



COPY RIGHT

2017 IJIEMR. Personal use of this material is permitted. Permission from IJIEMR must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. No Reprint should be done to this paper, all copy right is authenticated to Paper Authors

IJIEMR Transactions, online available on 25th Oct 2017. Link

[:http://www.ijiemr.org/downloads.php?vol=Volume-6&issue=ISSUE-9](http://www.ijiemr.org/downloads.php?vol=Volume-6&issue=ISSUE-9)

Title: **ADAPTIVE MODEL OF LOW COMPLEXITY FAULT TOLERANT PARALLEL FFTS USING PARTIAL SUMMATION**

Volume 06, Issue 09, Pages: 282–291.

Paper Authors

D.DEEPTHI, KONDA KIRAN KUMAR

Nalgonda institute of technology and science, Nalgonda



USE THIS BARCODE TO ACCESS YOUR ONLINE PAPER

To Secure Your Paper As Per **UGC Guidelines** We Are Providing A Electronic Bar Code

ADAPTIVE MODEL OF LOW COMPLEXITY FAULT TOLERANT PARALLEL FFTS USING PARTIAL SUMMATION

¹D.DEEPTHI, ²KONDA KIRAN KUMAR

¹M.Tech Scahloar, Dept of E.C.E(VLSI & ES), Nalgonda institute of technology and science, nalgonda.

²Assistant Professor, Dept of E.C.E, Nalgonda institute of technology and science, nalgonda.

ABSTRACT The increasing demand of low complexity and error tolerant design in signal processing systems is a reliability issue at ground level. Complex circuit is consistently affected by soft errors in modern electronic circuits. Fast Fourier transforms (FFTs) plays a key role in many communication and signal processing systems. Different algorithms have been used in earlier techniques for achieving fault tolerant coverage design. In real time application systems, numbers of blocks operating in parallel are frequently used. The proposed work exploits a technique to implement fault tolerance parallel FFT with reduced low complexity of circuit area and power. In Partial summation along with error detection and correction hamming code is used for designing soft error tolerant parallel FFT shelter. This new method achieves lower complexity proportional to that of FFT design size. Based on these two schemes, two modified preserve techniques that combine the use of error correction codes and Partial summation are proposed and evaluated. First method, the Parity-Partial Summation and ECC uses one FFT with minimum Partial Sum blocks for reducing hardware area. Secondly, Parallel Partial Summation ECC used for correcting errors in multiple FFTs protective methods. The result for 4-parallel and 6- parallel FFTs shows that the proposed technique effectively reduces area and power of fault tolerant design along with improved fault coverage.

KEYWORDS: Fast Fourier transforms (FFT), EIC, Partial Summation, and ABFT.

INTRODUCTION

The CMOS technology scaling has made today's designs more susceptible to radiation induced soft errors. Soft errors can alter the logical output of a circuit node creating an error that affects the system functionality [2]. The problem becomes more complexity of the soft error rate exponentially increases with that of circuits scaling. Single Event Upsets (SEU) also

affects the reliability of the circuit due to variation in set-up and hold time. Various methods have been adopted earlier to mitigate soft errors. Specifically designing libraries used for complex circuit and modified manufacturing process such as the silicon on insulator design also used for minimizing the error probability. Adding redundancy to keep the design free from

temporary errors is also adopted in existing designs. Five modular redundancy mitigation techniques recovers the Error Module used to overcome the unintended behavior of the system by adding redundancy [7], [8], [9]. It adds two identical designs and a voter along with the original design to produce correct results. It increased the area overhead which not suitable for complex designs. Some techniques have been introduced to eliminate this problem by making changes in the algorithm (ABFT) [10]. Based on algorithms, the use of the Parseval theorem or sum of squares check is one the frequently used method. The SOS check states that the SOSs of the inputs to the FFT are equal to the SOSs of the outputs of the FFT. This correlation can be used to detect errors while using multiple FFTs. As the growth of SoC and NoC becoming increasingly popular, it is frequent to find FFTs working in parallel. This happens especially for applications in orthogonal frequency demodulation multiplexing (OFDM) systems, such as IEEE 802.11a/g, Long Term Evolution (LTE), and Digital Video Broadcasting-Terrestrial (DVB-T) [11], [12]. A MIMOOFDM design also sees several FFTs operating parallel for obtaining diversity gain and to combat signal fading. So, keeping the FFTs secure was one of the important tasks in these kinds of applications. The FFTs in parallel increases the scope of applying error correction codes together. Generating parity together for parallel FFTs also helps in minimizing the complexity in some ECC [15]. By assuming that there can only be a single error on the

system in the case of radiation-induced soft errors and may be two in worst case. The proposed new technique is based on the combination of Partial Summation combined with parity FFT for multiple error correction. The proposed schemes have been implemented using Xilinx FPGA to evaluate the protections of soft error for minimizing the power and delay of the output depends upon the area overhead. The rest of the paper is organized as follows. Section 2 contains highly based on Fast Fourier Transformation. In Fault tolerant error coverage techniques for parallel FFTs are shelter in best discussions of Section 3. From the analyzed report implementations are in briefly discussed in the Section 4 and Section 5 concludes the evaluated research results.

Fast Fourier Transform

Fast Fourier Transform (FFT) algorithm converts a signal from time domain into a sequence in the frequency domain [13]. Fast Fourier transforms are widely used for many applications which include engineering, science, and mathematics. It computes transformations through DFT matrix. The FFT operation starts with decomposing N-point time domain signal and calculating N frequency spectra and finally forming a single spectrum.

The Discrete Fourier Transform (DFT)

Discrete Fourier Transform (DFT) is an important unit in many communication applications like OFDM, etc. DFT is also measured as one of the tools to act upon frequency analysis of discrete time signals.

The Discrete Fourier Transform is a continuous Fourier transform for the use of discrete functions. Given a real sequence as the input, the DFT outputs them as a sequence of complex numbers. The mathematical representation of the transform is

$$X(k) = \sum_{n=0}^{N-1} x(n)e^{-\frac{j2\pi kn}{N}}, n = 0, 1, \dots, N-1 \quad (1)$$

If an N – point DFT is implemented directly, the necessity of arithmetic units is of the order of O(N²) that is N² multiplications and N (N-1) additions. Thus FFT is used for designing the DFT. Depending on inputs being real or complex, the design of adders and multipliers are formed.

Divide and Conquer Approach to Computation of the DFT

The reduction of computational complexity algorithm for DFT is made possible by using a divide and conquers approach. This approach decomposes a larger DFT into smaller one forming a collective FFT algorithm. Let us consider N-point DFT. It is one of the well-organized ways to implement Discrete Fourier Transform (DFT) due to its compact use of arithmetic blocks. The FFT and inverse FFT of an N point signals are given below

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{kn} \quad (2)$$

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k)W_N^{-kn} \quad (3)$$

Where

$$W_N^{kn} = e^{-j\frac{2\pi}{N}kn}$$

From the above equation it is evident that both FFT and its inverse are almost same with small variation. Inverse FFT design is formed by using divide by N-point and taking the conjugate of the twiddle factors. Due to the frequent use of FFT in modern wireless designs, higher radix FFTs such as radix-4, radix-8, radix-2^k, split radix, etc. are designed for improving the performance timing and reducing the complexity. The difference between these designs is based on their butterfly units. Basic concept of 4-points DIF FFT circuit which applies Radix-2 architecture is in Figure 1 shows calculation signal flow graph about discrete Fourier coefficient of N=4. Here W₄⁰, W₄¹ are the twiddle factors of the four point Fast Fourier Transform. Note, under the arrow of is subtraction; twiddle factor at the top of line is multiplication. In butterfly processing element that is shown red line make be corresponding to next block in the slide diagram. Another butterfly processing element colors are same. This is the 4 points FFT circuit for parallel input. Generally, FFT analyzes an input signal sequence by using decimation-in-frequency (DIF) or decimation-in-time (DIT) decomposition to design an efficient signal-flow graph (SFG). Here, the paper work focuses DIF decomposition because it matches with various pipelined designs. x(0), x(1), x(2) and x(3) are the input time domain signals with 1, -1 and -j as the twiddle factors producing X(0), X(1), X(2) and X(3) as the frequency domain outputs.

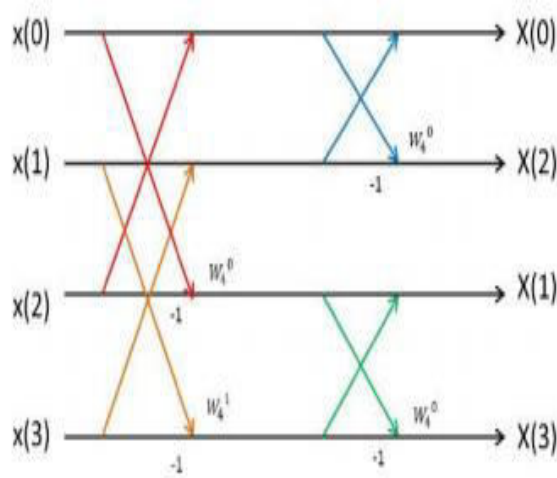


Figure 1. Signal Flow Graph of 4-point FFT

Error Tolerant Techniques for Parallel FFTs

Error Correction based on Hamming Codes
 The aim of error tolerant design is to protect parallel FFTs from errors. Various schemes have been proposed for error detection and correction in FFTs. One of the basic and simple methods is error correction using hamming codes. Unlike parity code which can detect only odd bit error, the hamming code can detect two bit errors and correct one error. Similar to other error correction codes, hamming codes also utilizes the parity bit which is generated for the corresponding input sequence for detecting errors [14]. It achieves higher code rate with minimum distance of three. The number of parity bits depends on the total number of data bits. For example, hamming code with 4 information bits produces 7 encoded data bits with its difference being the parity. In this case, the three parity bits h_1, h_2, h_3 are computed as a data bits c_1, c_2, c_3, c_4 as described below:

$$h_1 = c_1 \oplus c_2 \oplus c_3 \quad (4)$$

$$h_2 = c_1 \oplus c_2 \oplus c_4 \quad (5)$$

$$h_3 = c_1 \oplus c_3 \oplus c_4 \quad (6)$$

The limitations of is that, during the multiple error scenario hamming code will not be able to exactly identify the individual FFTs with error. Fault tolerant FFT based on Parseval's check 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.

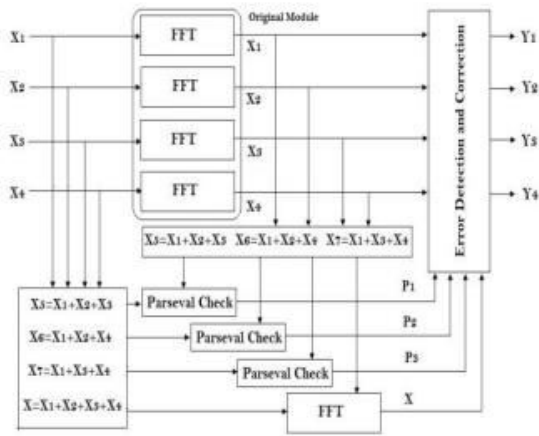


Figure 2. Parity-SOS-ECC fault-tolerant parallel FFTs

This method combines the feature of parity calculation of hamming codes and error detection process of Sum of Squares. Concurrent Error Detection (CED) schemes for the FFT are the Sum of Squares (SOS) check based on Pa theorem. The use of parseval check is exponentially reduced to the direct comparisons of FFTs inputs and outputs used to protect parallel FFTs

Proposed Error Indicator and Corrector Techniques

The proposed work focuses on two new techniques for reducing the hardware overhead and increasing the error correction capability. The Figure 3 describes the new technique which focuses on the existing systems limitations. The technique analyzed in the previous work has certain limitation due to the complexity of handling larger number of FFTs and Sum of Squares block. Instead of using Sum of Squares, Partial summation is used for calculating its parity at the input and the output side of the FFT. It sums all possible node values of 4-point FFT along with the twiddle factors.

Multiplication operation which leads to the complexity in Sum of Squares is thus eliminated using only the adder blocks. Technique 1 also uses 3 parallel redundant FFT in the case of 4-parallel FFT design for correcting multiple errors in fault tolerant for soft error.

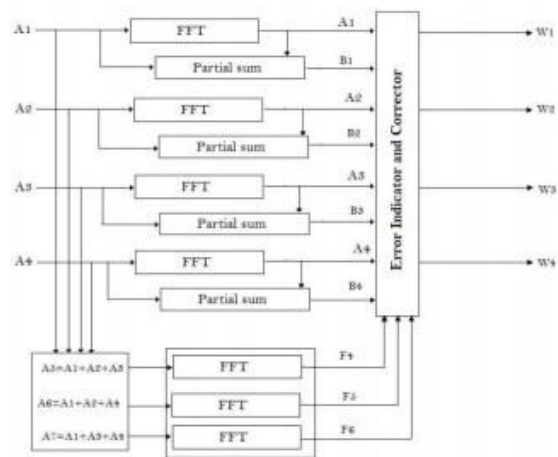


Figure 3. Parallel-PS-ECC fault-tolerant parallel FFTs

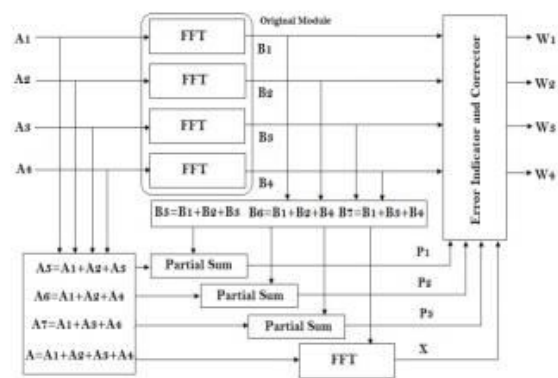


Figure 4. Parity-PS-ECC fault-tolerant parallel FFTs

Figure 4 illustrates the Parity Partial Summation block for less error prone applications. For example when the error occurs in A1 and A2 then it can be detected by the partial summation used individually for FFT blocks. The first check equation is

selected in such a way that the both error block signals are not present.

$$A5=A1+A2+A3 \quad (7)$$

$$A6=A1+A2+A4 \quad (8)$$

$$A7=A1+A3+A4 \quad (9)$$

The error in first FFT is corrected by using equation (9) as

$$B1=B7-B3-B4 \quad (10)$$

Once the error is detected using in the first FFT is corrected then equation (7) can be used for correcting error in the second one which is given as,

$$B2=B5-B1-B3 \quad (11)$$

The combination of parity PS-ECC fault tolerant parallel FFT reduces the number of additional FFTs to just one and also reduces the shelter overhead. In our implementation those blocks are protected with adders used to compute the inputs and the outputs to the redundant FFTs in the existing technique with a small impact on circuit complexity as they are much simpler than the FFT computation.

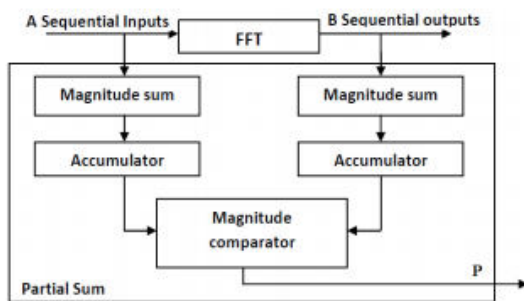


Figure 5. Implementation of Partial Summation

Figure 5 shows implementation architecture of partial summation block. The sequential input and output of the 4- point FFT is fed to the magnitude sum block. The magnitude comparator compares the input and output for verifying the FFT output. If both are equal then the FFT is soft error free.

Table 1. Overhead of Different Techniques

Types	FFTs	Partial-sum
ECC	$1+\log_2(k)$	0
PARALLEL-PS- ECC	$1+\log_2(k)$	K
PARITY-PS-ECC	1	$1+\log_2(k)$

The overhead of the two new proposed techniques along with the general ECC is shown in Table 1. For the less error prone applications technique two can be used with Partial summation block replacing the Parseval check. Both the new techniques proposed uses minimum hardware resources compared to the existing design by the modification of Partial summation block for Sum of Squares.

Evaluation Results

A four-point DIF-FFT FFT is computed for both 4-parallel and 6-parallel designs and results shows that the proposed Parallel Partial Summation ECC technique results reduced hardware resources such as gate count, slice and LUTs.

Table 2. Resource Utilization of FFT

Parameters	FFT
Gate count	1120
FlipFlops	10
LUT	15
Slices	16
Power(mW)	16

The resource utilized for single FFT is shown in Table 2. The two proposed techniques have been implemented on an FPGA and evaluated both in terms of overhead and power. The analysis of two new techniques with its area, power and

delay parameters is implemented using Xilinx ISE environment. With this alternative new scheme formulation, it is clear that the scheme can be used for any number of parallel FFTs for soft error detection and correction. Various results show the effectiveness of the proposed scheme in terms of various parameters.

Table 3. Resource Utilization for 4-Parallel FFT

Design Summary	Parity-SOS-ECC	Parallel-PS-ECC	Parity-PS-ECC
Gate count	9584	5330	3930
Flip Flops	225(1.046%)	175(0.813%)	159(0.739%)
LUT	388(1.804%)	161(0.7486%)	147(0.683%)
Slices	284(2.641%)	130(1.209%)	101(0.939%)
Power(mW)	252	246	243
Delay(ns)	7.049	2.682	2.729

It is inferred that the Table 3 shows that various parameters like Gate count, FlipFlops, LUT, Slices, power and delay have been analyzed and the comparative results of 4-Parallel FFT furnish the best performance fault tolerant error free result.

Table 4. Resource Utilization for 6-Parallel FFT

Design Summary	Parity-SOS-ECC	Parallel-PS-ECC	Parity-PS-ECC
Gate count	16349	7454	5658
Flip Flops	425(1.976%)	295(1.371%)	161(0.748%)
LUT	1448(6.733%)	270(1.255%)	199(0.925%)
Slices	886(8.240%)	211(1.962%)	157(1.460%)
Power(mW)	270	258	247
Delay(ns)	10.381	4.987	3.104

Table 4 shows the comparative results of 6-Parallel FFT Parity-SOS-ECC with two new techniques. Considerable improvement also achieved in terms of power and delay. The other proposed technique Parity Partial Summation ECC results again further

improved hardware design utility compared to Parallel Partial Summation ECC. From this analyzed view to reduce power and design area of the new proposed techniques using to the parity Partial Summation ECC than existing partial check ECC technique.

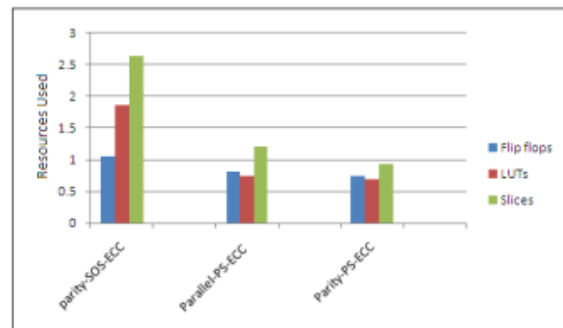


Figure 6. Comparison graph for four parallel FFTs

To further demonstrate this phenomenon, the number of slices required for the different schemes, LUTs and number of FFTs is plotted as shown in Figure 6 and Figure 7 for 4-Parallel and 6-Parallel designs respectively. For larger number of parallel FFTs, both the proposed techniques produce improved result. From the Figure 6 above it is inferred that efficiency of the proposed method results were minimum utilization of hardware resource such as Flip-Flop, LUT and Slices. Parity Partial summation produces the optimized result compared to the other methods.

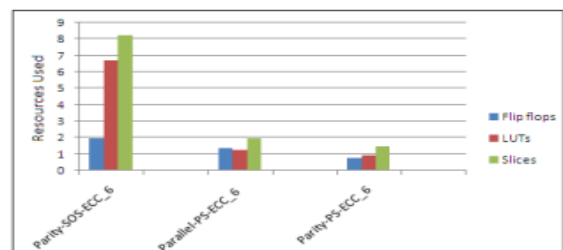


Figure 7. Comparison graph for six parallel FFTs

From the Figure 7 above it is inferred that efficiency of the proposed method results among the minimum utilization of hardware resources. First method, the Parity-PS-ECC uses one FFT with minimum Partial Summation (PS) block for reducing the complexity of the design. Second method of Parallel partial Summation ECC used for correcting errors in multiple FFTs. It can be observed that six parallel FFT is the value for which parity-PS and ECC have almost same resource utilization with better performance of power and delay values. The Figure 8 shows the output result of the design summary window obtained in Xilinx ISE. It contains the total number of various hardware resource utilized for the new proposed technique for four-parallel Parity Partial summation methods as already mentioned in the Table 3. The 4-FFT and the different fault tolerant schemes have been implemented using VHDL. Then, the design is synthesized using Virtex-4 xc4vlx80 Xilinx FPGA.

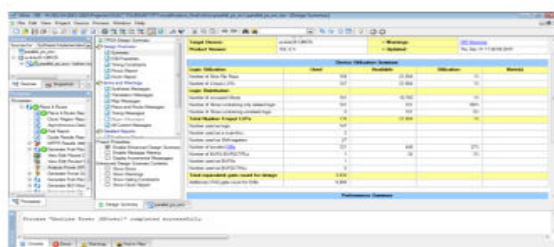


Figure 8. Design summary of 4-parallel FFTs



Figure 9. Power analysis of 4-parallel FFTs

From the Figure 9 shows the power report of the new proposed design obtained in Xilinx ISE. It contains the total power consumed for the proposed 4-parallel Parity Partial summation method as mentioned in the Table 3.

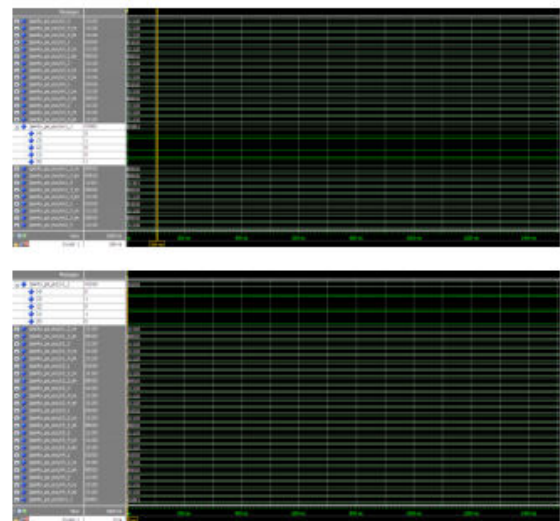


Figure 10. Simulation Results for Parity-PS-ECC Fault-Tolerant Parallel FFTs fault and corrected output

A final observation is that the ECC scheme can detect all errors that exceed a given threshold. On the other hand, the Partial summation technique is used to detect all errors. Therefore, to compare the two techniques for a given implementation, fault injection experiments should be done to determine the percentage of errors that are actually corrected. This means that an evaluation has to be done both in terms of overhead (power and area) and error coverage. The Figure 10 and Figure 11 shows the output result of the simulation and FPGA hardware after implementing of the new proposed design. The first screen shots the output of the first FFT with error and the second one shows its corrected output. The

actual output of the 4-point real input is 1, 2, 3 and 4 should produce 01010 at X (0). But, due to the error it produces the error output 01001. From the second screen shot of Figure 10 is inferred that the error is corrected and the actual output 01010 is displayed. From the second snapshot of Figure 11 is inferred that the error is corrected and the actual output 01010 is displayed through LED.

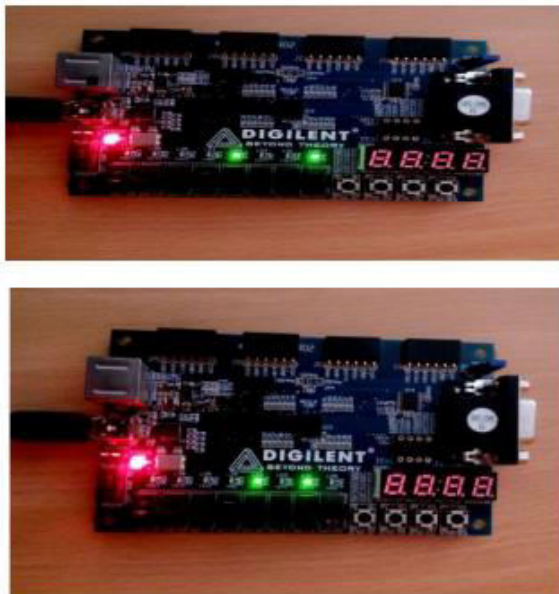


Figure 11. New Proposed schemes for fault and corrected output

Conclusion and Future Work

The fault secure design of parallel FFTs against multiple errors is analyzed and two new techniques based on Partial Summation is implemented. The evaluation result shows that both the approaches results in low complexity and low power consumption. The parallel Partial summation ECC can reduce the hardware resource utilized by the Sum of Squares and also correcting errors in multiple FFTs. The second technique

without hamming codes further reduces the area which is used less error prone applications. Compared to the existing counter methods, the result of proposed four-parallel and sixparallel shows the area optimization by 59% and 65% along with the 99.97 fault coverage criteria. In the future work, a self adaptive system to overcome hard errors can also be taken into consideration.

References

- [1] M. Nicolaidis, —Fault Tolerant Parallel FFTs Using Error Correction Codes and Parseval Checks,| IEEE transactions on very large scale integration (VLSI) systems, February 26, 2015.
- [2] R. Baumann, —Soft errors in advanced computer systems,| IEEE Des.Test Computer., vol. 22, no. 3, pp. 258–266, May/June. 2005.
- [3] M. Nicolaidis, —Design for soft error mitigation,| IEEE Trans. Device Mater. Rel., vol. 5, no. 3, pp. 405–418, Sep. 2005.
- [4] B. Shim and N. R. 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.
- [5] Riaz Naseer, Rashed Zafar Bhatti, Jeff Draper, —Analysis of Soft Error Mitigation Techniques for Register Files in IBM Cu-08 90nm Technology,| Information Sciences Institute University of Southern California Marina Del Rey, CA 90292 USA 2006 IEEE.

[6] Z. Gao et al., —Fault tolerant parallel filters based on error correction codes,|| IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 23, no. 2, pp. 384–387, Feb. 2015.

[7] Haryono, —Five Modular Redundancy with Mitigation Technique to Recover the Error Module,|| International Journal of advanced studies in Computer Science and Engineering IJASCSE, Volume 3, Issue 2, 2014.

[8] Pedro Reviriego, Chris J. Bleakley, and Juan Antonio Maestro., —Structural DMR: A Technique for Implementation of Soft-Error-Tolerant FIR Filters,|| IEEE transactions on circuits and systems—ii: express briefs, vol. 58, no. 8, august 2011.

[9] R. E. Lyons W. Vanderkul k., —The Use of Triple Modular Redundancy to Improve Computer Reliability,|| IBM JOURNAL APRIL 1962.

[10] A. L. N. Reddy and P. Banerjee, —Algorithm-based fault detection for signal processing applications,|| IEEE Trans. Computer., vol. 39, no. 10, pp. 1304– 1308, Oct. 1990.

[11] G. L. Stüber, J. R. Barry, S. W. McLaughlin, Y. Li, M. A. Ingram, and T. G. Pratt, —Broadband MIMO OFDM wireless communications,|| Proc. IEEE, vol. 92, no. 2, pp. 271–294, Feb. 2004.

[12] —Digital Video Broadcasting (DVB); Framing structure, channel coding and modulation for digital terrestrial television,||

Final draft ETSI EN 300 744 V1.5.1 (2004-06) European Standard (Telecommunications series).

[13] Jen-Chih Kuo, —VLSI Design of a Variable-Length FFT/IFFT Processor for OFDM-Based Communication Systems,|| EURASIP Journal on Applied Signal Processing.

[14] —Hamming Codes,|| James Fiedler Fall, 2004. [15] —Error detection and correction,|| Logic and computer design fundamentals.

Guide Details



Konda Kiran Kumar M.Tech (Vlsi & Sd) Kiran3712@Gmail.Com Nalgonda



D. DEEPTHI, I received B.Tech Degree in Electronics & communication Engineering (ECE) from GATE COLLEGE, Kodad, Nalgonda (Dist), and then completed her P.G in VLSI & Embedded systems (VLES) in NITS College, Nalgonda, TS, India.

Mail ID: vanam014@gmail.com



International Journal for Innovative Engineering and Management Research

A Peer Reviewed Open Access International Journal

www.ijemr.org