Design and Analysis of Fault Tolerant Parallel Filters based on Error Correction Codes

Ramya Sree Maharaj*, G Kumaraswamy**
*M.Tech student, Dept of ECE, CMRIT, Hyderabad, TS, India.
**Associate Professor, Dept of ECE, CMRIT, Hyderabad, TS, India.

Abstract- Digital filters are widely used in signal processing and communication systems. In some cases, the reliability of those systems is critical, and fault tolerant filter implementations are needed. Over the years, many techniques that exploit the filters' structure and properties to achieve fault tolerance have been proposed. As technology scales, it enables more complex systems that incorporate many filters. In those complex systems, it is common that some of the filters operate in parallel, for example, by applying the same filter to different input signals. Recently, a simple technique that exploits the presence of parallel filters to achieve fault tolerance has been presented. In this brief, that idea is generalized to show that parallel filters can be protected using error correction codes (ECCs) in which each filter is the equivalent of a bit in a traditional ECC. This new scheme allows more efficient protection when the number of parallel filters is large. The technique is evaluated using a case study of parallel finite impulse response filters showing the effectiveness in terms of protection and implementation cost.

Keywords- ECC, Faults, Digital signals, Fault tolerant, parallel filters, errors, efficiency.

I. INTRODUCTION

The complexity of communications and signal processing circuits increases every year. This is made possible by the CMOS technology scaling that enables the integration of more and more transistors on a single device. This increased complexity makes the circuits more vulnerable to errors. At the same time, the scaling means that transistors operate with lower voltages and are more susceptible to errors caused by noise and manufacturing variations [1]. The importance of radiation-induced soft errors also increases as technology scales [2]. Soft errors can change the logical value of a circuit node creating a temporary error that can affect the system operation. To ensure that soft errors do not affect the operation of a given circuit, a wide variety of techniques can be used [3]. These include the use of special manufacturing processes for the integrated circuits like, for example, the silicon on insulator. Another option is to design basic circuit blocks or complete design libraries to minimize the probability of soft errors. Finally, it is also possible to add redundancy at the system level to detect and correct errors. One classical example is the use of triple modular redundancy (TMR) that triples a block and votes among the three outputs to detect and correct errors. The main issue with those soft errors mitigation techniques is that they require a large overhead in terms of circuit implementation. For example, for TMR, the overhead is >200%. This is because the unprotected module is replicated three times (which requires a 200% overhead versus the unprotected module), and additionally, voters are needed to correct the errors making the overhead >200%. This overhead is excessive for many applications. Another approach is to try to use the algorithmic properties of the circuit to detect/correct errors. This is commonly referred to as algorithm-based fault tolerance (ABFT) [4]. This strategy can reduce the overhead required to protect a circuit. Signal processing and communications circuits are well suited for ABFT as they have regular structures and many algorithmic properties [4]. Over the years, many ABFT techniques have been proposed to protect the basic blocks that are commonly used in those circuits. Several works have considered the protection of digital filters [5], [6]. For example, the use of replication using reduced precision copies of the filter has been proposed as an alternative to TMR but with a lower cost [7]. The knowledge of the distribution of the filter output has also been recently exploited to detect and correct errors with lower overheads [8]. The protection of fast Fourier transforms (FFTs) has also been widely studied.

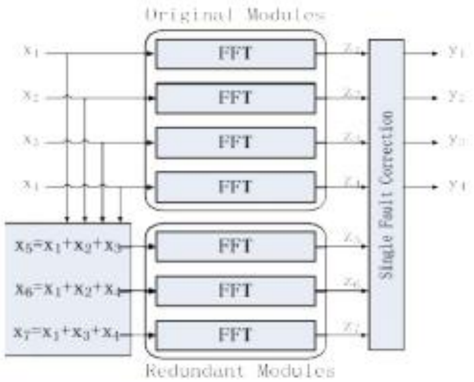

Fig.1: Block diagram.

II. RELATED STUDY

In this brief, a general scheme to protect parallel filters is presented. As in [11], parallel filters with the same response that process different input signals are considered. The new approach is based on the application of error correction codes (ECCs) using each of the filter outputs as the equivalent of a bit in and ECC codeword. This is a generalization of the scheme presented in [11] and enables more efficient implementations when the number of parallel filters is large. The scheme can also be used to provide more powerful protection using advanced ECCs that can correct failures in multiples modules. Fast Fourier transform is used to convert a signal from time domain to frequency & this is needed so that you can view the frequency components present in a signals. If you know the frequency components present in a signals you can play with the signals :) Let's say, u want to design a low pass filter and want to decide on the cut off frequency of the filter. If you have the frequency domain details for a signals u can clearly identify the frequency components which u want to retain & the ones which u want to take out[2]. Environmental interference and physical defects in the communication medium can cause random bit errors during data transmission. Error coding is a method of detecting and correcting these errors to ensure information is transferred intact from its source to its destination. Error coding is used for fault tolerant computing in computer memory, magnetic and optical data storage media, satellite and deep space commu nications, network communications, cellular telephone networks, and almost any other form of digital data communication. Error coding uses mathematical formulas to encode data bits at the source into longer bit words for transmission. The "code word" can then be decoded at the destination to retrieve the information. The extra bits in the code word provide redundancy that, according to the coding scheme used, will allow the destination to use the decoding process to determine if the communication medium introduced errors and in some cases correct them so that the data need not be retransmitted. Different error coding schemes are chosen depending on the types of errors expected, the communication medium's expected error rate, and whether or not data retransmission is possible. Faster processors and better communications technology make more complex coding schemes, with better error detecting and correcting capabilities, possible for smaller embedded systems, allowing for more robust communications.

III. AN OVERVIEW OF PROPOSED SYSTEM

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 [13]. 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 [14] 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: $p1 = d1 \oplus d2 \oplus d3$ $p2 = d1 \oplus d2 \oplus d4$ $p3 = d1 \oplus d3 \oplus d4$. (3) The data and parity check bits are stored and can be recovered later even if there is an error in one of the bits. This is done by recomputing the parity check bits and comparing the results with the values stored. In the example considered, an error on d1 will cause errors on the three parity checks; an error on d2 only in p1 and p2; an error on d3 in p1 and p3; and finally an error on d4 in p2 and p3. Therefore, the data bit in error can be located and the error can be corrected. This is commonly formulated in terms of the generating G and parity check H matrixes. Parseval's method is one of the techniques to detect errors parallel in multiple FFT. This is achieved with Sum of Squares (SOSs) check [5] based on Parseval's theorem. The error free FFT should have its Sum of Squares of the input equaling the Sum of Squares of its frequency domain output. This correlation can be used to identify errors with minimum overhead. For parallel FFTs, the Parseval's check can be combined with the error correction codes to minimize the area overhead. Multiple error detection and correction is achieved through this combination. One of the easy ways is to generate the redundant input for single FFT with all the four FFT inputs. To correct error the parity FFT output is XORed with fault free outputs of the FFTs. Compared to the previous schemes presented in the Fault Tolerant Parallel FFTs Using Error Correction Codes and Parseval Checks [1], this technique reduced the total number of Sum of Squares used. Another existing work done is by combining SOS checks with hamming codes instead of using Parseval's check individually as shown in Figure 2.

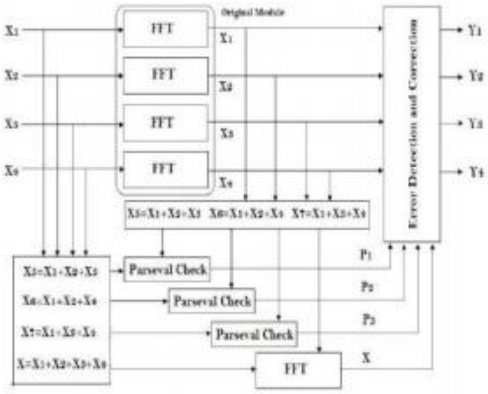

Fig.2: Proposed FFT model.

In all the techniques discussed, soft errors can also affect the elements added for protection. For the ECC technique, the protection of these elements was discussed. Similar correction equations can be used to correct errors on the other modules. More advanced ECCs can be used to correct errors on multiple modules if that is needed in a given application. The overhead of this technique, as discussed, is lower than TMR as the number of redundant FFTs is related to the logarithm of the number of original FFTs. For example, to protect four FFTs, three redundant FFTs are needed, but to protect eleven, the number of redundant FFTs in only four. This shows how the overhead decreases with the number of FFTs.

IV. CONCLUSION

Detecting and correcting errors such as critical reliability are difficult in signal processing which increases the use of fault tolerant implementation. In modern signal processing circuits, it is common to find several filters operating in parallel. Proposed is an area efficient technique to detect and correct single errors. This brief has presented a new scheme to protect parallel FFT using cordic that is commonly found in modern signal processing circuits. The approach is based on applying SOS-ECC check to the parallel FFT outputs to detect and correct errors. The SOS checks are used to detect and locate the errors and a simple parity FFT is used for correction. The 8 point FFT with the input bit length 32 is protected using the proposed technique. . The detection and location of the errors can be done using an SOS check per FFT or alternatively using a set of SOS checks that form an ECC. This technique can detect and correct only single bit error and it reduces area results in high speed compared to existing techniques.

V. REFERENCES

[1]. R. Baumann. 2005. Soft errors in advanced computer systems. IEEE Des. Test Comput. 22(3): 258-266.

[2]. M. Ergen. 2009. Mobile Broadband-Including WiMAX and LTE. New York, NY, USA: SpringerVerlag.

[3]. Z. Gao et al. 2015. Fault tolerant parallel filters based on error correction codes. IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 23(2): 384-387.

[4]. R. W. Hamming. 1950. Error detecting and error correcting codes. Bell Syst. Tech. J. 29(2): 147-160.

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

[6]. J. Y. Jou and J. A. Abraham. 1988. Fault-tolerant FFT networks. IEEE Trans. Comput. 37(5): 548-561.

[7]. N. Kanekawa, E. H. Ibe, T. Suga and Y. Uematsu. 2010. Dependability in Electronic Systems: Mitigation of Hardware Failures, Soft Errors, and ElectroMagnetic Disturbances. New York, NY, USA: Springer-Verlag.

[8]. E. P. Kim and N. R. Shanbhag. 2012. Soft Nmodular redundancy. IEEE Trans. Comput. 61(3): 323-336.

[9]. M. Nicolaidis. 2005. Design for soft error mitigation. IEEE Trans. Device Mater. Rel. 5(3): 405- 418.

[10]. S. Pontarelli, G. C. Cardarilli, M. Re and A. Salsano. 2008. Totally fault tolerant RNS based FIR filters. In Proc. 14th IEEE Int. On-Line TestSymp. (IOLTS). pp. 192-194.

[11]. A. L. N. Reddy and P. Banerjee. 1990. Algorithmbased fault detection for signal processing applications. IEEE Trans. Comput. 39(10): 1304- 1308.

[12]. P. Reviriego, C. J. Bleakley and J. A. Maestro. 2012. A novel concurrent error detection technique for the fast Fourier transform. In Proc. ISSC, Maynooth, Ireland. pp. 1-5.