Microelectronics Radiation Mitigation

R. Jansen – TEC-EDM

19/01/2023

Outline

Analysis

- **Mission Analysis**
- Component Evaluation

Mitigation

- Reliability
- Classification
- Implementation
- Verification/Validation

Conclusion

Mission Analysis – Introduction

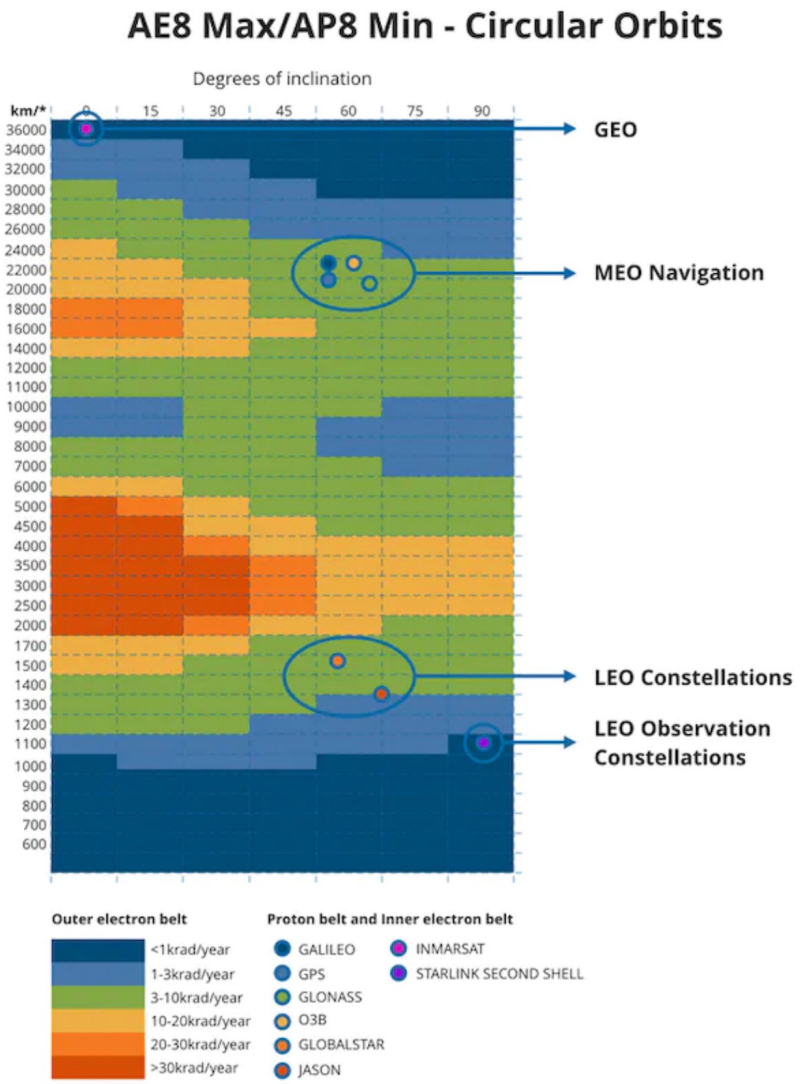
- Inputs for the selection of the appropriate microelectronics mitigation
  - Mission duration
  - Mission Environment
  - Functionality and performance requirement
  - Reliability and Availability

- System, Unit and Board level analysis yields additional requirements
  - Component selection (preliminary)
  - Verification and validation

- Mitigation at electronics level is a trade-off against
  - System performance (Speed, Latency, Availability, …)
  - Power consumption
  - Area utilisation
  - Engineering time and Cost

Mission Analysis – Mission & System Inputs

- Input for the selection of the appropriate component
  - Mission Classification – Product Assurance/Quality
    - Component class
  - Maximum SEE LET level
    - GEO – 60 MeVcm2/mg or
    - LEO – 36 MeVcm2/mg
  - Maximum TID dose (Flux/Time)
    - LEO (1 year) - 5krad
    - MEO (5 year) – 25krad
    - GEO (15 year) – 100krad
  - Availability



Mission Analysis - Components – FPGA (COTS+RHBD)

- AMD/Xilinx
  - 65nm – Virtex-5QV
  - 20nm – RT Kintex
  - 7nm – XQR Versal
- Lattice
  - Nexus CrossLink-NX
- Microchip
  - 130nm - ProASIC/RTAX2000/RTAX4000
  - 65nm - RTG4
  - 28nm – RT Polarfire
- Nanoxplore
  - 65nm – NG-Medium
  - 28nm – NG-Ultra
  - 28nm - Ultra300

Outline

Analysis

- Mission Analysis
- **Component Evaluation**

Mitigation

- Reliability
- Classification
- Implementation
- Verification/Validation

Conclusion

Components – COTS SRAM FPGA – SEU σ – Proton

- SEU proton (64MeV) cross-section (10^{-15} cm^2) reported in the literature
- Devices tested
  - 28nm - Artix 7
  - 20nm – Kintex
  - 16nm – Zynq Ultrascale+
  - 7nm – Versal ACAP
- Recorded cross-sections have been rounded
- Measurement from different test campaigns included
  - CRAM – Configuration RAM
  - BRAM – Block RAM
  - URAM – Ultra RAM
  - FF – Logical register

Technology	28nm	20nm	16nm	7nm
CRAM	5 to 8	1 to 2.5	0.12 to 3.5	3e-2
BRAM	1 to 7	2.5 to 4.5	0.6 to 1	1
URAM				0.3
FF	5	2	0.3	

- Noticeable is the reduction in cross-section with technology

Components – COTS SRAM FPGA – SEU σ – Neutron

- SEU neutrons (10MeV) cross-section (10^{-15} cm^2) reported in the literature
- Devices tested
  - 28nm - Artix 7
  - 20nm – Kintex
  - 16nm – Zynq Ultrascale+
  - 7nm – Versal ACAP
- Recorded cross-sections have been rounded
- Measurement from different test campaigns included
  - CRAM – Configuration RAM
  - BRAM – Block RAM
  - URAM – Ultra RAM
  - FF – Logical register
- Noticeable is the reduction in cross-section with technology
- Single Event Rate (SER) for CRAM provided for LEO orbit

Technology	28nm	20nm	16nm	7nm
CRAM	7	2.5	0.25 to 0.35	2.2e-2
BRAM	7		1 to 3	1.2
URAM				0.3
FF				

Configuration Memory Rates

	per bit, per day	Improve-ment*	Node
Virtex-II	3.99E-07	1	130 nm
Virtex-4	2.63E-07	1.517	90 nm
Kintex-7	1.41E-08	28.298	28 nm
UltraScale	7.56E-09	52.778	20 nm
UltraScale+	1.33E-09	300.000	16 nm

Components – COTS SRAM FPGA – SEU σ – HI

- SEU heavy ion saturation cross-section (10^{-9} cm^2) reported in the literature
- Devices tested
  - 28nm - Artix 7
  - 20nm – Kintex
  - 16nm – Zynq Ultrascale+
  - 7nm – Versal ACAP

Technology	28nm	20nm	16nm	14nm	7nm
CRAM	2	1 to 8			
BRAM	1.2				
FF					

- Recorded cross-sections have been rounded
- Measurement from different test campaigns included
  - CRAM – Configuration RAM
  - BRAM – Block RAM
  - FF – Logical register
- Noticeable is the reduction in cross-section with technology

Components – COTS Flash FPGA – SEU σ – HI

- SEU heavy ion saturation cross-section (10^{-9} cm^2) reported in the literature
- Devices tested
  - 28nm-M – RT Polarfire
  - 28nm-L – Nexus CrossLink-NX
- Recorded cross-sections have been rounded
- Measurement from different test campaigns included
  - BRAM – Block RAM
  - FF – Logical register

Technology	28nm - M	28nm - L
BRAM	1	1e-2
FF	1	0.2

- Please note that for COTS FPGAs also all the peripheral and processing blocks should also be radiation tolerant (i.e. no SEL and SEFI preferably)

Components – RHBD FPGA – SEU σ – HI

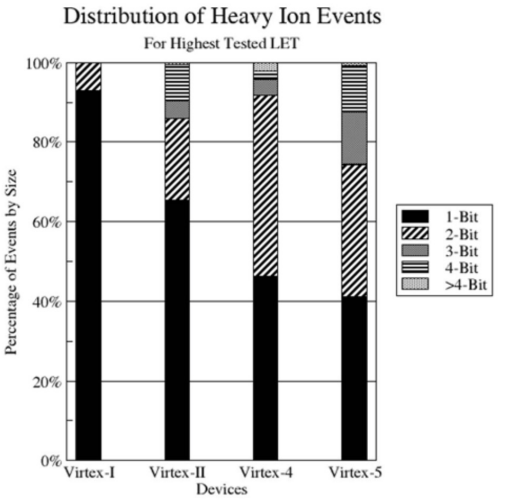
- SEU heavy ion saturation cross-section (10^{-9} cm^2) reported in the literature
- Devices tested
  - 150nm-M – RTAX2000
  - 65nm-M – RTG4
  - 65nm-X - Virtex-5QV
  - 65nm-NX – NG-Medium
  - 28nm-NX – NG-Ultra
- Recorded cross-sections have been rounded
- Measurement from different test campaigns included
  - CRAM – Configuration RAM
  - BRAM – Block RAM
  - FF – Logical register

Technology	150nm-M	65nm-M	65nm-X	65nm-NX	28nm-NX
CRAM		1 to 8	30	5	
BRAM		100	120	60	
FF	20	6.5	28 no Fil 3 with Fil	4	

Components – RHBD FPGA – SEU σ – MBU

- The number of transistors affected by a radiation event increases with
  - advancing technology node
  - Increasing SEE LET
- The increasing number of affected transistors causes a multiple bit upset (MBU) and/or multiple cell upset (MCU)
- These MBU/MCU make the recovery from an upset increasingly more difficult.
- Without careful attention of MBU and MCU the effectiveness of TMR would be limited by common cause failures (CCF)
- With careful analysis of the MBUs and memory cell placement in advanced technology nodes, the occurrence of MBUs at least for proton, neutron SEE can be contained

- Shown are the MBU generated in the 28nm Xilinx Zynq (F. Benevenuti et al.)



Type of memory	Type of SEU	Examples	α Particles	Heavy ions	Neutrons		
					14 MeV 0°	14 MeV 180°	(Epi)Thermal
BRAM	SBU 1-1-1		100.0%	82.0%	93.4%	97.1%	95.4%
	MBU 2-1-2		—	16.2%	4.7%	2.9%	—
	MBU 1-2-2		—	—	—	—	4.5%
	Others		—	1.8%	1.9%	—	0.1%
CRAM	SBU 1-1-1		97.6%	38.1%	76.7%	79.9%	78.1%
	MBU 2-2-2		2.4%	41.9%	16.9%	15.5%	0.0%
	MBU 2-1-2		0.0%	4.4%	3.5%	2.1%	0.0%
	MBU 1-2-2		—	—	0.3%	1.5%	17.8%
	MBU 2-2-3		—	3.0%	1.3%	0.5%	0.0%
	MBU 2-2-4		—	0.2%	—	0.5%	—
	MBU 2-3-4		—	8.3%	0.6%	—	0.0%
	MBU 2-3-5		—	0.6%	0.3%	—	0.0%
	Others		—	3.4%	0.3%	—	4.1%

Outline

Analysis

- Mission Analysis
- Component Evaluation

Mitigation

- **Reliability**
- Classification
- Implementation
- Verification/Validation

Conclusion

Reliability, MTTF and Availability - Introduction

- The COTS FPGAs are SEU susceptible and the availability/reliability requires to be an analysed
- The simplest model considers an operating state, a repairing state and a failed stated in case an additional failure occurs

- The reliability R(t) as a function of time for a constant failure rate λ is

$$r(t) = e^{-\lambda t}$$

- The mean time to failure (MTTF) is

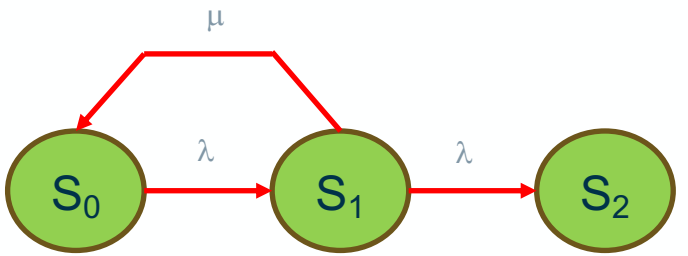
$$MTTF = \int_0^\infty r(t)dt = \frac{1}{\lambda}$$

- Given a constant repair rate μ

$$MTTF = \frac{\mu}{\lambda^2}$$

- Given the fixed mean time to repair (MTTR) the availability is

$$Availability = \frac{uptime}{uptime+downtime} = = \frac{MTTF}{MTFF+MTTR}$$



Reliability and MTTF – Memory Scrubbing

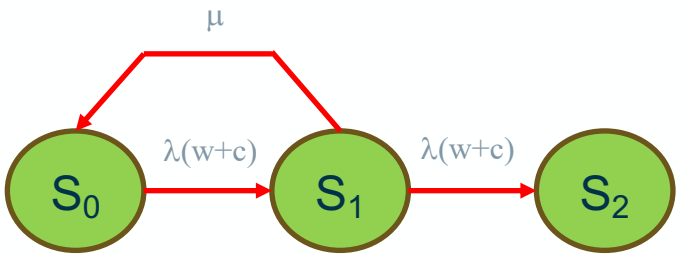
- Memory is scrubbed (blind) with a SEC-DED error protection scrubbing scheme with period is T
- Memory bit failure rate is λ for M words of width w'=w+c bits where w is the number of bits in the word and c the number of correction bits
- The reliability after each scrubbing cycle per bit is R(t)

$r_s(t) = R(T)^n r_0(t)$

R(T) = $r_0(T)$ and n = floor(t/N)

with $r_0(t) = e^{-\lambda w' t}$ - w'(1- $e^{-\lambda t}$) $e^{-\lambda(w'-1)t}$, with t<T

We have R(T) = 1- $\frac{M(\lambda w')^2}{2}$



- The mean time to failure (MTTF) is

$$MTTF_S = \int_0^\infty r_s(t)dt = \frac{2}{MT\lambda^2 w'^2}$$

Reliability and MTTF – TMR

- TMR is implemented with 3 registers and one voter
- The reliability R(t) as a function of time for a constant failure rate λ is

$$r_{TMR}(t) = 3e^{-2\lambda t} - 2e^{-3\lambda t}$$

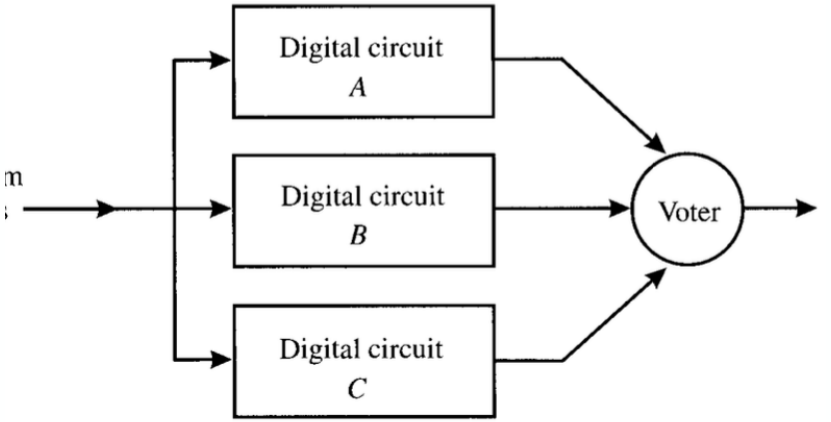
- The mean time to failure (MTTF) is

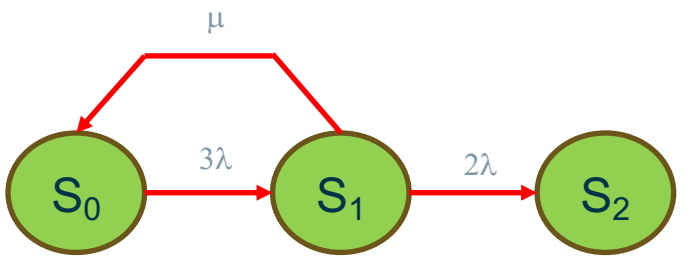
$$MTTF_{TMR} = \int_0^\infty r_{TMR}(t) = \frac{5}{6\lambda}$$

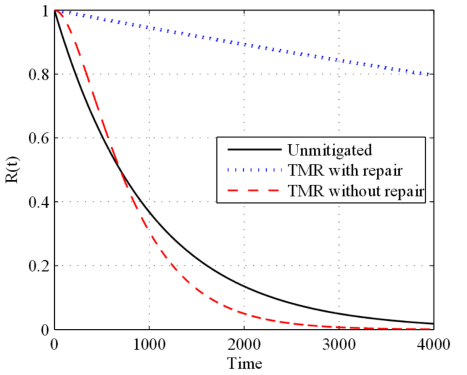
- Please note that this is less than for no TMR. However the reliability increases significantly when repair at rate μ is included
- The mean time to failure for this TMR configuration can be calculated to be

$$MTTF_{TMR+R} = \int_0^\infty r_{TMR+R}(t) = \frac{5}{6\lambda} + \frac{\mu}{6\lambda^2}$$

- The MTTF is significantly improved







Reliability and MTTF – TMR with scrubbing

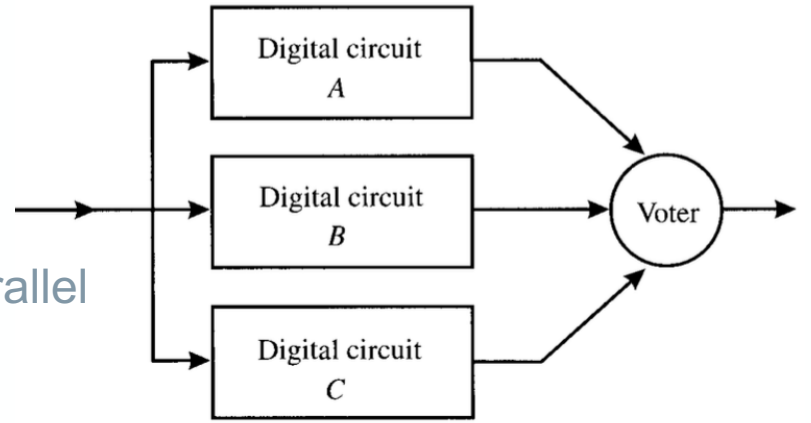
- TMR is implemented with 3 registers and one voter with repair
- The configuration memory for the TMR is scrubbed (blind)
- The reliability R(t) as a function of time for a constant failure rate λ is

$$r_{TMR+R+S}(t) = r_{TMR+R}(t) * r_S(t)$$

- The mean time to failure (MTTF) can be calculated by considering two parallel independent processes.

$$MTTF_{TMR+R+S} = \int_0^\infty r_{TMR+R+S}(t)dt$$

$$= \left(\frac{1}{MTTF_{TMR+R}} + \frac{1}{MTTF_S}\right)^{-1}$$



- Reliability calculations show also that the reliability and MTTF increases with increasing number of TMR stages

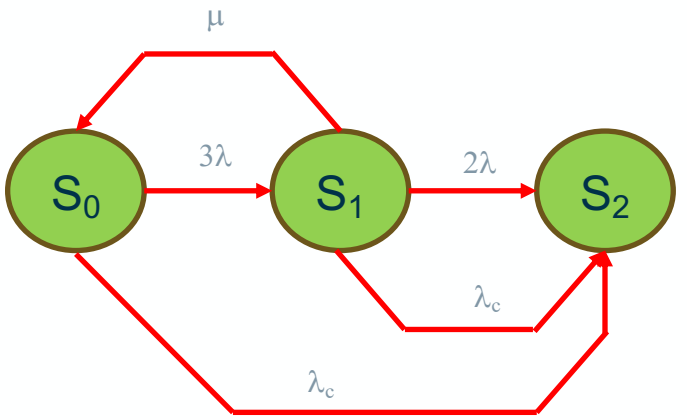
Reliability and MTTF – TMR + Common Cause Failure

- The application of TMR has the potential to improve the reliability of the system significantly based on the assumption that there is no common cause failure (CCF).
- In presence of the common cause failure the gains from the implementation of TMR are limited
- Causes for CCF are single point failure (SPF), multiple bit upsets (MBU), common mode failures (CMF)
- Given the falure rate λ, repair rate μ and CMF rate λ_C the mean time to failure (MTTF) can be calculated to be (after MJ Cannon et al.)

$$\text{MTTF} = \int_0^\infty R(t) = \frac{2\lambda+\lambda_C+\mu}{6\lambda^2+5\lambda\lambda_C+\lambda_C^2+\mu\lambda_C}$$

- With increasing repair and decreasing failure rate the TMR are limited by the CCF rate

$$\text{MTTF} = \int_0^\infty R(t) = \frac{1}{\lambda_C}$$



S_0 – Normal operation
S_1 – Impaired operation
S_2 – Failed state
μ – Repair rate
λ – Single module failure rate
λ_C – CCF failure rate

Reliability and MTTF – New ECSS ASIC+FPGA Standard

- The current standard ECSS-Q-ST-60-02C is going to be replaced by
  - ECSS-E-ST-20-40 – Engineering standard
  - ECSS-Q-ST-60-03 – Product Assurance standard
- The ASIC and FPGA will follow the same qualification flow as for space equipment and units
- This implies that dependability and with it availability and reliability analysis are required
- For COTS devices this will be of specific importance

Outline

Analysis

- Mission Analysis
- Component Evaluation

Mitigation

- Reliability
- **Classification**
- Implementation
- Verification/Validation

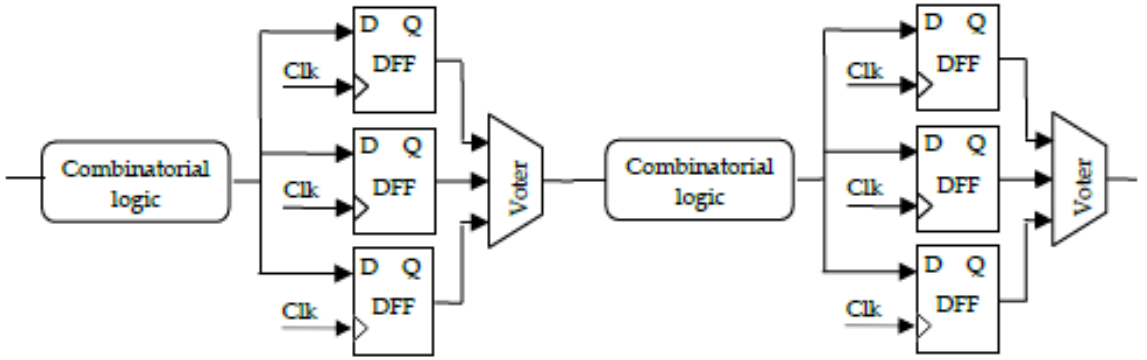
Conclusion

Local TMR

- Type of spatial redundancy where only the sequential elements (D Flip Flops) in the circuit are triplicated, and their outputs compared by a single majority voter.
- It can detect and correct SEUs in registers.
- Smallest area overhead penalty, since only registers are triplicated, not the combinational logic.
- Can be implemented by the designer at HDL or netlist level, with the appropriate synthesis tools.

Challenges

- Local TMR only protects against SEUs directly in the registers (DFFs).
- If an SET propagates through the combinational logic and is captured at a sampling clock edge, the voter will receive 3 identical, but false, values and the error won't be detected.



Note: TMR cannot be used by the designers for SEU protection in configuration memories of SRAM FPGAs. Other techniques are used in those cases (presented later)

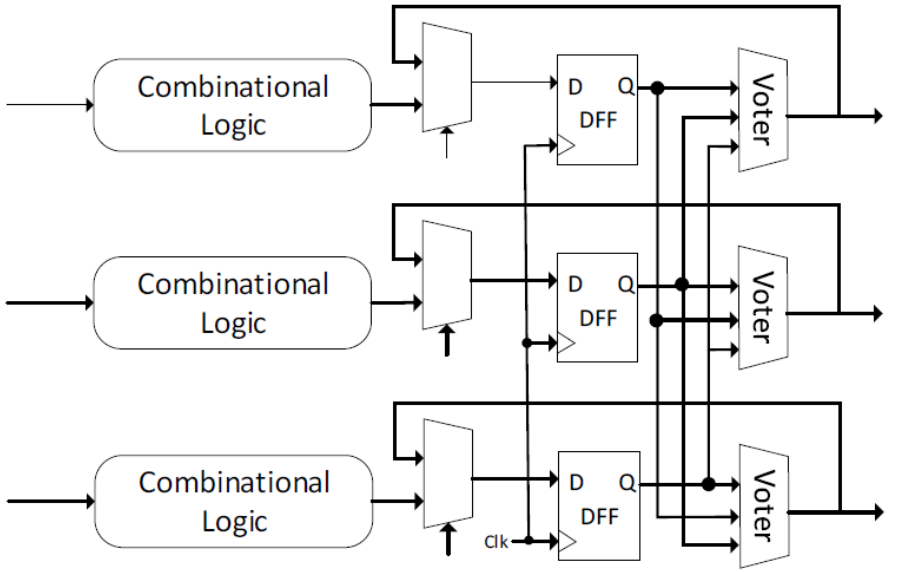
Distributed TMR

- Type of spatial redundancy where the complete computation paths are triplicated, including combinational logic, sequential elements, and voters.
- Single clock and reset lines are used.

- It can detect and correct upsets in registers and combinational logic and can clear errors via feedback to avoid their accumulation.
- Can be implemented by the designer at HDL or netlist level, with the appropriate synthesis tools.

Disadvantages

- Higher area and power consumption overheads, since all registers, CL and voters are triplicated.



Global (or Full) TMR

- Type of spatial redundancy where all circuit elements, including DFF, combinational logic and TMR voters are triplicated. The clock and reset trees are also triplicated.

- Triplicating the clock trees also gives protection against SETs in the clock generation logic (clock tree).

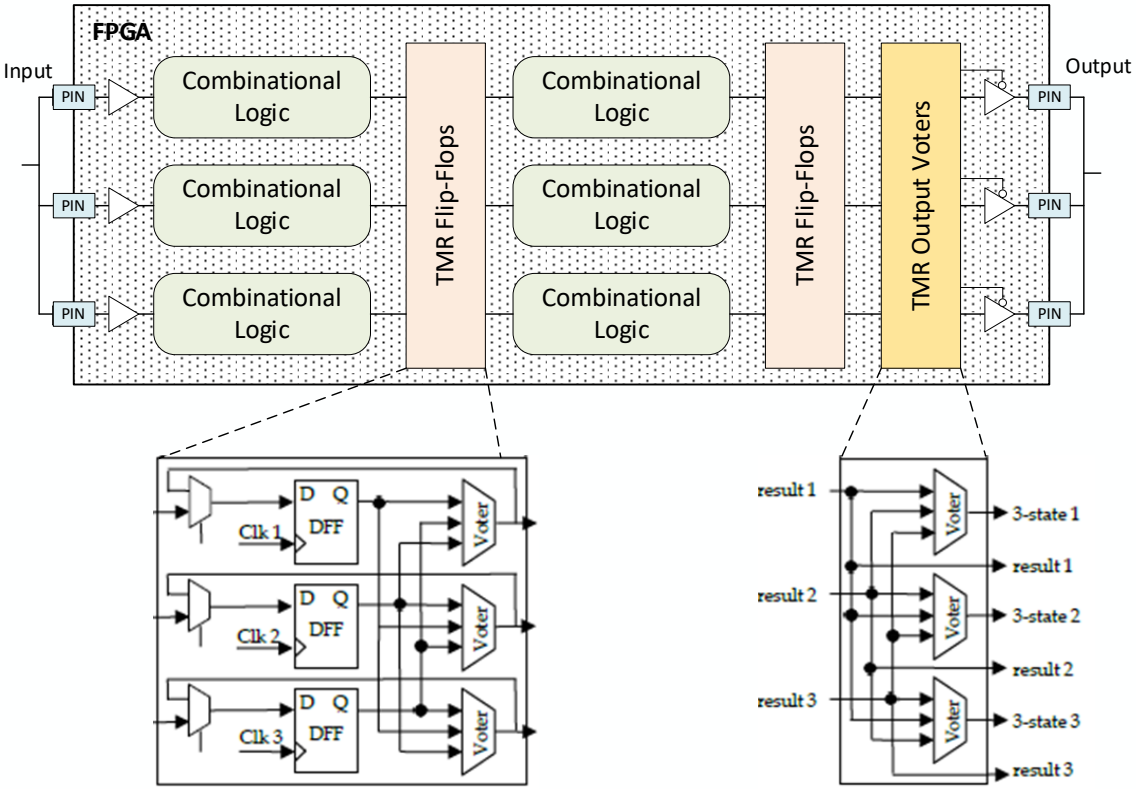
- Global TMR is the strongest TMR method for SEU mitigation (in principle), BUT…

Challenges

- Skew among the triplicated clock trees introduces further design challenges and may reduce mitigation strength.

- The additional circuit area required by the Full TMR scheme may even result in an actual increase on the error cross section of the circuit.

- The designer should confirm that the design tools properly support this TMR option and can manage the timing challenges, before using it.



Embedded user memory TMR

- Type of spatial redundancy where the embedded user memory blocks (BlockRAMs) are triplicated, and their outputs are voted.
- Voted results can be written back to the memories to correct the errors

- Data refresh via feedback only needed for longer time storage. May not be needed for regularly updated data.

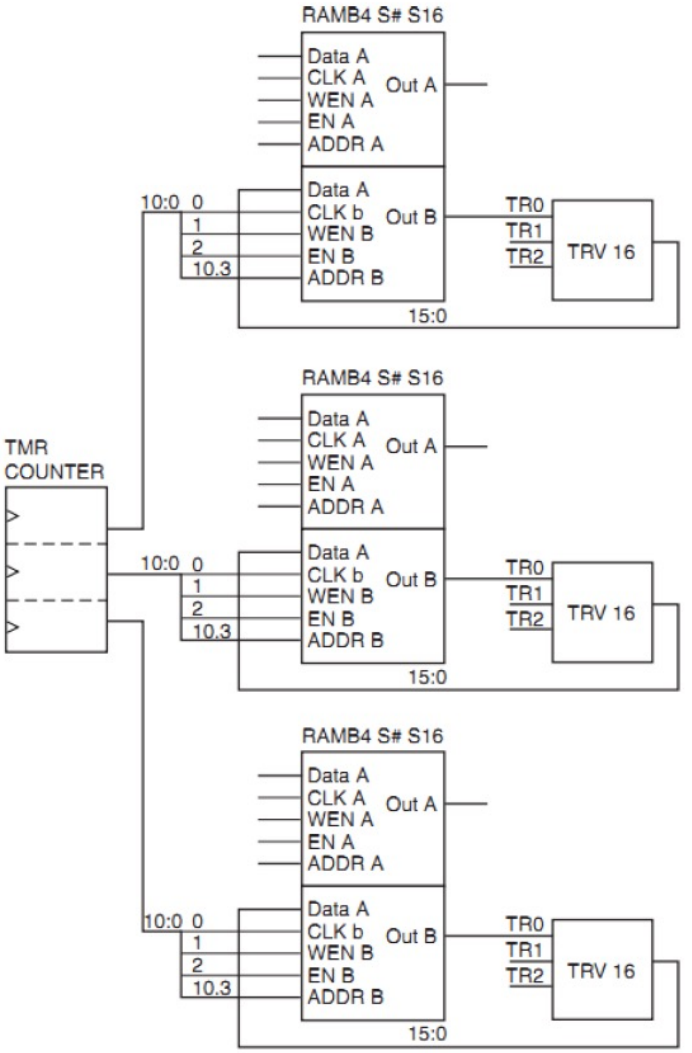
- Data refresh can also be done automatically with a counter, periodically going through the addresses and writing back the voted results.

Disadvantages

- Higher resource utilization overheads, since the memory blocks are triplicated, plus voters and counter logic.

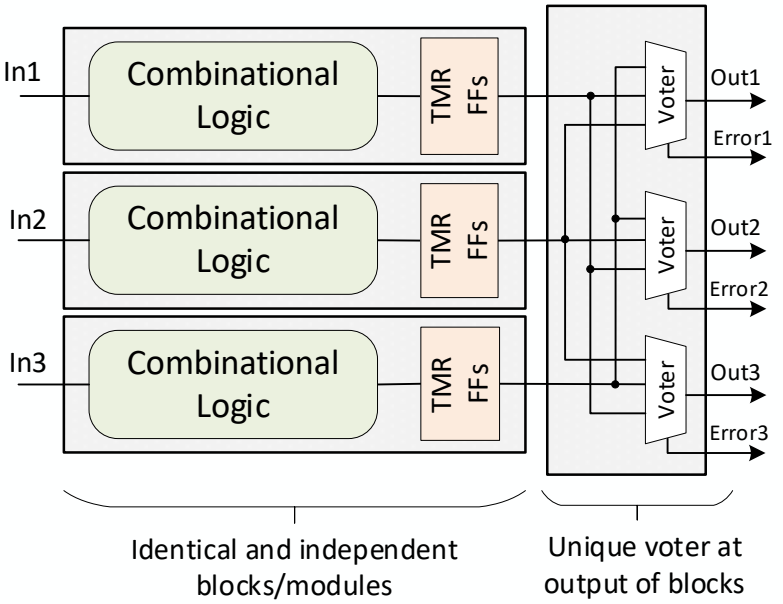
- Dual- port memories are needed for this scheme. But effectively they can only be used as single port memories by the user, due to the feedback used for the data refresh.

- Memory EDAC may be a more efficient solution, in terms of resources (discussed later)



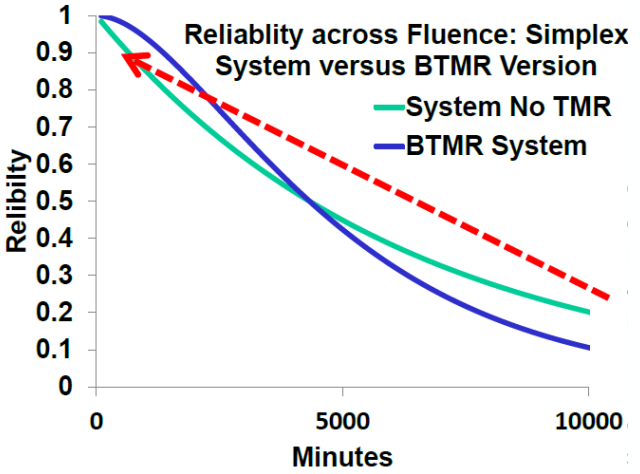
Block (Module) level TMR

- Type of spatial redundancy where complete circuit blocks or modules are triplicated and voted, rather than individual DFFs.
- Improved resilience to MBUs due to the physical separation of the DFFs in the different blocks, reducing the probability of upsetting the TMR sets.
- It can block errors from propagating to other areas of the system.
- Can use partial reconfiguration for the erroneous block, reducing overall scrubbing time and energy.
- Good solution for regularly reset/flushable systems



Challenges:

- Timing synchronisation (controlled skew) between the different functional blocks
- Re-synchronisation of the erroneous block with the others -> need additional detection signals to know when one of the blocks are in failure.
- Possible accumulation of errors if blocks are not regularly reset (or flushed).
- Reliability of BTMR systems actually drops over time faster than non-TMR systems (!) (*reference: M. Berg, SERESSA, 2019*)
- Regular resets may affect availability.



Graph: © Melanie D. Berg, 2019

Radiation Mitigation Techniques References

- Additional radiation mitigation techniques can be found
  - S. Habinc - Suitability of reprogrammable FPGAs in space applications (2001)
  - R. Weigand - SEE Analysis and Mitigation for SEE Analysis and Mitigation for
  - FPGA and Digital ASIC Devices (2005)
  - D Merodio Codinachs et al. – Overview of FPGA activities in the European Space Agency (2009)
  - F. Siegle et al. – Mitigation of Radiation Effects in SRAM-based FPGAs for Space Applications (2015)
- In addition, an ECSS handbook on ASIC and FPGA mitigation techniques has been published and presented
  - A. Fernandez-Leon - New ECSS Handbook on "Techniques for Radiation Effects Mitigation in ASICs and FPGAs" (2015)
- For COTS components for ESA missions a guideline has been published, which lists mitigation techniques for all relevant SEE (e.g. SEFI, SEL, …)
  - Guidelines for the utilization of COTS components and modules in ESA
- Each of the FPGA supplier have extensive literature and application on the implementation of mitigation techniques for their device

Outline

Analysis

- Mission Analysis
- Component Evaluation

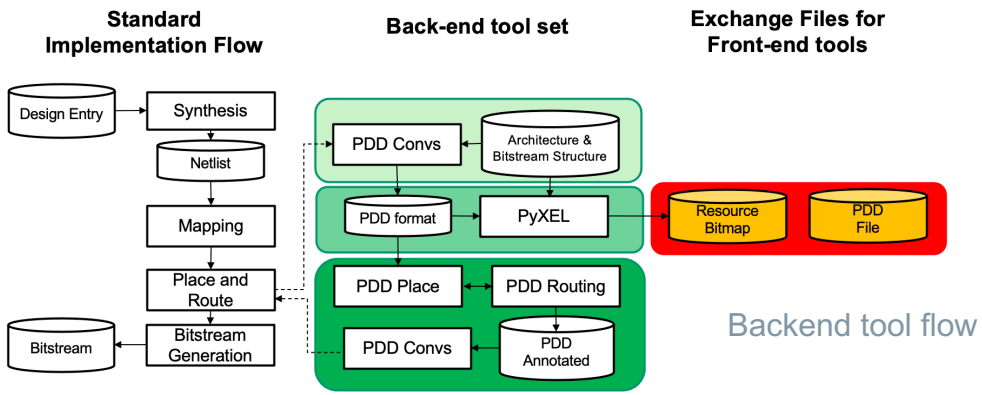
Mitigation

- Reliability
- Classification
- **Implementation**
- Verification/Validation

Conclusion

SEE mitigation implementation

- Commercial tools by Synopsys and Siemens/Mentor support TMR, Safe FSM, Hamming-3 mitigation schemes for different FPGA technologies
- Implementation of temporal redundancy and TMR is supported with different options as wells as by Siemens/Mentor with Precision HiRel, by Synopsys with Synplify and by Xilinx with XTMR.
- Research tools are underdevelopment to increase the reliability of the FPGA design against radiation SEE. E.g. with the Politecnico di Torino:
  - Physical Design Description Place and Router
  - PyXEL – tool to analyse the relationship between the configuration memory and the physical implementation
  - Veri-PLACE – tool for the analysis an mitigation of SEU effects in the FPGA configuration



Backend tool flow overview – L. Sterpone

Outline

Analysis

- Mission Analysis
- Component Evaluation

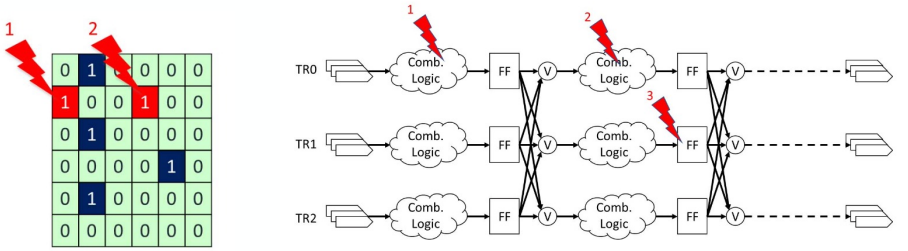
Mitigation

- Reliability
- Classification
- Implementation
- **Verification/Validation**

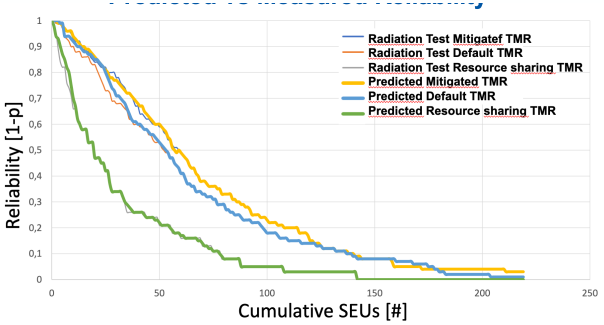
Conclusion

Verification and Validation

- The radiation measurements can be compared against the prediction from fault injectors: FLIPPER2, FT-UNSHADES2, XTRC-V5FI, TURTLE, UFRGS, ..
- The fault predictor should take into account
  - Effect of the configuration memory on the logic fabric
  - Multiple effects from a single upset
- With partitioning of the design onto the FPGA fabric the occurrence of SEMU can be minimised
- The radiation test data can also be correlated with the fault injection results by comparing the CRAM upsets per design upset with the CRAM upsets per scrubbing action.
- Tools are also provided by Siemens/Mentor to
  - Determine with formal verification the resilience against faults
- The Synopsys Z10X supports also fault simulations and coverage



Depiction of SEMU
Single Event Multiple Upset in configuraiton memory



Comparison of SEMU prediction versus radiation measurement from L. Sterpone, et al.

Outline

Analysis

- Mission Analysis
- Component Evaluation

Mitigation

- Reliability
- Classification
- Implementation
- Verification/Validation

Conclusion

Conclusion

- Mission requirements affecting microelectronics design listed
- For potential COTS and RHBD potential FPGA candidates listed
- Provided
    - SEE evaluation results for COTS FPGAs
    - Examples how radiation mitigation increases
        - Reliability
        - Mean time to failure
        - Availability
    - Overview of the different TMR architectures
- Discussed
    - Radiation mitigation implementation details and tools
    - Verification and Validation
- Hopefully provided a starting point for radiation mitigated digital design for space

Conclusion

- Grateful for contributions
  - D. Merodio-Codinachs
  - K. Marinis
  - M. Talis
  - L. Santos
  - A. Urbon