

Radiation Hardened Latch Designs for Multi-Node Upsets

Adam Watkins, *Member, IEEE*, Spyros Tragoudas, *Member, IEEE*

Abstract—As the process feature size continues to scale down, the susceptibility of logic circuits to radiation induced error has increased. This trend has led to the increase in sensitivity of circuits to multi-node upsets. In these circuits, the memory elements are the most vulnerable to error since data is held within the element. Previously, work has been done to harden latches against single event upsets (SEU). Currently, there has been a concerted effort to design latches that are tolerant to double node upsets (DNU). Future process technologies may lead to the likelihood of the a triple node upset (TNU). Based on this trend, a novel DNU tolerant design and TNU tolerant design are proposed. In addition to this, the DNU is designed to be robust for use in clock gating. This design is used as a basis for the TNU tolerant design. Through experimentation, it is shown that the DNU design is more efficient than all other DNU robust designs. The TNU tolerant latch is shown to have an overhead of (PUT NUMBERS HERE).

Index Terms—Soft Errors, Radiation Hardened Latch, Transient Pulses, Single Event Upset, Double Node Upset.

1 INTRODUCTION

As the transistor feature size continuously scales down to improve performance, modern circuitry continues to become more susceptible to radiation induced errors commonly referred to as a soft error. Terrestrial soft errors can manifest from either neutron particles originating from cosmic rays or alpha particles from packaging. In space, a soft error can come from gamma rays, protons, neutrons, electrons and heavy ions [1], [2]. A soft error of any source occurs when a particle hits the diffusion region of a reverse bias transistor. This, in turn, allows an “off” transistor to temporarily conduct current which can cause a voltage change in a node connected to the affected transistor. If the error occurs in combinational logic, the resulting voltage pulse may be propagated to a circuit output and captured by a flip-flop thus causing an error. Additionally, the error may occur directly on an internal latch of a flip-flop causing immediate data corruption. Due to this possibility, there is a need for new latch designs that can tolerate errors.

There has been extensive research in the field of hardening latches against single event upsets (SEU). The most straightforward hardening design is the use of triple modular redundancy (TMR). This design consists of 3 standard latches connected to a 3-input majority voting circuit. While this design is robust against errors, it has the drawback of high area, delay and power consumption. For this reason there have been many other designs proposed that offer high SEU reliability with lower area, delay and power consumption. The first and most common design is the DICE cell proposed in [3]. The design in [3] consists of 4 one-input c-elements connected in series. Two of the nodes are connected to a pass-gate which allows data to be loaded.

When an error occurs in the DICE latch, the struck node is flipped. The two nodes connected to the struck node are set to a high impedance state and one node remains error-free. Since the error-free node is strongly driven to the correct value, it drives the remaining nodes back to the correct value. While this design works well for a single node, it is not capable of handling multi-node errors.

While the DICE latch is efficient in area, it suffers from high delay. For this reason there has been a multitude of SEU tolerant devices that have been proposed. The SEU tolerant designs follow one of two approaches to hardening: sizing transistors such that the critical charge exceeds the maximum injected charge for the intended environment and by designing circuits that functionally tolerate the error. For the former designs, such as [4], they are typically performance and area efficient. The drawback with these types of designs is that they require accurate estimates for the maximum injected charge. If the maximum charge is found to be too high, a designer using this type of latch would have to choose between performance and reliability.

The latter type of latch, such as [5], [6], [7], [8], [9], [10], have the advantage of recovering from a SEU regardless of the injected charge due to the logical functions of the latch forcing recovery of affected node. In cases where the maximum injected charge is not excessively high, the latches have higher performance and area overheads. However, these type of latches are preferable in many cases since the maximum charge may be unknown or very high.

In modern processes, the transistor size is small enough that a radiation particle may strike multiple simultaneous transistors. Cases where this type of strike may occur are commonly referred to as a single event multiple upset (SEMU). In addition to the SEMU case, high radiation environments may allow for the manifestation of a multiple event multiple upset (MEMU). In this case, multiple radiation particles strike internal transistors simultaneously. When either a SEMU or MEMU strike a latch, they may upset multiple nodes. If two nodes are upset in the latch, this

- Adam Watkins is with Los Alamos National Laboratory, Los Alamos, NM 87545 and Southern Illinois University Carbondale, Carbondale, IL 62901
E-mail: acwatkins88@lanl.gov
- Spyros Tragoudas is with Southern Illinois University Carbondale, Carbondale, IL 62901
E-mail: spyros@siu.edu

Manuscript received March 1, 2017;

is referred to as a double node upset (DNU). If three nodes are upset, this is called a triple node upset (TNU). The DNU is currently of great concern as the feature size has allowed for a sharp increase in the occurrence of DNUs. Section (REFERENCE SECTION HERE) provides an overview of all existing DNU latches. To the author's knowledge, there has been no research done on the occurrence of the TNU. It can be inferred however, that future processes or high radiation environments, such as a mission to Mars or Europa, will allow for the TNU to be of large concern.

To save power, many modern circuit designs employ a technique commonly referred to as clock gating to further reduce the power consumption. Clock gating consists of shutting off the clock to a stable value or "gating" the clock. If clock gating is used in a latch, it may need to hold the stored value for many clock cycles. If the latch is struck by a radiation error while gated, it could lead to a loss of data. This may occur if the latch has high-impedance states after an error. If an error occurs, the high-impedance nodes may slowly discharge causing a loss in data. To remedy this issue, researchers have proposed the addition of output circuitry to hold the data. However, as shown in Section (REFERENCE SECTION HERE), the additional circuitry adds a large overhead to the delay and power consumption.

To solve this problem, we propose the HRDNUT (Highly Robust Double Node Upset Tolerant) latch which is an efficient DNU tolerant design that is capable of recovering all nodes after an error occurs. The recovery feature provides a distinct advantage over previous designs in cases where clock gating is used since it removes the need for the additional circuitry since no nodes are held to a high-impedance state after an error. Designs that are DNU tolerant and exhibit this behavior are referred to as DNU-robust. Any design that is DNU tolerant and does not have high-impedance states is referred to a DNU-non-robust. The proposed design is thoroughly compared to existing designs and is more efficient than the existing DNU-robust design in [11] in power, delay and area. The design is also compared to all existing DNU tolerant latches and the most common SEU tolerant latches.

In addition to the DNU latch, we also propose the TNU-latch which is a TNU tolerant latch that is based on the HRDNUT. While this latch is non-robust, it provides a simple and efficient solution suitable for high reliability applications. To the author's knowledge, the TNU latch is the first of its kind.

Lastly, even though there are masking factors in combination logic, depending on the design, the error-rate in combinational logic is not insignificant. In cases where an error occurs, the resulting voltage pulse may propagate to the circuit output. As proposed in [6], one approach to remove this type of error is to apply a filtering circuit on the combinational logic output. In the case of multi-node upsets, the design in [6] is vulnerable to error. An improved design is also proposed.

The paper is organized as follows: Section [REFERENCE] provides a discussion on existing DNU tolerant latches, Section [REFERENCE] discusses the HRDNUT, Section [REFERENCE] gives the TNU-latch, Section [REFERENCE] consists of the explanation for the pulse filtering

circuit, Section [REFERENCE] contains a comparison of the proposed latches to many existing designs and Section [REFERENCE] concludes the paper.

2 EXISTING MULTI-ERROR TOLERANT DESIGNS

In this section, we discuss the existing DNU tolerant designs and give a background for the TNU latch. First we will discuss the DNU tolerant designs. The first proposed design, named the DNUCS latch and given in [12], contains two DICE latches connected to a 2-input Muller C-element. An example of the C-element is given in Fig. 1. The idea behind this design is that it is impossible for a DNU to both DICE latches since they are SEU tolerant. More specifically, since the DICE latch is SEU tolerant, it requires a DNU to upset the data. The output Muller C-element only changes value when the inputs are unanimously the same value. Since both latches cannot be upset, the DNCS will tolerate a DNU. While this design is DNU tolerant, it is not DNU-robust since it may move to a high-impedance state after an error.

Another design, named the interception latch and proposed in [13], improves on the DNCS by providing lower power consumption, delay and area. The latch functions using 6 two input Muller C-elements connected in series. Every other node in the latch is fed to an output 3-input Muller C-element. In this design, a DNU can only flip two nodes within the design. Since the output is voted on by a Muller C-element, the output will not change value. Like the DNCS latch, a DNU will force the latch into a high-impedance state which implies the latch is DNU-non-robust.

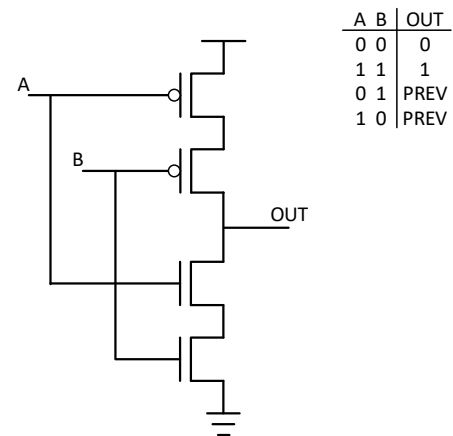


Fig. 1: Muller C-element

The most recent and efficient proposed latch is the HSMUF which is proposed in [14]. This latch uses the TP-DICE structure found in [15] which is an extended DICE cell with a total of 6 nodes. In the TP-DICE cell, every other node is connected to the input of a 3-input Muller C-element. When a DNU occurs in the worst case, two nodes are set to erroneous value, two nodes are set to high-impedance and two nodes hold the correct value. Since a Muller C-element is placed on the output, the high-impedance and correct nodes hold the output to the correct value. However, a drawback with this design is that it relies

on high impedance states for reliability thus the design is non-robust. A common way to mitigate this issue is to place a weak-keeper at the output of the latch as in 2. While this design does ensure the output is held, the Muller C-element must be sized such that the driving strength exceeds the that of the keeper. According to our simulations found in Section [REFERENCE RESULTS], the addition of the keeper substantially increases the delay, area and power consumption.

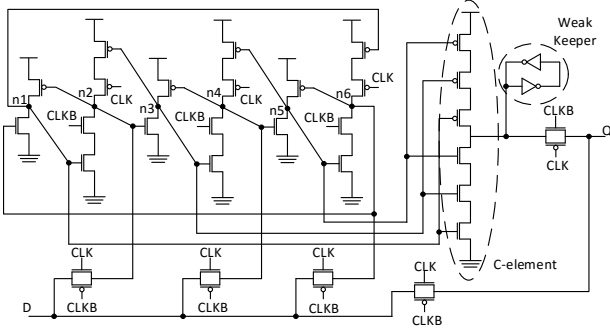


Fig. 2: HSMUF latch [14] with a weak keeper on the output.

It is shown in Section [REFERENCE] that designing a latch such that each node is capable of recovering all nodes allows for the latch to be usable in clock gating while also providing lower power, area and delay compared to the C-element approach. As stated in Section 1, latches that are capable of recovering all nodes after an error are called robust. This unique feature is desirable since it leads to more efficient latch designs that can recover from an error. One of the existing DNU-robust designs is the DONUT latch proposed in [11] which is the most area efficient design. This latch is based on the combination of four DICE latches creating twelve nodes. Each node is connected to cross-coupled transistors. Since the design is based on the DICE latch, it is able to exploit the recovery feature of the latch. One issue that was discovered during the testing of this latch is that it suffered from excessively high power and delay. It was found that the root of the problem was due to data contention on the data loading lines. To solve this issue, the latch was modified such that the data loading nodes were set to high impedance during the transparent mode. As shown in Section [REFERENCE RESULTS], this modification saved a large amount of power. This design is referred to as the DONUT-M and is given in Fig. 4.

In addition to DNU tolerant latches, we also investigate the TNU latch. The authors in [16] propose a latch design that shows limited TNU tolerance. Their design uses eight nodes and eight 2-input C-elements. There are four input signals which are each connected to the output of a C-element to load the data during the transparent mode. In the hold mode, the latch is fully tolerant to a DNU since at least one of the erroneous nodes will be driven by a C-element with error-free inputs. However, when a TNU occurs, the latch is only tolerant if all three errors are on adjacent nodes.

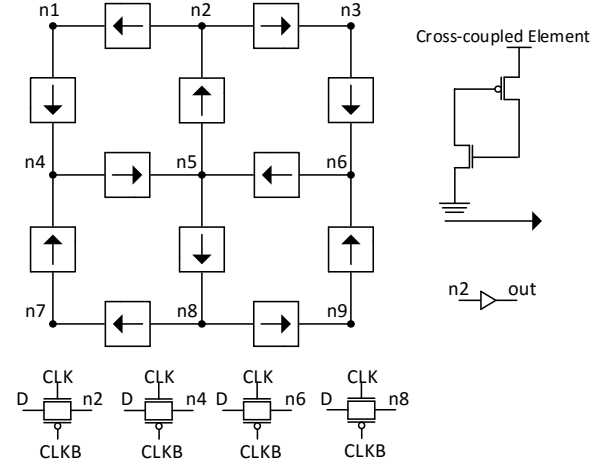


Fig. 3: DONUT latch as proposed in [11].

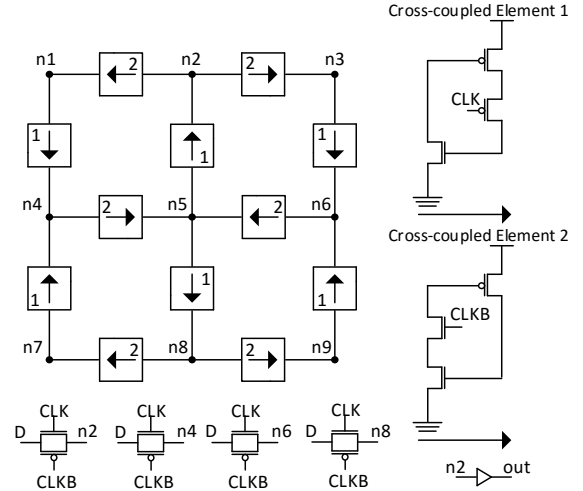


Fig. 4: Modified low-power DONUT latch.

3 DISCUSSION OF HRDNUT LATCH

This section discusses the HRDNUT latch. The HRDNUT latch is based on a basic storage loop given in Fig. 5 which contains a 3-input C-element connected to an inverter. This design is derived from a standard keeper element. In addition, the C-element contains two transistors driven by a CLK and CLKB. The purpose of the transistors is to set the output to high impedance during the transparent mode. As demonstrated by the DONUT-M, the addition of these transistors drastically reduces the power and delay at a cost of area. Once in the transparent mode, the output is loaded using a pass-gate which allows D to be set directly.

The basic block is then connected to form the block based latch in Fig. 6. To ensure that the nodes recover after an error, each C-element in the data loop is driven by the other two nodes. This type of configuration ensures that an error on a C-element output is not held. The latch in Fig. 6 is SEU tolerant and DNU tolerant when one error strikes the output

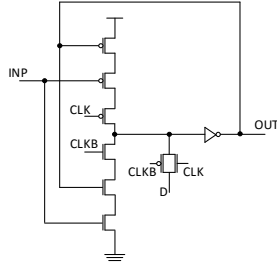


Fig. 5: Basic data storage loop block.

C-element. While this latch is not DNU tolerant, it forms the basis for the DNU robust design. To demonstrate the functionality of the latch, we will first describe its operation during normal operation.

In normal operation, the latch data is loaded during the transparent mode. In this mode the input clock CLK is high and $CLKB$ is low. At this stage, nodes $n1$, $n2$ and $n3$ are set to high impedance and each pass gate is activated allow the value at D to be loaded to the node. Once loaded, the data will propagate to the inputs of the output C-element. During the hold mode, the held data is reinforced by C-elements $C1$, $C2$ and $C3$. The output is then reinforced since the inputs of the output C-element are unanimously held the a single value.

In the case of an SEU in an internal node on the block based latch, the error will propagate to the other C-elements. However, since each C-element has at least one unaffected node, the data on the node will fully recover allowing full recovery of the latch state. In the case of a DNU on the internal nodes, the inputs of the unaffected C-element will be flipped due to the errors. This will ultimately flip the output leading to an error on the output. While this latch is not DNU tolerant, as stated before, it forms the basis for the HRDNUT.

Based on the block latch, the HRDNUT is modified such that no C-element drives itself. More specifically, we first add additional resiliency to latch by transforming the inverter on node $n2$ to a two-input C-element. The second input on this element is driven by the output. This allows for an error on $n2$ to be blocked by C-element $C4$. Next to add resiliency to nodes $n1$ and $n3$, elements $C5$ and $C6$ are added. In addition to DNU tolerance, the C-elements allow for robustness of the latch by ensuring that $C1$, $C2$ and $C3$ do not drive themselves. Note that on $C5$ and $C6$, one NMOS and one PMOS transistor is driven by OUT and $n2$. This ensures that the latch will recover all nodes since the node $n5$ drives the PMOS on $C7$ while the output drives the PMOS on $C5$. The idea behind this is that $n5$ only affects C-element $C7$ if $n5$ is a low value. However, an error on OUT does not prevent recovery since $C7$ is only affected if $n5$ is 0. $C6$ operates similarly but node OUT drives a NMOS.

Now that we have explained the design process, we will evaluate the HRDNUT latch during normal operation as in [17]. When the positive clock signal (CLK) has a high value and the negative clock signal ($CLKB$) has a low value, the latch is in transparent mode. At this stage, the transistors connected to the clock signal in C-element $C1$

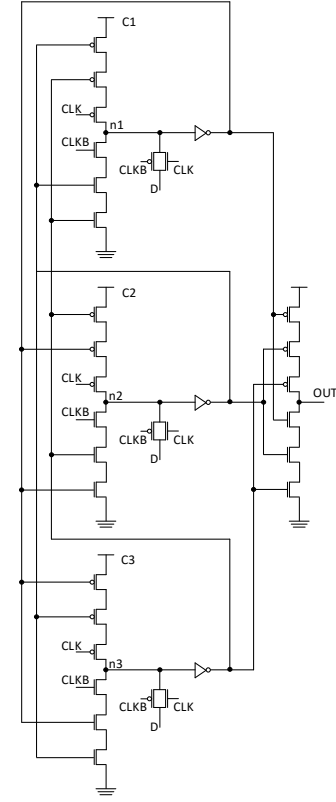


Fig. 6: Schematic of the block-based latch.

deactivates the PMOS and NMOS stacks thus causing the node $n1$ to be in a high impedance state. This, in effect, reduces data contention thus reducing delay and dynamic power consumption. Next, the data is loaded through the pass gates connected to nodes $n1$, $n2$ and out . Since the output node out is loaded directly, the data to out delay is minimized and all nodes are set to their respective error free values. When CLK changes to a low value and $CLKB$ to a high value, the latch moves into the hold mode. In this stage, the pass gates are deactivated and the state of the latch is held since each node is driven to the correct value using a C-element. Fig. 7 provides the waveforms of the CLK , D and OUT nodes for both the transparent and hold modes of operation.

In the case of an SEU, the HRDNUT retains excellent resiliency of the block based latch and the ability to recover every node after an error. In the case of any internal node being struck by an error, the latch will not change value due to all internal C-elements requiring at least 2 identical input values to change values. In the case of an error hitting the output node out , the latch fully recovers since out does not directly drive C-element $C7$.

Lastly, we will evaluate the latch in the case of a DNU. Note that unless otherwise stated, it is assumed that the analysis applies to both when $D=0$ and $D=1$. For our analysis, we categorize the possible DNU strike combinations into 9 distinct cases based on their effect in the HRDNUT latch. The categories are discussed in detail below.

- 1) Consider strikes at nodes $n1$ and $n2$. In this case, the

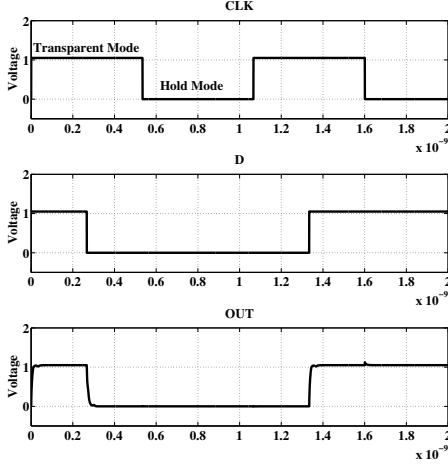


Fig. 7: Waveforms of the HRDNUT latch during normal operation.

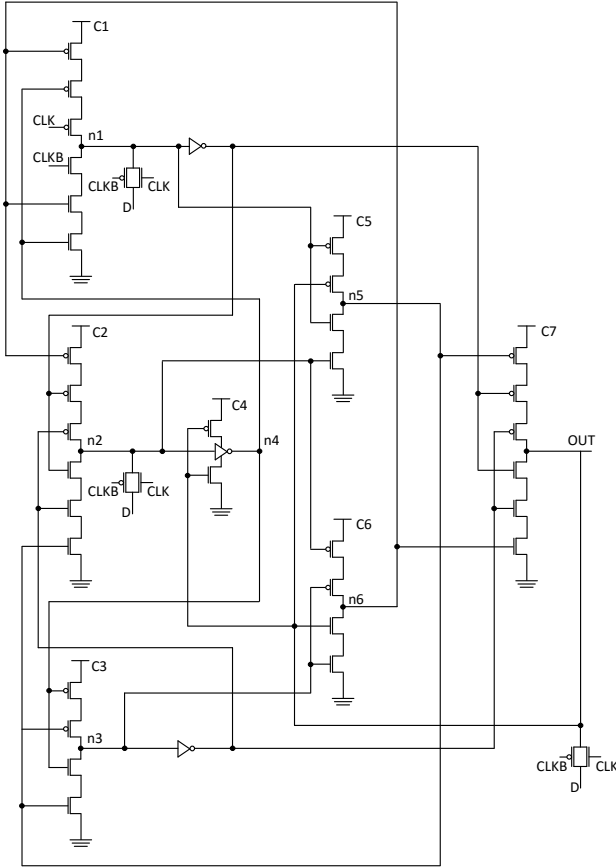


Fig. 8: Schematic of the HRDNUT latch.

error at $n1$ will propagate to C-elements $C5$ and $C7$ but will not cause a flip since the error at $n2$ will be blocked by C-element $C4$. Additionally, since the inputs of C-elements $C1$ and $C2$ are unchanged, the nodes will recover their initial values. This analysis can be applied to node combinations containing node $n2$ except for the combination with node out since the error will be blocked by C-element $C4$.

- 2) In the case of a DNU upsetting nodes $n2$ and out , the error at $n2$ will propagate through C-element $C4$. However, C-elements $C1$ and $C3$ will block the error and nodes $n1$, $n3$, $n5$ and $n6$ will hold their values thus driving node out to the correct state.
- 3) Consider when a DNU strikes nodes $n1$ and $n5$. In this case, the error at $n1$ hits the output of C-element $C1$ which is propagated to $C7$. The error on $n5$ is also propagated to C-element $C7$. Since node $n3$ and the inputs of C-elements $C1$ and $C5$ are unaffected by an error, the output retains the error-free value and the latch fully recovers the previous state. The above analysis also applies to the node combination $(n3, n6)$.
- 4) In the case of a DNU hitting nodes $n3$ and $n4$, the error at $n4$ is propagated to C-element $C3$ and the error at $n3$ is propagated to $C7$ and $C6$. After the error on $n3$ subsides, $C4$ will drive node $n4$ and, due to the connection at $C3$, node $n3$ back to the error-free value. The node combination $(n1, n1)$ can be analyzed similarly. For the node combinations of $(n4, n5)$ and $(n4, n6)$, the latch will also recover the previous result since the inputs to $C4$ are unchanged. This implies that after the error occurs at $n4$, the node will be driven back to the correct value thus also driving the nodes $n5$ or $n6$ back to the correct value.
- 5) When a DNU upsets the combination of $n4$ and out , the error at out is propagated to $C4$, $C5$ and $C6$ and the error at $n4$ to $C1$ and $C3$. Since none of the inputs to $C7$ are changed by the error, out is flipped back to its error-free value which drives $n4$ through $C4$ back to its previous state.
- 6) Consider when a DNU strikes nodes $n1$ and $n3$ being struck. In this case, the errors are propagated to C-elements $C2$, $C5$, $C6$ and $C7$. However, since the errors do not manifest into an error on any other node, the latch fully recovers from the error.
- 7) When a DNU strikes the nodes $n1$ and $n6$. The error at node $n6$ propagates to $C1$ and $C7$ while the error at $n1$ also propagates to $C7$. Due to the error-free node $n3$ driving $C7$, the previous value is held at the output by $C7$. Additionally, $n3$ will drive $C6$ back to its previous value thus driving $C1$ back to the error free state. This analysis can be applied similarly to the node combination of $(n3, n5)$.
- 8) In the case where a DNU strikes nodes $n5$ and out the error at $n5$ propagates to $C7$, $C2$ and $C3$ and the error at out goes to $C4$, a PMOS in $C5$ and a NMOS in $C6$. When the error-free value at out is 1, the value at $n5$ is 0. The error at the nodes change the values to 0 and 1 respectively and the erroneous value at out is propagated to the PMOS at $C5$ and the NMOS at $C6$.

This, in effect, causes the PMOS at C5 to be activated and the NMOS at C6 to be deactivated. However, since nodes $n1$ and $n2$ remain error-free, the NMOS stack of C5 will drive $n5$ back to the correct value. This, in turn, forces C7 to also drive *out* back to the error-free value. In the case where *out* has an ideal value of 0, the error will be fully recovered since the NMOS stack will be entirely driven by fault-free nodes. The above analysis can be applied to the node combination of ($n6$, *out*).

- 9) Lastly, we analyze the node combinations ($n1$, *out*), ($n3$, *out*) and ($n5$, $n6$). In these cases the errors do not cause a change on the inputs of any C-elements driving the node thus the previous value will always be recovered.

To evaluate the design, pulses were injected using the equation given in [19]. The equation is given below with τ as the technology dependent constant, Q_o as the injection current value and t as the variable for time. In the equation τ was set to 32×10^{-12} and Q_o was set to 5fC. The results of this simulation for each distinct case is given in Figs. 9-18.

$$I(t) = \frac{2Q_o}{\tau\sqrt{\pi}} \sqrt{\frac{t}{\tau}} e^{-\frac{t}{\tau}} \quad (1)$$

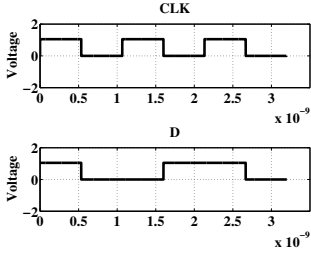


Fig. 9: Waveforms for CLK and D.

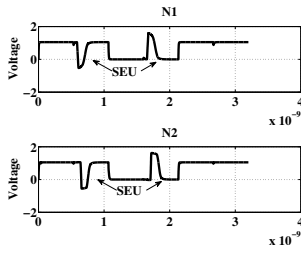


Fig. 10: Node pair $n1$ and $n2$ upset and recovery.

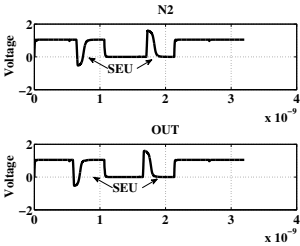


Fig. 11: Node pair $n2$ and *out* upset and recovery.

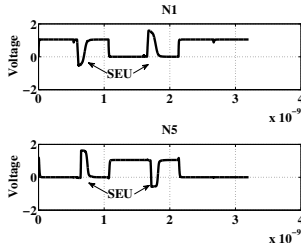


Fig. 12: Node pair $n1$ and $n5$ upset and recovery.

4 PROPOSED TNU TOLERANT LATCH

In this section we discuss the implementation of a non-robust triple node upset (TNU) tolerant latch named TNU-latch. The development of a robust TNU latch has been investigated but was fruitless due to the number of possible cases to verify. For example, a latch with two more nodes compared to the HRDNUT giving nine total nodes require the verification of 84 unique cases. Instead, we focused on

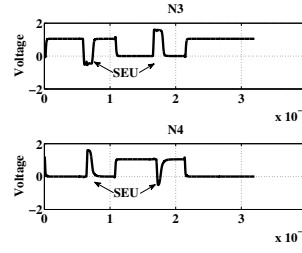


Fig. 13: Node pair $n3$ and $n4$ upset and recovery.

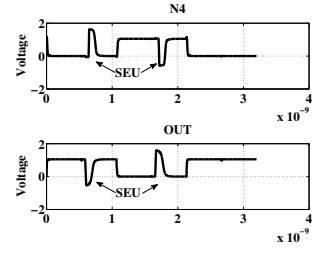


Fig. 14: Node pair $n4$ and *out* upset and recovery.

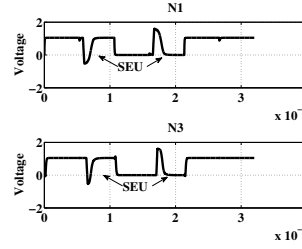


Fig. 15: Node pair $n1$ and $n3$ upset and recovery.

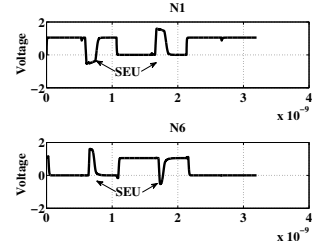


Fig. 16: Node pair $n1$ and $n6$ upset and recovery.

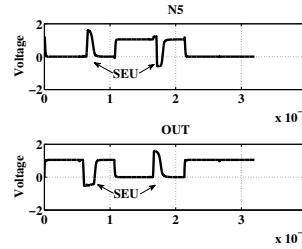


Fig. 17: Node pair $n5$ and *out* upset and recovery.

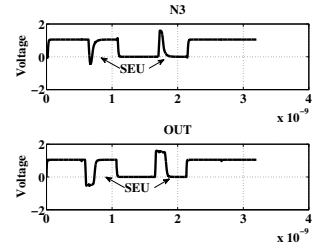


Fig. 18: Node pair $n3$ and *out* upset and recovery.

the development of a simple and efficient design that has [TRANSISTOR NUM]. This design has been verified for all possible TNU cases and is shown to be fully tolerant. To our knowledge, this design is the first fully TNU tolerant latch proposed in literature. While it is not clear if current process generation benefit more from interleaving SEU and DNU tolerant designs or directly using a TNU latch, it can be inferred that smaller processes in highly radioactive environments will benefit from a TNU tolerant latch.

First we will describe the design of TNU-latch. The latch consists of a base block latch as in the HRDNUT but with 5 storage blocks. Each storage block has a C-element with 4 inputs with each input connected to the other nodes. In addition 4 of the C-elements have two transistors connected to *CLK* and *CLKB* to ensure that the output node is set to high impedance during the transparent mode. Similar to the HRDNUT, this reduces the power consumption and the delay for a relatively small increase in area. To vote on the output, the nodes in the block latch are connected to two 3-input C-elements. These C-elements drive and 2-input output C-element. A schematic of the TNU-latch is given in [REFERENCE FIG].

The basis behind the latch is to first ensure that the C-elements in the block latch cannot be driven to an incorrect

value due to a TNU. To ensure this, each C-element has four inputs. If the latch was designed as in the block based latch for the HRDNUT, the output C-element would have 5 inputs. However, among experimentation with this design it was found that an error on the output element would lead to an unrecoverable error. To solve this issue, the output C-element was split into 2 3-input elements which drive a 2-input element. The is effectively removed the error since a TNU can, in the worst case, only flip a single C-element leaving one output element unaffected thus holding the data. This also allows for the latch to tolerate a TNU with an error on the output since only two errors will affect the internal nodes. None of the C-elements can be flipped due to an internal DNU thus allowing for the output to recover.

First we will evaluate the TNU-latch during the transparent mode. In this mode, the data is loaded to nodes $n1$, $n2$, $n3$ and $n4$. This is done when the clock is at a high value which sets the output node to high impedance and turns on the loading pass gates. Once the four nodes are loaded, all the inputs on C-element $C5$ are set such that $n5$ is set to the loaded value. Since nodes $n1$ - $n5$ are all loaded, C-elements $C6$ and $C7$ are set to the loaded value thus driving the output C-element. A waveform of the latch in transparent mode is given in Fig. [REFERENCE FIG].

We will now evaluate the latch for tolerance against soft errors. In the case of a SEU, the latch is tolerant since an SEU cannot change the state of a C-element. Additionally, the latch is DNU tolerant for similar reasons. When a TNU occurs, there are 56 total strike cases. Due to the simple design of the latch, we condense all cases into 6 distinct cases which are given in the following list. Waveforms were generated using equation 2 and the same simulation parameters as in Section 3 to model the pulse shape for each individual case. The waveforms are given in Figs. [REFERENCE FIGURES].

- 1) Consider 3 strikes on the internal nodes $n1$, $n2$ and $n3$. The errors will all propagate to C-element $C5$ but will not cause a change on node $n5$ since $n4$ is not affected. Additionally, the errors will propagate to the inputs of $C1$, $C2$, $C3$ and $C4$. However, since at least $n5$ will be unaffected, the C-elements will hold their state. Next, we look at C-elements $C6$ and $C7$ and note that since $n5$ does not change values, the C-elements hold the correct value on nodes $n6$ and $n7$. Since these nodes have the correct value, node OUT does not change. This analysis holds true for any TNU combination within the set of $[n1, n2, n3, n4$ and $n5]$.
- 2) Assume that nodes $n1$, $n2$ and OUT are struck by a TNU. The errors on the block latch can be treated as a DNU. Since no additional nodes in the block latch are flipped by the error, the C-elements $C6$ and $C7$ can only be affected by at most two errors. This implies that neither element will possibly flip due to an error. Since nodes $n6$ and $n7$ are error-free, $C8$ will drive OUT back to the correct state. All cases where two errors affect the set $[n1, n2, n3, n4, n5]$ will resolve similarly.
- 3) Consider errors on nodes $n1$, $n6$ and $n7$. The error on $n1$ will propagate to internal elements $n2$, $n3$,

$n4$ and $n5$. However, since only a single error is at the inputs of the C-elements they hold their correct value. Additionally, $C1$ is driven back to the error-free value. This allows for nodes $n6$ and $n7$ to also be recovered. All cases where there is one error on the internal block behave the same as this example.

- 4) In the case of a TNU striking nodes $n1$, $n2$ and $n6$, the errors on $n1$ and $n2$ propagate to C-elements $C1$, $C2$, $C3$, $C4$ and $C5$. Since each C-element is driven by 4 nodes, no additional nodes flip value. This leads to two of the nodes on $C6$ having two erroneous inputs. This ensures that the error on $n6$ is not recovered. However, $n7$ remains error-free thus allowing the output to be fully recovered. Any TNU combination that has two errors from the set $[n1, n2, n3, n4, n5]$ and a single error from $n6$ or $n7$ will evaluate similarly.
- 5) Consider errors on nodes $n1$, $n6$ and OUT . In this case the error on $n1$ will be fully recovered as in case 3. Since the error is recovered, all input nodes to $C6$ are error-free allowing for full recovery of the node. $n6$ and $n7$ are also error-free setting the output of $C8$ to the correct value. This behave similarly for any case which has one error in the set of $[n1, n2, n3, n4, n5]$ a single error on $n6$ or $n7$ and an error on OUT .
- 6) Lastly, assume errors on $n6$, $n7$ and OUT . The TNU-latch will fully recover since all of the inputs of $C6$ and $C7$ are error-free driving nodes $n6$ and $n7$ to the correct value. The nodes then drive the output of $C8$ to the previous error-free value.

5 SIMULATION RESULTS

The proposed latch designs were implemented using the 1.05V 32nm PTM library [18] and simulated in HSPICE. All transistor widths for all designs were set to minimum size which is 80nm for PMOS and 40nm for NMOS. All designs were operated at 1 Ghz. We compared the HRDNUT and TNU-latch to existing SEU and DNU tolerant designs. We did not compared to TNU tolerant designs since no other designs are known to exist. For the analysis, we compare to the following SEU tolerant latches: DICE [3], FERST [6] and HIPER [5]. Additionally, we also compare to the following DNU tolerant designs: DNCS [12], Interception [13], HSMUF [14] and DONUT [11]. All transistors for the implemented latches were set to minimum width and length except for the designs that use a C-element with a weak keeper. In these designs the C-element's PMOS width was set to $W=320\text{nm}$ and the NMOS width was set to $W=160\text{nm}$ and the weak keeper was sized to be at minimum width. The C-element was sized so that the output driving strength did not allow the keeper to drive an erroneous value in the event of an error.

For the comparison we measure the propagation delay, average power consumption and area of all design. We then categorize the designs based on the number of errors they can tolerate and if they are robust to the number of errors they tolerate. The delay for each design was calculated based on the difference between the time that input D is at $0.5 * V_{DD}$ and the output at the same point. The average

power was calculated over a 200 ns duration when the latch is error-free. For the calculation of the area, the unit size transistor (UST) metric as adopted in [12] was used. This metric quantifies the expected area based the total transistor area divided by the unit size. For this case, the unit size was set to be 40 nm. Table 1 gives the results of the simulation.

TABLE 1: SPICE Simulations of Existing Latches using the 1.05V 32nm PTM library

Latch	DNU Im-mune	DNU Ro-bust	Power (μ W)	Delay (ps)	Area (UST)
DICE	No	No	1.332	8.145	16
FERST	No	No	3.178	31.648	60
HIPER	No	No	1.292	2.221	27
DNCS	Yes	No	4.948	22.486	61
[13]	Yes	No	5.606	79.168	89
HSMUF	Yes	No	1.871	1.0626	51
HSMUF (Keeper)	Yes	No	3.787	3.945	78
DONUT [11]	Yes	Yes	4.021	14.722	54
DONUT-M (Section 2)	Yes	Yes	2.760	8.421	72
HRDNUT (Proposed)	Yes	Yes	2.450	2.310	66

According to Table 1 the only DNU robust designs are the two DONUT latch implementations and the HRDNUT. Compared to the modified DONUT latch, the HRDNUT provides DNU robustness while reducing the power consumption and number of transistors by 11.3% and 8.33% respectively while also reducing the delay by 72.5%. For the above reasons, the HRDNUT is the best design for clock gating applications due to its high robustness, even after a DNU occurs, and lower power, delay and area overheads.

REFERENCES

- [1] K. M. Zick and J. P. Hayes, "High-level vulnerability over space and time to insidious soft errors," in *2008 IEEE International High Level Design Validation and Test Workshop*, Nov 2008, pp. 161–168.
- [2] J. R. Schwank, M. R. Shaneyfelt, and P. E. Dodd, "Radiation hardness assurance testing of microelectronic devices and integrated circuits: Radiation environments, physical mechanisms, and foundations for hardness assurance," *IEEE Transactions on Nuclear Science*, vol. 60, no. 3, pp. 2074–2100, June 2013.
- [3] T. Calin, M. Nicolaidis, and R. Velazco, "Upset hardened memory design for submicron cmos technology," *IEEE Transactions on Nuclear Science*, vol. 43, no. 6, pp. 2874–2878, Dec 1996.
- [4] M. Nicolaidis, R. Perez, and D. Alexandrescu, "Low-cost highly-robust hardened cells using blocking feedback transistors," in *26th IEEE VLSI Test Symposium (vts 2008)*, April 2008, pp. 371–376.
- [5] M. Omana, D. Rossi, and C. Metra, "High-performance robust latches," *IEEE Transactions on Computers*, vol. 59, no. 11, pp. 1455–1465, Nov 2010.
- [6] M. Fazeli, S. G. Miremadi, A. Ejlali, and A. Patooghy, "Low energy single event upset/single event transient-tolerant latch for deep submicron technologies," *IET Computers Digital Techniques*, vol. 3, no. 3, pp. 289–303, May 2009.
- [7] P. Hazucha, T. Karnik, S. Walstra, B. Bloechel, J. Tschanz, J. Maiz, K. Soumyanath, G. Dermer, S. Narendra, V. De, and S. Borkar, "Measurements and analysis of ser tolerant latch in a 90 nm dual-vt cmos process," in *Custom Integrated Circuits Conference, 2003. Proceedings of the IEEE 2003*, Sept 2003, pp. 617–620.
- [8] R. Rajaei, M. Tabandeh, and M. Fazeli, "Single event multiple upset (semu) tolerant latch designs in presence of process and temperature variations," *Journal of Circuits, Systems and Computers*, vol. 24, no. 01, p. 1550007, 2015. [Online]. Available: <http://www.worldscientific.com/doi/abs/10.1142/S0218126615500073>
- [9] S. Lin, H. Yang, and R. Luo, "types o," in *IEEE Computer Society Annual Symposium on VLSI (ISVLSI '07)*, March 2007, pp. 273–278.
- [10] M. Zhang, S. Mitra, T. M. Mak, N. Seifert, N. J. Wang, Q. Shi, K. S. Kim, N. R. Shanbhag, and S. J. Patel, "Sequential element design with built-in soft error resilience," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 14, no. 12, pp. 1368–1378, Dec 2006.
- [11] N. Eftaxiopoulos, N. Axelos, and K. Pekmetzi, "Donut: A double node upset tolerant latch," in *2015 IEEE Computer Society Annual Symposium on VLSI*, July 2015, pp. 509–514.
- [12] K. Katsarou and Y. Tsiatouhas, "Soft error interception latch: double node charge sharing seu tolerant design," *Electronics Letters*, vol. 51, no. 4, pp. 330–332, 2015.
- [13] —, "Soft error immune latch under seu related double-node charge collection," in *2015 IEEE 21st International On-Line Testing Symposium (IOLTS)*, July 2015, pp. 46–49.
- [14] A. Yan, H. Liang, Z. Huang, and C. Jiang, "High-performance, low-cost, and highly reliable radiation hardened latch design," *Electronics Letters*, vol. 52, no. 2, pp. 139–141, 2016.
- [15] D. R. Blum and J. G. Delgado-Frias, "Schemes for eliminating transient-width clock overhead from set-tolerant memory-based systems," *IEEE Transactions on Nuclear Science*, vol. 53, no. 3, pp. 1564–1573, June 2006.
- [16] —, "Hardened by design techniques for implementing multiple-bit upset tolerant static memories," in *2007 IEEE International Symposium on Circuits and Systems*, May 2007, pp. 2786–2789.
- [17] A. Watkins and S. Tragouodas, "A highly robust double node upset tolerant latch," in *2016 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*, Sept 2016, pp. 15–20.
- [18] W. Zhao and Y. Cao, "Predictive technology model for nano-cmos design exploration," *J. Emerg. Technol. Comput. Syst.*, vol. 3, no. 1, Apr. 2007. [Online]. Available: <http://doi.acm.org/10.1145/1229175.1229176>
- [19] J. F. Ziegler, "Terrestrial cosmic rays," *IBM Journal of Research and Development*, vol. 40, no. 1, pp. 19–39, Jan 1996.

Adam Watkins Biography text here.

PLACE
PHOTO
HERE

Spyros Tragoudas Biography text here.

PLACE
PHOTO
HERE