IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 5, Issue 1, Ver. III (Jan - Feb. 2015), PP 45-48
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197
www.iosrjournals.org

Highly Reliable Parallel Filter Design Based On Reduced Precision Error Correction Codes

Booviga Priyadharsini.S[1], Saranya.T[2], Sneha.S[3], Suganya.C[4], Manjula.G[5,]
[1, 2, 3, 4]*(ECE, Sasurie College Of Engineering/Anna University, India)*
[5]*(Assistant Professor, ECE, Sasurie College Of Engineering/Anna University, India)*

Abstract: *The project is mainly focusing on multiple error detection and correction. Digital filters are widely used in signal processing and communication n systems. In some cases, the reliability of those systems is critical and fault tolerant implementations are needed. So that, the idea is generalized to show that parallel FIR filters can be protected using error correction codes. Triple Modular Redundancy (TMR) is the traditional mitigation techniques for Field-Programmable Gate Arrays (FPGAs) subject to Single-Event Upsets (SEUs) in high radiation environment. To overcome the problem, in our project we propose RFFF (Reduced Faultless FIR Filter) is a technique combined with TMR used to multiple errors are detected and corrected. TMR increases the parameters like area, power and delay. In RFFF, multiple errors are corrected and the comparison of parameters like area, power and delay of existing and the proposed technique is done.*

Keywords: *Error correction codes (ECCs), filters, Reduced Faultless FIR Filter (RFFF), Triple Modular Redundancy (TMR).*

I. Introduction

A number of techniques can be used to protect a circuit from errors. Those range from modifications in the manufacturing process of the circuits to reduce the number of errors to adding redundancy at the logic or system level to ensure that errors do not affect the system functionality. To add redundancy, a general technique known as triple modular redundancy (TMR) can be used. The TMR, which triplicates the design and adds voting logic to correct errors, is commonly used. However, it more than triples the area and power of the circuit, something that may not be acceptable in some applications. When the circuit to be protected has algorithmic or structural properties, a better option can be to exploit those properties to implement fault tolerance. Digital filters are one of the most commonly used signal processing circuits and several techniques have been proposed to protect them from errors. Most of them have focused on finite-impulse response (FIR) filters.

To operate reliably in space, a hardware mitigation strategy, such as triple modular redundancy (TMR), must be applied. TMR, however, is very expensive and requires three times more hardware resources than an unmitigated circuit. Motivated by the observation that an FPGA-based radio comprises mostly arithmetic operations, this paper explores the application of reduced faultless FIR filter (RFFF) to the problem. The metric used to evaluate the effectiveness of RFFF is the bit error rate (BER) achieved by the FPGA-based radio. To fully evaluate the benefits of RFFF on a communications system, the impact of ionizing radiation on BER must be well understood.

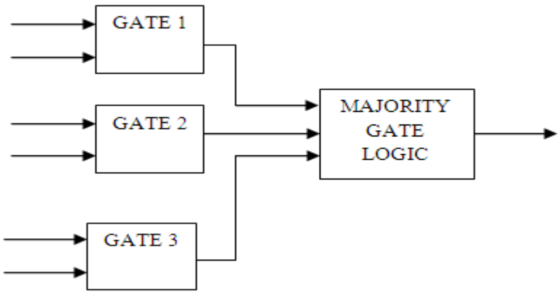

Fig. 1. Triple Modular Redundancy

II. Literature Survey

Triple Modular Redundancy has three identical logic circuits (logic gates) are used to compute the specified Boolean function. The set of data at the input of the first circuit are identical to the input of the second and third gates. In computing, triple modular redundancy, sometimes called triple-mode redundancy. TMR is a fault-tolerant form of N-modular redundancy, in which three systems perform a process and that result is processed by a majority-voting system to produce a single output. If any one of the three systems fails, the other two systems can correct and mask the fault. The TMR concept can be applied to many forms of redundancy, such as software redundancy in the form of N-version programming, and is commonly found in fault-tolerant computer systems. Some ECC memory uses triple modular redundancy hardware (rather than the more common Hamming code), because triple modular redundancy hardware is faster than Hamming error correction software.

TMR with Hamming Code

The new technique is based on the use of the ECCs. A simple ECC takes a block of k bits and produces a block of n bits by adding n−k parity check bits. The parity check bits are XOR combinations of the k data bits. By properly designing those combinations it is possible to detect and correct errors. As an example, let us consider a simple Hamming code with k = 4 and n = 7. In this case, the three parity check bits p1, p2, p3 are computed as a function of the data bits d1, d2, d3, d4 as follows:

$$p1 = d1 \oplus d2 \oplus d3$$
$$p2 = d1 \oplus d2 \oplus d4$$
$$p3 = d1 \oplus d3 \oplus d4$$

For the case of four filters y1, y2, y3, y4 and the Hamming code, the check filters would be

$$z_1[n] = \sum_{l=0}^{\infty}(x_1[n-l]+x_2[n-l]+x_3[n-l]).h[l]$$
$$z_2[n] = \sum_{l=0}^{\infty}(x_1[n-l]+x_2[n-l]+x_4[n-l]).h[l]$$
$$z_3[n] = \sum_{l=0}^{\infty}(x_1[n-l]+x_3[n-l]+x_4[n-l]).h[l]$$

Error Detection and Correction

$$z_1[n]= y_1[n]+ y_2[n]+y_3[n]$$
$$z_2[n]= y_1[n]+ y_2[n]+y_4[n]$$
$$z_3[n]= y_1[n]+ y_3[n]+y_4[n]$$

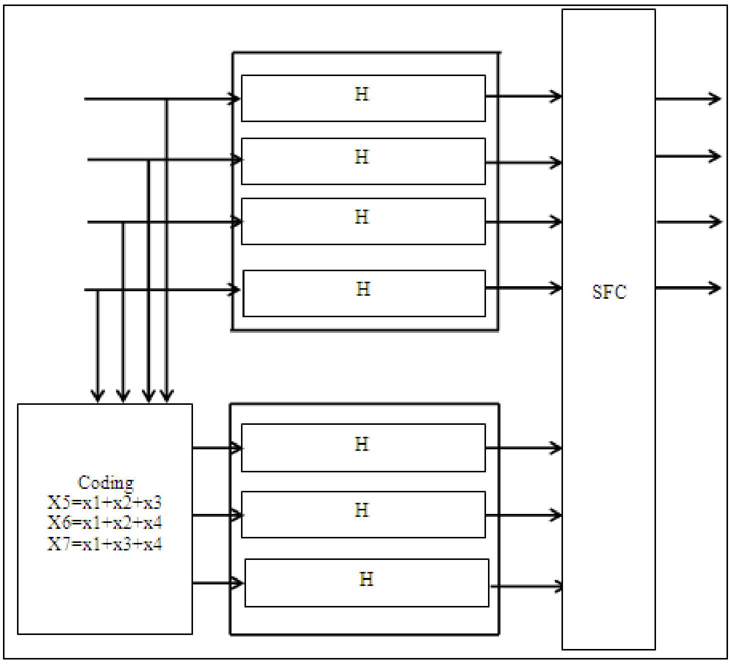

Fig. 2. Four filters and hamming codes

For example, an error on filter y1 will cause errors on the checks of z1, z2, and z3. Similarly, errors on the other filters will cause errors on a different group of zi. Therefore, as with the traditional ECCs, the error can be located and corrected. For the filters, correction is achieved by reconstructing the erroneous outputs using the rest of the data and check outputs. For example, when an error on y1 is detected, it can be corrected by making

$$y_{c1}[n]= z[n]- y_2[n]-y_3[n]$$

Similar equations can be used to correct errors on the rest of the data outputs and calculate s = yHT to detect errors. Then, the vector s is also used to identify the filter in error. In our case, a nonzero value in vector s is equivalent to 1 in the traditional Hamming code.

Table I: Error Location In The Hamming Code

s1 s2 s3	Error Bit Position	Action
000	No error	None
111	d1	Correct d1
110	d2	Correct d2
101	d3	Correct d3
011	d4	Correct d4
100	p1	Correct p1
010	p2	Correct p2
001	p3	Correct p3

III. Reduced Faultless Fir Filter

RFFF is a redundancy technique similar to TMR that requires less hardware overhead by using reduced-precision (RP) arithmetic in two of its three replicas. It takes advantage of the fact that RF arithmetic can be a good estimate of computations that use higher precision. When TMR protects the entire circuit and provides an error-free output, RFFF simply limits the error at the output of a module.

RFFF has an advantage over TMR when it is able to sufficiently limit the magnitude of the SEU-induced noise at a lower hardware cost. RPR is not suited to protect any type of circuitry as TMR is the decision hardware required

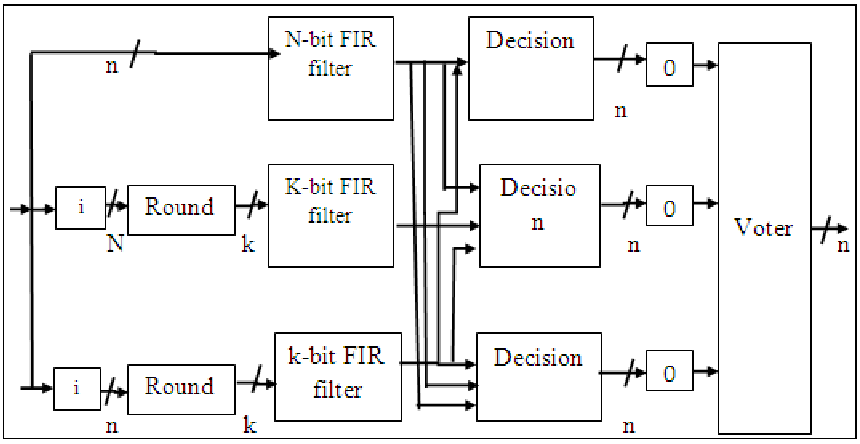

Fig. 3. Block diagram of an n-bit FIR filter protected with RFFF using k-bit RF filters (k<n)

Implementation of RFFF with TMR

It shows a block diagram of an n-bit finite impulse response (FIR) filter (a filter with n-bit registers and coefficients) protected with RFFF. Note that the decision blocks and outputs can be triplicates as well to avoid single points of failure in those modules. The outputs of the three identical decision blocks are voted on, as in the TMR system.

Table II: Implementation Result

PARAMETRS	EXISTING	PROPOSED
AREA	59.656	37.680
POWER	1364mw	1108mw
DELAY	14.873ns	11.485ns

IV. Conclusion

This project is developed for error detection and correction. In our concept TMR can do only one error at a time .To overcome this problem we undergo a new technique called RFFF-TMR .The concept RFFF with TMR is used to detect the multiple errors. It provides high frequency and low power, delay, area .Our project is only demonstrates RPR on a small set of test cases, these are the first experiment using fault injection that fully characterize the effect of each configuration SEU in FPGA.

Future work could include examinations of RFFF for protecting a wide array of computation modules as well as evaluations of cost and benefits of RFFF in lager systems. Future analysis could also focus on the trade offs of changing the RPR redundancy factor effectively applying mare or less protection to the system.

References

[1]. M. Nicolaidis, "Design for soft error mitigation," IEEE Trans. Device Mater. Rel., vol. 5, no. 3, pp. 405–418, Sep. 2005.

[2]. A. Reddy and P. Banarjee "Algorithm-based fault detection for signal processing applications," IEEE Trans. Comput., vol. 39, no. 10, pp. 1304–1308, Oct. 1990.

[3]. B. Shim and N. Shanbhag, "Energy-efficient soft error-tolerant digital signal processing," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 14, no. 4, pp. 336–348, Apr. 2006.

[4]. T. Hitana and A. K. Deb, "Bridging concurrent and non-concurrent error detection in FIR filters," in Proc. Norchip Conf., 2004, pp. 75–78.

[5]. Y.-H. Huang, "High-efficiency soft-error-tolerant digital signal processing using fine-grain subword-detection processing," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 18, no. 2, pp. 291–304,Feb. 2010.

[6]. S. Pontarelli, G. C. Cardarilli, M. Re, and A. Salsano, "Totally fault tolerant RNS based FIR filters," in Proc. IEEE IOLTS, Jul. 2008, pp. 192–194.

[7]. Z. Gao, W. Yang, X. Chen, M. Zhao, and J. Wang, "Fault missing rate analysis of the arithmetic residue codes based fault-tolerant FIR filter design," in Proc. IEEE IOLTS, Jun. 2012, pp. 130–133.

[8]. P. Reviriego, C. J. Bleakley, and J. A. Maestro, "Structural DMR: A technique for implementation of soft-error-tolerant FIR filters," IEEE Trans. Circuits Syst., Exp. Briefs, vol. 58, no. 8, pp. 512–516, Aug. 2011.

[9]. P. P. Vaidyanathan. Multirate Systems and Filter Banks. Upper Saddle River, NJ, USA: Prentice-Hall, 1993.

[10]. A. Sibille, C. Oestges, and A. Zanella, MIMO: From Theory to Implementation. San Francisco, CA, USA: Academic Press, 2010.

[11]. P. Reviriego, S. Pontarelli, C. Bleakley, and J. A. Maestro, "Area efficient concurrent error detection and correction for parallel filters," IET Electron. Lett, vol. 48, no. 20, pp. 1258–1260,Sep. 2012.

[12]. A. V. Oppenheim and R. W. Schafer, Discrete Time Signal Processing. Upper Saddle River, NJ, USA: Prentice-Hall 1999.

[13]. S. Lin and D. J. Costello, Error Control Coding, 2nd ed. Englewood Cliffs, NJ, USA: Prentice-Hall. 2004.

[14]. R. W. Hamming, "Error correcting and error detecting codes," Bell Syst. Tech. J., vol. 29, pp. 147–160, Apr. 1950.