IOSR Journal of VLSI and Signal Processing (IOSR-JVSP)
Volume 6, Issue 3, Ver. II (May. -Jun. 2016), PP 100-106
e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197
www.iosrjournals.org

Error Correction Technique for Multiple Errors in Parallel FIR Filters

Amritha P.S[1], Ashwath Rao[2]

[1]*ECE dept/ Sahyadri College of Engineering and Management, India*
[2]*Associate Professor ECE dept/ Sahyadri College of Engineering and Management, India*

Abstract: *Filters are broadly utilized as a part of communication and signal processing systems. At times, reliability becomes critical for such systems and fault tolerant usages are required. Over a long time, numerous procedures have been proposed that makes full use of filters structure and properties to accomplish fault tolerance. In certain complex systems, it is commonly found that filters work in parallel. For instance, by applying the same filter to various information signals. In this brief, idea is summed up to demonstrate that protection of parallel filters can be done by utilizing error correction codes (ECCs) by detecting and correcting multiple errors. This new scheme permits more effective assurance if the number of parallel filters is large. Proposed structure is coded using Verilog and synthesized in Xilinx 10.1 and cadence encounter is done.*

Keywords: *Error Correction Codes (ECCs), Filters, and multiple bits.*

I. Introduction

Electronic circuits are progressively present in medical, automotive, space applications etc where reliability of the circuit is critical. In order to protect circuits from errors, fault tolerant implementation need to be done. With the improvement in VLSI technology, the ongoing need of FIR filters with less hardware usage giving error free results and less latency has turned out to be more important. Various methods can be utilized to shield a circuit from errors. One among the general techniques which can be utilized to correct the error is triple modular redundancy (TMR) [1]. The TMR technique adds redundancy by tripling the data and utilizes voting logic to correct errors. Because of the triplication in the technique, it increases area and power of the circuit, something that may not be accepted in a few applications.

If the circuit which has to be protected has structural or algorithmic properties, other techniques can be used in protecting circuits from errors. One case is signal processing circuits for which Sum of Square (SOS) check method have been proposed [2]. This is an error detection technique. It cannot detect all the faults. A few systems have been proposed to protect filters from errors. Digital filters utilized as a part of the circuit are finite-impulse response (FIR) filters. For instance, reduced precision replicas were utilized in FIR filters to reduce the expense of implementing modular redundancy [3]. In [4], to recognize errors, relation between the inputs provided to a FIR filter and the memory elements of a FIR filter are utilized. To protect filters the utilization of residue number [5] and arithmetic codes [6] has likewise been proposed. Different implementation structure of FIR filter is used to correct the errors by using one redundant module has been proposed [7]. So far mentioned techniques are used in the protection of single filters.

It is quite common in complex systems to find many filters work in parallel. Several modern communication systems [8] and filter banks [9] has this situation. For these systems, protection of filters will be at higher level by treating parallel filters as a block that needs to be protected. This thought was explored in [10], where two parallel filters with same impulse response fed with different input sequences were considered. It was shown that with only one redundant copy, single error could be corrected. In [11], detection and correction of single fault in multiple modules of parallel FIR filter is done. So far considered schemes are used in detecting and correcting single errors.

In this brief, a simple technique to protect parallel FIR filters from multiple errors is exhibited. As mentioned in [11], same response given to parallel filters with distinctive input sequence is considered. The new scheme depends on the utilization of error correction codes (ECCs) such that each of the output from filter is similar to a bit in ECC codeword. This is a generalization of the plan exhibited in [11] and empowers more effective executions when the number of parallel filters is large. This scheme can likewise be utilized to provide multiple error detection and correction in parallel filter output.

II. Parallel Filters Having Same Impulse Response

FIR filter response is given by following equation

$$Y[n] = \sum_{l=0}^{\infty} X[n-l] \cdot h[l] \qquad (1)$$

Where X[n] is the input, Y[n] is the output, and h[l] is the impulse response of the filter. At the point when the impulse response h[l] is nonzero, just for a limited number of samples, then the filter is called as FIR filter.

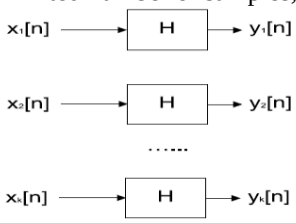

Fig.1. same impulse response given to the parallel filters

Here, a set of '*u'* parallel filters with different input signals and same response are taken. These parallel FIR filters are shown in fig 1. Communication systems which use several channels have these kinds of filters in it.
An intriguing property for these parallel filters is that the addition of any combination of the output $Y_i[n]$ can likewise be obtained by sum of the relating inputs $X_i[n]$ and by multiplying the subsequent signal with the same response h[l]. For instance,

$$Y_1[n] + Y_2[n] = \sum_{l=0}^{\infty} X_1[n-l] + X_2[n-l] \cdot h[l] \qquad (2)$$

This observation will be utilized in order to develop the proposed error correction scheme.

III. Proposed System

The new procedure depends on the utilization of the ECCs. A basic ECC considers a block of 'u' information bits and produces a block of 'n' bits by including n−u parity check bits. The parity check bits are XOR combination of the 'u' information bits. By correctly designing those combinations it is possible to identify and correct the errors. As a case, let us consider u = 4 and n = 7

For this case, three parity check bits are P_1, P_2, and P_3 that is obtained by the XOR combination of information bits d_1, d_2, d_3, d_4 as follows:

$$P_1[n] = d_1[n] \oplus d_2[n] \oplus d_3[n] \qquad (3)$$

$$P_2[n] = d_1[n] \oplus d_2[n] \oplus d_4[n] \qquad (4)$$

$$P_3[n] = d_1[n] \oplus d_3[n] \oplus d_4[n] \qquad (5)$$

The information and parity check bits are accumulated and can be recouped later regardless of the possibility that there are multiple errors. This is done by calculating the parity check filters and comparing the outcomes with the information stored. For example, errors in d_1 will cause errors on the three parity checks; errors on d_2 will cause errors in P_1 and P_2; errors on d_3 will cause errors in P_1 and P_3; and finally errors on d_4 will cause errors in P_2 and P_3. Therefore, errors in the information bits can be identified and the errors can be corrected.

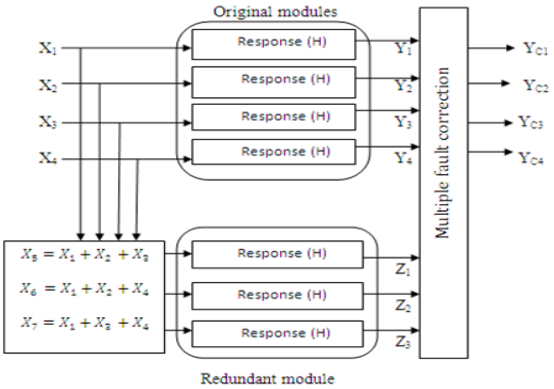

Fig.2. Proposed structure for four parallel FIR filters with redundant modules used for multiple error correction

The inputs given to four filters are X_1, X_2, X_3, and X_4 in which each input has four coefficients which is of 16 bits each. For example, X_1 has four coefficients i.e. $X_1[0]$, $X_1[1]$, $X_1[2]$, $X_1[3]$ in which each coefficient is of 16 bits. It is similar for the rest of the inputs.

The check filters Zj is obtained for the parallel filters by applying ECC scheme. Check filters is obtained by XOR combination of the inputs. Because of the combination of inputs, each of the check filter having four coefficients will be of 18 bits each. Thus each filter output will be having seven coefficients which are of 36 bits each.

In case of four filter outputs Y_1, Y_2, Y_3, and Y_4, the parity check filters are given below

$$Z_1[n] = \sum_{l=0}^{\infty} (X_1[n-l] + X_2[n-l] + X_3[n-l] \cdot h[l] \qquad (6)$$

$$Z_2[n] = \sum_{l=0}^{\infty} (X_1[n-l] + X_2[n-l] + X_4[n-l] \cdot h[l] \qquad (7)$$

$$Z_3[n] = \sum_{l=0}^{\infty} (X_1[n-l] + X_3[n-l] + X_4[n-l] \cdot h[l] \qquad (8)$$

Error detection method is performed to distinguish if faults has occured during the transmission of data starting with one point to other. In this system it can distinguish any number of faults in a module.

error detection is done by checking if

$$Z_1[n] = Y_1[n] + Y_2[n] + Y_3[n] \qquad (9)$$

$$Z_2[n] = Y_1[n] + Y_2[n] + Y_4[n] \qquad (10)$$

$$Z_3[n] = Y_1[n] + Y_3[n] + Y_4[n] \qquad (11)$$

For instance, faults in the filter output Y_1 will create errors on the check filter output Z_1, Z_2, and Z_3. In the similar manner errors in Y_2 will create errors in the output of check filter Z_1 and Z_2; errors in Y_3 will cause errors in Z_1 and Z_3; errors in Y_4 will cause errors in Z_2 and Z_3.

Overall system is shown in fig 2. It is observed that correction of multiple errors can be done by only three redundant filters. In filters, error correction is done by recreating the outputs which has multiple errors using rest of the information and parity check outputs.

For example, when errors on Y_1 is identified, it can be corrected by

$$Y_{C1}[n] = Z_1[n] - Y_2[n] - Y_3[n] \qquad (12)$$

When errors on Y_2 is identified, it can be corrected by

$$Y_{C2}[n] = Z_2[n] - Y_1[n] - Y_4[n] \qquad (13)$$

When errors on Y_3 is identified, it can be corrected by

$$Y_{C3}[n] = Z_3[n] - Y_1[n] - Y_4[n] \qquad (14)$$

When errors on Y_4 is identified, it can be corrected by

$$Y_{C4}[n] = Z_2[n] - Y_1[n] - Y_2[n] \qquad (15)$$

This technique can be used for any number of parallel filters. The methodology is more appealing when the filters 'u' is large in number. Because of this it slightly affects the complexity of the circuit.

The advantages are bigger for this second configuration. In that case, the relative number of included check filters (n −u)/n are smaller.

For example, when u = 11, only four redundant filters are required to correct multiple errors in a module. When u =11, n= 15, the inputs given to eleven filters are X_1, X_2, X_3, X_4, X_5, X_6, X_7, X_8, X_9, X_{10}, X_{11} in which each input has four coefficients which is of 16 bits each. For example, X_1 has four coefficients i.e. $X_1[0]$, $X_1[1]$, $X_1[2]$, $X_1[3]$ in which each coefficient is of 16 bits. It is similar for the rest of the inputs.

The check filters Zj is obtained for the parallel filters by applying ECC scheme. Check filters is obtained by XOR combination of the inputs. Because of the combination of inputs, each of the check filter having four coefficients will be of 19 bits each. Thus each filter output will be having seven coefficients which are of 38 bits each.

IV. Simulation Results

4.1 Results of four parallel filters

The proposed structure output of the four parallel FIR filter is analyzed. It is coded using Verilog and synthesized on Xilinx ISE 10.1 and cadence encounter is obtained. The proposed structure synthesized on Xilinx ISE 10.1 is shown in the fig. 3 and fig 4. Active low indicates there is error in the corresponding signal. Error detection is shown in fig. 5, error correction is shown in fig. 6, and physical design of the proposed scheme is shown in fig 7 respectively.

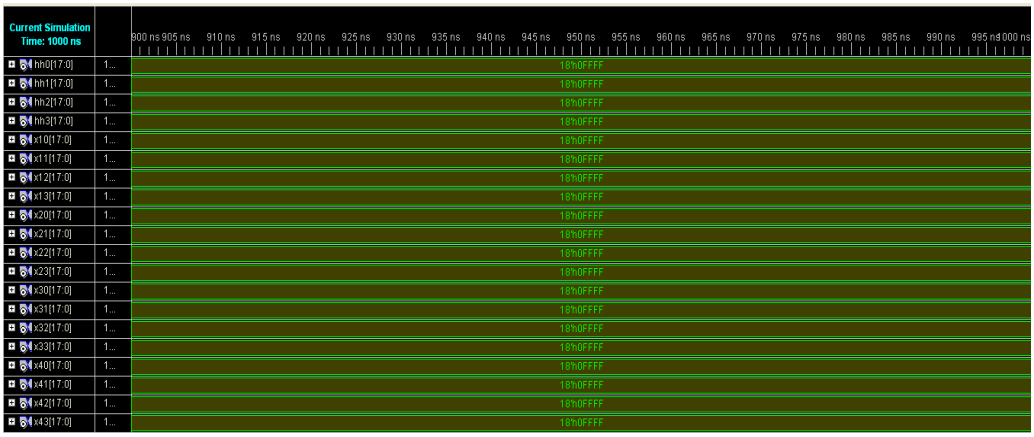

Fig 3. Inputs given to the filters having same impulse response

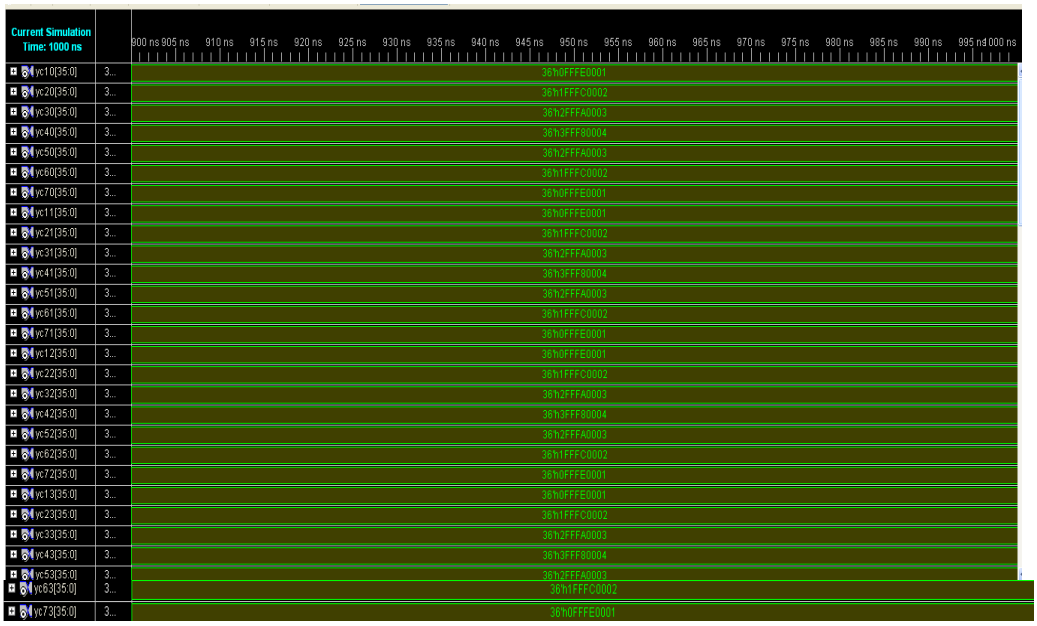

Fig 4. Error free output obtained for the input provided

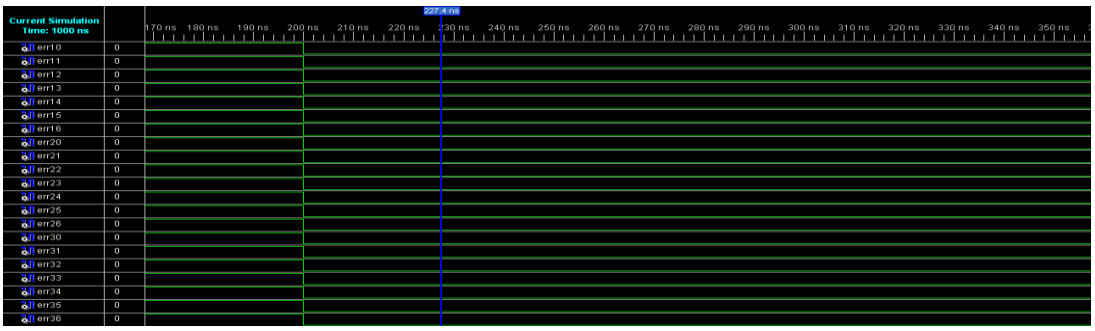

Fig 5. Errors in Y_1 causing errors in Z_1, Z_2, Z_3

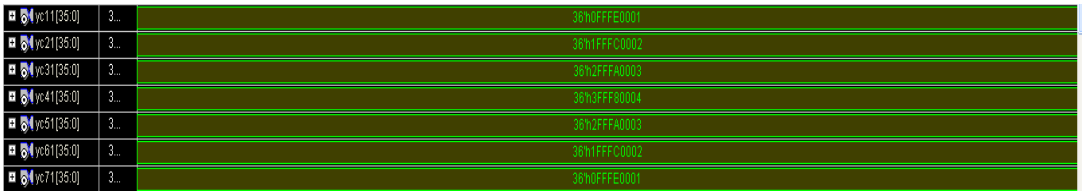

Fig 6. Errors on Y_1 is corrected and output is shown above

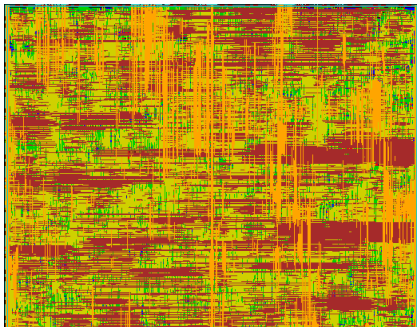

Fig 7. Physical design of four parallel FIR filters

4.2 Results of 11 parallel filters

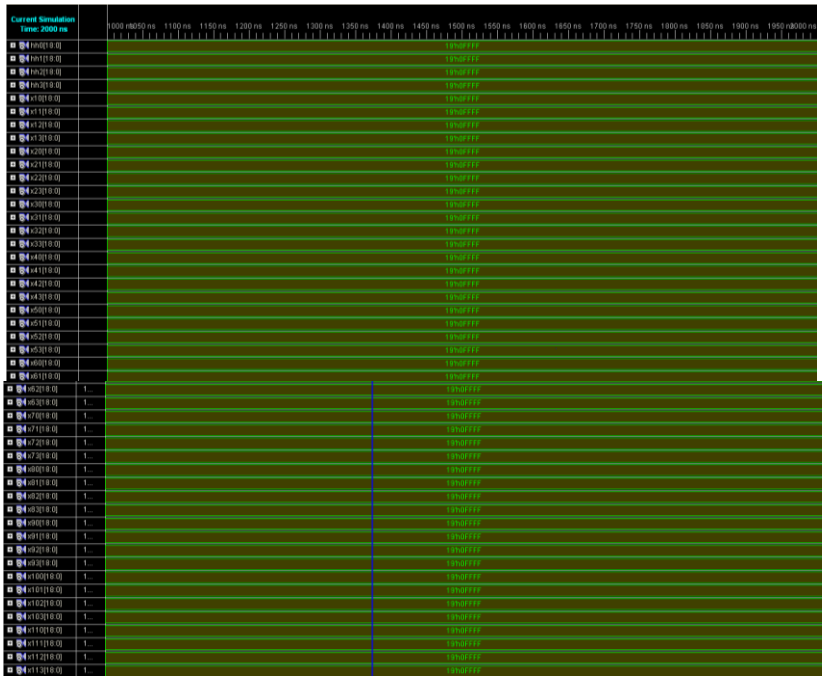

Fig 8. Inputs given to the filters having same impulse response

Fig 9. Error free output obtained for the input provided

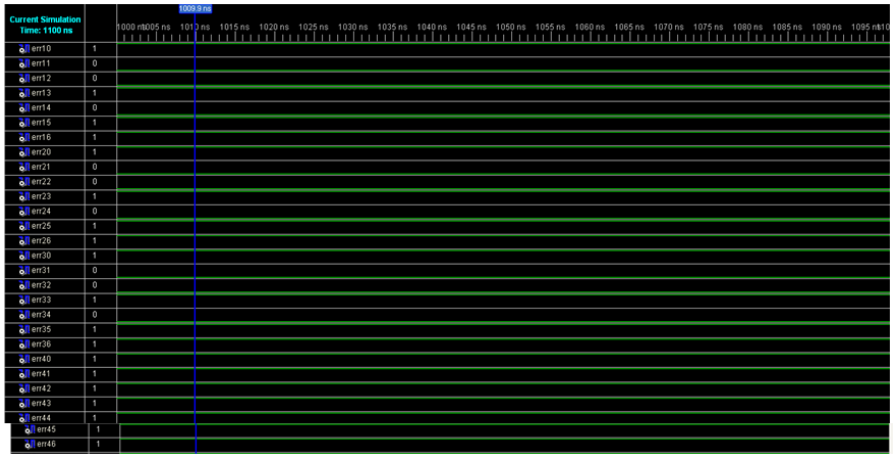

Fig 10. Error in $Y_2[1]$, $Y_2[2]$ and $Y_2[4]$ causing corresponding errors in Z_1, Z_2, Z_3

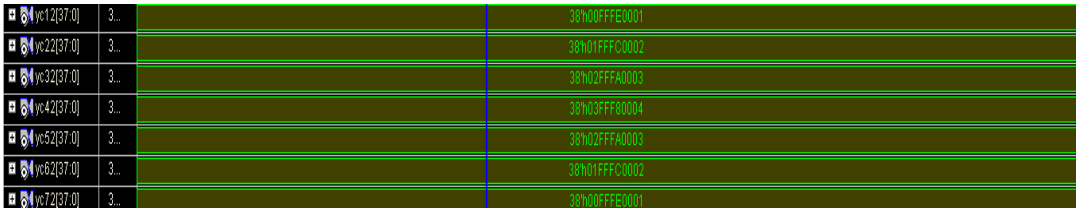

Fig 11. Errors on filter output Y_2 is corrected and corrected output is shown above

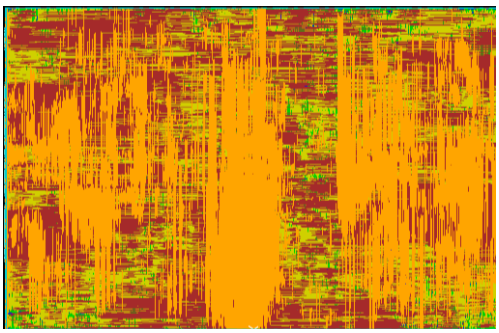

Fig 12. Physical design of eleven parallel FIR filters

V. Conclusion

A new technique has been introduced to protect parallel FIR filters from multiple errors that are generally found in modern signal processing circuits. The methodology depends on applying ECCs to the parallel filters outcomes to recognize and correct errors. This scheme can be utilized for any number of parallel filters with the same impulse response that process different input signals. This technique provides huge advantage if the number of parallel filters are more.

References

[1] M. Nicholaidis, "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] 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.
[6] 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.
[7] P. P. Vaidyanathan. Multirate Systems and Filter Banks. Upper Saddle River, NJ, USA: Prentice-Hall, 1993.
[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] A. Sibille, C.Oestges, and A. Zanella, MIMO: From Theory to implementation. San Francisco, CA, USA: Academic Press, 2010.
[10] P. Reviriego, S. Pontarelli, C. J. 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.
[11] Zhen Gao, Pedro Reviriego, et.al.,"Fault tolerant parallel filters based on error correction codes," IEEE Trans. Very Large Scale Integr. (VLSI) Syst,. vol. 23, no. 2, Feb. 2015