Hardening C-elements Against Metastability

Leandro S. Heck, Matheus T. Moreira, Ney L. V. Calazans
Pontifícia Universidade Católica do Rio Grande do Sul - Porto Alegre, Brazil
{leandro.heck, matheus.moreira}@acad.pucrs.br, ney.calazans@pucrs.br

***Abstract*—C-elements are non-standard sequential logic gates employed in the design of e.g. asynchronous circuits, clock generators and radiation-hardened circuits. Like any other bistable component as flip-flops and latches, C-elements are susceptible to the metastability phenomenon if marginal triggering conditions are satisfied. However, unlike flip-flops and latches, little research has specifically addressed C-elements metastability issues in the past. This work proposes for the first time an investigation of the susceptibility to metastability of four distinct C-element circuit architectures. Three of these are classical topologies and the last is one recently proposed. The paper also suggests optimizations to enhance the robustness to metastability of the studied structures, showing the effects of these changes in the mean time between failures (MTBF), considering implementations in a 65 nm technology. Results show improvements in τ ranging from around two to four times, when comparing baseline C-elements to their hardened versions.**

***Index Terms*—C-element, metastability, asynchronous design.**

I. Introduction and Related Work

C-elements are simple sequential components proposed a long time ago by Muller [1]. They display a straightforward behavior: in their simplest form, an n-input C-element forces 0 in its output when all inputs are 0, forces 1 in its output when all inputs are 1, and keeps its output with the previous value for any other input combination. This enables their use to construct several types of event synchronization circuits. Due to their main characteristic, they have been extensively employed in asynchronous design to build local handshake controllers [2]. Besides, they are also useful e.g. in synchronous design to detect single event transients (SETs) [3], in clock generation [4], in data synchronization between clock domains [5] and in timing error indication for resilient synchronous and asynchronous architectures [6].

As any other sequential component, C-elements are susceptible to metastability, which is the inability to decide between the two discrete voltage levels associated to the 0 and 1 logic values in bounded time [7]. This occurs when nodes on internal feedback loops of the device reach and remain for some time in voltage levels between those that define the logic values. The propagation of metastability from the internal component nodes through its outputs has unpredictable effects, potentially leading to system level failure and/or deadlock. For example, if a C-element in a clock generation circuit becomes metastable, violation of the circuit expected behavior may occur and even damaging the circuit is possible.

From the previous discussion, there is a clear need to make C-elements robust against such effects. The current literature reports various analyses on electrical metrics for C-elements. For example, the works conducted in [8]–[11] explore the design space of different C-element implementations, assessing power, area and delay trade-offs. In another work, Bastos et al. [12] evaluate transient-fault effects on these implementations, and a more recent work by Moreira et al. [13] explores the effects of voltage scaling on C-elements. Albeit these references perform extensive design space exploration, none of them addresses metastability effects, which can have irreversible effects in circuits, if overlooked. Accordingly, this paper evaluates the behavior of C-elements, proposes metastability-aware design modifications to their design and evaluate these through simulation.

The rest of this article comprises five sections. Section II presents the C-element topologies selected for this study, which are among the most frequently employed ones in the literature. Next, Section III analyzes the conditions under which C-elements enter metastability. Then, Section IV presents suggestions for hardening C-elements against metastability and details the experiments. Finally, Section V and VI provide a discussion and a set of conclusions for the work.

II. C-element Topologies

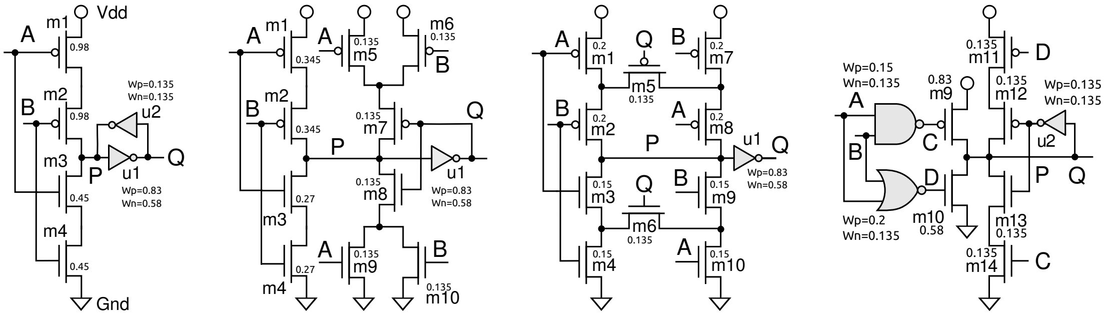
This paper investigates four 2-input C-elements topologies for CMOS technologies. Figure 1 depicts the schematics for each topology. Of these, three are the classical schemes discussed by Shams et al. in [8], and the fourth is a more recent proposal [14]. They are designated herein as the C-elements of: *i*) Martin (CMA), *ii*) Sutherland (CSU), *iii*) van Berkel (CVB), and *iv*) Moreira et al. (CMO) [14].

The CMA, in Figure 1(a), performs better for high-density designs and is reported to be the more robust to radiation effects. This design needs careful transistor sizing to avoid the possible conflict due to the multiple drive characteristic of node `P`. The CSU, in Figure 1(b), is useful in low-power applications and scales better than the other classical topologies with regard to the number of inputs. It has a transistor scheme (`m5`, `m6`, `m9` and `m10`) that disables the feedback inverter when it is switching, avoiding the conflicts reported for the CMA topology. Figure 1(c) shows the CVB, used in low-power and high speed designs. This topology does not scale easily for more than two inputs. Figure 1(d) depicts CMO, reported as best for voltage scaling applications [14].

III. Metastability on C-elements

Figure 2 presents the functionality of a two-input C-element using an extended state transition graph that considers the possibility of metastable transient states. A state is a combination of input and output values. White circles represent states with a stable output. Gray circles are transitory states, where the output is going to change after some expected propagation time t_{pd}, even without any input change taking place. Finally, black circles represent transitory metastable states, where the component can remain for an unbounded time duration. In both metastable transitory states the output is not at a defined logic value. In fact, the output in these states remains around a given (metastable) equilibrium voltage V_{ms}. Lines without

978-1-5386-1911-7/17/$31.00 ©2017 IEEE



(a) CMA (b) CSU (c) CVB (d) CMO

Fig. 1. The schematics for each of the analyzed two-input C-elements: (a) Martin, (b) Sutherland, (c) Van Berkel and (d) Moreira et al. Transistors have minimum length (60 nm) and the widths are indicated in the picture.

a label in the graph of Figure 2 represent a possible change in the output without a change on any input. In the analyzed C-elements, stable states where inputs are equal (`000` and `111`) are also called *driven states* because inputs force the output to remain at a determined value. States where the inputs differ (resp. `10-` and `01-`) are also called *memory states*, since the output value depends on the value stored on the component feedback loop, independent on the input values.

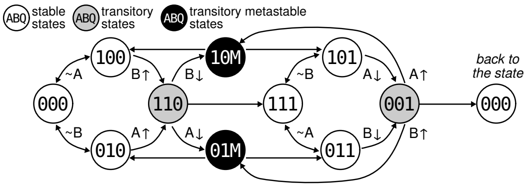


Fig. 2. Extended state transition graph for a two-input conventional C-element. A state is a combination of values for inputs `A` and `B` and for the output `Q`. Input transitions are signaled as rising (↑), falling (↓) and toggling (∼).

Starting from a stable state, two scenarios can lead to metastability generation. These occur when either: *i*) two inputs change in a short period of time; or *ii*) when a single input glitches. For example, from state `010`, if `A` switches, the next state is transitory state `110`, followed by stable state `111`. Assume however that while the component is in transitory state `110`, `B` switches. Then, the next state can be metastable state `10M`. Similarly, if during the same transitory state `A` switches back to 0, the next state can be metastable state `01M`. These scenarios may occur due to the cross-coupled inverters or similar structures used to generate memory states, present in all evaluated C-element designs. Cross-coupled inverters are explicit in the CMA design (see Figure 1(a)), formed by inverters `u1` and `u2`. The equivalent structures are not evident in the other topologies, due to the interposition of control transistors. In CSU, the `u2` inverter is formed by `m5`-`m10` transistors and in CVB by `m1`-`m10`. In CMO, transistors `m11`-`m14` form inverter `u1`. As transistors which compose the memory scheme of CSU, CVB and CMO are merged with those that implement the C-element logic function, hardening the component for metastability can impact the whole design. In addition, the key idea to hardening a bistable element is to keep its metastability resolution time constant τ small [15], to reduce the time required to resolve the voltage uncertainty. The next Section proposes modifications to harden the addressed C-elements against metastability.

IV. Hardening C-elements for Metastability

There is an agreement that it is possible to characterize the susceptibility to metastability of a system using the mean time between failures (MTBF) concept, often specified in millions of years or more. In addition, the MTBF of a system considers the MTBFs of all its component devices. The system MTBF is typically smaller than the MTBF of any metastability-susceptible device it contains.

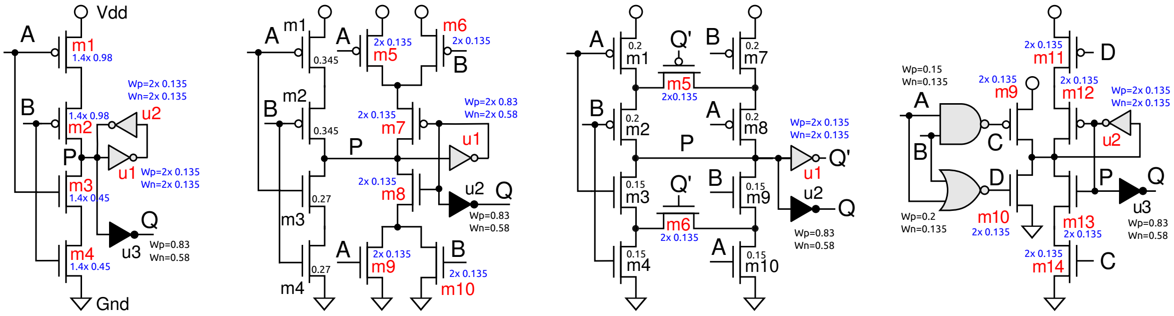
A. Relevant Metastability Parameters

Computing the MTBF of a C-element can be challenging. To address MTBF calculations in practice, this work relies on *MetaAce* [16], a commercial tool from Blendics that can measure metastability-related circuit characteristics. Note that this is a professional tool that automatically generates a large number of Spice-level descriptions with very small increments on metastability-related inputs, based on the bisection method [17]–[19]. This guarantees a tight correlation with the circuit finally fabricated and we can thus consider its results highly reliable, as attested by industrial uses of *MetaAce*.

Equation (1) allows computing the MTBF of a synchronizer [19]. Here, the MTBF of a C-element is calculated by the same equation. t_w represents the vulnerable window where changes on the inputs can lead to some metastability scenario. *MetaAce* replaces this term by an approximation that can be computed deterministically. This approximation is defined by the voltage-time gain g_{tv} and the voltage to escape from metastability v_e which is $\approx 0.3V$ considering a 1V supply.

$$MTBF = \frac{e^{(t_s/\tau)}}{t_w f_c f_d}, \qquad \text{where} \quad t_w = \frac{g_{tv}}{2v_e} \qquad (1)$$

All remaining parameters, on the other hand, are application specific and must be provided by the designer. Accordingly, f_c is the sampling frequency, f_d is the data frequency and t_s is the settling time, *i.e.* the time left for metastability resolve to a defined logic level. Here, f_c and f_d represent the average switching ratio of the C-element inputs. The τ parameter value computation is particularly relevant, because changes in this have an exponential influence in the MTBF.



(a) CMA (b) CSU (c) CVB (d) CMO

Fig. 3. Modifications including addition of the output inverter (inverter in black), transistor sizing (blue labels) and devices with low V_{th} (red labels).

B. C-elements Enhancement Proposal

As Equation 1 shows, the main way to enhance C-elements against metastability is reducing the τ of the circuit. Keeping it small basically involves the isolation of the cross-coupled inverters from the rest of the circuit and reducing the propagation delay of the component. For isolation, it is possible to add an extra inverter to the standard circuit version (named STD), which is responsible for driving the output load connected to node `P` in any of the C-element schematics of Figure 1. The transistors of the added inverter can be sized properly, given the drive strength of the C-element, and its output node can be called `Q`, while the previous `Q` node can be renamed `Q'`. This paper calls this the INV modification. Starting with the INV modification, it is possible to apply two other methods to increase the cross-coupled inverters speed. In the first method, called 2xW, the width of all transistors which compose the cross-coupled inverters are increased by a factor of 2 to boost the current through the transistor, leading to faster switching. Also, this modification is problematic to CMA during the occurrence of conflict in node `P`. The `u2` inverter must be weaker than the input transistors. In this way, for 2xW modification it was necessary to increase W for transistors `m1`-`m4` by 1.4×, to make the new CMA work properly. In the second method, called the LV$_{\mathrm{t}}$ modification, all cross-coupled inverter transistors are changed from standard V_t to their low V_t version. Note that optimizations 2xW and LV$_{\mathrm{t}}$ are applied separately to evaluate their individual effects on the design. A fifth version of modified C-elements is finally produced by combining the last two versions (2xW and LV$_{\mathrm{t}}$), which is accordingly named the ALL modification. Note that 2xW and LV$_{\mathrm{t}}$ started from the INV modification. Thus, ALL obviously includes the INV modification.

C. Simulation Environment

All circuits were designed to the layout level using an STM Bulk CMOS 65 nm technology. The *ROGen* tool, part of the ASCEnD-A standard cell library design flow [20] automates the process of cell sizing, while the *Astran* [21] tool of the same flow enables the automatic layout generation for each C-element topology. Layouts with LV$_{\mathrm{t}}$ transistors were manually edited, because *Astran* does not generate layouts with mixed threshold voltages. Parasitic capacitances were extracted with Cadence *Virtuoso* and the resulting circuit was used to evaluate performance, power and susceptibility to metastability figures. Area was collected from the layouts and, since the layout generation is not a fully deterministic task, it can vary a bit along the distinct modifications of the same circuit. Performance and power figures were analyzed through analog simulation using Cadence *Spectre* and metastability was analyzed by the Blendics *MetaAce* tool, which internally calls Synopsys *HSpice* and *Matlab*. In all experiments, circuits had a fixed output capacitance load equivalent to a fan-out of four (FO4). Note that metastability characterization needs to be made with post-extracted Spice to yield accurate figures, since delays related to the capacitance of the interconnections are crucial when simulating metastability. Furthermore, we evaluated metastability occurring from the item *i)* situation described in Section III, where two inputs change in a short period of time, because the tool currently does not allow to model glitches. This limitation does not jeopardize the generality of the analysis as these scenarios are just different conditions to enter metastability states and do not interfere in the recovery of the circuits from such states.

V. Experimental Results and Discussion

Table I presents the obtained figures, including metastability parameters τ and g_{tv}. It also shows the average values for propagation and transition delays resp. t_{pd} and t_{slew}), energy per operation e_{dyn}, leakage power p_{leak} and layout area. Considering the INV modification, it is observed a minimum improvement of 21.7% in τ for the CVB design and a maximum improvement of 55.5% for CMO. Such improvements come with a small cost in CMA, CVB and CSU, where performance, power and area overheads are typically below 10%. In some cases, there are even small improvements. Still considering the INV modification, the CMO design shows significant overhead in terms of propagation delay. This is due to the architectural modification done to ensure the output is driven by a separate inverter, which added an extra level of logic to the circuit. However, the loss of performance after the INV modification can be reduced for all designs by running *ROGen* again to resize the involved transistors.

The LV$_{\mathrm{t}}$ design reduces τ around 17% compared to the INV. In addition, it increases overheads due to the fact that adding LV$_{\mathrm{t}}$ devices increase the internal capacitances, compromising t_{pd}, e_{dyn} and p_{leak}. Still comparing to INV, 2xW reduces τ almost the same as LV$_{\mathrm{t}}$, except in the case of CMO. In

TABLE I
C-ELEMENT COMPARISONS.

C-Element		τ	g_{tv}	$\overline{t_{pd}}$	$\overline{t_{slew}}$	$\overline{e_{dyn}}$	$\overline{p_{leak}}$	Area
Ref.	Mod.	(ps)	(V/s)	(ps)	(ps)	(fJ)	(µW)	(µm^2)
CMA	STD	25.71	12.0	53.35	30.96	7.90	0.52	4.16
CMA	INV	14.83	23.0	51.70	27.44	10.70	0.51	5.20
CMA	LV$_t$	12.36	15.0	54.46	26.44	6.67	0.52	5.77
CMA	2xW	11.49	20.0	90.05	30.65	7.92	0.52	5.01
CMA	ALL	9.63	21.5	88.49	30.34	6.26	0.55	5.64
CSU	STD	28.99	22.0	44.82	26.83	8.37	0.47	5.72
CSU	INV	18.70	26.0	47.79	27.11	6.70	0.50	6.76
CSU	LV$_t$	15.57	26.0	46.82	26.22	7.33	0.48	7.72
CSU	2xW	14.23	21.0	53.83	27.67	7.20	0.50	6.24
CSU	ALL	12.06	24.0	52.01	26.31	6.98	0.50	7.08
CVB	STD	30.07	11.0	37.15	25.95	9.36	0.49	6.76
CVB	INV	21.19	22.0	40.36	26.73	12.20	0.53	8.32
CVB	LV$_t$	17.86	29.0	40.62	26.74	12.49	0.53	8.60
CVB	2xW	17.80	20.0	43.13	28.04	8.42	0.48	7.80
CVB	ALL	16.68	15.0	43.66	28.09	9.87	0.47	8.63
CMO	STD	40.86	13.0	40.09	27.79	7.94	0.48	7.28
CMO	INV	18.20	8.9	73.25	26.38	7.22	0.48	8.32
CMO	LV$_t$	15.01	15.0	64.31	24.93	6.95	0.48	8.32
CMO	2xW	11.94	12.0	66.54	23.62	7.46	0.46	7.80
CMO	ALL	9.96	14.0	59.13	22.85	7.92	0.45	8.22

this case overheads were also typically below 10% for the evaluated designs.

The obtained results indicate a clear trade-off between performance, power and area figures and the susceptibility of C-elements to become metastable. Considering the different topologies without any optimization, it is possible to infer that the CMA topology arises naturally as the most suited C-element for metastability-sensitive applications, as it provides the best τ values overall. Indeed, CMA always presents a τ smaller than the other topologies for any of the optimization options proposed here. However, it is important to highlight that CMA is not the best choice for low power applications [14] regardless its design simplicity. CMA is substantially slower than e.g. the CVB design, the fastest C-element topology [14]. Designers can explore the use of topologies like INV, LV$_t$ and 2xW optimizations based on their requirements.

VI. Conclusions

The article proposes a set of optimizations to reduce susceptibility of typical and modern C-elements to enter in metastable states. Results indicate that over 2× improvements in τ are achievable in these designs, by adopting the proposed optimizations individually, which have as consequence improved MTBF values for C-elements. Moreover, as they are independent from one another, the suggested optimizations can be employed jointly to further optimize designs. The paper advances the state-of-the-art in C-elements, and adds possibilities to effectuate design space exploration of these in real designs. This is of relevant to the electronics design community, as C-elements are gaining relevance and being employed in a variety of applications as reported in [3]–[5].

References

[1] D. E. Muller, “Theory of Asynchronous Circuits,” Digital Computer Laboratory, University of Illinois, Tech. Rep. No. 66, Dec. 1955.

[2] J. Cortadella, M. Kishinevsky, A. Kondratyev, L. Lavagno, and A. Yakovlev, *Logic Synthesis for Asynchronous Controllers and Interfaces*. Springer-Verlag, 2002.

[3] D. G. Toro, M. Arzel, F. Seguin, and M. Jézéquel, “Soft Error Detection and Correction Technique for Radiation Hardening Based on C-element and BICS,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 61, no. 12, pp. 952–956, Dec. 2014.

[4] E. Yahya, O. Elissati, H. Zakaria, L. Fesquet, and M. Renaudin, “Programmable/Stoppable Oscillator Based on Self-Timed Rings,” in *International Symposium on Asynchronous Circuits and Systems (ASYNC)*, 2009, pp. 3–12.

[5] C. Aarthi and R. K. Gnanamurthy, “Low-power Delay Buffer Design Using Asymmetric C-element Gated Clock Strategy,” in *International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, Jul. 2014, pp. 870–875.

[6] M. T. Moreira, D. Hand, P. A. Beerel, and N. L. V. Calazans, “TDTB Error Detecting Latches: Timing Violation Sensitivity Analysis and Optimization,” in *International Symposium on Quality Electronic Design (ISQED)*, 2015, pp. 379–383.

[7] T. Polzer, A. Steininger, and J. Lechner, “Muller C-Element Metastability Containment,” in *International Workshop on Power And Timing Modeling, Optimization and Simulation (PATMOS)*, 2012, pp. 103–112.

[8] M. Shams, J. C. Ebergen, and M. I. Elmasry, “A Comparison of CMOS Implementations of an Asynchronous Circuits Primitive: the C-element,” in *International Symposium on Low Power Electronics and Design (ISLPED)*, Aug. 1996, pp. 93–96.

[9] M. Shams, J. Ebergen, and M. Elmasry, “Modeling and Comparing CMOS Implementations of the C-element,” *IEEE Transactions on VLSI Systems*, vol. 6, no. 4, pp. 563–567, Dec. 1998.

[10] O. Elissati, E. Yahya, S. Rieubon, and L. Fesquet, “Optimizing and Comparing CMOS Implementations of the C-element in 65nm Technology: Self-timed Ring Case,” in *International Workshop on Power And Timing Modeling, Optimization and Simulation (PATMOS)*, 2010, pp. 137–149.

[11] M. T. Moreira, B. S. Oliveira, F. G. Moraes, and N. L. V. Calazans, “Impact of C-elements in Asynchronous Circuits,” in *International Symposium on Quality Electronic Design (ISQED)*, Mar. 2012, pp. 437–343.

[12] R. P. Bastos, G. Sicard, F. Kastensmidt, M. Renaudin, and R. Reis, “Evaluating Transient-fault Effects on Traditional C-element’s Implementations,” in *IEEE International On-Line Testing Symposium (IOLTS)*, 2010, pp. 35–40.

[13] M. T. Moreira and N. L. V. Calazans, “Voltage Scaling on C-elements: A speed, power and energy efficiency analysis,” in *International Conference on Computer Design: VLSI in Computers and Processors (ICCD)*, Oct. 2013, pp. 329–334.

[14] M. T. Moreira, M. E. Arendt, A. R. Guazzelli, and N. L. V. Calazans, “A New CMOS Topology for Low-Voltage Null Convention Logic Gates Design,” in *International Symposium on Asynchronous Circuits and Systems (ASYNC)*, 2014, pp. 93–100.

[15] D. Li, P. I.-J. Chuang, D. Nairn, and M. Sachdev, “Design and Analysis of Metastable-Hardened Flip-Flops in Sub-Threshold Region,” in *International Symposium on Low Power Electronics and Design (ISLPED)*, Aug. 2011, pp. 157–162.

[16] J. Cox, T. Chaney, and D. Zar, “Simulating the Behavior of Synchronizers,” White Paper, Blendics, Inc, Fev 2011. [Online]. Available: http://blendics.com/wp-content/uploads/2015/05/Blendics_Synchronizer13Feb2011.pdf

[17] S. Yang and M. Greenstreet, “Computing Synchronizer Failure Probabilities,” in *Design Automation & Test in Europe (DATE)*, 2007, pp. 1361–1366.

[18] ——, “Simulating Improbable Events,” in *ACM/IEEE Design Automation Conference (DAC)*, 2007, pp. 154–157.

[19] I. W. Jones, S. Yang, and M. Greenstreet, “Synchronizer Behavior and Analysis,” in *International Symposium on Asynchronous Circuits and Systems (ASYNC)*, 2009, pp. 117–126.

[20] M. T. Moreira, M. Arendt, A. M. Z. Jr., R. A. L. Reis, and N. L. V. Calazans, “Automated Synthesis of Cell Libraries for Asynchronous Circuits,” in *Symposium on Integrated Circuits and Systems Design (SBCCI)*, 2014, pp. 183–188.

[21] A. M. Ziesemer and R. A. L. Reis, “Simultaneous Two-Dimensional Cell Layout Compaction Using MILP with ASTRAN,” in *IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, Jul. 2014, pp. 350–355.