
    - Configurable in various widths
  - I/O blocks (IOB)
    - Many I/O standards supported
    - I/O registers
FPGA Architecture (Xilinx Vertex)

- RAM utilization
  - Configuration dominates
  - Sparsely utilized
    - Rarely more than 30%
    - Even in designs where logic is fully utilized
  - Still dominates by an order of magnitude

Virtex XCV1000 memory Utilization

<table>
<thead>
<tr>
<th>Memory Type</th>
<th># of bits</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configuration</td>
<td>5,810,048</td>
<td>97.4</td>
</tr>
<tr>
<td>Block RAM</td>
<td>131,072</td>
<td>2.2</td>
</tr>
<tr>
<td>CLB flip-flops</td>
<td>26,112</td>
<td>0.4</td>
</tr>
</tbody>
</table>
FPGA Architecture (Xilinx Vertex)

- Half-latch or weak keepers
  - Provide constants
  - Save logic resources
  - Used throughout device
  - Subject to SEU upset
    - Can reset over time
  - Not observable
    - Not defined by configuration bits
  - Reinitialized as part of device initialization
    - Full reconfiguration required
Failure Modes

• Latchup
  – Parasitic bipolar transistors
    • Created as a by product of CMOS fab techniques
    • When activated, short power to ground
      – Can burn out the device
  – Epitaxial processing eliminates parasitics
    • Eliminates latchup completely
  – Lower Vcc decreases vulnerability
    • Bipolar transistors barely forward biased
      – Xilinx V2 (1.5 Vcc) is latchup immune to 160MeV
Failure Modes

• Single event upsets (SEUs)
  – Logic Content
    • Usually manifested as a “glitch”
    • Can be persistent in a feedback element
      – Counter or ALU
  – Logic Configuration
    • Altered logic definition
    • Always persistent
      – Usually results in undesirable operation
  – Routing
    • Statistically most probable
    • Always persistent
      – Least likely to result in logic failure
Failure Modes

• Single event functional interrupts
  – Power on reset or other global function
    • Usually results in immediate functional interrupt
      – Device needs to be reconfigured
  – JTAG or other configuration interface
    • Can inhibit or corrupt readback operations
      – Device reset required to restore test functionality

• Multiple bit upsets (MBUs)
  – Multiple configuration bits altered
    • Can defeat fault tolerant design (TMR)
Mitigation Techniques

• Scrubbing
  – Readback and verification of configuration
    • Sets limits on duration of upsets
  – Partial configuration
    • Supported by Vertex family
    • Allows fine grained reconfiguration
    • Does not reset entire device
      – Allows user logic to continue to function
  – Complete reconfiguration
    • Required after SEFI
    • No user functionality for the duration of reconfiguration
Triple Module Redundancy

- Simple triple module redundancy
  - Three copies of user logic
  - Two of three voting on output
    - Counter example
  - Simple TMR handles faults
    - Cannot resynchronize on the fly
    - Requires logic reset after repair
    - OK for stateless logic
Triple Module Redundancy

- Feedback TMR
  - Three copies of user logic
  - State feedback from voter
    - Counter example
  - Handles faults
  - Resynchronizes
    - Operational through repair
  - Speed penalty due to feedback
  - Desirable for state based logic
Triple Module Redundancy

• Feedback TMR can be SEU immune
  – Must TMR clocks as well
  – Scrubbing frequency provides upset rate tolerance
  – For low SEU rates, fault probability becomes SEFI rate
  – Xilinx has automated TMR tool in beta test

• Unfortunately, MBUs also occur
  – Can defeat TMR
  – Current TMR tools do not floorplan
  – Occur .1% on vertex, up to 2% on vertexII
  – Implications still under investigation
Triple Module Redundancy

- TMR costs
  - Triple logic utilization
    - At least 3x logic utilization
    - Need to floorplan for MBU resistance
      - Also for operation during repair
    - No fully automated tool at present
  - Triple power consumption
    - SRAM devices already inefficient
  - Slower operation
    - Feedback TMR inherently slower
    - Worse when floorplanning requirements taken into account
Other TMR Techniques

• Selective TMR
  – Identify persistent, or state based logic
  – TMR only these sections
    • Other critical sections may also be TMRed
      – Application dependent
  – Subject of ongoing development and test
    • 90% of full TMR performance (preliminary result)
    • Much lower device utilization, power, etc
    • Automated tool in development
Other Pitfalls (virtex)

- Half-Latches
  - Unobservable failure mode
  - Requires device reinitialization to reset
  - Design tools insert automatically
    - No switch to stop software from inserting them
  - Los Alamos has developed removal tool
    - Works on completed design
      - Can fail when design is heavily utilized
      - Too memory inefficient for largest virtexII devices
Other Pitfalls (virtex)

- Block RAM has shared output register
  - Readback can collide with user logic
    - RAM cannot be verified by scrubbing
    - User logic must handle RAM verification
- Distributed RAM has shared output as well
  - Similar collision problem
- Clock delay lock loop module
  - Status bits inaccurate during upset related failures
Alternatives

• Antifuse
  – Configuration based on physical shorts
    • Invulnerable to upset
    • Cannot be altered
  – Over 90% smaller upset cross section for comparable geometry
  – Signal routing more efficient
    • Much lower power dissipation for similar device geometry
  – Lags SRAM in fabrication technology
    • Usually one generation behind
    • Latch up more of a problem than in SRAM devices
Alternatives

• Rad-hard Antifuse
  – All flip-flops TMRed in silicon
    • Unmatched reliability
    • High cost
    • Unimpressive performance
      – Feedback TMR built in
      – Usually larger geometry
      – Not available in highest densities offered by antifuse
  – Some devices even have TMRed RAM
    • Not ECC, but self correcting feedback TMR
When to Use Antifuse

• Where requirements are well known
  – Also stable over time
• Logic density does not exceed what is available
  – About 2M gates currently
• Where power consumption is critical
  – Also low noise
  • Many mixed mode designs and analog/digital front ends
When to use SRAM

- In system reprogrammability required
  - Unstable requirements
  - Desire for generic hardware
- Cost of TMR and scrubbing tolerated
  - Schedule does not allow for proper system engineering
  - NRE for TMRed hardware small compared to total system NRE
    - Fluid hardware/software functional tradeoff
Conclusion

- FPGAs can be used in elevated Radiation
  - Errors can be detected and corrected
  - Fault tolerant design can be utilized
    - TMR can produce designs virtually immune to upset
- SRAM devices are the only choice for in system reprogrammability
- Antifuse is naturally more radiation tolerant
  - A natural choice if reprogrammability not required