

Soft Error Analysis and Optimizations of C-elements in Asynchronous Circuits

Balaji Vaidyanathan, Yuan Xie, N. Vijaykrishnan
CSE Department
The Pennsylvania State University
University Park, PA 16802, USA
Email: {bvaidyan, yuanxie, vijay}@cse.psu.edu

Hao Zheng
CSE Department
University of South Florida
Tampa, FL 33620, USA
Email: {zheng}@cse.usf.edu

Abstract— Control circuit in an asynchronous design is comprised mostly of Muller C-elements. Previous work has concentrated on power, performance, and area issues of various CMOS implementations of the C-element. In this paper we carried out a thorough soft error analysis of four popular CMOS implementations of the Muller C-element. It shows that SIL implementation has the best soft error resilience. Optimization techniques to improve the soft error resilience of C-elements are proposed. Results show 2x improvements in critical charge by using our techniques. Finally, analysis of power, performance, and area tradeoff is carried out for the optimized C-element.

I. INTRODUCTION

WITH technology scaling the clock signal reach-ability across the chip and clock power have become major issues in synchronous design. Asynchronous designs operate without global clocks, and they display several advantages over the synchronous design: there are no clock related skew or power issues; they offer average case performance; there are no global timing issues; they are less affected by technology and process [1]. Currently, asynchronous designs are adopted in various small and large circuits [2]. Asynchronous circuits are broadly classified as Delay insensitive (DI), Quasi Delay Insensitive (QDI) and Speed Independent (SI) circuits [1]. These Asynchronous circuits have unbounded gate delay assumption, which provides them with inherent tolerance to a broad class of delay faults. However, in certain cases, faults occurring in asynchronous circuits can have catastrophic effects due to the event ordering constraints and might cause circuit failure and can sometimes lead to deadlock [3].

Another issue with technology scaling is the increased vulnerability of the CMOS circuits to soft errors due to particle strike [19] attributed to the decreasing voltage, nodal capacitance, and high density of devices. Soft error tolerance schemes at device, circuit, architecture and algorithm levels have been extensively researched for clocked systems [22], [23]. In comparison, transient fault sensitivity of asynchronous circuits has been sparingly researched.

Control circuit in an asynchronous circuit is comprised mostly of Muller C-elements [11]. In this paper, we present a soft error analysis of C-elements in asynchronous circuits and propose methods to harden these circuits against soft errors.

The rest of the paper is organized as follows: Section 2 reviews related work; Section 3 introduces the asynchronous circuits and Muller C-element and their soft error vulnerability is examined in Section 4; Section 5 provides various methods to enhance the soft error tolerance of Muller C-element. Finally we summarize our work in section 6.

II. RELATED WORK

Work in [7], [8] has dealt with the unidirectional and asymmetric errors occurring in the asynchronous buses. They use Error Detection and Correction (EDC) codes to detect transmission completion, error occurrences and correct them. [6] shows the VLSI implementation of asynchronous decoder for unordered codes for DI communication. A more robust EDC codes for DI and Semi Delay Insensitive (SDI) circuits is provided in [9]. The work presented in [5] deals with converting circuit errors and invalid tokens to deadlock and also presents layout techniques to mitigate delay faults in QDI circuits. In [4] transient fault sensitivity of the QDI circuits is analyzed using logic level fault models and circuit duplication is proposed to eliminate single transient faults. In [5] the authors have used duplicated nodes along with double checking to prevent Single Event Upset (SEU) in QDI. Apart from the asynchronous design, Muller C-elements are also proposed in synchronous design to protect flip flops against soft errors by efficient usage of scan flip flops [23]. However, the Muller C-element itself is not protected against soft error in this design.

While all of the previous work has achieved fault tolerance by encoding the transmission channel or using circuit duplication, we provide here with QDI circuit hardening techniques against soft errors without any duplication or encoding.

III. ASYNCHRONOUS CIRCUITS

According to the underlying timing models, asynchronous circuits can be classified as Huffman Circuits where the delay of gates and wires are bounded, delay insensitive circuits where the delay of gates and wires are unbounded, speed-independent circuits where gate delays are unbounded and wires delays are negligible compared with gate delays, and quasi-delay insensitive (QDI) circuits which assume unbounded gate and wire delays with isochronic forks. Among

these design styles, QDI circuits have unique advantages in that they make the least timing assumptions (no gate and wire delay assumptions) on the implementation while the circuits function correctly in a wide range of operating environment with variations [10].

Since there is no clock in asynchronous designs, the synchronization between the sender and receiver is controlled by some handshaking protocol [11] as shown in Figure 1a. The handshaking protocol can be implemented in either two-phase or four-phase. A timing diagram for the four-phase handshaking protocol is shown in Figure 1b.

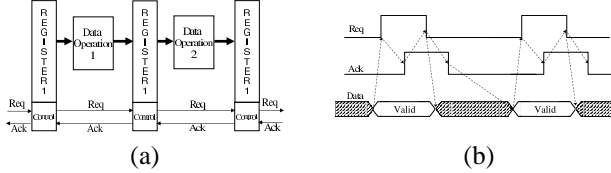


Fig. 1. (a) Asynchronous communication. (b) 4-Phase handshake protocol.

The Muller C-element [11] is a fundamental circuit element widely used for control synchronization in asynchronous designs. In general, a C-element is a state holding circuit which is transparent when all its inputs are equal, and holds the previous output otherwise. Table I shows the truth table for a 2-input Muller-C element.

Input 1	Input 2	Output
0	0	0
0	1	Prev-output
1	0	Prev-output
1	1	1

TABLE I
TRUTH TABLE FOR MULLER-C ELEMENT

Since C-elements are so important in asynchronous design, we study their sensitivity to particle strike, and discuss the optimization techniques to increase their soft error tolerance. In this paper, four static single rail C-element implementations referred in [13] are taken for analysis as shown in Figure 2.

Figure 2a shows one of the C-element implementation with conventional pull-up and pull-down circuit followed in [16]. Figure 2b shows the C-element implementation with an inverter latch [15], and Figure 2c with a modified inverter latch. Finally Figure 2d shows a C-element implementation with a symmetric topology as in [14].

IV. SOFT ERROR ANALYSIS AND RESULTS

The nodes which are taken for soft error analysis are numbered as shown in Figure 2. The particle strikes at these nodes is modeled as a double exponential current source [18] with a fast rise and a slow decay characterized by equation 1 as shown in Figure 3.

$$I(t) = I_{peak} \times (e^{-\frac{t}{\tau_a}} - e^{-\frac{t}{\tau_b}}), \quad (1)$$

Where $I_{peak} = \frac{Q}{\tau_a - \tau_b}$, and Q is the charge collected due to particle strike. τ_a and τ_b are ion-track establishment time constants respectively.

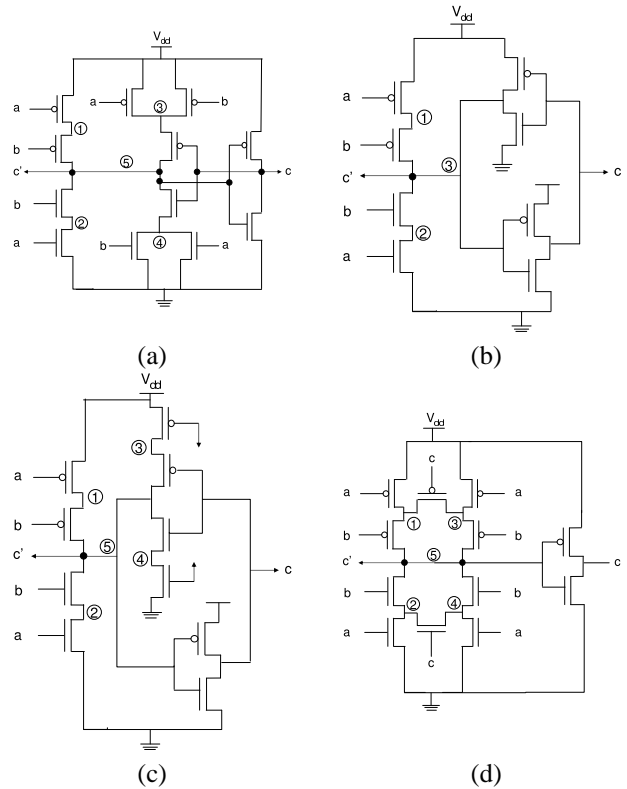


Fig. 2. (a) SC, (b) SIL, (c) MSIL, (d) SS.

Figure 3 shows a snapshot of the C-element with injection of the negative spike at node 'c' and the transient voltage caused by this current spike. By varying the peak of this waveform we try to find the minimum height for an error to appear at the output of the C-element.

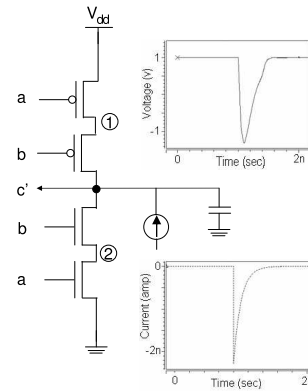


Fig. 3. Soft Error Injection.

The minimal amount of charge necessary to create a bit flip at the output is called critical charge or Q-critical (or Q_{crit}). Here we observe bit flip at the output of the C-element namely 'c' when it is in state zero and $a = \text{high}$ and $b = \text{low}$. The Q_{crit} for SC, SIL, MSIL and SS are measured at 45, 70, 100, 130 and 180nm technology using Berkeley Predictive Technology Model [20]. Figure 4a shows the Q_{crit} at node 1 when the C-element is in state zero. Drastic reduction in Q_{crit} evinces a need for higher SEU tolerant design in the future technologies.

We observe bit flip at the output of the C-element namely

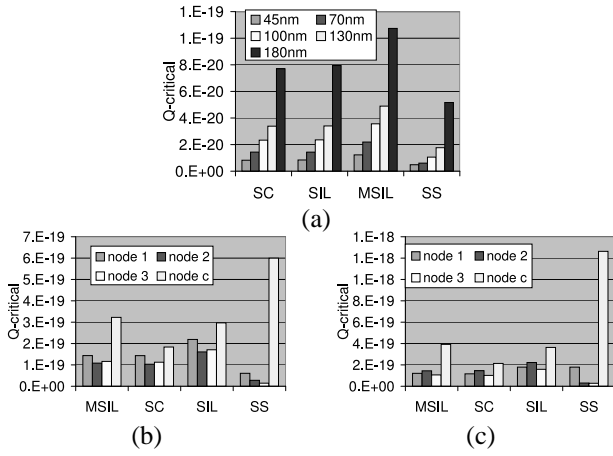


Fig. 4. (a) Q_{crit} at node one, (b) Q_{crit} for C-element when node c is in state zero, and (c) Q_{crit} when node c is in state one.

'c' when it is at state zero and one. We compare only the Q_{crit} at nodes 1, 2, 3, and c, since they are common to all implementations. Figure 4b (4c) shows the Q_{crit} for each of the nodes when the C-element stores a zero (one) in 70nm.

We observe that for all the nodes except the output node 'c', the Q_{crit} for SIL is more than 1.5 times as other implementation. This analysis gives a relative comparison of Q_{crit} for different implementations of the C-element, and helps in picking up a design which will obtain maximum benefits from circuit optimization for soft errors.

V. OPTIMIZATION TECHNIQUES

This section presents 3 optimization techniques applied to the SIL circuit. The SIL circuit contains a keeper circuit at the output to hold the previous value. The first technique is to modify the keeper circuit in this C-element in a way as shown in Figure 5a. This new keeper circuit implementation is similar to that in a SER tolerant SRAM called DICE memory cell [17]. Here the transistors taking inputs 'a' and 'b' are sized up to twice the normal size to drive the DICE cell.

The second and the third techniques are inserting explicit capacitor at the sensitive node to increase the resilience to SEU. A capacitor is formed by connecting both the drain and the source of the NMOS (PMOS) to GND (VDD), as explained in [21]. This capacitor can be connected to the SER susceptible node to increase the node capacitance. Cell level explicit capacitance insertion technique has been studied in [21]. Node capacitance is formed by connecting NMOS and PMOS capacitance in parallel to the SER susceptible node as will be explained in techniques two and three.

In the second technique we implement SIL with an explicit capacitance introduced at node 3 which is the most critical node, since it has the least Q_{crit} among other nodes as seen from Figure 4b and 4c. Figure 5b shows the SIL with the explicit capacitance enclosed in circle.

In the third technique we implement SIL with the stacked nodes in the feedback inverter in the latch along with the explicit capacitance in the stacked nodes and node 3 as shown in Figure 5c.

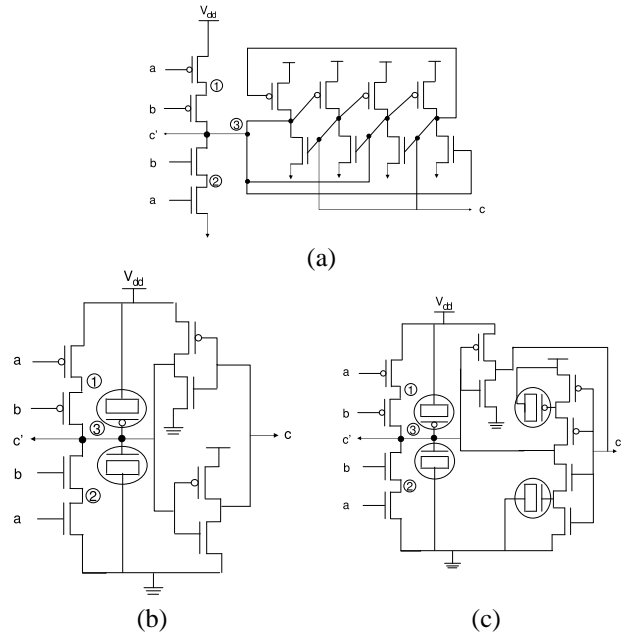


Fig. 5. (a) SIL with modified keeper circuit, (b) SIL with explicit capacitance at node 3, (c) SIL with explicit capacitance and stack nodes.

Figure 6 shows the Q_{crit} for nodes in the optimized SIL normalized over the un-optimized SIL. Figure 5a shows the results for SIL circuit with modified keeper circuit. The Q_{crit} values are doubled at all nodes except at the output node 'c' as expected. The Q_{crit} at node c decreases by 10% when it is at high. Improvement in Q_{crit} at Nodes 1 and 2 are due to sized up transistor as mentioned before. Improvement at node 3 is attributed to sizing up of the transistors and also due to the DICE memory cell which adds four gate-capacitances to node 3.

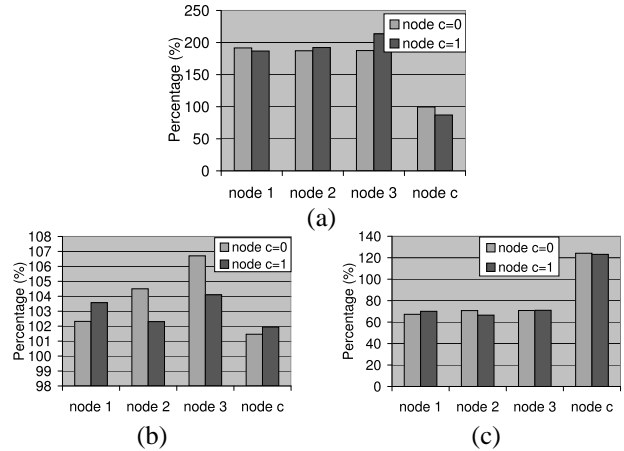


Fig. 6. Q_{crit} for (a) SIL with modified keeper circuit, (b) SIL with explicit capacitance at node 3, (c) SIL with explicit capacitance and stack nodes.

Results for the SIL circuit with the explicit capacitance insertion are shown in Figure 5b and 5c. Figure 5b shows a 3-8% improvement in the Q_{crit} . Figure 5c shows the results for the SIL circuit with the stacked nodes and the explicit capacitance. Here we actually observe that the Q_{crit} at nodes 1, 2 and 3 decreases.

This decrease in Q_{crit} at nodes 1, 2 and 3 is due to the

increase in the sensitivity of the latch to writing 0/1. We observe that the 0/1 writing time for the latch in this C-element decreases 50-66% compared to 0/1 writing time for the SIL implementation in Figure 2b. The node 'c' shows 20% improvement due to additional 2 gate capacitance at the output compared to SIL implementation in Figure 2b. Q_{crit} for nodes 4 and 5 (for the SIL circuit with the stacked nodes and the explicit capacitance) are not shown in Figure 5c as they aren't as critical as nodes 1, 2 and 3. Q_{crit} for nodes 4 and 5 are 1.5-2.5x times the Q_{crit} of nodes 1, 2 and 3.

VI. TRADEOFFS

In [12], power and performance characteristics of SIL, SS and SC are compared. The MSIL shares similar characteristics of power and performance to SIL. SIL implementation differs from SS and SC by 10% less in performance and 50% more in power, respectively. Since the SIL implementation has at least 1.5 times better SER tolerance than SS and SC, it is chosen for optimization. We compare here the performance and power for the original SIL implementation and three SER-optimized implementations. SIL, SIL-1, SIL-2, SIL-3 are those shown in Figure 5a, 5b, 5c, and 5d respectively. Figure 7a shows the timing for storing 0 and 1 for the four SIL implementations. Figure 7b shows the average power consumption and area for the four SIL implementations.

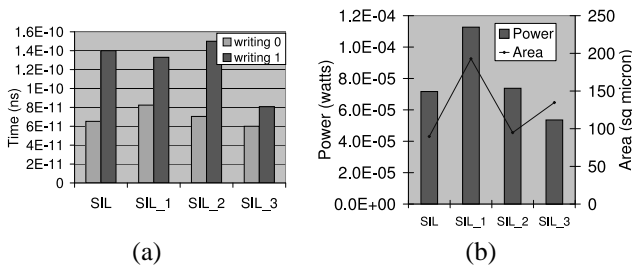


Fig. 7. (a) Timing. (b) Power and area.

Figure 7a shows that the time for writing 0/1 for SIL, SIL-1, and SIL-2 are within 10-25% of difference. SIL-3 shows 75% improvement in the time for writing-1 while an improvement of 10% for writing-0 compared to SIL. The improvement in the performance for SIL-3 is the decrease in the strength of the feedback inverter due to stacking.

The SIL circuits were implemented in 70nm technology. The power consumption of SIL-1 was expectedly high due to the complex keeper circuit implementation. Similar trend is observed for area of SIL-1 as shown in Figure 7b.

Overall SIL-1 implementation has higher reliability against soft errors with negligible performance degradation, at the expense of 1.5x power consumption and 2.15x area penalty compared to SIL implementation. The writing 0/1 time for SIL-1 is 25%/5% more than the SIL.

VII. CONCLUSION

Soft error analysis of four popular CMOS implementations of the Muller C-element has been carried out. SIL implementation is found to dominate over other CMOS implementation of C-elements in terms of overall resilience to soft errors.

Hardening techniques to improve the soft error resilience of C-elements were analyzed. SIL implementation with DICE memory cell was shown to have higher soft error resilience. Results show 2x improvements in Q_{crit} for our proposed techniques with 1.5X power and 2.15X area overhead and 10% performance overhead on average.

REFERENCES

- [1] S. Hauck, "Asynchronous design methodologies: An overview," In *Proceedings of the IEEE*, 1995
- [2] C. H. V. Berkel, M. B. Josephs, and S. M. Nowick, "Scanning the Technology: Applications of Asynchronous Circuits," In *Proceedings of the IEEE*, Volume 87, Issue 2, Feb. 1999.
- [3] C. LaFrieda, R. Manohar, "Fault Detection and Isolation Techniques for Quasi Delay-Insensitive Circuits," In *Proc. of International Conference on Dependable Systems and Networks*, Italy, June 28 - July 01, 2004.
- [4] Y. Monnet, M. Renaudin, R. Leveugle, "Asynchronous circuits transient faults sensitivity evaluation," In *Proceedings of the 42nd annual conference on Design automation*, 2005.
- [5] W. Jang, A. J. Martin, "SEU-Tolerant QDI Circuits," In *Proc. of 11th IEEE International Symposium on Asynchronous Circuits and Systems*, New York City, USA, March 13-16, 2005.
- [6] V. Akella, N. H. Vaidya, G. R. Redinbo, "Limitations of VLSI implementation of delay-insensitive codes," In *Proc. of the Twenty-Sixth Annual International Symposium on Fault-Tolerant Computing*, June 25-27, 1996
- [7] M. Blaum, J. Bruck, "Unordered error-correcting codes and their applications," In *Digest of papers: The 22th Int. Symp. Fault-Tolerant Comp.*, July 1992.
- [8] L. Tallini, L. Merani, B. Bose, "Balanced codes for noise reduction in VLSI systems," In *In Digest of papers: The 24th Int. Symp. Fault-Tolerant Comp.*, June 1994.
- [9] F.-C. Cheng, S.-L. Ho, "Efficient systematic error-correcting codes for semi-delay-insensitive data transmission," In *Proc. International Conf. Computer Design*, November 2001
- [10] R. Manohar, A. J. Martin, "Quasi-delay-insensitive circuits are Turing complete," In *Proc. International Symposium on Advanced Research in Asynchronous Circuits and Systems*, IEEE Computer Society Press, 1996.
- [11] N. R. Poole, "Self-timed logic circuits," In *Proc. of Electronics and Communication Engineering Journal*, 1994.
- [12] M. Shams, J. C. Ebergen, M. I. Elmasry, "Comparison of CMOS Implementations of an Asynchronous Circuits Primitive: the C-Element," In *Proc. Of the Intl Symposium on Low Power Electronics and Design*, 1996.
- [13] M. Shams, J. C. Ebergen, M. I. Elmasry, "Optimizing CMOS implementations of the C-element," In *Proc. International Conf. Computer Design*, October 1997.
- [14] K. V. Berkel, "Beware the isochronic fork," In *Proc. of Integration*, the VLSI journal, June 1992.
- [15] A. J. Martin, "Formal program transformations for VLSI circuit synthesis," In E.W. Dijkstra, editor, *Formal Development of Programs and Proofs*, UT Year of Programming Series, Addison-Wesley, 1989.
- [16] I. E. Sutherland, "Micropipelines," In *Communications of the ACM*, June 1989.
- [17] T. Calin, M. Nicolaidis, R. Velazco, "Upset Hardened Memory Design for Submicron CMOS Technology," In *Proc. of IEEE Tran. On Nuclear Science*, Vol. 43, No. 6, Dec 1996.
- [18] Q. Zhou, K. Mohanram, "Cost-Effective Radiation Hardening Technique for Combinational Logic," In *Proc of IEEE/ACM International Conference on Computer Aided Design*, Nov 2004.
- [19] R. C. Baumann, "Soft errors in advanced semiconductor devices-part I: the three radiation sources," In *IEEE Transactions on Device and Materials Reliability*, Volume: 1 Issue: 1, March 2001.
- [20] "<http://www.eas.asu.edu/ptm/>"
- [21] T. Karnik, S. Vangal, V. Veeramachaneni, P. Hazucha, V. Erraguntla, S. Borkar, "Selective Node Engineering for Chip-Level Soft Error Rate Improvement," In *VLSI Circuits Digest of Technical Papers*, 2002. Symposium on 13-15 June 2002.
- [22] D. Ernst et al., "Razor: A Low-Power Pipeline Based on Circuit-Level Timing Speculation," In *Proc. of 36th Intl. Symposium on Microarchitecture*, December 2003.
- [23] S. Mitra, N. Seifert, M. Zhang, Q. Shi, K. S. Kim, "Robust System Design with Built-In Soft-Error Resilience," In *Computer*, vol. 38, no. 2, February 2005