Modified Null Convention Logic Pipeline to Detect Soft Errors in Both Null and Data Phases

F. K. Lodhi and O. Hasan
Sch. of Elect. Engg. and Comp. Sc.
National University of Sciences and Technology (NUST)
Islamabad, Pakistan
{faiq.khalid, osman.hasan}@seecs.edu.pk

S. R. Hasan
Department of Electrical and Computer Engineering
Tennessee Technological University
Cookeville, TN, USA
shasan@tntech.edu

F. Awwad
Faculty of Information Technology
United Arab Emirates University
Al-Ain, UAE
f_awwad@uaeu.ac.ae

***Abstract*—Glitches due to soft errors can act as a severe deterrent to asynchronous circuit operations. To mitigate soft errors in quasi delay insensitive (QDI) asynchronous circuits, built-in soft error correction in NULL convention logic (NCL) has been introduced [9]. However, this technique cannot detect errors during the NULL phase of NCL pipeline, and also cannot avoid error propagation into the pipeline after its detection. This paper provides a modified approach to overcome these limitations with, on average, comparable power and latency costs. This work also analyzes the temperature variation effects on latency and power consumption of the proposed design. The modified NCL pipeline is implemented in IHP 90nm CMOS technology and analyzed under various operating temperatures. It is found that the proposed design survives well in the worst case operating temperatures and does not propagate soft errors.**

I. INTRODUCTION

Due to shrinkage in modern deep sub-micron technologies, the secondary neutron particles from cosmic rays have become a major source of inducing transient errors in semiconductors devices [1]. These transient errors may corrupt data and control signals, which can cause temporary or permanent soft errors (SE) in both synchronous and asynchronous digital systems. Different techniques have been devised to mitigate soft errors, most of which are categorized as either introducing redundancy (e.g. triple modular redundancy (TMR)) or error correction codes (ECC) [2]. However, these conventional techniques cannot be readily applied to asynchronous systems, because of their inherent combinational feedback signals.

Recently, Quasi Delay Insensitive (QDI) asynchronous circuits [3, 4] are widely used. QDI can overcome the traditional asynchronous circuit limitations, such as their event driven nature and the data control delay mismatch problem due to the combinational feedback circuits. However, to the best of our knowledge, very limited support for detecting and correcting soft errors [2] in QDI circuits exists in the open literature so far. Monnet et al. [2, 5] observed that soft errors can affect the threshold voltages that lead to timing non-idealities. They coined a term "sensitive time", to evaluate the sensitivity of the QDI asynchronous circuit against soft errors. Single event upset (SEU) tolerant QDI circuits have also been proposed in [6, 7]. But these solutions compromise the performance of the design, which makes them unsuitable for high speed applications. It has been proposed in [8] that the encoding redundancy of the QDI can be utilized for checking validity of the data. It allows the use of existing hardware but while detecting SE many transient disruptions occur that result in an invalid data token value [8]. This idea was further extended by Kaung et al [9] by presenting a built-in soft error correction (BISEC) technique using the handshaking protocol and dual-rail encoding in QDI circuits. This technique has two major limitations. Firstly before any action is taken against the detected fault, the fault may propagate in the NCL pipeline [10]. Secondly, this technique is not able to detect the soft errors in the NULL phase.

The foremost goal of this paper is to resolve the above mentioned issues in Kuang et al's work. We propose a modified built-in soft error correction (BISEC) technique that overcomes the fault propagation and is able to detect the soft errors in the NULL phase. It has been experimentally demonstrated that the soft errors in the combinational block, which could have gone undetected in the existing technique, are detected properly with our proposed architecture. The introduction of modified reset mechanism enables the proposed design to eliminate the threat of the propagation of 'detected' faults in the NCL pipeline. Moreover, an analysis is performed for temperature variations on timing parameters (NULL-DATA latency mismatch) and power consumption. The electrical simulation results show that the proposed modified BISEC technique offers more robustness against NULL-DATA latency mismatch while attaining the same performance and power consumption like the conventional BISEC technique.

The rest of the paper is organized as follows: Section II provides the implementation details of the proposed modifications into the built-in soft error detection and correction scheme. In Section III, we present the experimental

results of proposed architecture along with comparisons with the existing BISEC technique. Finally, Section IV concludes the paper.

II. PROPOSED ARCHITECTURE

This section explains the methodology and architecture of our proposed design in detail. The proposed design is an extended version of the built-in soft error technique by Kuang et al. [9] which is implemented using the NCL pipeline. A traditional NCL pipeline architecture has three main blocks: NCL based registers, the combinational logic block and a completion detection scheme as shown in Fig. 1 [9]. In this architecture, two stages interact via a request and acknowledgement protocol. In Fig. 1, request and acknowledgement are based on Ki and Ko signals. This pipeline works in two different states: DATA phase and NULL phase. DATA phase represents the DATA0 (0,1) and DATA1 (1,0) while NULL represents the NULL (0,0). This architecture detects soft errors by utilizing the dual-rail encoding and checks whether the circuit generates the prohibited "11" state. The soft error detection block identifies this state by providing a "0". For correcting such errors an intermediate register (middle register in Fig. 1) is used along with a reset circuitry. When a soft error is detected in the combinational block, the input data is blocked by setting the output of the reset circuit 'q' equal to 0. An inverted delayed version of acknowledge, Ki1 in Fig. 1, is used to block the input data

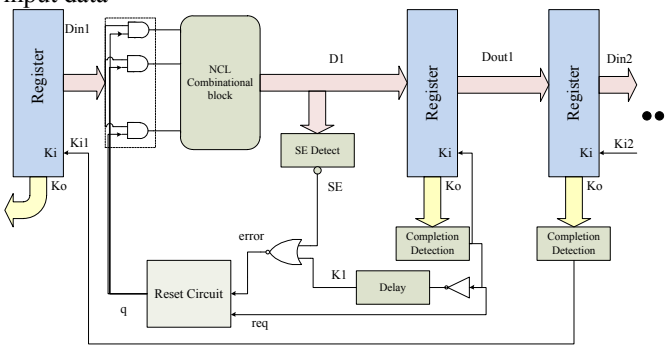

Figure 1. Built-in Soft Error Correction and Detection Scheme [9]

One of the major limitations of this circuit (shown in Fig. 1) is that it fails to detect errors during the NULL phase. Hence, due to soft errors it sends either "01" or "10" if the actual output is `00'. Such undetected errors can result in sending data in a NULL phase and in turn lead to system malfunctioning. To alleviate this problem, we propose a modified BISEC architecture, which is shown in Fig. 2. This proposed design includes two new blocks and a unique utilization of SE block output to reset the right most register. The state detection block is used to determine whether the system is in the DATA phase or the NULL phase. Based on the state, then it sends STATE signal with the delay of a 2-input AND gate and the combinational block to match the data timing of D1. This way, the system can foresee what values are invalid at any given time. As is the case with the conventional architecture [9], the premise of the design is that only the combinational block is prone to soft errors.

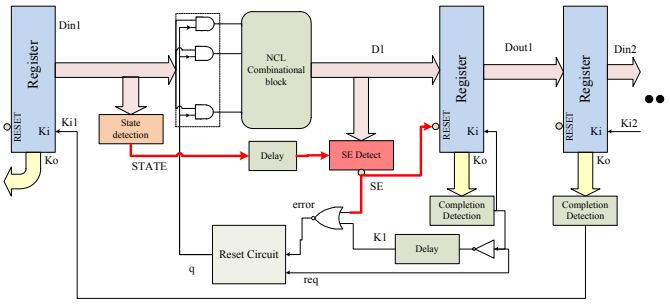

Figure 2. Modified Implementation of Null Convention Logic

As pointed out in Section I, the second main limitation of the architecture of Fig. 1 is that the detected fault can propagate to the output. For example, even if a fault is detected by the soft error block in Fig. 1 the output Din2 can still propagate such a fault to the next stage in the pipeline. Hence, the error detection result would be required to propagate throughout the pipeline to discard the data in error state. This limitation leads to a performance penalty which is equivalent to the number of stages in the pipeline and could even result in transferring the wrong data to the receiver end. To overcome this limitation, a novel approach to reset the NCL registers is used in the proposed architecture. As soon as the SE detection block detects an error at the output of the combinational block, it forbids the input from proceeding into the circuit and again starts computing the same data. Moreover, the architecture resets the intermediate register and thus blocks the corrupted data from going out at the output Din2 in Fig. 2. A NULL appears at the output during this phase. Fig. 3 illustrates the above-mentioned optimization in the proposed architecture by showing that upon detecting a fault the proposed architecture gives a NULL at the output during the re-computational phase while the conventional architecture propagates the faults.

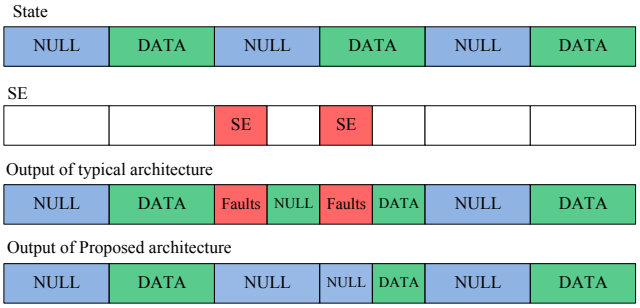

Figure 3. Output comparison of two architectures

As described earlier, the proposed architecture has an additional State Detection block and a modified SE Detection unit, which are described below:

A. State Detection Block:

The state detection block determines the state of the system by using the input data bits. This block sets the STATE signal low when system is in the NULL phase. This block is implemented with 4 input OR gates equivalent (using three 2 input OR gates).

B. Soft Error Detection and Correction

The difference between the proposed and the existing soft error correction and detection schemes lies in detecting the soft errors during the NULL phase. SE detection block detects the soft errors on the basis of the STATE signal. When the system is in the DATA phase and the combinational block generates the output "11" (INVALID DATA) or it gives "00" (NULL) then the SE detection block detects the soft error by setting the SE signal low. Similarly, when the system is in the NULL phase and the combinational block does not generate the output "00" (NULL) then the SE detection block detects the soft error by setting the SE signal low. Hardware implementation of this block is based on the truth table given in Table 1 where A0 and A1 are the active low NULL detectors of the respective bits. The gate level implementation of this block is shown in Fig. 4.

TABLE I. TRUTH TABLE OF SE DETECTION BLOCK

Truth Table of SE Detection Block							
STATE	***A0***	***A1***	***SE***	***STATE***	***A0***	***A1***	***SE***
0	*0*	*0*	*1*	*1*	*0*	*0*	*0*
0	*0*	*1*	*0*	*1*	*0*	*1*	*0*
0	*1*	*0*	*0*	*1*	*1*	*0*	*0*
0	*1*	*1*	*0*	*1*	*1*	*1*	*1*

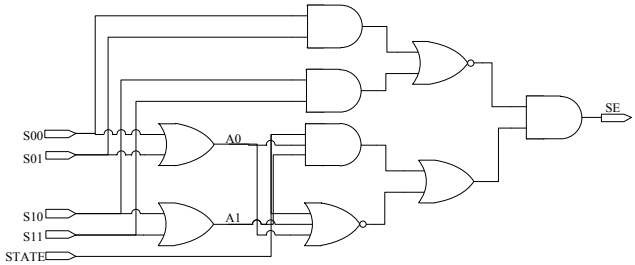

Figure 4. Gate level implementation of SE Detection block

III. EXPERIMENTAL RESULTS AND DISCUSSION

This section analyzes the simulation results of the proposed architecture using 90nm CMOS IHP technology. For the gate level implementation, static CMOS logic has been used. Transistor sizing in different threshold gates, except the feedback inverter of each gate, has been done using the traditional mobility method. The feedback inverter is sized on the basis of gate capacitances of the feedback nMOS and pMOS transistors. In order to model the soft error in this architecture, we have implemented the trapezoidal approximation of current equation provided in reference [9]. I_{max} for this trapezoidal approximation is found to be 425uA for IHP 90nm CMOS technology. Therefore, to induce soft errors in our proposed architecture, this current model is included in threshold gates that are utilized in combinational logic of Fig. 2. Experimental results include functional verification and also the temperature effects on the two important metrics of the asynchronous circuits, i.e., the latency, and the average power consumption of the entire system.

Fig. 5 illustrates the functional correctness of the proposed architecture. Due to space limitations, we are showing only one of the two bits of data. In Fig. 5, I00, I01 and O00, O01 are the input and output bits, respectively, whereas SE and Ki1 are the soft error detection and acknowledgement signals, respectively. Soft error in both NULL and DATA phase are shown in Fig. 5. When soft error occurs in the DATA phase, it is detected by setting SE low as shown in Fig. 5. When the error is detected then NULL appears at the output instead of the corrupted data, hence showing the functional correctness of our design. Fig. 5 shows that after the re-computation of data, as the correct data appears at the output, it sends the data acknowledge signal by setting Ki1 to low. Soft errors in the NULL phase are also handled in a similar fashion.

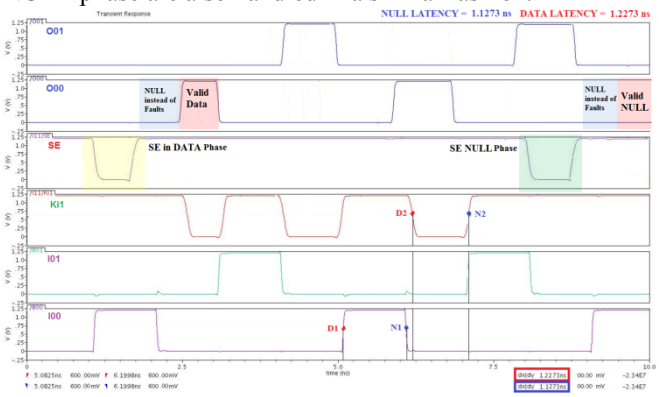

Figure 5. Simulation results of proposed Architecture

The latency of the proposed architecture is described using two ways in Fig. 5. The NULL latency is the time difference between N1 (when NULL phase is started) and N2 (when NULL acknowledgement (Kix = 1) is received; where x represents the number of pipeline stages). While the DATA latency is the time difference between D1 (when DATA is started) and D2 (when DATA acknowledgement (Kix = 0) is being received). Figure 6 shows that the DATA latency is 1.18 ns as compared to 1.12 ns for NULL latency, which is a difference of approximately 5%. Hence, a physical designer engineer should also consider this variation into account for timing closure.

A. Temperature Variations in Latency

In modern DSM technologies, considerable temperature variations in the nodes have become inevitable [11], therefore effect of temperature variation is also studied. In the threshold gates (logic family [9] used to make registers of NCL) transistors do not follow typical sizing of complementary logic due to the feedback transistors. Therefore, threshold gates have different latencies for different input/output combinations. This results in different propagation delays for low to high and high to low transitions. In turn, this results in different values for NULL and DATA latencies. This difference may be catastrophic if no prior analysis is performed, because it can even lead to misinterpretation of NULL phase and DATA may be sent to the next pipeline stage when it is not ready to receive it. As timing parameters may vary with temperature, therefore, we performed our simulations for wide temperature range and observed the difference in DATA and NULL latencies. We varied the temperature from -80 to 80 ^{0}C for both architectures (the

conventional one and the proposed). The effects of temperature on DATA and NULL latencies of both architectures are shown in Fig. 6, where it can be seen that both the architectures have almost equal latency and it increases linearly with the increase in temperature. Ideally we try to keep the difference in the latency close to zero. It is observed that for the conventional architecture [9], the difference between the NULL latency and the DATA latency is zero between 20 to 40 ^{0}C but for the proposed architecture, this range is wider, i.e., 15 to 45^{0}C, as shown in Fig. 6.

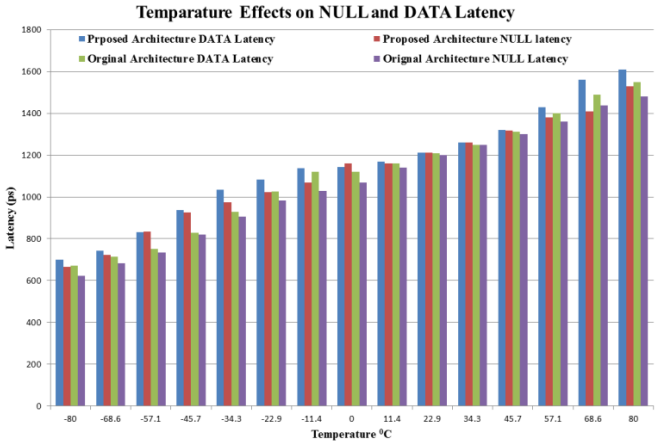

Figure 6. Temperature Effect on NULL and DATA Latency

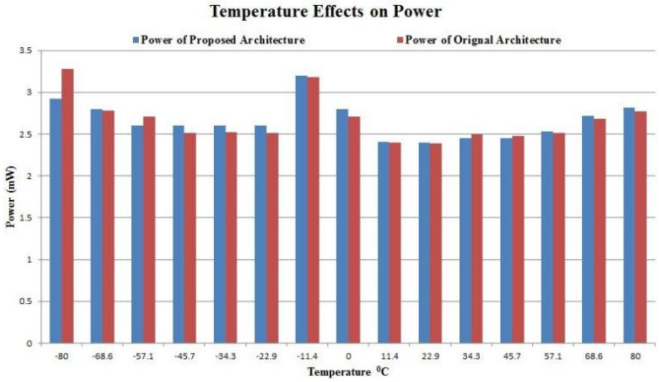

Figure 7. Temperature effect on Power consumption

B. Temperature Variations in Power Consumption

We utilized the worst case conditions for our power estimation. The worst case occurs when the data traffic is fastest (switching factor is highest), which is possible when the receiver is ready to receive the data whenever the sender sends it. Fig. 7 shows our simulation results for this case. It is observed that the average power consumption with temperature varying from -80 to 80 ^{0}C for both the architectures is almost equal. The power consumption is also affected by temperature variations as is the case with latency. Fig. 7 shows that the power consumption decreases with an increase in the temperature up to 25^{0}C and then it starts to increase. Both architectures follow the same trend. It is also observed that both architectures consume the lowest power within 10 – 40 ^{0}C range.

IV. CONCLUSION

Soft errors due to neutron strikes are becoming a serious threat in asynchronous circuits due to the significant reduction in the size of transistor in modern DSM technologies. QDI asynchronous circuits are widely used due to their built-in error detection property. In this paper, we proposed a modified NCL pipeline architecture to detect the soft errors in both the NULL and DATA phases in QDI asynchronous circuits in contrast to BISEC [9], which can only detect errors in the DATA phase. A modified soft error detection block is conceptualized, designed and implemented, and electrical simulations have proven the validity of the design in both NULL and DATA phases. It is also demonstrated that unlike the conventional NCL pipeline, the proposed architecture avoids the fault propagation as when error is detected it sends out NULL data instead of sending out corrupt data bits. The proposed architecture is implemented in IHP 90nm CMOS technology and analyzed and compared with the conventional architecture, under the worst case conditions. This analysis shows that this architecture operates in the same range of temperature, almost has the same latency and also consumes, on average, same amount of power as the conventional technique. Further investigation on its effect on other process variation parameters is under study.

REFERENCES

[1] K. J. Kulikowski, M. G. Karpovsky, A. Taubin, Zhen Wang, "Concurrent Fault Detection for Secure QDI Asynchronous Circuits", 2nd Workshop on Dependable and Secure Nanocomputing, 2008, pp. 1-6

[2] Y. Monnet, M. Renaudin, R. Leveugle, "Asynchronous circuits sensitivity to fault injection," in Proc. 10th IEEE International On-Line Testing Symposium, 2004, pp.121-126.

[3] C. Junchao, C. Kwen-Siong, G. Bah-Hwee and J.S Chang, "An ultra-low power asynchronous quasi-delay-insensitive (QDI) sub-threshold memory with bit-interleaving and completion detection," International NEWCAS confrence, 2010, pp. 117-120.

[4] Z. Rong, C. Kwen-Siong, G. Bah-Hwee and J.S Chang, "Quasi-delay-insensitive compiler: Automatic synthesis of asynchronous circuits from verilog specifications ", Int. Midwest Symp. on circuit and syst., 2011, pp. 1-4.

[5] Y. Monnet, M. Renaudin, R. Leveugle, "Asynchronous circuits transient faults sensitivity evaluation," Design Automation Conference , 2005, pp. 863-868.

[6] W. Jang and A. J. Martin, "SEU-Tolerant QDI circuits", in Proc. IEEE Int. Symp. Asynchronous Circuits Syst., 2005, pp. 156–165.

[7] K. T. Gardiner, A. Yakovlev and A. Bystrov, "A C-Element Latch scheme with increased Transient Fault Tolerance for Asynchronous Circuits," in Proc. 13th IEEE Int. On-Line Test. Symp., 2007, pp. 223–230.

[8] Hazucha and C. Svensson, "Impact of CMOS technology Scaling on the Atmospheric Neutron Soft Error Rate," IEEE Trans. Nucl. Sci., vol. 47, no. 6, 2000, pp. 2586–2594.

[9] W. Kuang, P. Zhao, J. S. Yuan, and R. F. DeMara, "Design of Asynchronous Circuits for High Soft Error Tolerance in Deep Submicrometer CMOS Circuits," IEEE Trans. On VLSI vol. 18. 2010, pp. 410-422.

[10] S. C. Smith and J Di, "Designing asynchronous circuits using NULL convention logic (NCL)," Morgan & Claypool, 2009. ISBN-13: 978-1598299816

[11] S. R. Hasan, B. Pontikakis and Y. Savaria, "An All-digital Skew-adaptive Clock Scheduling Algorithm for Heterogeneous Multiprocessor Systems on Chips (MPSoCs)," Int.Symp. of Circuits and Syst., 2009, pp. 2501-2504.