Increasing Reliability of Latches Through SEU and DNU Tolerant Designs

Nathan Villacampa
Department of Electrical and Computer Engineering
University of Central Florida
Orlando, FL 32816-2362

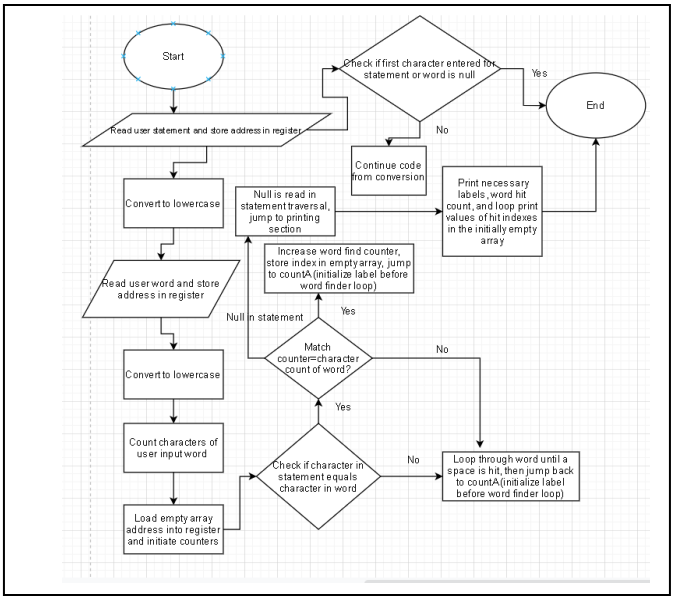
***Abstract*—Circuit elements such as latches are vulnerable to defects and single event/double node upsets (SEU and DNU) at small scale levels. Manufacturing defects and radiation issues all attribute to lower reliability in device performance and inefficiency in power consumption. This article will explore various designs to increase reliability and beneficial trade-offs in the performance of latches by making them more tolerable to SEUs and DNUs with techniques like masking and redundancy. Program inputs used to demonstrate energy consumption include a statement provided with expected output (to test accuracy of count and indexes as well as functionality of the code), a null case, and a case with a statement containing variations of the sought word. The output will be the number of appearances of the sought word within the statement and the indexes or locations of where the word was found. Using the dynamic instruction count of the program, energy consumption was calculated and used as the fundamental metric for comparison between the designs reviewed. The total energy consumption of the best design (DNU-latch[1]) was 20.27pJ of the assembly code using test case 1 presented below.**

Keywords—SEU(single event upset),DNU(double node upset),TMR(triple modular redundancy),DNCS-SEU(Double node charge sharing SEU), Soft error.

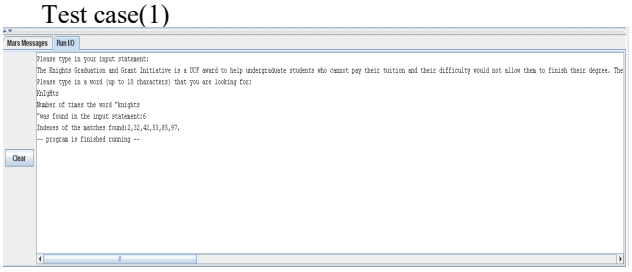
I. Introduction

The coding method used was a cascade of loops and labels to detect the sought word in the user input statement. The code finds words by loading byte values into registers from the statement and the word and running through branch conditionals dependent on whether the values are equivalent. These conditionals are connected to labels that increment the word found counter if the main loop successfully iterates for the character count amount of times (ie. every character in the word and statement was equal for a word traversal), move to next word, or jump to printing section.

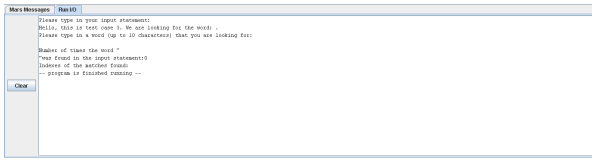
The first test input will be the statement and sought word provided with expected outputs to test the functionality of the code. The next two inputs will be nothing entered for the statement and random word and vice versa to test if the code correctly terminates without entering an infinite loop. The last input will be a random statement with multiple variations of the sought word. The code outputs for each of the test cases should be the correct number of occurrences of the sought word in the



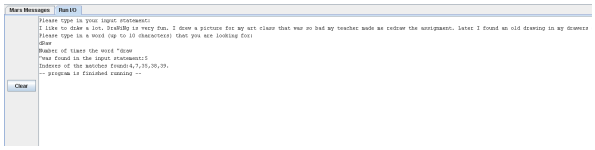
Test case(1)



Test case (3)



Test case (4)



statement as well as the correct indexes where the word was found.

A. Project Design

The code begins by filling in empty arrays with the user input statement and sought word sought. After checking if anything was entered (if nothing, end program to avoid infinite loops), the code then converts all letters in the statement to lowercase, and converts the word to lowercase after reading the word so that all variations of lowercase and uppercase letters that form the word get counted as an occurrence. A character count loop is then running to store the amount of characters into a register to be used in the main word finding loop. After initializing counters, the code jumps to the word finding loop which operates by loading individual bytes from the statement and word and comparing them via a list of branch conditionals which branch if either the bytes do not match, null is reached in the statement, or word find loop has successfully iterated with no branches resulting in a find. If the bytes do not match, the loop branches to another loop which traverses through the statements current word before resetting and initializing above the word find loop. If null is reached in the statement, the loop is finished and branches to the printing section of the code. If a word is found, then the loop branches to increment hit counter and store the index. During all of this, several counters for index, word find, character traversal etc. are being incremented. After the word find loop ends, the print label contains code that prints all necessary strings and values. Another loop is contained in this section to print all the indexes where the sought word was found before ending the program.

Input statement:	Sought word:
(1) The Knights Graduation and Grant Initiative is a UCF award to help undergraduate students who cannot pay their tuition and their difficulty would not allow them to finish their degree. The Knights Success Grant is the most well-known program inside the Knights Graduation and Grant Initiative. In order to be awarded the Knights Success Grant, you need to be referred but it does not mean that all students who are referred will be awarded the grant. The students who want to apply for the Knights Success Grant need to submit a required application and complete the Knights Success Grant web course. For more information, you can stop by their office in the Registrar's Office on the main campus of UCF.	KnIgHts Expected output: 6 occurrences, indexes: 2,32,42,53,85,97
(2)	pizza
(3) Hello, this is test case 3. We are looking for the word: .	
(4) I like to drAw a lot. DraWiNg is very fun. I drew a picture for my art class that was so bad my teacher made me redraw the assignment. Later I found an old drawing in my drawers drawn from scratch back when I was in middle school.	dRaw

B. Test cases

The first set of input statement and word was chosen because the correct expected output is known, and the very long statement allows us to see if the code is accurate in detecting and storing the correct indexes of the sought word. The second and third sets were tested to make sure never-ending loops didn't occur if nothing was entered by the user for any of the inputs. The last test was to test the sensitivity of the word finder.

II. Reliability Bit-Cells

Around the nano-scale level, radiation is a common challenge for reliability in technological devices. Single event upsets or SEUs are caused by radiation, decreasing the reliability of flip flops and latches. In manufacturing, due to the small scale of the devices, devices like latches are vulnerable to cluster defects and variations of length and other physical measurements discussed in [2]. A future solution is through nano-fabrication techniques in which devices are accurately created by using photolithography presented in [2].

A conventional solution to the issue of radiation is spatial redundancy, which utilizes duplicating latches. Triple Modular Redundancy involves using three duplicated latches connected to a "voter" circuit that selects the value that most of the latches provide [1]. Although this improves the reliability and tolerance of radiation, the trade-off is more space is required and more power is consumed. Implementing the voter circuit also introduces a reliability on the voter. [2] presents the Inherent Voter Based Scheme which uses inverters and interconnection between columns to remove the voter block from a centralized position, making TMR implementation more tolerable to voter errors (error recovery and "Bad Row" identification algorithms). This design is significant because the reliability of the voter contained in the critical data-path of a circuit is very important. A single fault may result in a much bigger soft error in SRAM or other systems that utilize multiple latches.

A more specific concern towards SEU issues is double node charge sharing SEUs. DNCS-SEUs affect latch operation and bring soft errors (consequently an accelerating decrease in reliability) towards electronic devices. This issue is a major concern to SRAM devices because it invokes vulnerability to soft errors, which can cascade into a lot of software faults caused by hardware. A design technique to increase tolerance of DNCS-SEUs proposed in [3] is a latch that uses two DICE cells. Each DICE cell contains 2 nodes so that, if a transient change or pulse occurs, the correct value can still be returned. The trade-off with this design, is that a large amount of space is consumed in comparison to the initial latch by itself.

III. Results and Discussion

For the energy consumption calculations, test case 1 is used because the input statement is relatively large, making it easier to discern changes in energies consumed between differing designs.

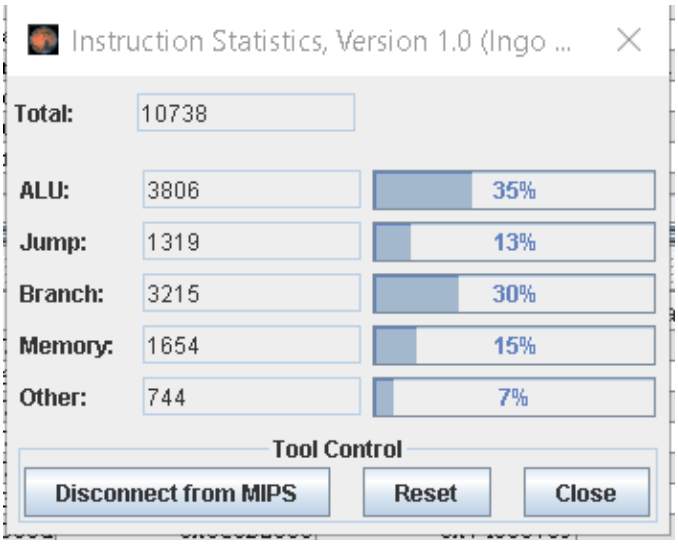


Table I: Energy consumption for a single bit-cell memory in the designs provided in [1-3].

Design	Energy consumption of a Single Bit-Cell Memory
SEU-Latch [1]	0.88 fJ
DNU-Latch [1]	0.28 fJ
[2]	6.96 fJ
[3]	1.51 fJ

alu	3806	3806 fj	Table 1:	Energy consumption (test 1)
jump	1319	2638 fj	Design	memory per design
branch	3215	9645 fj	SEU-Latch[1]	1455.52 fj
memory	1654	Table 1	DNU-Latch[1]	463.12 fj
other	744	3720 fj	[2]	11511.84 fj
total	10738	fj	[3]	2497.54 fj
exmemtotal		19809 fj		

Grand energy consumption:	
SEU-Latch[1]	21264.52 fj
DNU-Latch[1]	20272.12 fj
[2]	31320.84 fj
[3]	22306.54 fj

Table II: Total Energy consumption for the assembly program using designs provided in [1-3].

Design	Total Energy Consumption
SEU-Latch [1]	21.26452 picojoules
DNU-Latch [1]	20.27212 picojoules
[2]	31.32084 picojoules
[3]	22.30654 picojoules

IV. Conclusion

Various designs that promote defect resilience have been exposed through sources [1-3]. From the energy values calculated in Table II using the assembly program with test case 1, it can be concluded that the best design in terms of energy consumption and space is the DNU-latch design presented in [1]. The trade-off in comparison to other designs is that there's reliance introduced from implementing a voter into the critical path of the circuit and doesn't account for DCNS-SEU. Technical topics learned from this project include radiation induced upsets (SEU,DNU), spatial redundancy, approaches to increase tolerance (TMR, Inherent Voter Based Scheme), designs to improve reliability (DICE), and growing innovation of use of photolithography to decrease manufacturing defects and other measurement issues presented in [2].

References

[1] F. S. Alghareb, R. Zand and R. F. Demara, "Non-Volatile Spintronic Flip-Flop Design for Energy-Efficient SEU and DNU Resilience," in IEEE Transactions on Magnetics, vol. 55, no. 3, pp. 1-11, March 2019, Art no. 3400611.

[2] H. Pourmeidani and M. Habibi, "Hierarchical defect tolerance technique for NRAM repairing with range matching CAM," 2013 21st Iranian Conference on Electrical Engineering (ICEE), Mashhad, 2013, pp. 1-6.

[3] K. Katsarou and Y. Tsiatouhas, "Double node charge sharing SEU tolerant latch design," 2014 IEEE 20th International On-Line Testing Symposium (IOLTS), Platja d'Aro, Girona, 2014, pp. 122-127.