How Reliability is improved by the use of Triple Modular Redundancy

Gabriela Pinedo
Department of Electrical and Computer Engineering
University of Central Florida
Orlando, FL 32816-2362

Abstract— ***In this paper, we are looking at how Triple Modular Redundancy is used to improve the reliability of a digital system. As always there are tradeoffs that have to be scarified in one area in order to improve another, this one being the amount of energy and space being consumed within a system in order to make it more able to output the desired result that the system was designed for. Inputs in our system are the word that is being searched within the statement that the user also provides. The outputs are the amount of times the word is found the indices of where they were found within the statement.***

Keywords— **Triple Modular Redundancy (TMR), critical datapath, spintronic based memory, reliability, voter.**

I. INTRODUCTION

A. Project Design

In this project the first thing that is asked of is for the user to input a statement that is going to be used to search how many times a particular word appears within said statement. After that statement has been given, the user will then input the word that is being searched for within the statement. After those inputs have been given, a counter for the amount of times the word has been discovered is initialized to 0, the statement addresses, and word addresses are loaded, and the word is searched for by seeing if each character matches within a word of the statement. If there is a mismatch, the program goes onto the next word of the statement and the word that is being searched is reloaded to start looking for a match at the beginning of its spelling. If there is a match the counter goes up by one, the index of where it was found in the statement is recorded, and it moves onto the next word in the statement to see if that matches the word we are searching for. This continuously repeats until it reaches the end of the statement, and a final output is displayed. Fig. 1 shows a flowchart of the assembly program that is described above.

Fig.1: Flowchart of the assembly program.

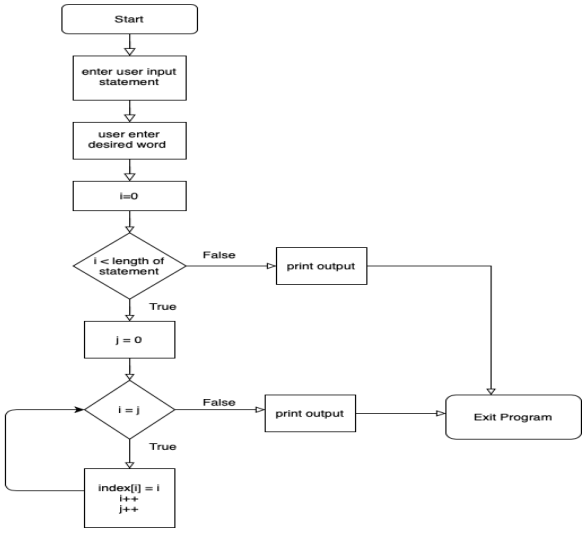

Fig.2: Sample outputs of the program.

```
# Please type in your input statement:

  The fact that there's a stairway to heaven and a highway to hell explains life well.

# Please type in a word (up to 10 characters) that you are looking for
(e.g Knight or KnIGhT, knight,...):

 t0

# Number of times the word " t0" was found in the input statement: 2

# Indexes of the matches found:
```

```
# Please type in your input statement:

  She did a happy dance because all of the socks from the dryer matched.

# Please type in a word (up to 10 characters) that you are looking for
(e.g Knight or KnIGhT, knight,...):

 SUckS

# Number of times the word " SUckS" was found in the input statement: 0

# Indexes of the matches found:
```

```
# Please type in your input statement:

  He bit a bitter bite.

# Please type in a word (up to 10 characters) that you are looking for
(e.g Knight or KnIGhT, knight,...):

 bIt

# Number of times the word " bIt" was found in the input statement: 3

# Indexes of the matches found:
```

B. *Test Cases*

To test my program, I selected for there to be 3 different size length statements to be input as the statement to see how length of the statement is not a factor that affects my program. I as well input a word that is found in the statement as my word that is being searched for, as well as a word that is not found in my statement, and the words are all different size lengths as well. I wanted to give a variety of inputs for it to show that my program can run under any circumstance that is being tested with. Another thing was that regardless of the way the user typed in either the statement or the word we are searching for as far as in Uppercase or Lowercase or a combination of both doesn't matter, as long as it is the same word it will catch it. Therefore, case sensitivity does not apply.

II. Reliability Bit-Cells

Triple Modular Redundancy (TMR) is a technique that helps improve reliability of digital systems by having the use of a two-out-of-three-voting concept at a low level. When using a TMR approach, you can increase improvements in reliability but since there will be added hardware in this process, more energy will be used throughout the program. Reliability within the voter within the critical Datapath is essential to the overall circuit operation because if there are failures within the voters then the overall reason of reliability is also bound to fail. All of the outputs will be considered corrupted and the use of extra hardware and energy would have gone to waste. TMR will still work if one voter fails, but anything more than that, will cause for the whole technique to be unsuccessful. The circuit will no longer produce the desired output.

III. Results and Discussion

1) *ALU = 1 fJ*
2) *Branch = 3 fJ*

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

3) *Jump = 2 fJ*
4) *Memory = Refer to Table I*
5) *Other = 5 fJ*

= (1X1248)+(3X462)+(2X811)+(.88X193)+(5X736)= 8105 fJ
= (1X1248)+(3X462)+(2X811)+(.28X193)+(5X736)=7990 fJ
= (1X1248)+(3X462)+(2X811)+(6.96X193)+(5X736)=9279fJ
= (1X1248)+(3X462)+(2X811)+(1.51X193)+(5X736)= 8227fJ

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

Design	Total Energy Consumption
SEU-Latch [1]	8105 fJ
DNU-Latch [1]	7990 fJ
[2]	9279 fJ
[3]	8227 fJ

IV. Conclusion

The main thing I observed was that triple modular redundancy is a common method that is used to make systems dependable on outputting the desired result needed. The total energy consumption of the best design is the DNU-Latch design with 7990fJ. Things I have learned about this project include one of the main topics of triple modular redundancy and its purpose of being used in a system. Another thing I have learned is about how critical data paths are what determines how fast a system is. The next thing I learned is what voters are and how they are crucial to the TMR technique.

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.

[4] R. E. Lyons and W. Vanderkulk, "The Use of Triple-Modular Redundancy to Improve Computer Reliability," in *IBM Journal of Research and Development*, vol. 6, no. 2, pp. 200-209, April 1962.

[5] J. F. Wakerly, "Microcomputer reliability improvement using triple-modular redundancy," in *Proceedings of the IEEE*, vol. 64, no. 6, pp. 889-895, June 1976. This article explains the improvement aspect of

[6] U. Afzaal and J. Lee, "A Self-Checking TMR Voter for Increased Reliability Consensus Voting in FPGAs," in *IEEE Transactions on Nuclear Science*, vol. 65, no. 5, pp. 1133-1139, May 2018