

# The Effects of Race Conditions when Implementing Single-Source Redundant Clock Trees in Triple Modular Redundant Synchronous Architectures

Melanie D. Berg, *Member, IEEE*, Hak S. Kim, Anthony D. Phan, Christina M. Seidlick, Kenneth A. LaBel, *Member, IEEE*, Jonathan A. Pellish, *Member IEEE*, Michael J. Campola, *Member IEEE*

**Abstract**— We present the challenges that arise when using redundant clock domains due to their time-skew. Radiation data show that a singular clock domain provides an improved triple modular redundant (TMR) scheme over redundant clocks.

**Index Terms**—Field Programmable Gate Array (FPGA), Application Specific Integrated Circuit (ASIC), Single Event Upset (SEU), Xilinx, Triple Modular Redundancy (TMR).

## I. INTRODUCTION

In many circumstances, application specific integrated circuits (ASICs) and field programmable gate array (FPGA) devices require some level of mitigation to operate reliably in radiation environments. Triple modular redundancy (TMR) is the process of (1) triplicating circuitry into three redundant domains; and, (2) performing a majority vote on the redundant domains. TMR has proven to be a reliable single event upset (SEU) mitigation strategy [1-6]. Subsequently, it is utilized in a significant number of missions targeted to operate in radiation environments.

The state of a synchronous system is defined to be stored within the system's sequential logic (DFFs). When the state of the system is perturbed, i.e., the system is in an unexpected state, malfunction can occur. In a radiation environment, ionizing particles can disrupt system-states by affecting: data-path combinatorial logic (CL) (from a single event transient (SET)), data-path DFFs (from an SEU), configuration bits (from an SEU only affecting SRAM based FPGAs), or global routing logic.

---

Manuscript received July 6, 2013; revised October 23, 2013. This work was supported in part by the NASA Electronic Parts and Packaging Program (NEPP), NASA Flight Projects, and the Defense Threat Reduction Agency (DTRA) under IACRO 10-49771 and 11-43951.

M. D. Berg is with ASRC Federal Space and Defense, Greenbelt, MD USA. She supports the NASA Goddard Space Flight Center, Greenbelt, MD 20771 USA phone: 301-286-2153; fax: 301-286-4699; e-mail: melanie.d.berg@nasa.gov.

H. S. Kim, A. D. Phan, and C. M. Seidlick are with ASRC Federal Space and Defense, Greenbelt, MD USA. They support the NASA Goddard Space Flight Center, Greenbelt, MD 20771 USA.

K. A. LaBel, J. Pellish, and M. Campola are with the NASA Goddard Space Flight Center, Greenbelt, MD 20771 USA.

In order to reduce circuit area and timing overhead, a designer can select from a variety of TMR schemes. The TMR methodology is defined by which logic is triplicated and by which components have direct connections to voters. Each triplicate is referred to as a TMR domain. The following are descriptions of three commonly used TMR schemes:

- Local TMR (LTMR): Only flip-flops are triplicated. Voters are inserted and placed after the DFFs. Clock domains are not triplicated.
- Distributed TMR (DTMR): The entire data-path of the design is triplicated (CL and DFFs). Clock domains are not triplicated.
- Global TMR (GTMR): The entire design is triplicated. Clock domains are triplicated (redundant).

Theoretically, it would be optimal to implement TMR schemes with three separate clock-sources that create three separate clock domains. Hence, if an SET were to affect one clock network, the other two would still be intact. However, due to relative clock-source drift, using redundant clock-sources to create redundant clock domains has proven to be an ineffective solution [7]. Alternate solutions include: (1) use of a single clock-source that connects to one and only one clock tree (one clock domain per source); or (2) use of a single clock-source that spans to three redundant clock-trees (three clock domains per).

Regarding both TMR clocking solutions (mentioned above), the use of a single clock-source eliminates relative clock-source drift. Regarding the second solution, it is expected that using a single clock-source fanning out to redundant clock trees will strengthen mitigation. However, we show that this expectation is not always true.

With older, slower devices, the use of single clock-source redundant clock trees has been successfully implemented. Alternatively, we show that with technological advancements in data-path routing and CL delays ( $T_{\text{comb}}$ ), the use of single clock-source spanning to redundant clock trees (although assumed to be synchronized) has become an ineffective TMR solution for complex modern designs. The rationale is that race conditions are created due to unmanageable clock skew ( $T_{\text{skew}}$ ) between the three redundant clock trees. In addition,

we show that some of these race conditions exist in small variable pockets and can be undetectable during the system verification process.

We present the challenges that arise when using redundant clock trees due to their clock-skew. Radiation data show that a singular clock tree provides an overall improved TMR methodology versus redundant clock trees. Although this theory holds true for ASICs and FPGAs, for this study, an SRAM-based FPGA (Xilinx Kintex-7 FPGA (XC7K325T)) [8] was used to obtain heavy-ion data.

## II. BACKGROUND

### A. Clock Skew, Data-path Delay, and Synchronous Data Capture

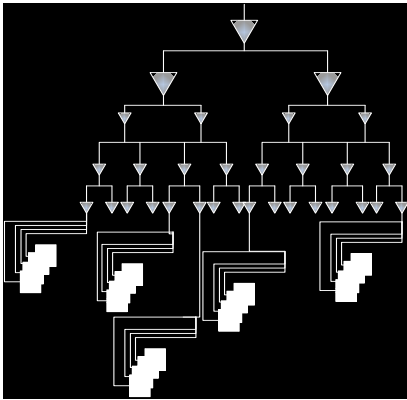


Fig. 1 Clock path clock tree.

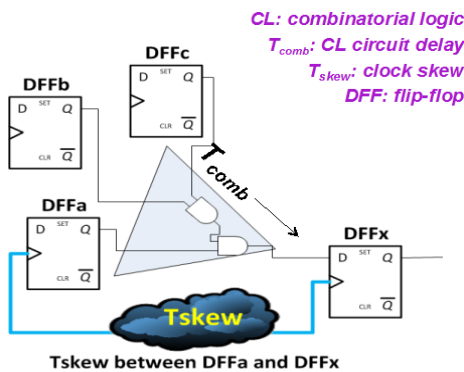


Fig. 2: Data-path with three launch-DFFs (DFFa, DFFb, DFFc) fanning into a capture-DFF (DFFx).

In a synchronous design, there are two logic paths that are managed to achieve reliable data capture: a clock path (clock domain) as illustrated in Fig. 1; and a data-path as illustrated in Fig. 2.

Clock paths start from a clock-source, propagate through a clock tree, and end at a DFF. A clock tree, as depicted in Fig. 1, is a network of routed buffers that have terminal connections to flip-flops (DFFs). Regardless of clock tree source, i.e., single or separate sources, each clock tree in a synchronous design is considered a separate clock domain. In a synchronous design, it is mandatory to balance a clock tree domain so that all connected DFFs receive controlling clock edges at virtually the same moment in time [9-10]. The difference in time of a clock edge's arrival to one DFF with

respect to its arrival to another DFF is defined as clock skew ( $T_{skew}$ ).

In a synchronous data-path: (1) data is launched from a DFF in clock cycle N, (2) data is manipulated by combinatorial logic and routes, and (3) the result of the data manipulation is then captured by a DFF at clock cycle N+1. DFF data capture is controlled by a clock edge (from the balanced clock tree path) and requires data be stable during DFF setup ( $T_{setup}$ ) and DFF hold times ( $T_{hold}$ ). A marginal time  $T_{margin}$  is defined to include additional constraints for the data-path accommodating voltage shifts, temperature effects, and clock jitter ( $T_{jitter}$ ).  $T_{margin}$  is generally set at 5% to 10% of the clock period ( $T_{clk}$ ). Equation (1) describes the required relationship for reliable data capture in a synchronous system with negligible DFF clock-to-Q delay. Equation (1) sets a limit on how slow a data-path can be and is referred to as max-path analysis. As a clarification, the delay of a single-cycle data-path must be faster than one clock cycle.

$$T_{clk} > T_{comb} + T_{setup} + T_{margin} - T_{skew} \quad (1)$$

Alternatively, data-paths cannot be too fast; otherwise, race conditions can violate  $T_{hold}$ . Equation (2) sets min-path analysis constraints.

$$T_{skew} < T_{comb} - T_{hold} - T_{jitter} \quad (2)$$

Both (1) and (2) (max-path and min-path constraints) shall be satisfied for reliable data capture [9-10]. It is important to note that a system containing large  $T_{skew}$  will have trouble meeting min-path constraints. In addition, changing the clock period (slowing down the clock or speeding up the clock) will not affect skew and, hence, will not fix the problem. The absence of  $T_{clk}$  in (2) emphasizes this point.

When using a single clock domain (clock tree),  $T_{skew}$  exists but is minimized by clock tree balancing. Alternatively, when using redundant (yet separate) clock trees,  $T_{skew}$  is significantly increased.

### B. TMR and Clock Skew

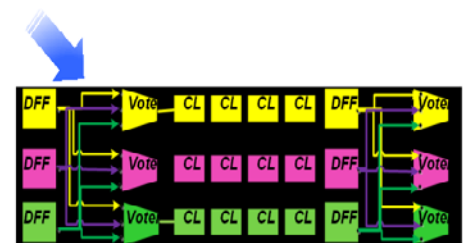
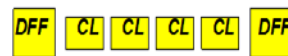


Fig. 3: Triplication of one data-path and formation of its mitigation window.

Fig. 3 illustrates triplication of one data-path in a DTMR and GTMR design. Voter insertion is also shown. We define a data-path from a launch DFF-voter pair to a capture DFF-voter pair as a mitigation window (MW).

Each replication is a TMR domain. For DTMR, each TMR domain shares the same clock tree. In GTMR, each TMR domain has a unique clock tree.

As previously stated, due to clock drift, only a single clock-source should be used in a TMR scheme. Subsequently, connecting a single clock-source to three separate clock trees creates redundant GTMR clock domains. In this case, as long as there are no shared resources (after the single clock-source), SETs that occur in one clock tree should not affect another clock tree.

In reality, the use of multiple clock trees in a redundant system increases skew. In this case  $T_{skew}$  results from the following: routing differences from the clock-source to each clock tree, skew between the clock trees, and  $T_{skew}$  within one clock tree. The designer, to a certain extent, can minimize routing differences and internal clock tree  $T_{skew}$ . When using redundant clock trees in an ASIC, skew between clock trees can be managed in smaller less complex designs; however, skew between clock trees is nearly impossible to manage for modern systems. In an FPGA, skew between clock trees is dependent on the manufacturer and is marginally manageable by the designer.

With GTMR, in the absence of SEUs, there is a possibility of having broken MWs because of  $T_{skew}$ . As an example, one TMR domain may have positive skew and is capturing one state ahead of the other TMR domains. The problem is that with the occurrence of an SEU, broken MWs have weakened mitigation (masking and correction cannot be guaranteed). It is important to note that, within a GTMR design,  $T_{skew}$  will vary between MWs and hence MW reliability will also vary. Consequently, pockets of broken MWs can exist, can change over voltage and temperature, and can be difficult to identify even with the usage of analysis tools. Alternatively with DTMR, if designed correctly, there are no broken MWs.

### C. Timing: Old Devices versus New Devices

In older devices,  $T_{comb}$  is large enough (i.e., logic is slow) to accommodate  $T_{skew}$  and min-path constraints (Eq (2)). With the advances in technology,  $T_{comb}$  is becoming smaller due to faster routes and faster combinatorial logic cells. Hence, min-path  $T_{skew}$  constraints are becoming stricter with new technology. For this reason, although redundant clock domains were successfully implemented in older technologies, this is not the case for newer technologies.

## III. HEAVY-ION TESTING AND DATA ANALYSIS

Heavy-ion testing was performed at Texas A&M Cyclotron Institute. All tests regarding data provided in this manuscript were performed at room temperature. The following sections provide accelerated heavy-ion data and describe data analysis.

### A. Device under Test and Designs under Analysis

The device under test (DUT) for this study is the Xilinx Kintex-7 FPGA (XC7K325T-1FBG900). It is a high speed FPGA manufactured on a 28nm process.

The designs under analysis (DUA), for this manuscript, are three versions of the NASA Electronics Parts and Packaging (NEPP) developed counter arrays [6]. For statistical purposes, each counter array contains 200 8-bit counters. The DUAs vary by TMR schemes as follows: (1) No-TMR reference design, (2) DTMR design, and (3) GTMR design. LTMR was not implemented because it has been shown, in Xilinx non-radiation-hardened FPGA devices, that the LTMR methodology will cause higher SEU susceptibility than designs that do not contain mitigation [11].

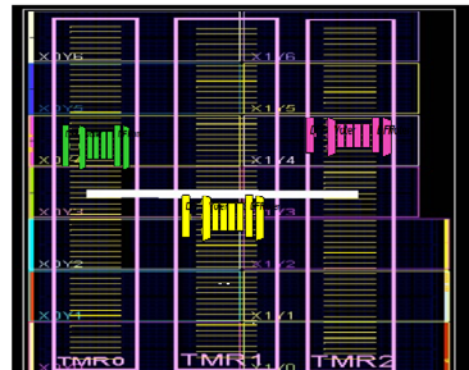
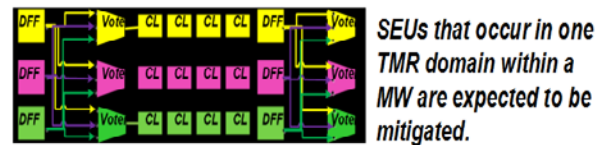


Fig. 4: Example of how a mitigation window is partitioned in the DUT.

The DTMR and GTMR designs are partitioned in order to reduce shared resources. This in turn can potentially reduce single points of system failure from SEUs. Fig. 4 illustrates the logical partitioning of the DUAs within the DUT.

The goal for this test-suite is to investigate the impact of  $T_{skew}$  to TMR'd designs within high-speed devices; i.e., devices that have fast (small)  $T_{comb}$ .

### B. Heavy-Ion Data Metrics

We use fluence-to-failure (particles/( $cm^2 \cdot design$ )) as a metric for comparing mitigation strategies. The formulation of the metric is as follows: for each test, the calculated effective fluence associated with the first observed failure is noted; at each linear energy transfer (LET MeV $\cdot cm^2/mg$ ) value, the mean fluence to first failure (MFTF particles/( $cm^2 \cdot design$ )) is calculated. The MFTF for each of the DUAs is illustrated in Fig. 5. As MFTF increases, the mitigation strength increases; i.e., a stronger mitigation strategy is able to withstand more particles. As a note, the average SEU cross-section ( $\sigma_{SEU}$ ) per LET for each DUA is the inverse of MFTF.

### C. Heavy-Ion Data Analysis

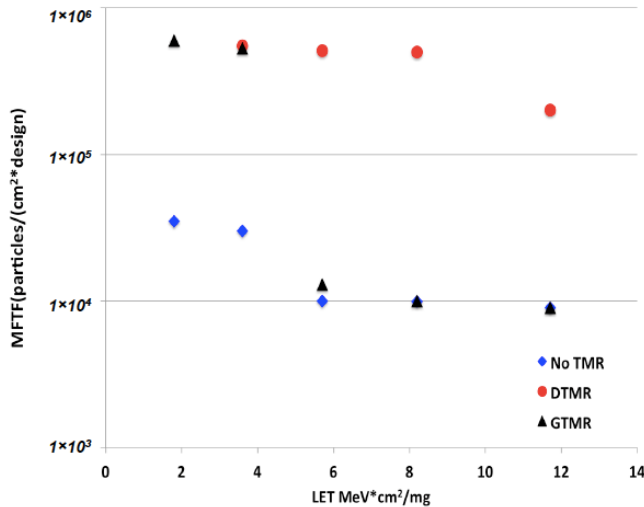


Fig. 5: Heavy-ion data representing mean fluence to failure (MFTF) versus linear energy transfer (LET). Data are for three versions of counter-array designs: (1) counter-array with no-TMR, (2) counter-array with GTMR, and (3) counter-array with DTMR.

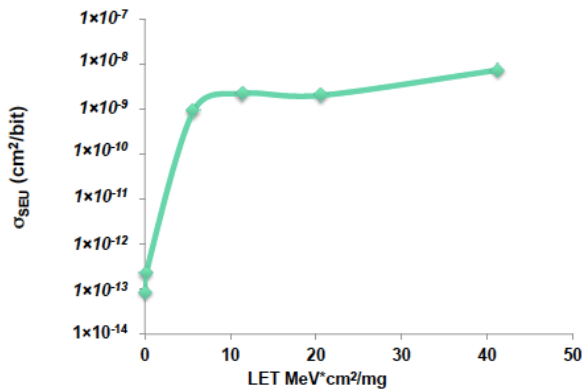


Fig. 6: Heavy-ion data representing configuration SEU cross-section ( $\sigma_{SEU}$ ) versus linear energy transfer (LET).

The following sections are analyses of the SEU MFTF data represented in Fig. 5 and Fig. 6. The analyses are based off of the following notes:

- It is not definitive that an SEU in a configuration bit will cause a system failure because not all configuration bits are used within a DUA [4-6]. If the configuration bit SEU causes a system failure, then the configuration bit will have to be corrected (scrubbed) prior to fixing the state of the system [6].
- If an SEU affects the data-path or global tree, e.g., a captured SET or SEU, then the system state can be corrected without configuration scrubbing. Hence a difference between an SEU in the data-path versus an SEU in configuration is that data-path SEUs can be corrected without scrubbing.
- Both DTMR and GTMR schemes are partitioned; and are partitioned using the same methodology and spacing. Hence, configuration bits that control circuitry in more than one of the redundant TMR domains are minimized.

- There are roughly  $3.3 \times 10^7$  configuration bits in the Kintex-7 XC7K325T [8].
- All MFTF tests were run up to a fluence of  $1 \times 10^6$  particles/cm<sup>2</sup>.
- During testing a variety of no-TMR reference designs [6][11], global SEUs are first observed at  $1.8 \text{ MeV} \cdot \text{cm}^2/\text{mg}$ . However, the observed system failures were small perturbations. This suggests that SETs at low LETs mostly affect lower leaf clock tree buffers (buffers closer to individual DFFs). This can be explained because, at low LET values, SETs are not strong enough to propagate from higher clock tree leaves (buffers) to the DFFs. As LET is increased, SET strength is increased and larger system perturbations occur. Hence, at higher LET values, SETs that occur at higher clock tree buffers can propagate and affect terminal leaf DFFs.
- Due to the partitioning scheme, lower leaf clock tree SEUs will only affect DFFs that are physically near each other.

#### 1. Data Analysis $LET = 1.8 \text{ MeV} \cdot \text{cm}^2/\text{mg}$

At LET equal to  $1.8 \text{ MeV} \cdot \text{cm}^2/\text{mg}$ , as shown in Fig. 5, no SEU system failures were observed with DTMR counters up to a particle fluence of  $1 \times 10^6 \text{ cm}^2/\text{mg}$ . GTMR MFTF is about a decade higher than No-TMR counters. GTMR and DTMR have MWs. However, GTMR has the possibility of having broken MWs because of  $T_{skew}$ . For no-TMR there are no MWs because nothing is mitigated.

Regarding Fig. 6. and [11], there are very few configuration SEUs in this LET range. We attribute the majority of system failures at an  $LET = 1.8 \text{ MeV} \cdot \text{cm}^2/\text{mg}$  to SETs that occur in low leaf clock tree buffers. Because no-TMR has no mitigation, clock SETs can affect the system. Because of the implemented partitioning scheme and the fact that low leaf clock SETs will only affect DFFs physically near each other, DTMR is not perturbed. GTMR can only be affected by low leaf clock SETs if the design contains broken MWs. Fig. 5 shows GTMR is affected at an  $LET = 1.8 \text{ MeV} \cdot \text{cm}^2/\text{mg}$ . Hence, it is assumed that due to  $T_{skew}$ , GTMR has broken MWs. Because there is a decade of difference between GTMR and no-TMR MFTF, suggests that not every MW in the GTMR design is broken. Hence the GTMR design contains pockets of broken MWs.

#### 2. Data Analysis at $2 \text{ MeV} \cdot \text{cm}^2/\text{mg} < LET < 4 \text{ MeV} \cdot \text{cm}^2/\text{mg}$

As shown in Fig. 5, at LET values between  $2 \text{ MeV} \cdot \text{cm}^2/\text{mg}$  and  $4 \text{ MeV} \cdot \text{cm}^2/\text{mg}$ , DTMR and GTMR MFTF are statistically equivalent. We attribute the system failures to be mostly due to SETs in the clock tree. However due to the increase in LET, higher leaf clock tree buffers are affected.

In a DTMR scheme, there is only one clock tree. Higher leaf clock tree SETs can span across TMR domains and cause a DTMR system failure. In a GTMR design, the clock trees are redundant. However, as previously mentioned, if the GTMR implementation has broken MWs, a clock tree SET can cause system malfunction.

### 3. Data Analysis $4\text{MeV}\cdot\text{cm}^2/\text{mg} < \text{LET} < 10\text{MeV}\cdot\text{cm}^2/\text{mg}$

As shown in Fig. 5, at LET values greater than  $4\text{MeV}\cdot\text{cm}^2/\text{mg}$  GTMR approaches No-TMR. Configuration bit SEUs are now significant. This is because there are roughly  $3.3 \times 10^7$  configuration bits with a  $\sigma_{\text{SEU}}$  of  $1.0 \times 10^{-9} \text{cm}^2/\text{bit}$  (as shown in Fig. 6). It is noted that each design only utilizes a percentage of the configuration bits.

For DTMR, because of the partitioning scheme and the singular clock tree, there are no broken MWs and most configuration SEUs do not affect system operation. For this reason, the DTMR is only affected by higher leaf clock tree perturbations.

### 4. Data Analysis $\text{LET} > 10\text{MeV}\cdot\text{cm}^2/\text{mg}$

As LET increases, the number of global route SETs increase. In addition, the SETs become stronger and can affect higher leaf clock buffers. There also is the possibility of configuration multiple bit upsets (MBUs). Due to both phenomena, the DTMR scheme starts to break down and approach No-TMR and GTMR.

## IV. CONCLUSION

Theoretically, GTMR should be the strongest TMR mitigation scheme. For this reason, it has been suggested as the TMR strategy of choice for SRAM-based FPGAs. However, the uncontrollable clock skew between GTMR clock domains can cause race conditions that inevitably weaken GTMR mitigation. For small (less complex) designs implemented in FPGAs that contain clock trees with minimal  $T_{\text{skew}}$ , GTMR can be realizable [2-6]. As device and design area increase, as with modern devices such as the Xilinx Kintex-7, GTMR clock skew also increases. The increase in skew increases the potential for race conditions. Some race conditions can be uncontrollable and unrecognizable by manufacturer-supplied design tools. Consequently, Kintex-7 GTMR versus DTMR heavy-ion data show that GTMR or XTMR is an ineffective and unreliable mitigation solution. In conclusion, we suggest that DTMR is a more applicable TMR strategy for larger commercial SRAM-based FPGA devices.

## V. REFERENCES

- [1] Microsemi Datasheet: "RTAX-S/SL RadTolerant FPGAs" [http://www.actel.com/documents/RTAXS\\_DS.pdf](http://www.actel.com/documents/RTAXS_DS.pdf), V5.2, October 2007
- [2] Sani Habinc, "Functional Triple Modular Redundancy (FTMR)VHDL Design Methodology for Redundancy in Combinatorial and Sequential Logic," Geisler research website: [http://www.gaisler.com/doc/fpga\\_003\\_01-0-2.pdf](http://www.gaisler.com/doc/fpga_003_01-0-2.pdf), December 2002.
- [3] C. Ramamurthy, S. Chellappa, V. Vashishtha, A. Gogulamudi and L. T. Clark, "High Performance Low Power Pulse-Clocked TMR Circuits for Soft-Error Hardness," in *IEEE Transactions on Nuclear Science*, vol. 62, no. 6, pp. 3040-3048, Dec. 2015.
- [4] C. Carmichael, E. Fuller, P. Blain, and M. Caffrey, "SEU Mitigation Techniques for Virtex FPGAs in Space Applications," in MAPLD Proceedings, September 1999.
- [5] F. Lima, L. Carro, R. Reis, "Designing fault tolerant systems into SRAM-based FPGAs," *Design Automation Conference, 2003. Proceedings*, vol., no., pp.650,655, 2-6 June 2003
- [6] M. Berg, "FPGA SEE Test Guidelines", NASA Radiation Effects and Analysis Group Website: [https://nepp.nasa.gov/files/23779/FPGA\\_Radiation\\_Test\\_Guidelines\\_2012.pdf](https://nepp.nasa.gov/files/23779/FPGA_Radiation_Test_Guidelines_2012.pdf), July 2012. Berg M, "Selection of Integrated Circuits for Space Systems Section V: Example 1: Trading ASIC and FPGA Considerations for System Insertion," NASA Radiation Effects and Analysis Group Website: [https://nepp.nasa.gov/files/22744/nsrec09sc\\_Berg.pdf](https://nepp.nasa.gov/files/22744/nsrec09sc_Berg.pdf), July 2009.
- [7] D. Davies, J. Wakerly, "Synchronization and Matching in Redundant Systems," in *IEEE transaction in Computers* vol., c-27, no. 6, pp. 531-540, June 1978.
- [8] Xilinx data sheet. "Kintex-7 FPGAs Data Sheet: DC and AC Switching Characteristics," Website: [http://www.xilinx.com/support/documentation/data\\_sheets/ds182\\_Kintex\\_7\\_Data\\_Sheet.pdf](http://www.xilinx.com/support/documentation/data_sheets/ds182_Kintex_7_Data_Sheet.pdf), September 2012.
- [9] C. Maxfield, *The Design Warrior's Guide to FPGAs*. Burlington, MA Elsevier, 2004.
- [10] D. Smith, *HDL Chip Design*, Doone Publications, Madison, AL, USA, 1996
- [11] M. Berg, "Single Event Effects in FPGA Devices 2015-2016", NASA Radiation Effects and Analysis Group Website: <http://nepp.nasa.gov/workshops/etw2015/talks/25%20%20Thu/0920%20-%20BergFPGA2015.pdf>