Journal of Engineering Science and Technology
Vol. 18, No. 2 (2023) 862 - 879
© School of Engineering, Taylor's University

SOFT ERROR MITIGATION IN MEMORY SYSTEM

NORHUZAIMIN JULAI*,
FARHANA MOHAMAD ABDUL KADIR, SHAMSIAH SUHAILI

Faculty of Engineering, University of Malaysia Sarawak,
Jalan Datuk Mohammad Musa, 94300 Kota Samarahan, Sarawak, Malaysia
*Corresponding Author: jnorhuza@unimas.my

Abstract

Technology downscaling has increased the sensitivity of circuitry to being corrupted by single event upsets. To provide more solutions for the issue, a method of error detection and correction is provided in this study. The double exponential model was used to simulate the single event upset current transient. The amplitudes of the transient current from the single event upset were varied until a change in logic value is achieved. A single rail with inverter latch (SIL) circuit configuration is injected in three vulnerable nodes to formulate their respective soft error sensitivities, with the parameters of temperature and voltage supply varied to observe their effects on the critical charge of each node. The temperatures were ranged from -50ºC to 200 ºC, while the supply voltage was varied from 0.7 V to 1.5 V. Decreases in temperature from the range of 200ºC to -50ºC cause the critical charge to increase. Critical charge increases with voltage supply increase from 0.7 V to 1.5 V. A shadow latch was implemented in Cadence and Quartus for error detection and correction. The shadow latch was able to successfully detect the presence of an error and restore the original data from voltages of 0.8 V to 1.2 V.

Keywords: Error correction, Error detection, Latch, Soft error.

1. Introduction

Soft errors have been an increasing worrying problem with the trend of downscaling technology. Soft errors are the result of interactions of energised particles interacting with the electrons in electronic circuitry [1]. There are many sources of these energised particles against which designers must take into account when designing circuits. These include alpha particles, cosmic neutrons, and radiation from the interactions of boron and cosmic neutrons. Alpha particles as a soft error source are borne from the decay of radioactive impurities [2, 3]. Electronics in close proximity to nuclear reaction have little in the way of protection from high energy neutrons and are prone to the occurrence of soft errors [4]. Furthermore, space environments in which high performance equipment used in satellites and aerospace fields, are exposed to a spectrum of ionised particles must also operate in a soft error rich environment.

Secondary particles can be formed when cosmic rays interact with the atmosphere. Neutrons are the primary source of soft errors at the terrestrial level from cosmic rays [5]. Another primary source of radiation in semiconductors is the interaction of cosmic neutrons with boron, which consequently emits alpha particles. These sources are capable of producing soft errors in electronics in vulnerable regions in devices via charge generation, which is the result of the traversal of a particle across a susceptible circuit node. Transistors in the "off" state in CMOS circuitry are highly prone to single effect upsets. Single event upsets can be created by a particle strike in the sensitive region of an NMOS or PMOS transistor. There will then be a current pulse which produces charge, the amount of which can produce a single event transient if it were to reach the critical charge (Q_{crit}). If an energised particle that has also been ionised within close proximity to a vulnerable circuit node, electron-hole pairs can be formed along its trajectory. This can result in charge collection event that will generate a current transient. There are numerous methods that have been used in soft error mitigation, several of which will be discussed in the next section.

In this paper, we used a C-element circuit which is single rail with inverter latch configuration (SIL) as our case study and we demonstrated the technique to detect and correct soft errors that occur in a system. SIL is chosen as case study for memory system and it hold the previous value if the two inputs are not equal.

2. Related Work

2.1. Radiation hardness by design

One method is by employing radiation hardening by design. This is done by implementing changes to a circuit at the design level for memory elements and can involve modifications to layout or circuit design. Pown and Lakshmi [6] proposed a 6T double gate tunnel field effect transistor static random access memory circuit. The design features a resistor and capacitor (RC) component attached between the data nodes of the SRAM, assists data recovery by dampening voltage transients and upping the critical charge. Similar implementations involve integration of resistor-capacitor filtering or selective redundancy for radiation mitigation [7].

The method of radiation hardness by design can also be applied to other existing techniques such as triple modular redundancy circuits [8]. The study provides a SEE-tolerant TMR circuit architecture for space applications. The proposed D-flip-

flop circuit method is totally digital and uses a local SET filter to mitigate transients on the data path. Chen et al. [9] proposed that full NMOS or PMOS transistors encapsulate the cell storage nodes to lower the count of vulnerable nodes and increase the reliability of the circuit. The latch presents acceptable recovery from SEU, and a relatively reduced transfer time for the sacrifice of size and power use, but ultimately improved performance.

Radiation hardness assurance has also been designated as a process to selectively analyse components in accordance with the requirements for operation in a radiation heavy space environment [8, 9]. Space radiation particles are characterized by their linear energy transfer characteristics to identify their propensity for triggering single event upsets in satellite operations. Requirements for components are evaluated for their robustness or sensitivities against single event upsets, their operational performance in orbit, and the risk analysis under operating conditions [10, 11].

The design and material composition of the circuit can also come into play in radiation hardening design. Silicon on insulator devices have demonstrated a considerable resilience to radiation effects due to isolated individual transistors, reducing leakage current and immunising the design to single event effects. However, hole accumulation in fully isolated transistors become a problem. Hara et al. [12] suggested a buried well design that would reduce the total ionisation dose effect.

2.2. Triple modular redundancy

One well known method is the triple modular redundancy technique [13]. This technique triplicates targeted circuitry and then selects an output through a majority voter. Due to the repetition of the circuit design, this technique imposes a large area overhead which can be problematic with the downscaling of technology.

Furthermore, the majority voter itself has a vulnerability to soft errors. Some iterations of this technique a dual rail design instead to reduce the circuit area while increasing timing and error rate reduction. Another method is by employing an approximate triple modular redundancy (ATMR) method known as reduced precision redundancy [14], wherein the circuit is simplified but still performs the approximate logic function. This is used in tandem with a full precision module. One study surveyed ATMR design strategies. AC's importance for ATMR fault masking was underlined. Due to the interdependence of the three modules and the operating principle of ATMR, approximate circuits require a dedicated tool for problem generation [15].

Some innovations of the triple modular redundancy techniques have been developed to undermine the inherent weakness of the majority voter. A voter circuit algorithm has been developed wherein the voter mechanism is triplicated [16]. An additional technique is developed in the same study where the module, switching circuitry, voter and triple modular system is implemented to act as a reserve in the event of a malfunction in the original module.

GPUs are used to accelerate DNNs but can be affected by reliability issues and transient effects. A GPU-employed model was run and analysed to find vulnerabilities. The TMR technique was then used to selectively safeguard sensitive model parts [17]. The mitigation strategy reduces malfunction errors from

6.463% to 0.21%. The technique is compared to ABFT, DMR, and TMR (TMR). The proposed solution displays only 0.3035 percent overhead compared to these strategies while resolving 84.8 percent of the SDC mistakes in DenseNet201.

2.3. Parity checking

Parity checking is another method used to detect and correct soft errors. Parity codes protect memory data. Parity codes protect flip-flops from single-event upsets. Extra computation time to find parity affects the critical path badly. Parity generation requires an additional half-XOR gate per bit. Receiving parity checking costs half of an XOR gate in addition to a comparison with the stored parity. The parity flip-extra flop's expense is offset by protecting all other flip-flops. An OR tree with a defined cost penalty is required to extract fault information from parity bit clusters. Parity codes hide memory errors. Positioning methods prevent memory or flip-flop bit disruptions [18].

Thangavelu [19] analysed the software technique to safeguard safety systems from SEU malfunctions and serves as a means for continuous control system execution during SEU fault detection. This software solution increased the aviation system's reliability via automation of the detection and correction process with a control system.

The imposition of cost by parity check bits per word in single error correction can be downplayed by a significant amount in the model proposed by Reviriego et al. [20]. The read/write processes require the same access quantity as the usage of product code for bit checking and correction. Ergo, the model can be an alternative method for the employment of single bit error correction on per word parity bit memory.

2.4. Error correction code

Two-dimensional error correction codes (TECED) were used to reduce hardware costs and increase energy efficiency, and the design was evaluated based on system performance. Relative to the classic Hamming code, the TECED cuts most of the memory's area overhead and power usage by 55%. Horizontal-Vertical Parity And Diagonal Hamming (HVPDH) technique is suggested for the identification of errors ranging from 1 to 8-bit in any combination [21].

There have also been efforts to ease the design of error correction codes in light of the rapid development of newer and more compact memory blocks via the development of an Automated Error Correction Code Design Tool (AEDT) [22]. The tool facilitates automated design of block codes for memory protection and allows users to define the location of parity bits and execution time, allowing the code to flexibly function around the patterns of current and future errors.

The development of newer error correction codes has also been considered for the soft error mitigation of memristors [23], checking for error correction code along the diagonals for logic support. Alternatively, fault tolerance can be implemented through an algorithm in the case of a neural network protection [24]. Convolutional neural networks benefit from algorithm-based fault tolerance (ABFT) better than conventional error correction code (ECC) due to the inability of ECC to protect computational components.

Table 1 is summarising the analysis of the several mitigation techniques.

Table 1. Mitigation technique weaknesses.

Mitigation techniques	Weakness
Radiation hardening	The technique of radiation hardening involves the specific application of changes to the manufacturing design or materials used in a particular device. This technique may involve the usage of wide band-gap substrates, which requires deeper in-depth research on the quality of the crystals used. Furthermore, radiation hardening can apply to shielding electronics from radiation through the use of materials to lessen device exposure. Unfortunately, the type of shielding used may be situational and may not see use across electronics of differing applications.
Triple modular redundancy	Triple modular redundancy has the downside of a high area overhead due to the triplication of entirety of the selected circuitry. Furthermore, the majority voter itself is vulnerable to single event upset, further decreasing the reliability of this method as it can produce an erroneous result at output. While the voter can undergo mitigation to reduce its susceptibility to error, this will further increase the area overhead on already dense chip area.
Parity checking	Parity checking, while used as the basis for other techniques such as error correction code, is in and of itself a point of weakness as an error detection technique. Depending on the implementation of the parity checking in circuitry, the parity bit itself may become corrupted and therefore allow the error to go undetected. Furthermore, in the event of multiple bits being corrupted, the parity checking may fail to detect the error if the parity matches. The implementation of this technique would also necessitate the use of redundant bits that would be used to store the parity bits.
Error correction code	While this technique can be used to detect as well as correct errors, this is traded in for more expenses and slower processing. Therefore, error correction code may only be used in high reliability circuitry where unerring operation is paramount. Error correction code must also be implemented with a cyclic redundancy check to detect the data for errors, adding further impact on processing resources.

3. Methodology

Figure 1 shows the flowchart of methodology employed in the duration of the study.

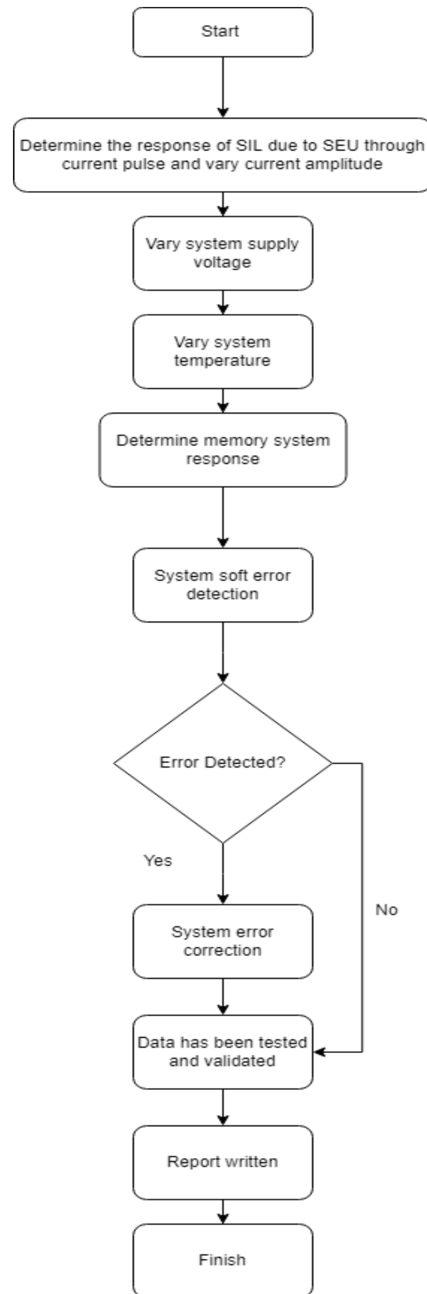

Fig. 1. Flowchart of methodology.

The study employs a double exponential current pulse model to simulate the transient current produced from a single event upset as shown below:

$$I(t) = \frac{Q_{total}}{\tau_\alpha - \tau_\beta}\left(e^{-t/\tau_\alpha} - e^{-t/\tau_\beta}\right) \quad (1)$$

where Q_{total} denotes the charge collected in the affected node

The junction collection time constant is denoted by τ_α while the initiation time constant for the ion track is represented by τ_β. The trapezoidal shape produced from the model has a fast rise time and slow fall time. The rise and fall times are 50ps and 164ps [25]. This model will define the shape and characteristics of the current pulse injected into the SIL circuit to simulate a transient current borne from a single event upset. Figure 2 shows the single rail with inverter latch configuration which consists of 2 NMOS transistors, 2 PMOS transistors and 2 inverters. The inverter loop is used to maintain the output, which is labelled OUT, in the event that the loop is disconnected from the main circuit. This will function as memory. The input of the interlocking inverters corresponds to the logic value produced at node (iii). The functionality of SIL in its entirety will be explained as follows. Suppose that both inputs A and B are at logic low. This causes transistors T1 and T2 to be turned ON, and T3 and T4 to be turned OFF, causing the logic level at node (iii) to be high. The signal is inverted as it passes through I2 in the inverter loop, and the logic level at node OUT will be low.

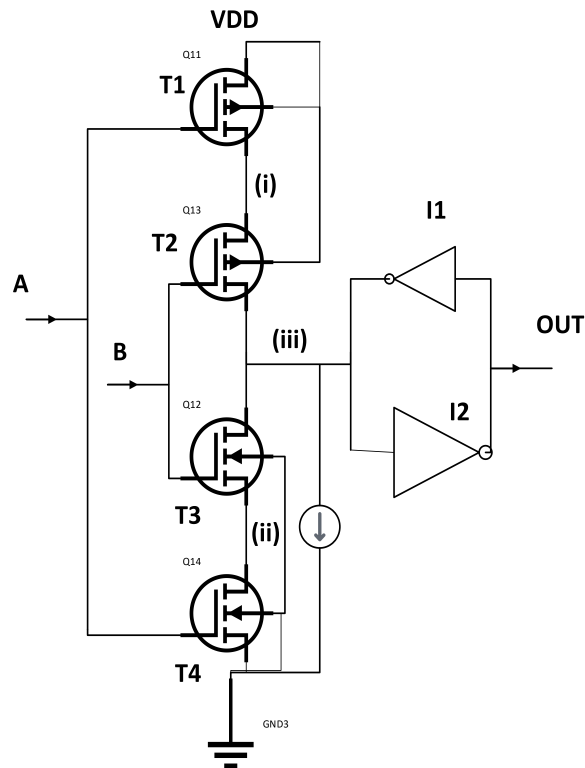

Fig. 2. Single rail with inverter latch configuration (SIL).

Alternatively, if both inputs A and B are high, transistors T1 and T2 are turned OFF and T3 and T4 are ON. This causes the current to be discharged to ground. Therefore, at node (iii) is low and inverted to high at node OUT. If either A or B are not equal, the system enters a hold state, wherein the node Out value is maintained by inverter I1 and I2. The vulnerable nodes were established as nodes (i), (ii) and (iii) as illustrated in Fig. 2.

The current is injected into these nodes until the original state is successfully overwritten. The results from the soft error injection were recorded and detailed as follows. The scenarios through which a soft error can occur are displayed in Figs. 3 and 4. These instances can be categorised as, the pulse produced not being significant and does not cause a state change as shown in Fig. 3 and 4 labels (a). Secondly, the pulse is more than 20% of the original value, does not cause state change but still propagate into the system as shown in Figs. 3 and 4 labels (b). Thirdly, the soft error causes the state to change, propagates in the system as shown in Figs. 3 and 4 labels (c).

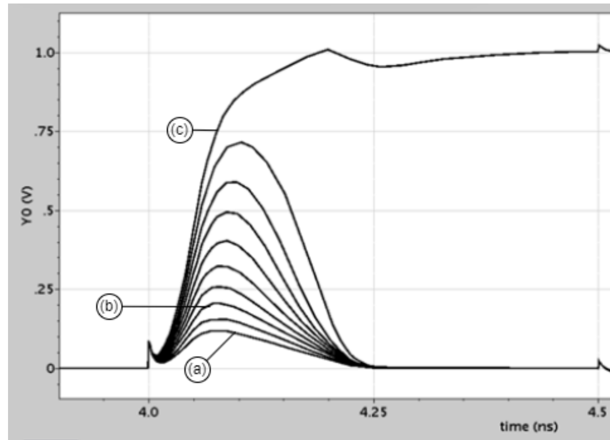

Fig. 3. Simulation of 0-1 error.

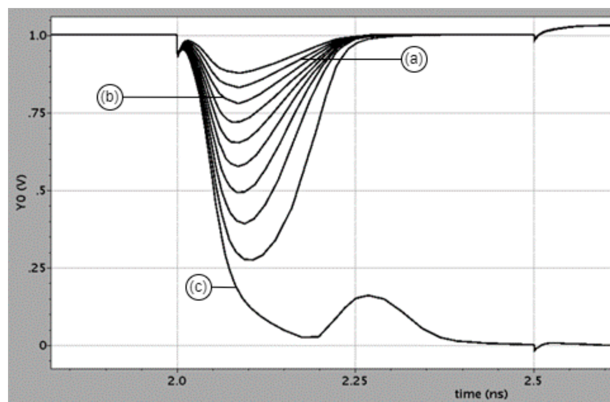

Fig. 4. Simulation of 1-0 error.

To compare the vulnerability of nodes with soft error, two presumptions of the of simulation must be made, that the current pulse is trapezoidal in shape with the aforementioned fall and rise times and that the current pulse affects the connection between PMOS and NMOS at the drain. The amount of critical charge is observed with variations to parameters which are that the voltage is ranged from 0.7 V to 1.5 V with increments of 0.1 V with the temperature set to 27ºC. For temperature variation the supply voltage is set to 1 V. The temperatures are ranged from -50 ºC to 200 ºC at the points -50 ºC, 0 ºC, 27 ºC, 125 ºC and 200 ºC.

3.1. Error detection and correction

Figure 5 shows the proposed latch consists of two shadow latches to provide the error signal and to feed the correct values in the event of soft error as in Fig. 5. The implementation is designed in Cadence environment. It is desired to design error detection and correction that can detect and correct errors in all the nodes. The delay in the shadow latch is needed to ensure the DATA propagates in the C-element and shadow latch arrive at the same time to XOR gate and MUX. The functionality of latch is explained as follows.

When DATA is ‘1’ and clock denoted at CLK is high, a ‘1’ appears at node 1, node 2 and node 3. A ‘1’ appears at the output of C-element, and at node 4, and propagates to multiplexer, MUX. No error is detected and therefore the value of S1 is selected and a ‘1’ appear at OUT.

When CLK is low, the previous DATA is maintained and propagate to the MUX and selected to appear to node OUT. In the event of soft error strikes the node in the C-Element, and when DATA is ‘1’ and CLK is high, a ‘1’ appears at node 1, node 2, node 3 and node 4. A ‘1’ is delayed and appears at node 6 and node 7. However, due to error, a ‘0’ appear at node 4. The values at node 4 and node 5 is compared and if there are not equal, a’1’ is produced at node 6 indicating an error is high. The value of S2 is selected and propagate to OUT.

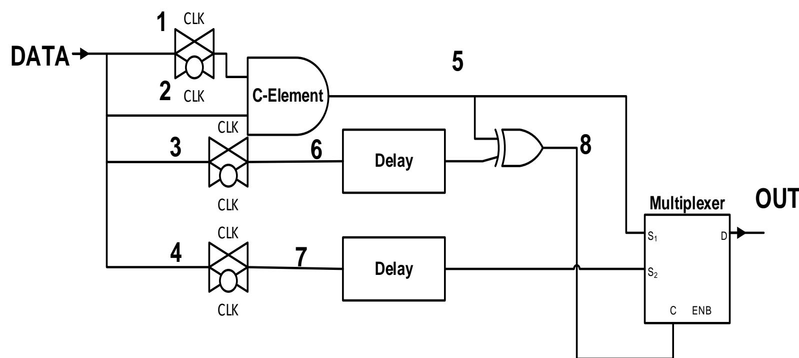

Fig. 5. Shadow latch model.

In order to test the proposed latch with error detection and correction capabilities, the system is re-designed by using Quartus II. In Cadence environment, the types of C-element used is SIL configuration and the shadow latches are used as shown in the rectangle of Fig. 5. Eight unit of latches of Fig. 5 is cascaded to form pipeline denoted by PL1 and PL2.

Soft error is injected at the nodes and the output is observed. In Quartus II however, the error is injected the output of C-element by using XOR gate. If the output of C-element is a ‘1’ and error of a ‘1’ is injected, the output, OUT C1 now is a ‘0’. This corresponds to the 1-0 error in Cadence. Similarly, If the output of C-element is a ‘0’ and error of a ‘1’ is injected, the output, OUT C1 now is a ‘1’. This corresponds to the 0-1 error in Cadence. The error detection and correction latch design modelled in Quartus is as shown in Fig. 6.

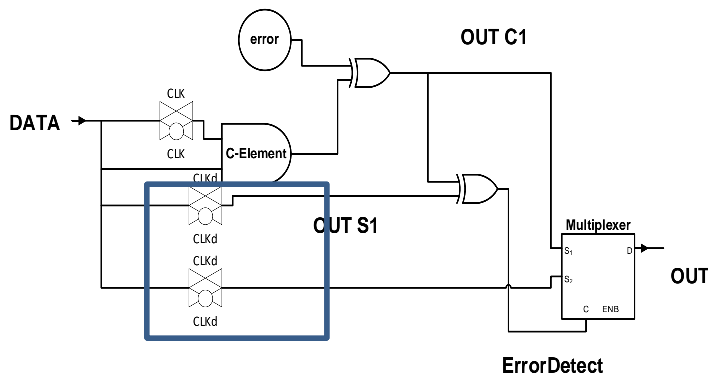

Fig. 6. Latch design with error detection and correction in Quartus environment.

3.2. Error detection and correction in adder system

In order to demonstrate functionality of the latches in an adder system, the latches are cascades in series to form 8 b-bit of latch and the output of 8-bit of latches are applied into parallel prefix adder (PPA) and in this case we use Brent Kung adder as shown in Fig. 7. Error is injected and the output of adder is observed.

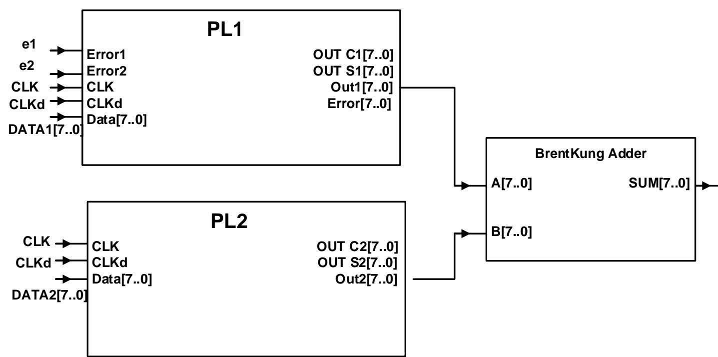

Fig. 7. Error detection and correction in adder system.

4. Methodology

This section involves the discussion of subsequent findings produced from the methodology of the study.

4.1. Transient response temperature and voltage supply

First, the error is injected in node (i),(ii) and (iii) of Fig. 2. The change of critical charge with respect to voltage supply change was graphed in Fig. 8. The graph shows that the critical charge for node (iii) at 1-0 and (i) increases at regular rate with as the voltage supply increases.

Therefore, the probability of the soft error affecting the aforementioned nodes would decrease as the voltage supply increases. From the voltage supply of 0.7 V

to 1.5 V, the critical charge for the node (i) increases by 173.8% and for node (iii) at 1-0 increases by 163.95%. The change in critical charge for the nodes (i) and (ii) is smaller from the voltage supply of 0.7 V to 0.9 V but then sees an increased rate of change from after 0.9 V onwards. Ultimately, the critical charge for the nodes (ii) and (iii) at transition 0-1 see a change of 142% and 130% respectively.

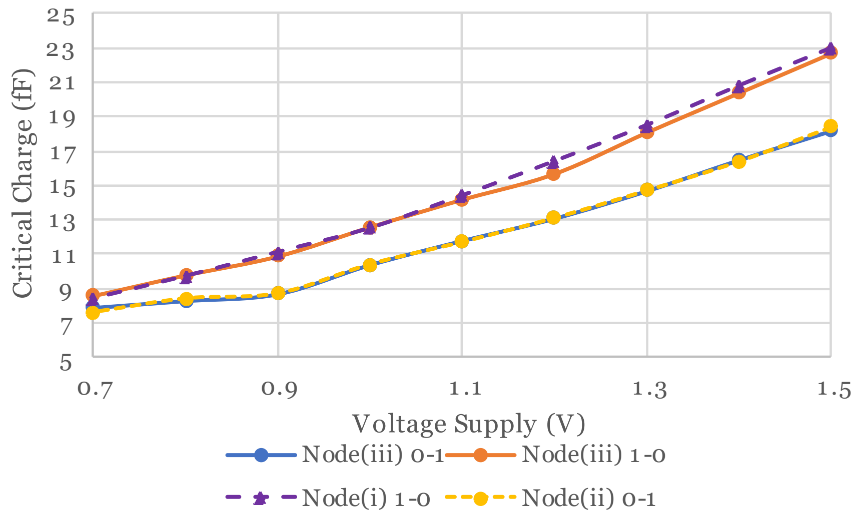

Fig. 8. Critical charge vs. voltage supply.

The graph for the critical charge vs temperature is illustrated as in Fig. 9. The critical charge decreases with the increase in temperature, making each node more vulnerable to soft errors. The node (iii) at logic changes from 0-1 and 1-0 see a change in critical charge of about 12.9% and 22.3% respectively. The nodes (i) and (ii) see a reduction of 20% and 13% respectively.

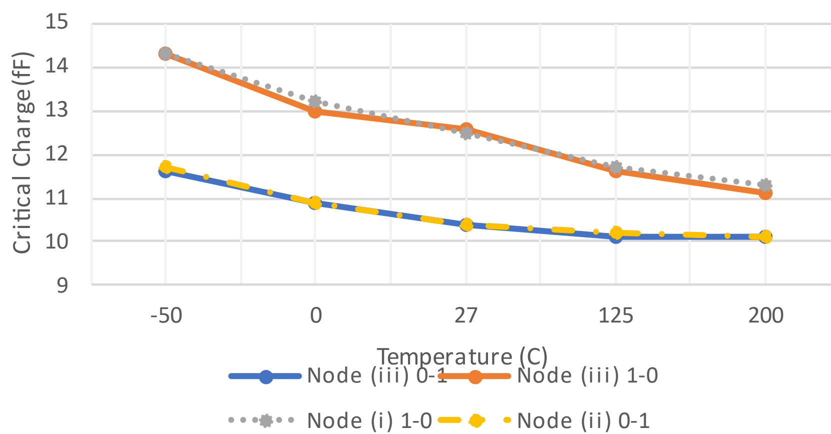

Fig. 9. Critical charge vs. temperature.

The standard deviation of the critical charge with variation to temperature and voltage supply was obtained to compare its effects towards soft error. Figure 10 shows the standard deviation with respect to voltage supply. The voltage supply imposes a higher deviation to the critical charge at all the node. However, the nodes (i) and (iii) for 1-0 are more susceptible to change in critical charge as the voltage supply increases. This shows that the drain of PMOS is more vulnerable than the drain of NMOS against soft errors with voltages change. The standard deviation for temperature increase has a reduced effect on the critical charge change as compared to the voltage supply. The standard deviation is lower than the voltage supply deviation across all nodes with the deviation at node (i) being incrementally higher than node (ii). This shows that the drain of PMOS is more vulnerable than the drain of NMOS against soft errors with temperature change.

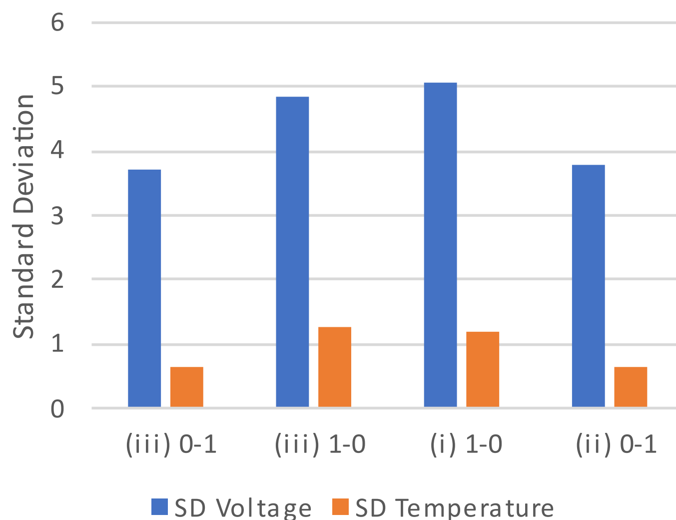

Fig. 10. Standard deviation of critical charge.

4.2. Error detection and correction

To demonstrate the functionality of latch against soft error, the latch as shown in Fig. 5 was simulated and inject soft error in the node. Figure 11 shows the simulation of fault free and 1-0 error. At T1, when CLK is high, and DATA is propagated to node 4. No error is detected as shown by node 6. Value in node 4 propagate to the OUT. At T2, the soft error is injected in C-element and When CLK is high, the output of C-element is change from high to low as shown by node 4 at point (i). Error is detected as shown by node 6 at point (ii). The corrected value is shown by OUT at point (iii).

Figure 12 shows the simulation of fault free and 0-1 error. At T1, when CLK is high, and DATA is propagated to node 4. No error is detected as shown by node 6. Value in node 4 propagate to the OUT. At T2, when CLK is high, the output of C-element is change from high to low as shown by node 4(i). Error is detected as shown by node 6(ii). The corrected value is shown by OUT (iii).

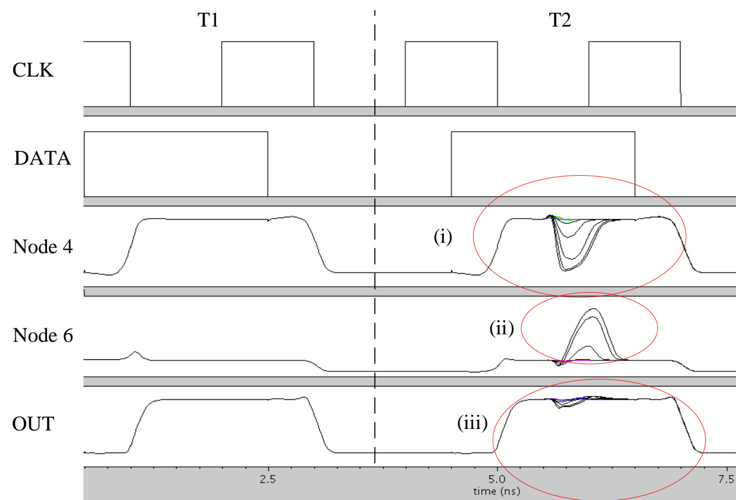

Fig. 11. Shadow latch model.

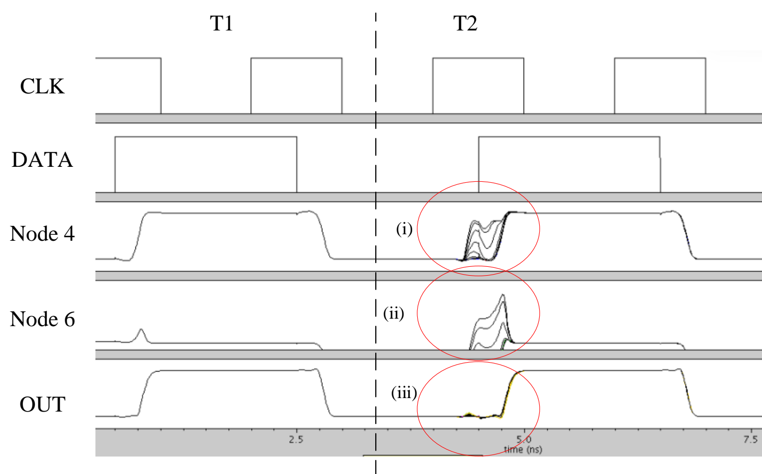

Fig. 12. Simulation of fault free and 0-1 error correction.

Figure 13 shows the simulation for (c) of fault free and 0-1 error. Voltage is changed from 0.8 V to 1.2 V. Delay is observed as voltage is reduced. At T1, when CLK is high, and DATA is propagated to node 4. No error is detected as shown by node 6. Value in node 4 propagate to the OUT. At T2, when CLK is high, the output of C-element is change from high to low as shown by node 4 (i). Error is detected as shown at node 6 (ii). The corrected value is shown by OUT (iii).

Latch as shown in Fig. 6 is injected with soft error. Figure 14 shows the simulation of fault free and 0-1 error and 1-0 error in Quartus. At time T1 when CLK is high, and DATA is propagated to OUTC1 and OUTS1. No error is detected and value in OUTC1 propagate to the OUT. At T2, when CLK is high, error is injected as shown in (i), the output of C-element is change from high to low as shown by (ii). Error is detected as shown by (iii). The output of shadow latch OUT S1 is selected and the correct output is propagate to Out as shown by (iv). At T3,

when CLK is high, error is injected as shown in (v), the output of C-element is change from low to high as shown by (vi). Error is detected as shown by (vii). The output of shadow latch OUT S1 is selected, and the correct output is propagate to Out as shown by (x).

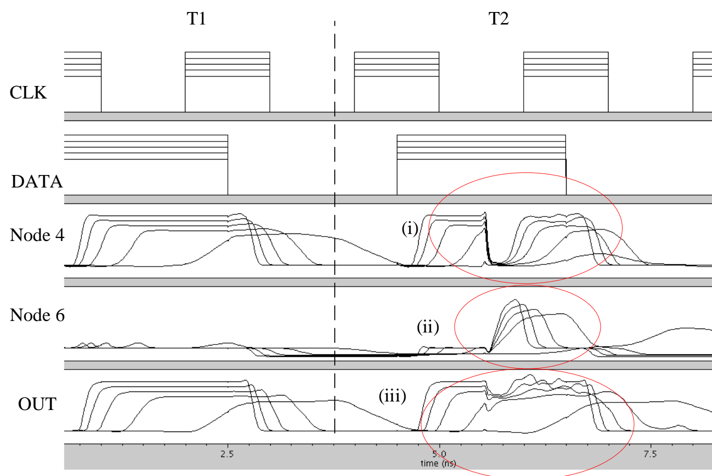

Fig. 13. Simulation of fault free and 0-1 error from 0.8 V to 1.2 V.

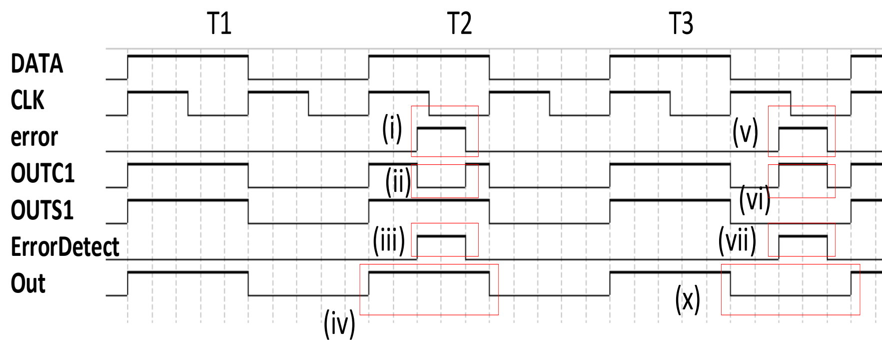

Fig. 14. Simulation of fault free and 0-1 error and 1-0 error in Quartus.

Adder system as shown in Fig. 7 is injected with soft error Figure 15 shows the simulation of the adder system equipped with error detection and correction capabilities At time T1, two eight-bit binary numbers, ‘00000001’ and ‘00000100’ are propagated from pipeline1 (PL1) and pipeline2 (PL2). No error is injected. The same values a ‘00000001’ and a ‘00000100’ propagate correctly at the output of C-element OUT C1 and OUT C2 and towards OUT1 and OUT 2. These values are correctly added and produced a ‘00000101’ at the SUM.

At time T2, two-eight bit binary numbers, a ‘00000010’ and a ‘00000101’ are propagated from pipeline1 (PL1) and pipeline2 (PL2). Error is injected as shown in (i) and Error signal as shown in (ii) indicate an error at the second bit of data. This is an example of 1-0 error. The output of c-element OUT C1 is now contained a ‘00000000’ instead of ‘00000010’as shown in (v). At this instance, the value from shadow latch is selected and the correct value is propagated to OUT1 as shown in

(iv). The correct values, a ‘00000010’ and a ‘00000101’ propagate correctly to the adder and correctly added to produce a ‘00000111’ at the SUM as shown in (v).

At time T3, two-eight bit binary numbers, a ‘00000100’ and a ‘00000111’ are propagated from pipeline1 (PL1) and pipeline2 (PL2). Error is injected at the most significant bit (MSB) as shown in (vi) and error signal as shown in (vii) indicate an error at the eight bit of data. This is example of 0-1 error. The output of c-element OUT C1 now contains ‘10000100’ instead of ‘00000100’ as shown in (viii). At this instance, the value from shadow latch is selected and the correct value is propagated to OUT1 as shown in (x). The correct values, a ‘00000100’ and a ‘00000111’ propagate correctly to the adder and correctly added to produce a ‘00001101’ at the SUM as shown in (xi).

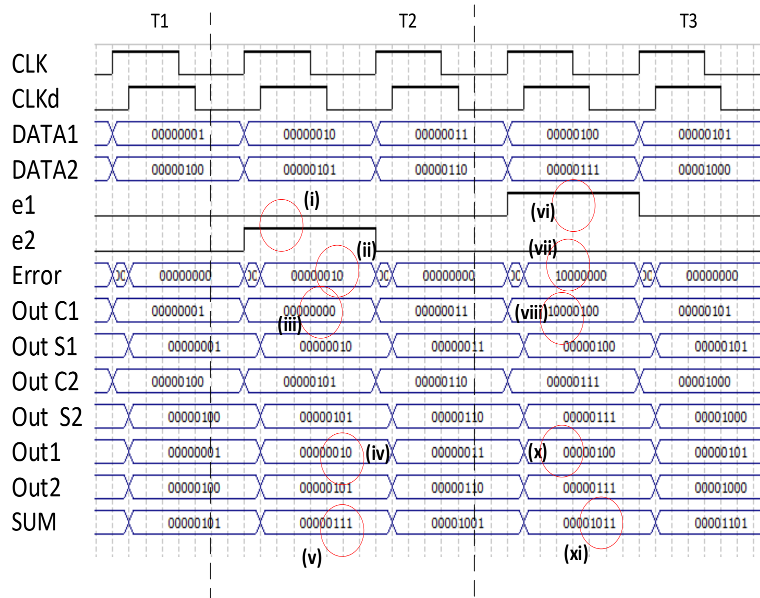

Fig. 15. Simulation of fault free and 0-1 error and 1-0 error in adder system.

5. Conclusion

The effects of soft errors in SIL configuration were modelled in Cadence. A current pulse of varying amplitude was injected into vulnerable node to achieve an observable change in logic value. The parameters under change were the temperature and voltage supply to observe their effect on the critical charge at each node. The temperature increase invoked a decrease in critical charge, therefore making each node more vulnerable to soft error. Alternatively, the critical charge increases linearly with increase in voltage supply. The paper has successfully proposed soft error mitigation by using shadow latches and the design were developed and implemented both in Cadence and Quartus environment. The simulation shows that soft errors can be detected when the circuit is operating from a range of 0.8 V to 1.2 V, as well as from operating temperatures of -50ºC to 200 ºC. The shadow latch was successful in detection of soft errors and in restoring the original shape of the logic signal.

Acknowledgement

The author would like to thank Ministry of Higher Education, Malaysia, Fundamental Research Grant Scheme (FRGS/1/2020/TK0/UNIMAS/02/11) and Universiti Malaysia Sarawak (F02/FRGS/2035/2020) for supporting this work.

Nomenclatures

fD	Collection charge for inner well NMOS
fV	Variation in circuit bias from 0 V to the value of supply voltage
L_G	Gate length
$Prob_N$	Probability of soft error to cause flip
$Qcoll$	Collection charge

Greek Symbols

ρ_{ENV}	Atmospheric neutron cross section per unit area for N and P type drains
τ_f	Fall time
τ_r	Rise time

Abbreviations

CLK	Clock
SEE	Single event effect
SET	Single event transient
SEU	Single event upset
SIL	Single rail with inverter latch
SRAM	Static random access memory
TMR	Triple modular redundancy

References

1. Wang, F.; and Agrawal, V.D. (2008). Single event upset: An embedded tutorial. *Proceedings of the* 21*st International Conference on VLSI Design* (*VLSID* 2008), Hyderabad, India, 429-434.
2. Claeys, C.; and Simoen, E. (2013). *Radiation effects in advanced semiconductor materials and devices*, 53(9). Springer Berlin Heidelberg.
3. Baumann, R.C. (2001). Soft errors in advanced semiconductor devices-part i: the three radiation sources. *IEEE Transactions on Device and Materials Reliability*, 1(1), 17-22.
4. Abe, S.-i.; Ogata, R.; and Watanabe, Y. (2014). Impact of nuclear reaction models on neutron-induced soft error rate analysis. *IEEE Transactions on Nuclear Science*, 61(4), 1806-1812.
5. Mori, H.; Uemura, T.; Matsuyama, H.; Abe, S.I.; and Watanabe, Y. (2015). Critical charge dependence of correlation of different neutron sources for soft error testing. *Proceedings of the* 2015 *IEEE International Reliability Physics Symposium*. Monterey, CA, USA, 2C.3.1-2C.3.5.
6. Pown, M.; and Lakshmi, B. (2020). Investigation of radiation hardened tfet sram cell for mitigation of single event upset. *IEEE Journal of the Electron Devices Society*, 8, 1397-1403.

7. Díez-Acereda, V.; Khemchandani, S.L.; Del Pino, J.; and Mateos-Angulo, S. (2019). RHBD techniques to mitigate SEU and SET in CMOS frequency synthesizers. *Electronics*, 8(6), 690.
8. Schrape, O.; Breitenreiter, A.; Schulze, C.; Zeidler, S.; and Krstic, M. (2021). Radiation-hardness-by-design latch-based triple modular redundancy flip-flops. *Proceedings of the* 2021 *IEEE* 12*th Latin America Symposium on Circuits and System* (*LASCAS*), Arequipa, Peru , 1-4.
9. Chen, Z.; Zhao, Y.; Lu, J.; Liang, B.; Chen, X.; and Li, C. (2022). TECED : a two-dimensional error-correction codes based. *Electronics*, 11(10), 1638.
10. Li. P.; Zhen, L.; Li, X.; Yang, J.; Zhang, H.; Sun, Y.; Mei, B.; LV, H.; Mo, R.; Yu, Q.; and Tang, M. (2021). Radiation hardness assurance of single event effects on components for space application. *Proceedings of the* 2021 4*th International Conference on Radiation Effects of Electronic Devices*, *ICREED* 2021, Xi'an, China, 1-6.
11. Velazco, R.; McMorrow, D.; and Estela, J. (2019). *Radiation effects on integrated circuits and systems for space applications*. Springer.
12. Hara, K.; Aoyagi, W.; Sekigawa, D.; Iwanami, S.; Honda, S.; Tsuboyama, T.; Arai, Y.; Kurachi, I.; Miyoshi, T.; Yamada, M.; and Ikegami, Y. (2019). Radiation hardness of silicon-on-insulator pixel devices. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators*, *Spectrometers*, *Detectors and Associated Equipment*, Volume 924, 426-430.
13. Sielewicz, K.M.; Rinella, G.A.; Bonora, M.; Giubilato, P.; Lupi, M.; Rossewij, M.J.; Schambach, J.; and Vanat, T. (2017). Experimental methods and results for the evaluation of triple modular redundancy SEU mitigation techniques with the Xilinx Kintex-7 FPGA. *Proceedings of the* 2017 *IEEE Radiation Effects Data Workshop* (*REDW*), New Orleans, Los Angeles, USA, 1-4.
14. Arifeen, T.; Hassan, A.S.; and Lee, J.-A. (2019). A fault tolerant voter for approximate triple modular redundancy. *Electronics*, 8(3), 332.
15. Arifeen, T.; Hassan, A.S.; and Lee, J.-A. (2020). Approximate triple modular redundancy: a survey. *IEEE Access*, 8, 139851-139867.
16. Santhiya, M.; Saranya, S.; Vijayachitra, S.; Lavanya, C.B.; and Rajarajeswari, M. (2021). Application of voter insertion algorithm for fault management using triple modular redundancy (TMR) technique. *Proceedings of the* 2021 *Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks* (*ICICV*), Tirunelveli, India, 578-583.
17. Adam, K.; Mohamed, I. I.; and Ibrahim, Y.A. (2021). Selective mitigation technique of soft errors for DNN models used in healthcare applications: DenseNet201 case study. *IEEE Access*, 9, 65803-65823.
18. Venkatesha S.; and Parthasarathi R. (2022). A survey of fault models and fault tolerance methods for 2d bus-based multi-core systems and TSV based 3D NOC. *arXiv preprint arXiv*:2203.07830. Retrieved June 21, 2022, from https://arxiv.org/ftp/arxiv/papers/2203/2203.07830.pdf
19. Thangavelu, K. (2022). Control system software execution during fault detection. *Proceedings of the* 2022 6*th International Conference on Intelligent Computing and Control Systems* (*ICICCS*), Madurai, India, 1-5.

20. Reviriego, P.; Pontarelli, S.; Maestro, J.A.; and Ottavi, M. (2013). Reducing the cost of single error correction with parity sharing. *IEEE Transactions on Device and Materials Reliability*, 13(3), 420-422.
21. Raha, P.; Vinodhini, M.; and Murty, N.S. (2017). Horizontal-vertical parity and diagonal hamming based soft error detection and correction for memories. *Proceedings of the* 2017 *International Conference on Computer Communication and Informatics* (*ICCCI*), Coimbatore, India, 2-6.
22. Li, J.; Reviriego, P.; Xiao, L.; and Wu, H. (2021). Protecting memories against soft errors: the case for customizable error correction codes. *IEEE Transactions on Emerging Topics in Computing*, 9(2), 651-663.
23. Leitersdorf, O.; Perach, B.; Ronen, R.; and Kvatinsky, S. (2021). Efficient error-correcting-code mechanism for high-throughput memristive processing-in-memory. *Proceedings of the* 2021 58*th ACM/IEEE Design Automation Conference* (*DAC*), San Francisco, California, USA, 199-204.
24. Zhao, K.; Di, S.; Li, S.; Liang, X.; Zhai, Y.; Chen, Y.; Ouyang, K.; Cappello, F.; and Chen, Z. (2021). FT-CNN: Algorithm-based fault tolerance for convolutional neural networks. *IEEE Transactions on Parallel and Distributed Systems*, 32(7), 1677-1689.
25. Cha, H.; and Patel, J.H. (1993). A logic-level model for/spl alpha/-particle hits in CMOS circuits. *Proceedings of the* 1993 *IEEE International Conference on Computer Design ICCD'*93, Cambridge, Massachusetts, USA, 538-542.