LETTER

Design and verification for CDC synchronization based on TMR

Yuyang Fan[1, a)] **and Zhi Deng**[1]

Abstract Triple modular redundancy (TMR) is widely used in FPGA/ASIC circuits to protect circuits against single event upsets (SEUs). However, because of the interference of metastability on signal transmission across clock domains, the TMR circuits' capability against SEUs is reduced greatly. In order to solve this problem, a cross-clock transmission solution which could be applied in TMR circuits are presented. In addition, simulation-based verification which combined protocol assertions, metastable injection and forced inversion is proposed to succeed in verifying the availability of the solution based on TMR circuits.

Keywords: single event upsets, clock domain crossing (CDC), triple modular redundancy

Classification: Integrated circuits (memory, logic, analog, RF, sensor)

1. Introduction

With scaling down of the feature size of chips, field programmable gate arrays (FPGAs) and ASIC are more susceptible to the effects of ionizing irradiation, especially SEUs, which seriously threaten the reliability of circuits [1, 2, 3]. To reduce impacts of SEUs, triple modular redundancy (TMR), a highly efficient mitigation technique, is the most used method [4, 5, 6]. Metastability caused by CDC is another problem which threaten the reliability of circuits. With the increases of clock frequency, asynchronous signals and signals transmitted across asynchronous clock boundaries, circuits are easily affected by metastability [7, 8, 9].

When synchronizers and their corresponding transmission protocols are used in TMR circuits, synchronizers may be interfered by metastability and SEUs at the same time. Consequently, TMR circuit including the synchronizers [10] loses its original function—voters in TMR circuits may fail. Therefore, the specific modifications for synchronizers and their corresponding transmission protocols in TMR applications are required.

At present, majority of researches focus on anti-irradiation design [11, 12, 13, 14, 15, 16, 17], irradiation test [18, 19, 20, 21, 22], CDC design [7, 23, 24], metastable parameter test [25, 26, 27]. However, they have not addressed the problem combining both effects of SEUs and metastability. Li Yubo in [28] described a single-bit CDC solution based on TMR. However, multi-bit solution was not mentioned, and methods for simulation-based function verification were lacked as well. Melanie Berg in [29] designed a FIFO synchronization strategy based on TMR technique, but no verification solution was provided. In addition, even though asynchronous FIFO has great applicability when synchronizing across clock domains, it occupies a large area and has complex design [30]. In particular, after taking the TMR technique, the circuit scale will be even larger. For further progress, we provide a multi-bit solution in this paper. Comparing to asynchronous FIFO, the solution is simpler and its area is smaller.

To verify the CDC synchronizers based on TMR, metastability and SEUs should both be taken into consideration. However, because the event that metastability and SEUs happen simultaneously is rare in practical projects, traditional methods for detecting metastability and SEUs are time-consuming and costly [31]. Moreover, if the problems are found after the design tape out, designers need redesign and tape out chips again. In this situation, it is obvious that the cost and time consumption increase tremendously. Therefore, new methods are in demand. This thesis presents a simulation method, which takes metastability, SEUs and constraint application conditions of synchronizers into consideration. Then a testbench is designed to verify its correctness. Through this testbench, the correctness of cross-clock domain synchronizers based on TMR technique can be judged by simulating, reducing the cost of testing.

2. Problem description

2.1 Discrepancy caused by metastability

Because of metastability, traditional simulator can not reflect all the situations of signals in the actual hardware [32]. As shown in the Fig. 1, due to setup time, in the receiving clock domain, actual hardware signal may delay one clock cycle, instead of corresponding to ideal simulator's signal.

Fig. 1 shows two different situations of the output data of actual hardware circuit. In scenario 1, the actual hardware's

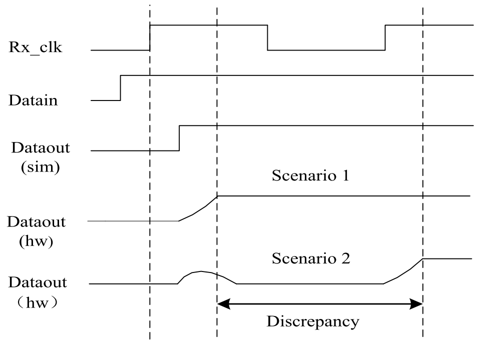


Fig. 1 Two different scenarios due to setup time.

[1] Civil Aviation University of China, Tianjin Key Laboratory for Civil Aircraft Airworthiness, Tianjin 300300, China

[a)] fanyuyang1988@126.com

DOI: 10.1587/elex.17.20200287
Received August 21, 2020
Accepted September 3, 2020
Publicized September 16, 2020
Copyedited November 10, 2020

output data is consistent with the simulator's output data. In scenario 2, the output data of actual hardware circuit is delayed by one clock cycle. This phenomenon is caused by metastability introduced by setup time violation. Similar with the scenario due to setup time violation, hold time violation may cause the actual hardware's output data to advance one clock cycle.

Because of the differences between actual hardware circuit and simulator, traditional simulation can not be used to simulate actual hardware circuit behavior in this situation. Therefore, simulation methods for CDC need to be improved.

2.2 Combinatorial problem

When using TMR on the CDC circuits, the combination of metastability caused by CDC and SEU caused by ionizing irradiation will bring combinatorial problem. The problem is described in detail below.

See Fig. 2, Tx_sig, a single-bit signal, passes through triplicated registers, and then they are sent to triplicated registers in receiving clock domain. Ideally, the duration of Delay1–3 are identical, and Tx_clk, Rx_clk in different registers are in the same phase, so data in register B1, B2 and B3 are consistent on condition of no SEU. However, Delay1, Delay2 and Delay3 exist differences in practical application and Tx_clk inputting to A1–A3 registers and Rx_clk inputting to B1–B3 registers are not completely in the same phase, resulting in inconsistency of data in receiving clock domain. When q1_Txclk, q2_Txclk and q3_Txclk are directly sent into two flip-flop synchronizers, q1_Rxclk, q2_Rxclk and q3_Rxclk may change asynchronously like case2 and case3 of Fig. 3 (a). In case2, q1_Rxclk changes one clock cycle ahead because of metastability. In case3, q3_Rxclk is delayed by one clock cycle due to the metastable state. In these cases, the capacity of TMR technique against SEU is reduced. As shown in Fig. 3 (b), in case3, SEU happens, q1_Rxclk stuck at 0 fault. In this situation, the voter cannot render a valid vote, resulting in the output data being 0.

To solve the combinatorial problem, we designed a new synchronization strategy, and corresponding verification is conducted.

3. Strategy for CDC synchronizer based on TMR

The most common solution for single-bit signal CDC is to design a two flip-flop synchronizer. When using TMR technique on the two flip-flop synchronizers, if synchronized signals only last one clock cycle in receiving clock domain, the TMR circuit may lose its capacity against SEU in the presence of metastability. In order to ensure enable signals in receiving clock domain maintain at least two clock cycles, when using two flip-flop synchronizers, the period of the enable signals in sending clock domain needs to be calculated. Suppose that the signal in sending clock domain last n * T1(T1 stands for sending clock cycle), and the signal in receiving clock domain includes at least two rising edge of receiving clock signal. Taking the worst case into consider, $n * T1 > 2 * T2$ (T2 stands for receiving clock cycle), $n > 2 * T2/T1$ (n is a positive integer). Therefore, the transmitted signal must keep $\lfloor (2 * T2 / T1) \rfloor + 1$ times T1.

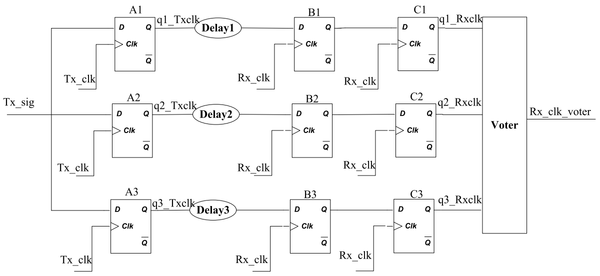
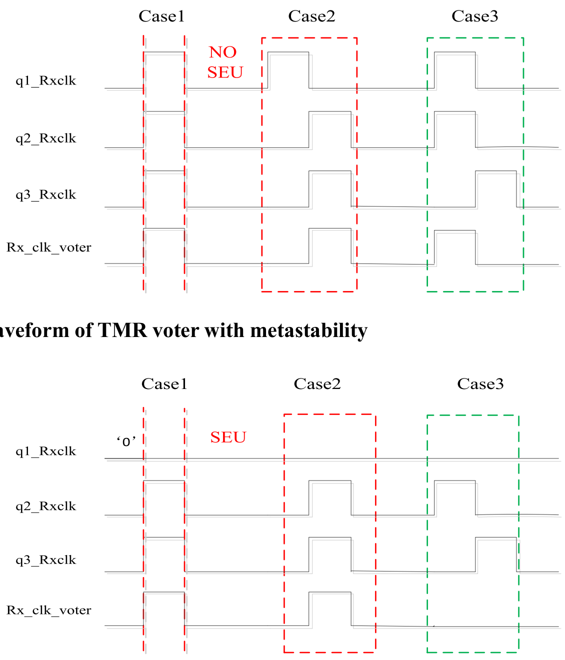


Fig. 2 Different delay situations in TMR circuits.

There are many ways for multi-bit data transmission across clock domain, such as asynchronous FIFO, data selector (DMUX), handshake protocol. Similar with a two flip-flop synchronizer, when using TMR technique on synchronizers, multi-bit synchronizers need to meet certain transmission protocols to work properly.

The basic principle of this strategy is to ensure the consistency of three data and three enable signals in one or one more receiving clock cycle on condition of no SEU. When one data or an enable signal is affected by SEU, TMR voter can still vote the correct data and enable signal. To achieve this goal, the data and enable signals are extended. The rule of extension is as follow: data or enable signal need to keep at least K (K is related to the kind of synchronizer) times T1, K is inversely proportional to T1 and is proportional to T2. This strategy can be used for all CDC synchronizers based on TMR, then we take data selector (shown in Fig. 4) as an example for explaining the strategy.

DMUX needs to meet transmission protocols to work properly: asynchronous input data data_txclk must remain



(a) Waveform of TMR voter with metastability

(b) Waveform of TMR voter with combinatorial problem

Fig. 3 Application problem based on TMR in CDC.

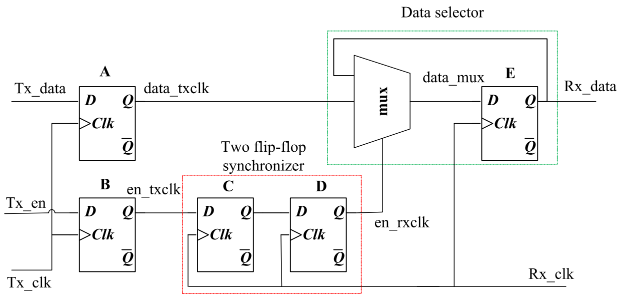


Fig. 4 DMUX synchronizer.

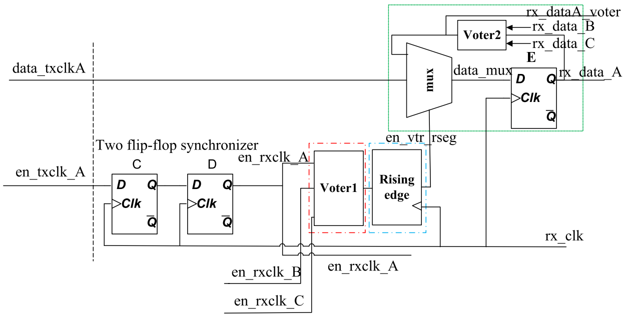


Fig. 5 DMUX synchronizer based on TMR.

unchanged not only during the time when enable signal (Tx_en) in sending clock domain is valid, but also during the time when selection signal (en_rxclk) in receiving clock domain is valid. Thus, asynchronous input data (data_txclk) must be kept valid for a long time. At the same time, enable signal (en_txclk) need to meet the two flip-flop synchronizer's transmission requirements.

As for the DMUX synchronizer, when using TMR technique on the synchronizer, two voters need to be added. As shown in Fig. 5, Voter1 is used for voting enable signal. The other two input data (en_rxclk_B, en_rxclk_C) of Voter1 come from redundant modules. Voter2 is used to vote received data. The rx_data_B and rx_data_C are from the other two redundant modules. In addition, a rising edge voting device is added after Voter1, so that the enable signal in receiving clock domain only keeps one receiving clock cycle, avoiding sampling a single data (data_txclk_A) twice in receiving clock domain.

When using TMR technique on DMUX circuit, en_rxclk_A–C needs to keep at least two receiving clock cycles. So in sending clock domain, en_txclk_A–C need to last $\lfloor 2 * T2/T1 \rfloor + 1$ times T1. However, data signal needs to be maintained longer. Ideally, it takes three receiving clock cycles to ensure that the corresponding data data_txclk_A transmit to the output register E. When metastability is generated, at least four receiving clock cycles are required for transmitting the data. Suppose that sending data keeps unchanged within $n * T1$. To ensure the data are received by output register E successfully, at least four rising edge of receiving clock are required during $n * T1$. In the worst case, $n * T1 > 4 * T2$, $n > 4 * T2/T1$. So signal Tx_sig must be maintained longer than $\lfloor (4 * T2/T1) \rfloor + 1$ times T1. So in order to ensure the function of DMUX, data_txclk_A need to last $\lfloor 4 * T2/T1 \rfloor + 1$ times T1.

4. Simulation techniques

Simulation techniques are used to verify the correctness of the CDC strategy, and they are applicable for all CDC synchronizers based on TMR. Three phases were designed to simulate CDC circuits based on TMR technique. Firstly, using assertion to check the application protocol of the synchronizer. The assertions are judged automatically in dynamic simulation, which ensure that the synchronizer meets requirements of TMR technique. Secondly, metastability was injected to simulate the delay of signals in registers. Lastly, a forced inversion solution is designed to simulate SEU. Any one triplicated signal in TMR circuit is randomly forced to upset. These three verification techniques can be randomly combined, ensuring any disturbed scenario is covered.

4.1 Protocol verification

In a specific application scenario, a cross-clock synchronizer's use conditions need to conform to its transmission protocol, so as to ensure that data can be transmitted correctly and voter can be effective even if a triplicated signal is inverted.

1. The protocol of two flip-flop synchronizer based on TMR

According to the analysis before, when using two flip-flop synchronizer in TMR circuit, in sending clock domain, duration time of transmission signals need to last long enough, and it's effective time is at least $\lfloor 2 * (T2/T1) \rfloor + 1$ times T1. Assertion is described as follow:
@(posedge clk1) $rose(en_txclkA) |->(en_txclkA[* ((2 * `T2)/`T1 + 1)]);

2. The protocol of DMUX based on TMR

Data need to meet the transmission protocol in order to be transmitted correctly in DMUX synchronizer. Transmission protocol is as follow: when selection signal of sending clock domain is valid, input data need to remain stable. After synchronization, in receiving clock domain, the corresponding input data need to remain stable when selection signal of receiving clock domain is valid. According to the analysis in section 3, assertion is designed:
@(posedge clk1) $rose(en_txclkA) |->(data_txclkA[* ((4 * `T2)/`T1 + 1)]);

Besides, enable signals also need to be verified, and the assertion for enable signals need to remain at least $\lfloor 2 * (T2/T1) \rfloor + 1$ times T1. Thus, assertion is as follow:
@(posedge clk1) $rose(en_txclkA) |->(en_txclkA)[* ((2 * `T2)/`T1 + 1)];

As shown in Fig. 6, clk1 and clk2 are the input clock, the clock cycle of clk1 lasts 4 ns, the clock cycle of clk2 lasts 10 ns. data_txclkA–C are input data of module A–C respectively; en_txclkA–C are enable signals in sending clock domain of module A–C respectively; rx_dataA–C are output data of module A–C respectively; en_rxclkA–C are enable signals in receiving clock domain of module A–C respectively. The en_vtr_rseg is the enable signal generated by rising edge circuit. When en_txclkA–C are valid, data_txclkA–C remain stable. We can see the data can be transmitted successfully when the protocol assertion is

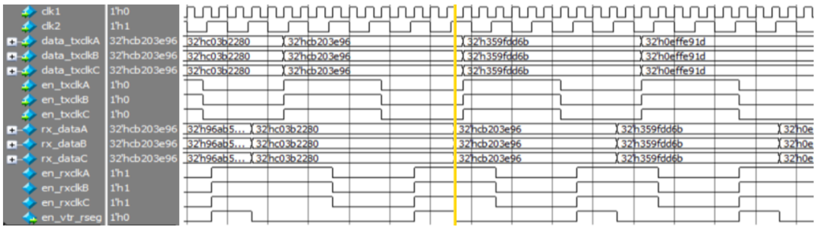


Fig. 6 Data transmission waveform of DMUX synchronizers.

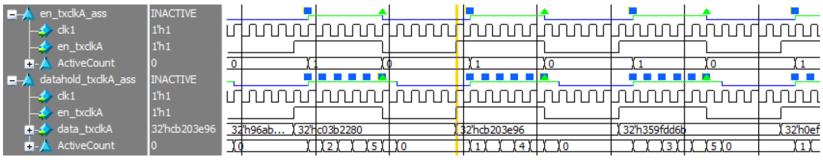


Fig. 7 Protocol verification waveform of DMUX synchronizers.

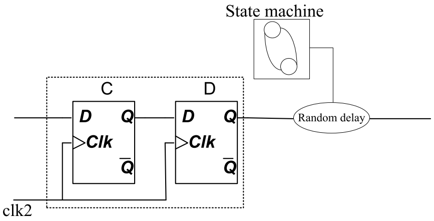


Fig. 8 Location of metastable injection.

passed.

According to Fig. 7, when assertion is passed, data can be transmitted successfully.

4.2 Metastable injection

When using TMR technique on synchronizers, there exists convergent points where voters vote the data and enable signals. To check if the convergence phenomenon affects the triplicated protocol-compliant synchronizers, simulating the application scenario with metastability is significant. Because signal cannot advance or delay at the same time; in addition, any one triplicated signal has the same effects on the circuit when it advances or delays one clock cycle, we choose to simulate the situation: a triplicated signal delays one clock cycle.

The metastable injection position is selected at the output of the second register of the two flip-flop synchronizer, as shown in Fig. 8. To ensure the enable signal will not be shortened due to the uncertainty of random delay, state machine was used to trigger the random delay. Additionally, during effective time of a enable signal, the signal triggering the random delay remained unchanged. This ensured the enable signal will not be shorter when metastability was injected.

In Fig. 9, when receiving the data 32’hcb203e96, metastability is not injected into en_rxclkB (enable signal in receiving clock domain). But en_rxclkA and en_rxclkC both delay one clock cycle because of metastability. In this case, the data in sending clock domain remain stable, no data is missed. When receiving the data 32’h359fdd6b, metastability is not injected into en_rxclkA and en_rxclkC. But en_rxclkB delay one clock cycle because of metastability. In this case, the data still pass successfully. In summary, Fig. 9 shows that the synchronizer designed is effective when metastability happens.

4.3 Forced inversion

Synchronizers affected by metastability should still have the ability against SEUs. In order to simulate SEUs, a triplicated signal in TMR circuits is forced to be inverted. Forced inversion can occur in the enable signal register or data registers in DMUX.

See Fig. 10, en_txclkA and en_txclkC are in normal state, but en_txclkB is stacked to 0. In this situation, the rx_dataA–C are right, proving that data can be transmitted successfully. So synchronizers in this design have the ability against SEUs.

4.4 Comprehensive simulation

As shown in Fig. 11, a comprehensive simulation testbench is composed of driver, scoreboard, monitor, protocol assertion IP, metastable injection and forced inversion module. A driver generates a dynamic stimulus. Protocol assertion IP is used to ensure that the synchronizer meets the requirements

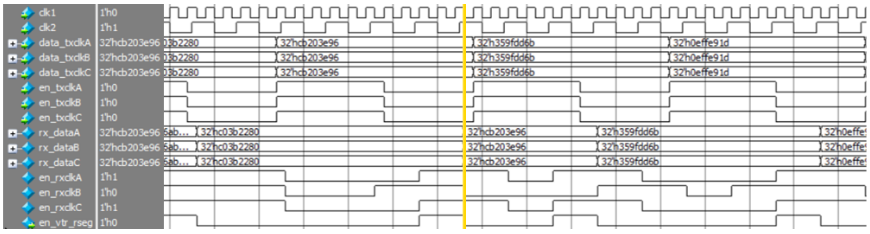


Fig. 9 Oscillogram of DMUX with metastable injection.

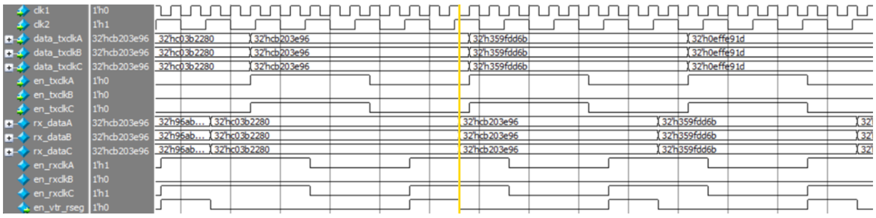


Fig. 10 Oscillogram of synchronizers with forced inversion.

Table I Data tests

Scenario	Number of sending data	Number of SU	Number of metastable injections (M)	Number of MSU MMSU MMMSU	Number of assertion pass	Number of correct data
T1>T2	1000	200	1400	60/80/30	6000	1000
T1=T2	1000	200	1400	60/80/30	6000	1000
T1<T2	1000	200	1400	60/80/30	6000	1000

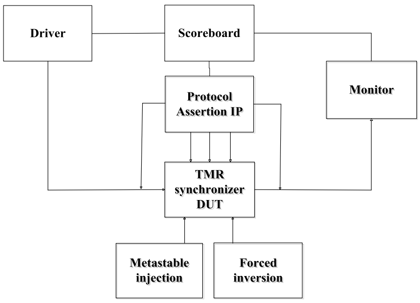


Fig. 11 Comprehensive simulation testbench.

of transmission protocol. At the same time, metastable injection and forced inversion modules are used to simulate the disturbance from metastability and SEUs. By metastable injection a delayed signal is generated. Meanwhile, in the testbench, key register's signals are randomly inverted to simulate SEU. The scoreboard can be used to automatically count the number of circuit operation errors.

The test results are shown in Table I. The event when any one triplicated signal is inverted is denoted as SU. The event when metastability is injected into only one triplicated CDC path is denoted as M. As for the event when metastability is injected into one triplicated signal and one triplicated signal is inverted, we denoted it as MSU. As for the event when metastability is injected into two triplicated signals and one triplicated signal is inverted, we denote it as MMSU. As for the event when metastability is injected into all the signals and one triplicated signal is inverted, we denote it as MMMSU.

We can adjust the periods of sending clock or receiving clock through a parametric testbench, and the simulation build three kinds of scenarios. SU, M, MSU, MMSU and MMMSU were constructed in each kind of scenario. When protocol assertions are passed, automated comparison in scoreboard is used to indicate the results of the DUT circuit.

According to Table I, all data can be transmitted successfully in above situations, no data is missed when the protocol assertions are passed. Taken together, these figures and this table show that the synchronizer designed in this thesis is effective.

5. Conclusion

This thesis presents a strategy to solve combinatorial problem in CDC synchronizers based on TMR. Then we design two synchronizers based on this strategy. Finally, simulation techniques are designed to verify this strategy, and we apply simulation techniques to DMUX as an example. It is important to note that when using simulation techniques on other synchronizers, the assertions used in protocol verification ought to be modified. Through this simulation verification method, designers can verify the synchronizers at the very early stage of design cycle.

Acknowledgments

This research is supported by Aeronautical Science Foundation of China under Grant 20182667009 and the Foundation Research Funds for the central Universities under Grant 3122018D033.

References

[1] R. Jasinski: "Fault-tolerance techniques for SRAM-based FPGAs," Comput. J. **50** (2007) 248 (DOI: 10.1093/comjnl/bxl060).

[2] N. Gaspard, *et al.*: "Technology scaling comparison of flip-flop heavy-ion single-event upset cross sections," IEEE Trans. Nucl. Sci. **60** (2013) 4368 (DOI: 10.1109/TNS.2013.2289745).

[3] D. Tang, *et al.*: "Soft error reliability in advanced CMOS technologies-trends and challenges," Sci. China Technol. Sci. **57** (2014) 1846 (DOI: 10.1007/s11431-014-5565-6).

[4] K.S. Morgan, *et al.*: "A comparison of TMR with alternative fault-tolerant design techniques for FPGAs," IEEE Trans. Nucl. Sci. **54** (2007) 2065 (DOI: 10.1109/TNS.2007.910871).

[5] R. Yao, *et al.*: "State synchronization technique based on present input and healthy state for repairable TMR systems," IEICE Electron. Express **13** (2016) 20161000 (DOI: 10.1587/elex.13.20161000).

[6] C. Ramamurthy, *et al.*: "High performance low power pulse-clocked TMR circuits for soft-error hardness," IEEE Trans. Nucl. Sci. **62** (2015) 3040 (DOI: 10.1109/TNS.2015.2498919).

[7] Altera Co.: White Paper: understanding metastability in FPGAs (2009).

[8] D. Rennie, *et al.*: "Performance, metastability, and soft-error robustness trade-offs for flip-flops in 40 nm CMOS," IEEE Trans. Circuits Syst. I, Reg. Papers **59** (2012) 1626 (DOI: 10.1109/TCSI.2012.2206505).

[9] Y. Feng, *et al.*: "Clock domain crossing fault model and coverage metric for validation of SoC design," 2007 Design, Automation & Test in Europe Conference & Exhibition (2007) 1385 (DOI: 10.1109/date.2007.364491).

[10] C.E. Cummings: *Clock Domain Crossing Design & Verification Techniques Using SystemVerilog* (Boston, 2008).

[11] J. Jason: "Fault-tolerant sequencer using FPGA-based logic designs for space applications," Ph.D. Dissertation, Naval Postgraduate School, Monterey (2013).

[12] L. Ciani and M. Catelani: "A fault tolerant architecture to avoid the effects of single event upset (SEU) in avionics applications," Measurement **54** (2014) 256 (DOI: 10.1016/j.measurement.2014.02.018).

[13] Z. Shen, *et al.*: "Study on FPGA SEU mitigation for the readout electronics of DAMPE BGO calorimeter in space," IEEE Trans. Nucl. Sci. **62** (2015) 1010 (DOI: 10.1109/TNS.2015.2427293).

[14] F. Siegle, *et al.*: "Mitigation of radiation effects in SRAM-based FPGAs for space applications," ACM Computing Surveys **47** (2015) 1 (DOI: 10.1145/2671181).

[15] Xilinx Co.: Triple module redundancy design techniques for virtex FPGAs (2006).

[16] W.G. Xiao and B.D. Yao: "Design of an anti-irradiation beam-steering unit based on ASIC circuit," The Journal of Engineering **2019** (2019) 5810 (DOI: 10.1049/joe.2019.0047).

[17] R.Q. Song, *et al.*: "MSIFF: A radiation-hardened flip-flop via interleaving master-slave stage layout topology," IEICE Electron. Express **17** (2020) 20190708 (DOI: 10.1587/elex.17.20190708).

[18] S. Li, *et al.*: "Study on the single-event upset sensitivity of 65-nm CMOS sequential logic circuit," IEICE Electron. Express **17** (2020) 20200102 (DOI: 10.1587/elex.17.20200102).

[19] M.V. O'Bryan, *et al.*: "Compendium of recent single event effects for candidate spacecraft electronics for NASA," 2013 IEEE Radiation Effects Data Workshop (REDW) (2013) 1 (DOI: 10.1109/REDW.2013.6658188).

[20] G. Borghello, *et al.*: "Dose-rate sensitivity of 65-nm MOSFETs exposed to ultrahigh doses," IEEE Trans. Nucl. Sci. **65** (2018) 1482 (DOI: 10.1109/TNS.2018.2828142).

[21] W. Chen, *et al.*: "Single-event upsets in SRAMs with scaling technology nodes induced by terrestrial, nuclear reactor, and monoenergetic neutrons," IEEE Trans. Nucl. Sci. **66** (2019) 856 (DOI: 10.1109/TNS.2019.2912021).

[22] Z. Mohseni, *et al.*: "Reliability characterization and activity analysis of lowRISC internal modules against single event upsets using fault injection and RTL simulation," Microprocess. Microsyst. **71** (2019) (DOI: 10.1016/j.micpro.2019.102871).

[23] Q.C. Zhang: "Application of static formal method in CDC and reset verification," China Integrated Circuit **28** (2019).

[24] D. Konstantinou, *et al.*: "The mesochronous dual-clock FIFO buffer," IEEE Trans. Very Large Scale Integr. (VLSI) Syst. **28** (2020) 302 (DOI: 10.1109/TVLSI.2019.2946348).

[25] B.M. Rogina, *et al.*: "Metastability testing at FPGA circuit design using propagation time characterization," 2010 East-West Design & Test Symposium (EWDTS) (2010) 80 (DOI: 10.1109/EWDTS.2010.5742050).

[26] Y. Tian, *et al.*: "Metastable parameter measurement method based on FPGA," Microelectronics & Computer **33** (2016) 46 (DOI: 10.19304/j.cnki.issn1000-7180.2016.10.010).

[27] Y. Luo, *et al.*: "Impact of total ionizing dose on low energy proton single event upsets in nanometer SRAM," IEEE Trans. Nucl. Sci. **66** (2019) 1848 (DOI: 10.1109/TNS.2019.2922501).

[28] Y. Li, *et al.*: "Synchronization techniques for crossing multiple clock domains in FPGA-based TMR circuits," IEEE Trans. Nucl. Sci. **57** (2010) 3506 (DOI: 10.1109/TNS.2010.2086075).

[29] M. Berg: "Embedding asynchronous FIFO memory blocks in Xilinx Virtex series FPGAs targeted for critical space system applications," Military and Aerospace Programmable Logic Devices (MAPLD) Conference.

[30] A. Kammoun, *et al.*: "An efficient and unified 2D-inverse integer cosine transform (IICT) FPGA-hardware implementation for HEVC standard," Analog Integr. Circuits Process. **101** (2019) 431 (DOI: 10.1007/s10470-019-01469-z).

[31] A.K. Sutton, *et al.*: "SEU error signature analysis of Gbit/s SiGe logic circuits using a pulsed laser microprobe," IEEE Trans. Nucl. Sci. **53** (2006) 3277 (DOI: 10.1109/TNS.2006.886232).

[32] N.V. Toan, *et al.*: "Measurements of metastability in MUTEX on an FPGA," IEICE Electron. Express **15**(2018) 20171165 (DOI: 10.1587/elex.14.20171165).