ISSN 2349-6193
Impact Factor: 2.805

IJESMR

International Journal OF Engineering Sciences & Management Research

AN EFFICIENT FAULT-TOLERANT FIR FILTERS USING CODING MATRIX

Malavika M S[*1], Rohith S[2] and Sudarshan T A[3]

[*1]M.tech, Department of E&C, VLSI design and embedded system Nagarjuna college of engineering,
[2]Assistant professor, Department of Electronics and Communication Nagarjuna college of engineering, Bangalore-562164
[3]Assistant Professor, Department of Mechanical Engineering, New Horizon College of Engineering, Bangalore-560103.

Keywords: ABFT techniques, parallel filters, 4x4 Matrix Keypad

ABSTRACT

A scheme based on error correction coding has been recently proposed to protect parallel FIR filters. In this scheme, each filter is treated as a bit, and redundant filters that act as parity check bits are introduced to detect and correct errors. This reduces the protection overhead and makes the Number of redundant filters independent of the number of parallel filters. The proposed scheme is first described and then illustrated with two case studies. Finally, both the effectiveness in protecting against errors and the cost are evaluated for a field-programmable gate array implementation. This Proposed System Implemented using Verilog HDL and Simulated by Modelsim 6.4 c and Synthesized by Xilinx tool. The proposed system implemented in FPGA Spartan 3 XC3S 200 TQ-144.

INTRODUCTION

The complexity of communications and signal processing circuits increases every year. This is made possible by the Digital technology scaling that enables the integration of more and more gates on a single device. This increased complexity makes the circuits more vulnerable to errors. To ensure that soft errors do not affect the operation of a given circuit, a wide variety

Of techniques can be used 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, 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). This strategy can reduce the overhead required to protect a circuit. Rest of the parts are organised as follows in section II objective of the work is given, existing system is discussed in section III. Design of the proposed system is discussed in section IV

Objective

In this brief, the idea of applying coding techniques to protect parallel filters is addressed in a more general way. In particular, it is shown that the fact that filter inputs and outputs are not bits but numbers enables a more efficient protection. This reduces the protection overhead and makes the number of redundant filters independent of the number of parallel filters. The proposed scheme is first described and then illustrated with two case studies. Finally, both the effectiveness in protecting against errors and the cost are evaluated for a field-programmable gate array implementation.

Existing System

Efficient coding schemes for fault-tolerant parallel filters
General: Elliptic curve cryptography (ECC) is an approach to public-key cryptography based on the algebraic structure of elliptic curves over finite fields. ECC requires smaller keys compared to non ECC cryptography (based on plain Galois fields) to provide equivalent security. Elliptic curves are applicable for encryption, digital signatures, pseudo-random generators and other tasks.

ISSN 2349-6193
Impact Factor: 2.805

IJESMR

International Journal OF Engineering Sciences & Management Research

Design Overview And Description

- FIR Filter
- Single Fault Correction Unit
- Original Module
- Redundant Module
- 4X6 Coding Matrix Block

Fir Filter: In signal processing, a finite impulse response (FIR) filter is a filter whose impulse response (or response to any finite length input) is of finite duration, because it settles to zero in finite time. This is in contrast to infinite impulse response (IIR) filters, which may have internal feedback and may continue to respond indefinitely (usually decaying). FIR filter performs a linear convolution on a window of N data samples which can be mathematically expressed as follows with input x(n) and output y(n)

…..(1)

H(n) is the set of filter coefficients ,also known as tap weights, that make up the impulse response. A direct form implementation of an FIR filter can be readily developed from the convolution sum as shown in fig.1.

Direct form FIR filters are also known as tapped delay line or transversal filters. N-tap FIR filter consists of N delay elements, N multipliers and N-1 adders or accumulator.

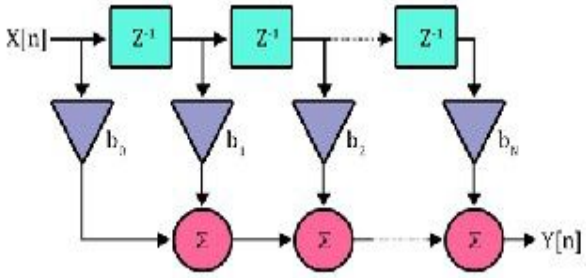

Fig:1 Direct form implementation FIR filter

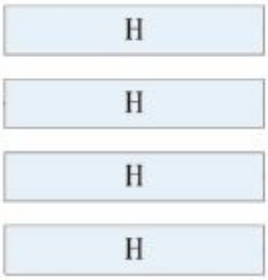

Fig:2 1 FIR Filter Block In Our Base Papers

Single Fault Correction Unit

Error detection vector by defining

$$\begin{cases} p_1 = z_5 - (z_1 + z_2 + z_3 + z_4) \\ p_2 = z_6 - (z_1 + 2z_2 + 3z_3 + 4z_4) \end{cases}$$

…(2)

the check vector can be expressed as This simplified checking can be also derived by noting that p1 and p2 simply check if the output values of the redundant filters (z5 and z6) match the value reconstructed using the outputs of the original filters (z1, z2, z3, and z4). Therefore, in the absence of errors, both p1 and p2 will be zero. From the coding matrix, for nonzero p1 and p2, it becomes clear that an error on the first filter will make p1 = p2 as both a51 and a61 are one. An error on the second filter will make 2*p1 = p2 as a52 = 1 and a62 = 2 and so on. Therefore, the vector can be used to identify the filter

ISSN 2349-6193
Impact Factor: 2.805

IJESMR

International Journal OF Engineering Sciences & Management Research

in error using the mapping shown in Table. For four nonzero values, the faulty filter between the fifth and sixth filters can be identified by checking whether p1 or p2 is nonzero. In fact, to check against Table, the following simplified vector with lower complexity can be used:

$$\bar{e}_{\text{sim}} = \begin{bmatrix} p_1 - p_2 \\ 2p_1 - p_2 \\ 3p_1 - p_2 \\ 4p_1 - p_2 \end{bmatrix}.$$

…..(3)

This provides a simple implementation as only three multiplications are needed and two of them are by powers of two and only require a shift. Another three multiplications are needed to compute p2. Thus, in total, the scheme requires only six multiplications. This shows that the error location logic can be efficiently implemented. Finally, when an error is detected, it can be corrected by recomputing the affected filter output using z5 and the remaining original filter outputs. For example, for an error in filter 1, correction is implemented as

$$z_{\text{corrected1}} = z5 - (z2 + z3 + z4) \text{…..(4)}$$

Original module

The Original Module is designed by Four FIR Filters and the Input is given by the 4 X 6 Code matrixes Block. And its produce the Filter output are Z1, Z2, Z3, and Z4.

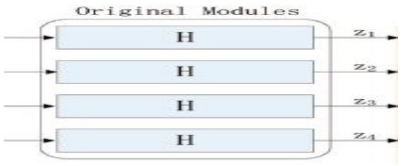

Fig:3 Original module

Redundant module

The Redundant Module is designed by Two FIR Filters and the Input is given by the 4 X 6 Code matrixes Block. And its produce the Filter output are Z5, Z6.

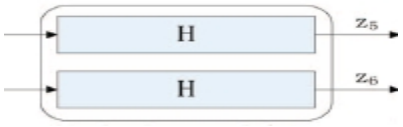

Fig:4 Redundant module

4x6 Coding Matrix Block

The proposed scheme is illustrated in Fig. 2 for the case of four parallel filters. The input signals are encoded using a matrix with arbitrary coefficients to produce the signals that enter the four original and two redundant filters. In its more general form, this coding matrix *A* can be formulated as

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \\ a_{51} & a_{52} & a_{53} & a_{54} \\ a_{61} & a_{62} & a_{63} & a_{64} \end{pmatrix}.$$

..(5)

Therefore, the input signal to the i^{th} filter is of the form (i = 1, 2, . . . , or 6), i.e. The corresponding *A* matrix is the identity matrix on the first four rows, and only the last two rows have generic coefficients. The matrix is

ISSN 2349-6193
Impact Factor: 2.805

IJESMR

International Journal OF Engineering Sciences & Management Research

$$A = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ a_{51} & a_{52} & a_{53} & a_{54} \\ a_{61} & a_{62} & a_{63} & a_{64} \end{pmatrix}. \quad \text{....(6)}$$

To simplify the implementation, those rows should have values that minimize the complexity of multiplications and the increase in the dynamic range in the redundant filters. To that end, the following values can be selected for the last two rows: [a51 a52 a53 a54] = [1111] and [a61 a62 a63 a64] = [1234].

PROPOSED SYSTEM

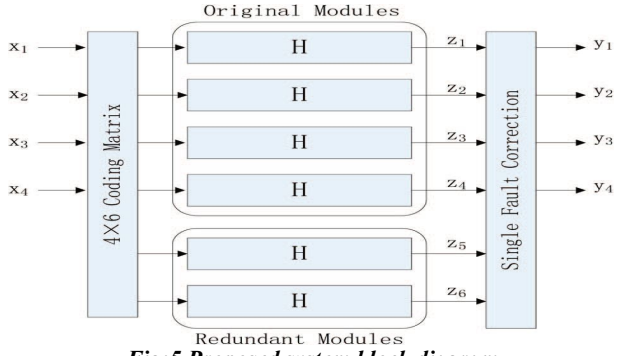

Fig:5 Proposed system block diagram

The proposed scheme is illustrated in above Fig5. for the case of four parallel filters. The input signals are encoded using a matrix with arbitrary coefficients to produce the signals that enter the four original and two redundant filters. The Filter Have A Multiplication Co-efficient Value For Fir Filter inner Operation. The Coding Block Encoded the$ Data into 6 Data then the Value is Passes through the Original Modules and Redundant Modules. Then the Outputs are Passes through the Fault Correction Unit. Then the Fault Values will be Find and Corrected with the help of matrix Values.

PROPOSED SYSTEM ALGORITHM

The SOS check can be combined with the ECC approach to reduce the protection overhead. Since the SOS check can only detect errors, the ECC part should be able to implement the correction. This can be done using the equivalent of a simple parity bit for all the FFTs. In addition, the SOS check is used on each FFT to detect errors.

Table:1 error detection

$c_1 c_2 c_3$	Error Bit Position
0 0 0	No error
1 1 1	z_1
1 1 0	z_2
1 0 1	z_3
0 1 1	z_4
1 0 0	z_5
0 1 0	z_6
0 0 1	z_7

ERROR CORRECTION CODES

GENERAL: In1948 Claude Shannon published al and mark paper “A mathematical theory of communication” that signified the beginning of both information theory and coding theory. Given a communication channel which may corrupt information sent over it, Shannon identified an number called the capacity of the channel and proved that arbitrarily reliable communication is possible at any rate below the channel capacity. For example, when transmitting image so f

ISSN 2349-6193
Impact Factor: 2.805

IJESMR

International Journal OF Engineering Sciences & Management Research

planets from deep space, it is impractical to retransmit the images. Hence if portions of the data giving the images are altered, due to noise arising in the transmission, the data may prove use less. A communication channel is illustrated in Figure6. At the source, a message, denoted x in the figure, is to be sent. If no modification is made to the message and it is transmitted directly over the channel, any noise would distort the messages o that it is not recoverable.

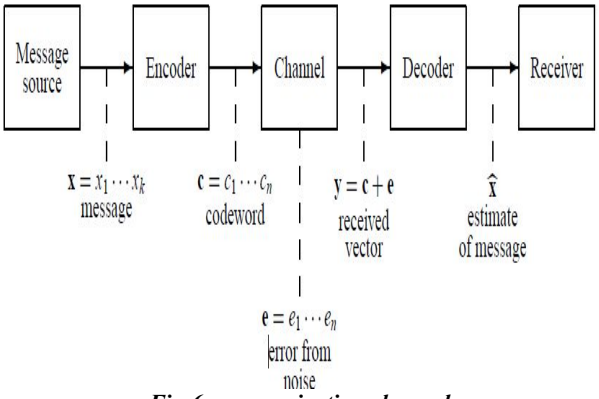

Fig:6 communication channel

ERROR IMPLEMENTATION

Error correction may generally be realized in two different ways

A. Automatic repeat request (ARQ): This is an error control technique whereby an error detection scheme is combined with requests for retransmission of erroneous data. Every block of data received is checked using the error detection code used, and if the check fails, retransmission of the data is requested – this may be done repeatedly, until the data can be verified.

B. Forward error correction (FEC): The sender encodes the data using an error-correcting code (ECC) prior to transmission. The additional information (redundancy) added by the code is used by the receiver to recover the original data. In general, the reconstructed data is what is deemed the "most likely" original data. ARQ and FEC may be combined, such that minor errors are corrected without retransmission, and major errors are corrected via a request for retransmission: this is called hybrid automatic repeat-request.

ERROR DETECTION SCHEMES

Error detection is most commonly realized using a suitable hash function (or checksum algorithm). A hash function adds a fixed-length tag to a message, which enables receivers to verify the delivered message by recomposing the tag and comparing it with the one provided. There exists a vast variety of different hash function designs. However, some are of particularly widespread use because of either their simplicity or their suitability for detecting certain kinds of errors.

FPGA Module

Device used SPARTAN3 (XC3S200-TQ144) 200,000-gate Up to 97 user defined I/O signals

Memory : 4MB - PROM
Clock : 50MHz crystal

SIMULATION SNAPSHOTS, DESIGN AND RESULTS

In this section simulation results of proposed system is given .The below figures shows the simulation results

A. Simulation results

ISSN 2349-6193
Impact Factor: 2.805

IJESMR

International Journal OF Engineering Sciences & Management Research

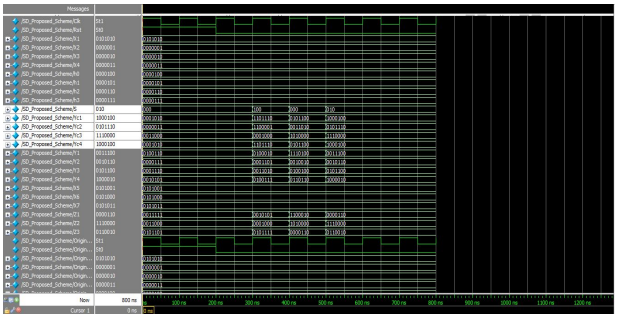

Fig 7 Existing Scheme 7 bit:

Minimum period: is 4.666ns (Maximum Frequency: 214.316MHz), Minimum input arrival time before clock: 20.532ns, Maximum output required time after clock: 31.462ns, Maximum combinational path delay: 36.117ns

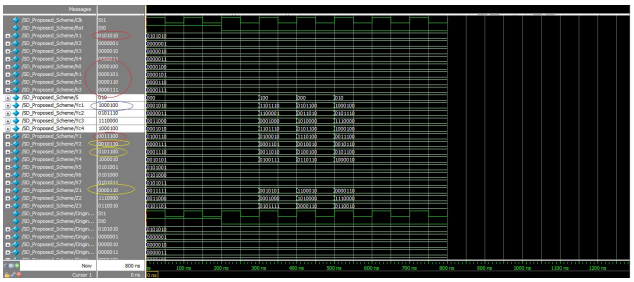

Fig: 8 E Values:

The fig 8 above shows the error value and it is corrected by the below proposed scheme

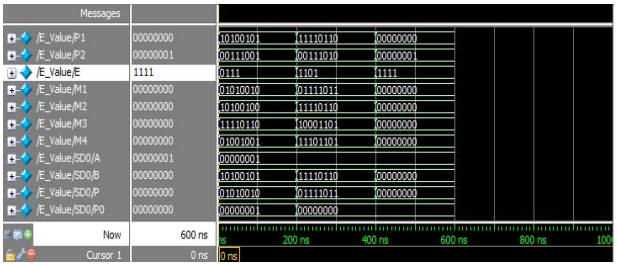

Fig: 9 Proposed Scheme Simulations:

The above fig shows the Timing constraint and Default period analysis for Clock 'Clk' Clock period is 4.666ns (frequency: 214.316MHz), Total number of paths / destination ports: 636 / 96, Source Clock is Clk rising Delay obtained is 4.666ns (Levels of Logic = 3)

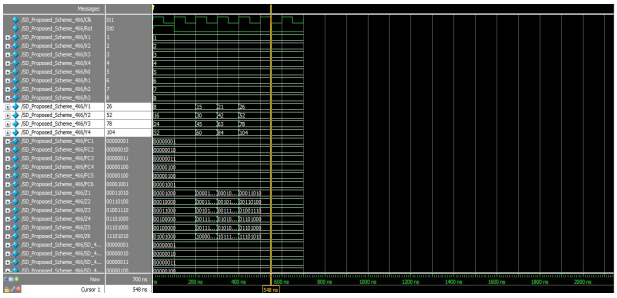

Fig: 10 Fault Selection:

ISSN 2349-6193
Impact Factor: 2.805

IJESMR

International Journal OF Engineering Sciences & Management Research

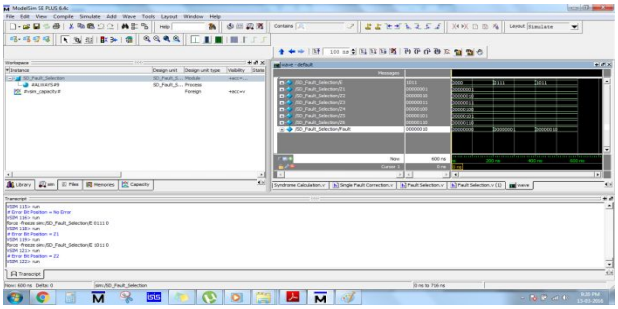

Fig 11Main Single Fault Correction

Single_Fault_Correct Corrected output obtained is 1 0.551 0.801, Total delay 40.110ns (20.057ns logic, 20.053ns route)

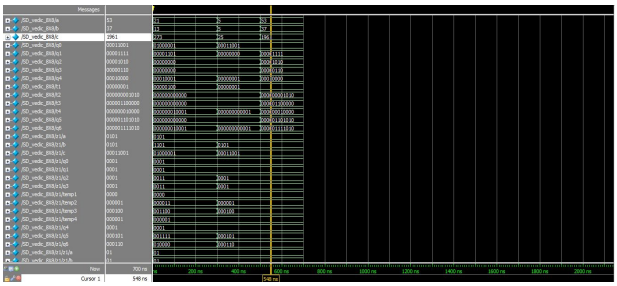

Fig 12 Modified 4X6 Coding Scheme:

RTL Design

The design below is the snapshots of RTL designs

Here encoder is used as one of the basic module in this work. The output waveform of the encoder module which has seven primary inputs with reset enable is shown below.

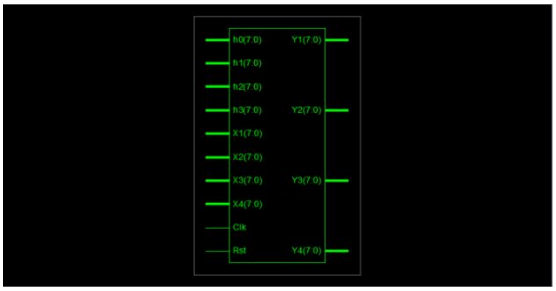

Fig: 13 Main Architecture:

The syndrome is used to detect the error and correct the error bit Device used 3s1500lfg320-4

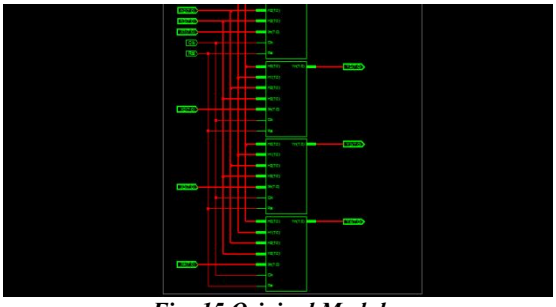

Fig: 15 Original Modules

ISSN 2349-6193
Impact Factor: 2.805

IJESMR

International Journal OF Engineering Sciences & Management Research

The fig above is the original module consisting of 4 encoders

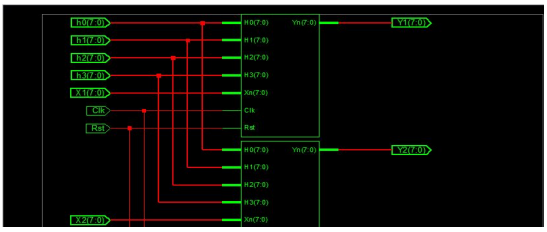

Fig: 16 Redundant Modules

The fig above is the redundant module consisting of 2 encoders and offset is 24.192ns

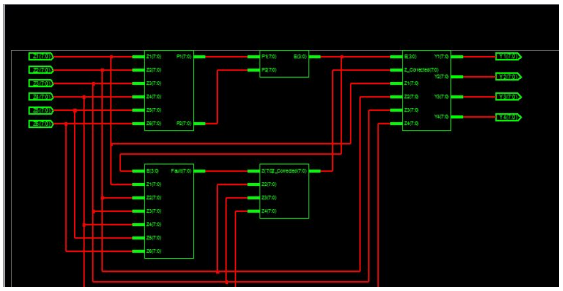

Fig: 17 Single Fault Correction Units:

Selected Device: 3s1500lfg320-4
Number of Slices used 1040 out of 13312
Number of Slice Flip Flops 144 out of 26624
Number of 4 input LUTs 1861 out of 26624
Number of IOs 98
Number of bonded IOBs 98
Number of GCLK 1 out of 8

Comparison Table

The table below provides the result and comparison with Other proposed system

Method Name	**Area in Number of LUT**			**Memory in Kilobytes**	**Delay**		
ECC Type	**LUT**	**Gate Count**	**Slices**	**S**	**Delay**	**Gate or Logic Delay**	**Path or Route Delay**
4X6 Coding (Proposed or Base Paper)	1861	12522	144	238376	36.117ns	17.181ns	18.936ns
Modified (Modified)	1650	13053	144	255080 kilobytes	40.110ns	40.110ns	20.053ns route

ISSN 2349-6193
Impact Factor: 2.805

IJESMR

International Journal OF Engineering Sciences & Management Research

CONCLUSION

A new method to implement fault-tolerant parallel filters has been presented in this brief. The proposed scheme exploits the linearity of filters to implement an error correction mechanism. In particular, two redundant filters whose inputs are linear combinations of the original filter inputs are used to detect and locate the errors. The coding of those linear combinations was formulated as a general problem to then show how it can be implemented. The practical implementation was illustrated with two case studies that were evaluated for an FPGA implementation and compared with a previously proposed technique. That technique relies on the use of ECCs such that each filter is treated as a bit in the ECC. The results show that the proposed scheme outperforms the ECC technique Therefore; the proposed technique can be useful to implement fault tolerant parallel filters. Future work will consider applying the scheme to parallel filters that have the same input signal but different impulse responses. The Proposed system is implemented using Verilog HDL and Simulated and Synthesis by Modelsim and Xilinx. In terms of error protection, fault injection experiments show that the ECC scheme can recover all the errors that are out of the tolerance range.

REFERENCES

1. Kanekawa, E. H. Ibe, T. Suga, and Y.Uematsu, Dependability in Electronic Systems: Mitigation of Hardware Failures, Soft Errors, and Electro-Magnetic Disturbances. New York, NY, USA: Springer-Verlag, 2010.
2. R. Baumann, “Soft errors in advanced computer systems,” IEEE Des. Test Compute, vol. 22, no. 3, pp. 258–266, May/Jun. 2005.
3. M. Nicolaidis, “Design for soft error mitigation,” IEEE Trans. Device Mater. Rel., vol. 5, no. 3, pp. 405–418, Sep. 2005.
4. A. L. N. Reddy and P. Banerjee, “Algorithm-based fault d1etection for signal processing applications,” IEEE.