International Journal of Engineering & Technology, 7 (4) (2018) 2338-2343

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET
doi: 10.14419/ijet.v7i4.14106
Research paper

Design of low- area, power fault tolerant parallel FFTs using trellis codes

B.NagaSaiLakshmi [1] *, RajaSekhar.T [1]

[1] Velagapudi Ramakrishna Siddhartha Engineering College, Vijayawada,India-520007
**Corresponding author E-mail:nagasailakshmi06@gmail.com*

Abstract

Present day electronic circuits are generally affected by the delicate mistakes. To maintain the reliability of the complex systems few techniques have been proposed. For few applications, an algorithmic - based fault tolerance (ABFT) system has attempt to abuse the algorithmic properties to identify and adjust mistakes. One example FFT used. There are various protection schemes to identify and adjust errors in FFTs. It is normal to discover various blocks are working in parallel. Recently; a new method is exploiting to implement a blame tolerance in parallel. In this work, same method is first applicable to parallel FFT and then secured methods are merged that the use of error correction codes (ECCs) and parseval checks are used to detect and correct a single bit fault. Trellis code is applied to parallel FFTs to protect the errors which are used to detect and correct a multibit faults are proposed and evaluated. The 4-point FFT is protected with the input32-bit length .Simulation and Synthesis report for FFT using ECC,SOS,ECC-SOS,Trellis codes are obtained in Xilinx software14.2v.Area,power,delay is analyzed in cadence using 90nm & 180nmTechnology.

Keywords:*Convolution Encoder; Error Correction Codes (ECCS); Fast Fourier Transform (FFTS); Soft Errors; Viterbi Decoding.*

1. Introduction

In the present Digital Communications Systems, it is extremely possible that information or message get defiled during transmission and reception through a noisy channel medium which leads to the infusion of arbitrary bits in to the original message and degenerate the original message. Error correction codes (ECCs) procedures have been generally used to rectify transient errors and enhance the reliability of memories. ECC words in memories consist of information bits andextra check bits because the ECCs used in memories and they are from a class of linear block codes. In communication and signal processing applications the difficulty of electronic circuits increases every year. It is only possible by the CMOS technologyscaling that authorize the combination ofadditional number of transistor on a isolated gadget. Complexity of electronic circuits makes unprotected from delicate errors. In the meantime, the scalingimplies that transistor work with the lower voltages and are more vulnerable to mistakes caused by noise andfabricate variations.[1].Based on technology scaling ,radiation –induced delicate errors are also expanded[2]. Delicate mistakes can alert the sensible value of a circuit junction making a temporary mistake that can affect the
System activity. A widevariety of methods can be used, to protectthe delicate blunders andit does notattack the working of a given circuit [3].Triple Modular Redundancy (TMR) is a popular method thattriples a block and apply the voting logic among the three outputs to identify and adjustblunders. However, the drawback of this TMR is it consumes more area and power of the circuit .To protect the electronic circuit from delicate blunders an alternative method is try to exploit the algorithmic or structural propertiesof the circuit to identify and adjust mistakes[4]. It is usually referred as a algorithmic –based fault tolerance (ABFT).

Over the many years, Signal processing and communications circuits few ABFT methods have been implemented [5]. For the protection of Digital filters various works have been used [6]. Recently, the distribution of the FIR filter output has been used to identify and adjust blunders [7].

In digital signal processing, fast Fourier transform plays a major role which increases the computing efficiency for obtaining large discrete Fourier transform (DFT) [8]. As the difficulty of signal processing circuits becomes more, it isgenerally to find various filters or FFTs performing in parallel or multiple-inputmultiple-output (MIMO)[9].In Digital Filters ,ABFT techniques are implementedfor theentire setof parallel Filters or FFTs modules in-place ofsingle module[10].At recently a new technique is proposed which is based on Error –Correction codes (ECCs). The idea behind in this method, each filter must be equivalent of a bit in ECC and parity check bits it can be calculated by using addition. The output of the sum of various inputs is the sum of the single outputs; this technique is used for operations. For sample, the discrete Fourier transform (DFT) is a linear operation for any FFT [11].
Earlier, the protection of parallel FFTs is studied. At any time, it is consider that there can only be a single fault on the system. There are three fundamental principles are offering .they is:

1) The estimation of the ECC method [12] for the secure of parallel FFTs displays its function in terms of overhead and protection effectiveness.
2) A new method is proposed which is based on the use of Parseval checks or sum of Squares (SOSs) checks merged with the redundant (parity) FFT [4].
3) The proposition of another method on which the ECC is utilized on the SOS checks rather than on the FFTs.

The proposition of two methods produce a new approaches to secure parallel FFTs that can be more area efficient than compare to single FFTs independently.

The rest of this paper is explained as follows .Section 2, shows the two protection schemes (Existing System). In Section 3 ,presents the proposedsystem of the convolutional encoder and viterbi decoder (Trellis codes) for the secure of parallel FFT. Section 4, presents results and Discussion comparison graphs. Section 5 and 6 gives the conclusion of the project and future scope of this paper.

2. Existed system

2.1. Parallel FFT protection using ECC

The main objective of fault tolerant is to secure the parallel FFTs from errors. Different protection techniques have been proposed for error

recognization and identification in FFTs. In digitalfilters a newprotection methodis used which is based on theuse of ECCs[12].Thistechnique is shown in Fig .1.for case, the basic and simple method is error correction hamming codes is used[13].Redundant (parity) bit is used to detect errors which are used to perform XOR operation with the data bits. let us assume a simple Hamming code[12]. In Fig.1. In this, original system contains four FFT modules with inputs x1,x2,x3and x4 and the outputs of the FFT modules are y1,y2,y3 &y4.The redundantsystem contains three FFTmodules with inputs x5,x6, & x7are used perform XOR operationwith the inputsx1,x2,x3 & x4 and the outputs of theFFTare z1,z2 & z3 . The outputs of the FFT are z1, z2 & z3 are also used to perform XOR operation with the outputs of the original FFT modules y1, y2, y3& y4.

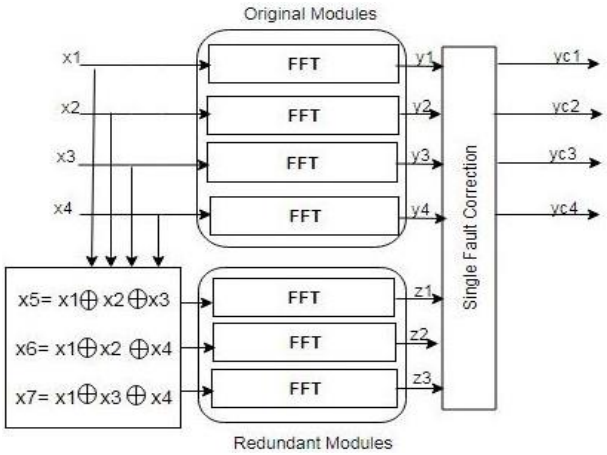

Fig. 1: Parallel FFT Protection Using ECCS.

For case, the input to the first, second &third redundant modules are

$$X5=x1x2x3 \quad (1)$$

$$X6=x1x2x4 \quad (2)$$

$$X7 \oplus x1x \oplus x4 \quad (3)$$

The DIF -FFT is used perform a linear operation, so its output z1, z2 & z3 are

$$Z1=y1y2y3c1 \quad (4)$$

$$2=y \oplus 2y4 \rightarrow c2 \quad (5)$$

$$Z3 \oplus y1 \oplus y4 \rightarrow c3 \quad (6)$$

By comparing the three redundant FFT outputs z1,z2 & z3and z1,z2 & z3 are XOR combinations oforiginal FFT outputs y1,y2,y3 & y4 and it is denoted as checks c1,c2 & c3. To observe the differences between the each and every check on which module fault can be identified. The various patterns and related faults are organized in Table 1.To identify the fault in the module, it can be corrected by rebuild its output utilizing the remaining modules. For case, a fault occurred in z1, this can be corrected as follows:

$$y1c[n] = z1[n] - y2[n] - y3[n] \quad (7)$$

$$y2c[n] = z2[n] - y1[n] - y4[n] \quad (8)$$

$$y3c[n] = z3[n] - y1[n] - y4[n] \quad (9)$$

Table 1: Error Location in the Hamming Code

c1 c2 c3	Error Bit Position
0 0 0	No error
1 1 1	z1
1 1 0	z2
1 0 1	z3
0 1 1	z4
1 0 0	z5
0 1 0	z6
0 0 1	z7

2.2. Parity-SOS fault-tolerant parallel FFTS

Parity-SOS Fault tolerant parallel FFT is the first technique .Sum of Squares (SOS) check is also depends on the Parseval theorem which is used to detect mistakes & it is defined as the SOSs of the inputs is equal to SOSs of the outputs of the FFT.

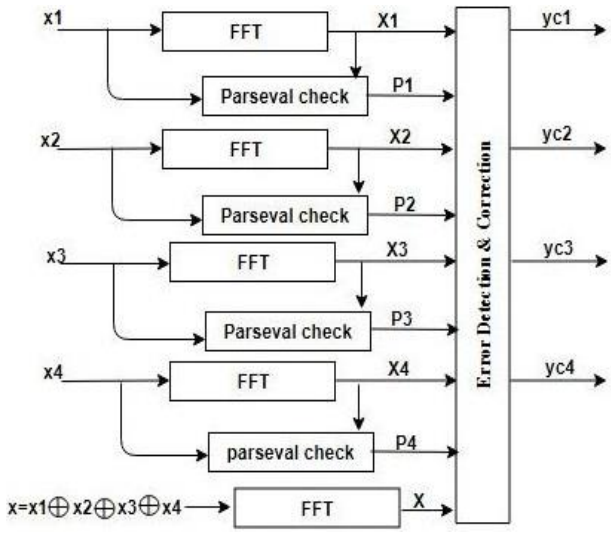

Fig. 2: Parity-SOS Fault Tolerant Parallel FFT.

In this first technique, it consists of four FFT modules with inputs x1, x2, x3 & x4 and outputs X1, X2, X3 & X4 and also parity (redundant) FFT is addition of all original FFT inputs. SOS check is merged with the ECC.SOS check is used on each FFTto detect mistakes .suppose an fault is detected ,the output of the parity FFT is used to correct that errors andECC is used to correct that mistakes. For example an error is detected using P1, P2, P3 & P4 and it can be corrected by rebuild the FFT with fault

By utilizing the parity FFT(X) output and the remaining FFT outputs.

For case, a fault occurred in the first FFT, then P1 is set as one and it can be corrected by using equations

$$X1c=X-X2-X3-X4 \quad (10)$$

X2c=X-X1-X3-X4(11)

X3c=X-X1-X2-X3 (12)

X4c=X-X1-X2-X3(13)

2.3. Parity-SOS-ECC fault tolerant parallel FFTS

Parity-SOS-ECC Fault tolerant parallel FFT is a second technique which is combined with SOS check and ECC replace by using an SOS check per FFT .Parity-SOS is used to detect an error and ECC part is used to correct that mistake. In this second technique, the advantage of the first Parity-SOS technique is minimizing the SOS checks. The advantages of these two techniques are minimizing the number of SOS checks needed. In this second technique, it contains four FFT original modules with inputs x1,x2,x3 & x4 and outputsX1,X2,X3 & X4 as shown in fig.3 .The x5,x6,x7 and X5,X6,X7 are the inputs to the parseval check and it used to perform XOR operation with the inputsand outputs of the original FFTs. Parseval check is used to compare theboth inputs ,if we are getting same outputs then it is represented as a no error. Suppose, if we are detect an error in X1 then the inputs to the parseval check performs comparison if it is not equal then it is represented as an error in the module .An Parity-FFT is used to correct that error .

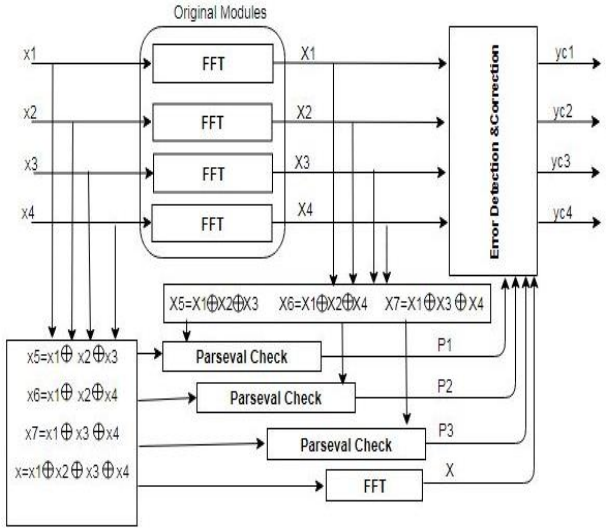

Fig. 3: Parity-SOS-ECC Fault-Tolerant Parallel FFT.

In this Parity-SOS-ECC (second technique) fault tolerant parallel FFT an error is corrected by using equations which is same as first technique. In this second technique it minimizes the number of SOS checks.

Table 2: Different Schemes to Protect K FFTS

	FFTs	SOS checks
ECC	$1+\log_2(k)$	0
Parity-SOS	1	k
Parity-SOS-ECC	1	$1+\log_2(k)$

This table is summarized as a set ofk original modules and let us assumeskis a power of two.

2.4. Architecture of the single FFT processor

The Block Diagram contains Address Generator Unit (AGU), Selector, Dual Port RAM, four-point decimation-in-frequency FFT, Cordic and Twiddle Factor generation unit as shown in fig.4.

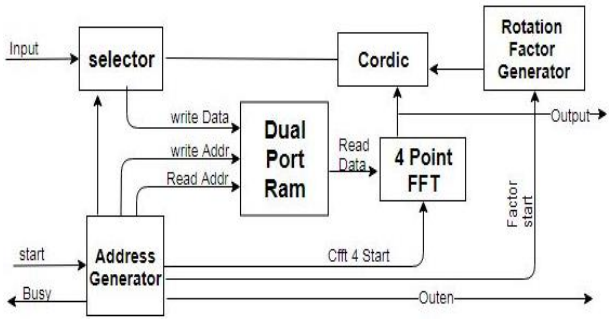

Fig. 4: Architecture of the FFT Implementation.

The Multiplexer (selector) is also called as a memory path buffer which computes the respective memory of input samples. Whenever, we can assert an active signal and there some input data is available. For each input sample, the address generator unit block assigns a memory location. From the Address Generation Block (AGU), the Dual Port RAM gets write Address signal and it protects both the memory path along with the respective input samples. The FFT has a Butterfly Unit within it.

At the same time, start signal is enabled both 4-point FFT and Rotation Factor Block, the FFT block sends a signal to Cordic block for calculatethe required twiddle factors containing sine-coseterms. so, the Cordic block is controlled by RF generator block.Whenever,the address generator block enable a read address signal to Dual Port RAM(DRAM),it passes the stored data samples along with memory path in FFT block. At lastly, Twiddle factor values are applicable to the output of the butterflies and a bit reverse scram is done.

The formula is,

Stage=log4 (computing point)

The input points to the FFT are 32-points for computing ($\log 4^{32}$ = 2) with two stages, totally 2*32=64 cycles are required to compute the FFT for 32 points.

2.5. Percevalchecks (SOS)

In this block diagram contains, Magnitude square, Accumulator, Magnitude Comparator and the FFT are shown in fig.5. The inputs and outputs of the FFT are sequential order .The FFT inputs and outputs are given to the Magnitude square which performs squaring operation and the outputs of the magnitude square isgiven to the input of theaccumulator which performs adding operation and the outputs of the accumulator are given to the inputs of the comparator which compares both the inputs, if both inputs are same then it represents Parseval check as zero means no error, if it is not equal it represents as a error .

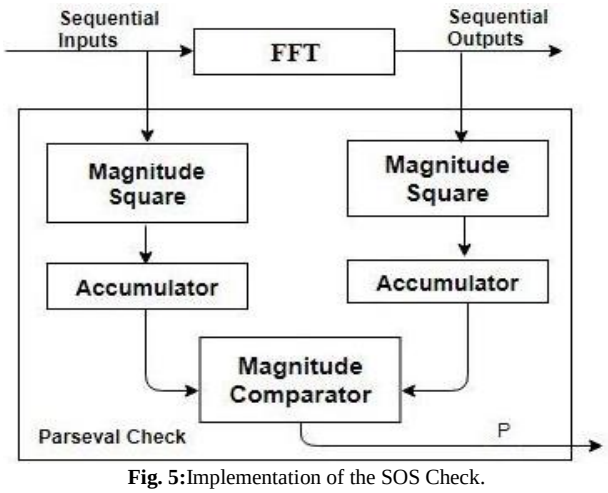

Fig. 5: Implementation of the SOS Check.

Parseval check is also called as Sum of Squares (SOS) and it is defined as the Addition of Squares of the input to the FFT is equal to the Addition of Squares of the output of the FFT. Sum of Squares is used to detect an error.

3. Proposed system

3.1. Parity-CE-VD fault-tolerant parallel FFTS

In this Block Diagram consists of Parallel FFT, Convolutional Encoder and Viterbi Decoder as shown in Fig.6.The inputs to the parallel FFT are x1,x2,x3 & x4 and outputs of the FFT are X1,X2,X3&X4.

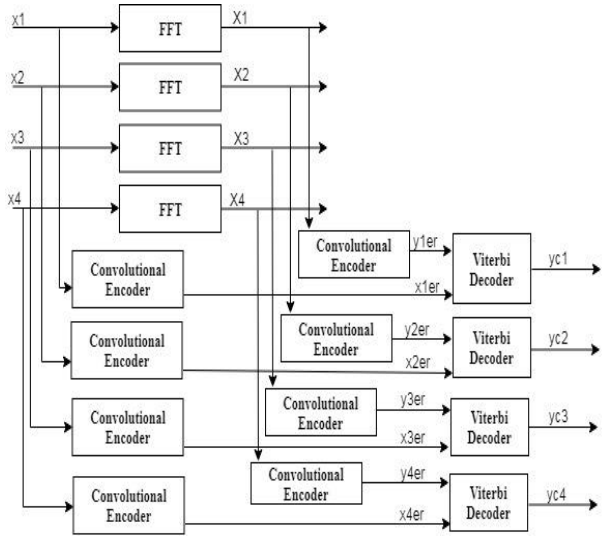

Fig. 6:. Parity-CE-VD Fault Tolerant Parallel FFT.

The inputs and outputs of the FFT are given as inputs to the convolutional encoder .In the Convolutional encoder some information bits are given to theshift registers. The output bits are encoded by modulo-2-addition (EXCLUSIVE-OR operation) for the input bits information and previous information bits in the shift register. The working of Convolutional encoder
Can be explained in different ways those are

1) State Diagram representation
2) Tree Diagram representation and
3) Trellis Diagram representation.

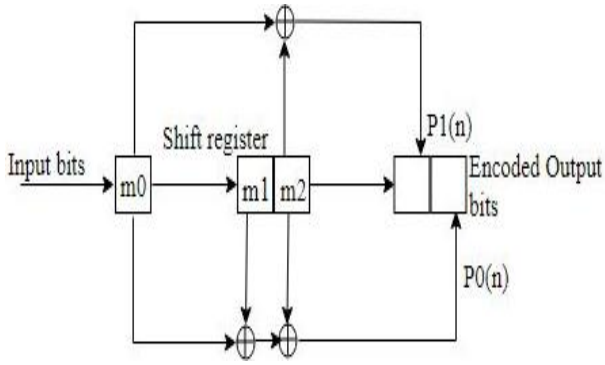

Fig. 7:A Convolutional Encoder with K=1, N=2, R=1/2.

In this Block Diagram m0, m1& m2 are the shift registers p0 (n), p1(n) are the encoded output bits.For the input sequence (01011001) the encoded output bits are (p0, p1) = (00 11 11 01 00 01 10 10).
State Diagram & its Truth Table:
In the State Diagram, the transition takes place from one state to another state; it is denoted as b/p0p1.

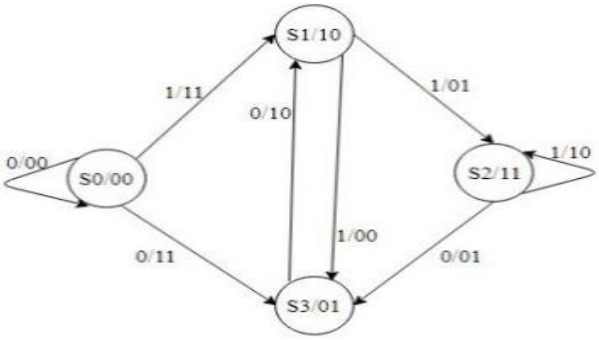

Fig. 8: State Diagram Representation.

Based on the input sequence (01011001) and the output sequence is(00 11 11 01 00 01 10 10) and the state truth table is as shown belowTable III.

Table 3: State Transition Table

Input bits	Present State	Next State	Output bits
0	00	00	00
1	00	10	11
0	00	01	11
1	10	11	01
1	10	01	00
0	11	01	01
1	11	11	10
0	01	10	10

4. Viterbi decoder (trellis)

The Decoding Algorithm use Two Metrics: Path Metric (PM) and Branch Metric (BM). The outputs of the convolutional encoder are given as input to the Viterbi Decoder (x1er, x2er ,x3er,x4er & y1er,y2er,y3er,y4er) as shown in Fig.6.Branch Metric is defined as the distance between the what was transmitted and what was receivedat each arc in the trellis .Path Metric is defined as the number of bit errors detected. Finally, path metric and Branch Metric are used to find errors it can be corrected by using Viterbi decoding. In this Parity-CE-VD technique multiple numbers of bits are detected and corrected.

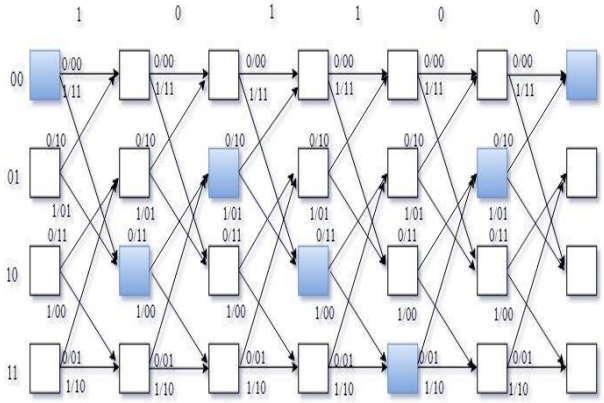

Fig. 9: Trellis Diagram Representation.

Hamming Distance (BM) is calculated by using this equation

BM_α= Hamming Distance (P_α, P_received)

BM_β= Hamming Distance (P_β, P_received)

Path Metric is calculated by using equation as shown below

PM[s, i+1] =min (PM [α, i] +BM [αs]

PM [β, i] +BM [β➔s])

5. Results and discussion

TheResults for theprotection Schemes, FFT , Parseval Checks and Trellis codeare implementedusingVerilogXilinx ISE software 14.2 version .The 4-pointFFT with the input bit length 32 is used for the Parallel FFT protection ,Parity-SOS fault tolerant Parallel FFT ,Parity-SOS-ECC Techniques and Parity –CE-VDfault tolerant parallel FFT techniques .The area, power, delayis analyzed by using cadence 90nm &180nm . Resource utilization of Single FFT and Parseval Check is shown in below Table IV.

Table 5: Resource Utilization of Single FFT &SOS Check

parameters	FFT	SOS check
Slices	70	96
Flip-Flop	136	128
LUT-4	32	192
Delay(ns)	3.806(ns)	7.055(ns)
Frequency(MHz)	413.206(MHz)	286.058(MHz)

Table 6: Resource Usage for Virtex-5 Device

Parameters	ECC protected	Parity-SOS protected	Parity-SOS-ECC protected	Parity-CE-VD protected
Slices	269	256	345	291
Flip-Flop	803	880	812	450
LUT-4	582	408	582	673
Delay(ns)	2.826(ns)	2.832(ns)	2.826(ns)	2.830(ns)
Frequency(MHz)	302.586(MHz)	126.21(MHz)	111.352(MHz)	280.654(MHz)

Table 7: Resource Usage for Virtex-6 Device

Parameters	ECC protected	Parity-SOS protected	Parity-SOS-ECC protected	Parity-CE-VD protected
Slices	233	233	263	251
Flip-Flop	803	880	812	406
LUT-4	569	471	592	359
Delay(ns)	0.659(ns)	0.657(ns)	0.659(ns)	0.664(ns)
Frequency(MHz)	505.229(MHz)	125.264(MHz)	116.711(MHz)	373.320(MHz)

Table 8: Resource Usage for Virtex-7 Device

Parameters	ECC protected	Parity-SOS protected	Parity-SOS-ECC protected	Parity-CE-VD protected
Slices	335	352	423	242
Flip-Flop	803	880	812	406
LUT-4	953	846	965	354
Delay(ns)	0.575(ns)	0.575(ns)	0.575(ns)	0.580(ns)
Frequency(MHz)	361.232(MHz)	121.682(MHz)	133.847(MHz)	423.250(MHz)

Table V-VII shows the comparison analysis of FFTs, Parity-SOS, SOS-ECC, and Parity-CE-VD for the devices Virtex-5, Virtex-6 and Virtex-7. Table VIII shows the comparative results of Existing and Extension methods for Parallel FFT Parity-SOS-ECC and Parity-CE-VD.

Table 9: Comparison of Parity-SOS-ECC &Parity-CE-VD

Target Device	Design Technique	Method	No. of Slices	No. of Slice registers	No. of Slice LUTs	Frequency (MHz)	Delay(ns)
Virtex-5	Parity-SOS-ECC	Existing	345	812	582	111.352MHz	2.826(ns)
	Parity-CE-VD	Extension	291	450	673	280.654MHz	2.830(ns)
Virtex-6	Parity-SOS-ECC	Existing	263	812	592	116.711MHz	0.659(ns)
	Parity-CE-VD	Extension	251	406	359	373.320MHz	0.644(ns)
Virtex-7	Parity-SOS-ECC	Existing	423	812	965	133.847MHz	0.575(ns)
	Parity-CE-VD	Extension	242	406	354	423.250MHz	0.580(ns)

Some improvement also achieved in the Area, Delay (ns), Frequency (MHz). To further demonstrate this thing, the number of Slices are necessary for various methods, number ofFlip-Flops and number of 4-input LUTs are plotted as shown in below Figure.10 for Virtex-5 ,Figure.11 for Virtex-6 and Figure .12 for Virtex-7 respectively.

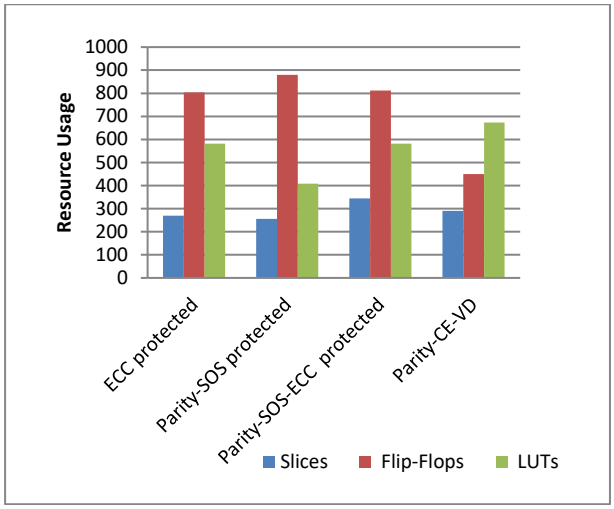

Fig. 10:. Comparison Graph for Different Schemes Virtex-5.

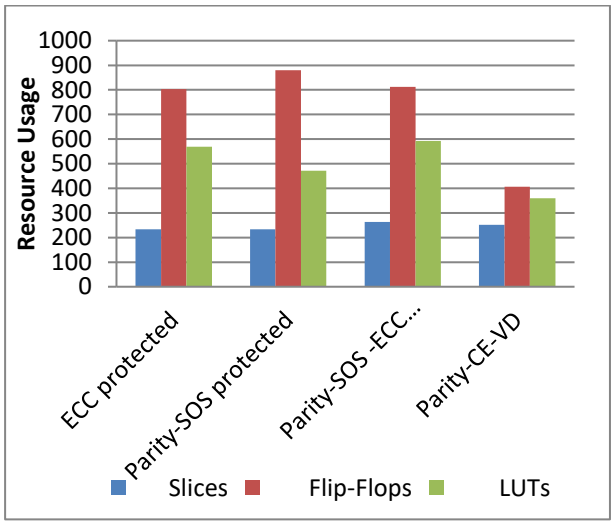

Fig. 11:. Comparision Graph for Various Schemes Virtex-6.

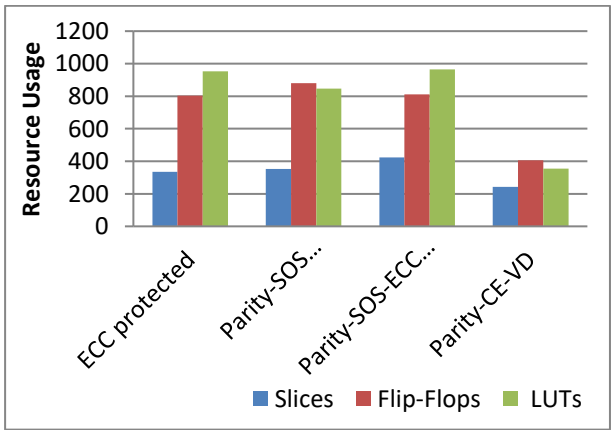

Fig. 12:. Comparision Graph for Various Schemes Virtex-7.

Below Figure shows the output result for the existing and proposed technique obtained in Xilinx ISE. It contains the total number of various hardware resource utilized for the new proposed technique for four-parallel Parity-CE-VD methods as already mentioned in the above Tables . The 4-point DIF FFT and the various fault tolerant schemes have been implemented by using Verilog HDL. In this existingtechniquedetect and correct single bit fault andin the proposed techniques which is used to detect and

correct a multi bit faults by using trellis code and it also reduce the area ,power & delay is analyzed by using cadence report 90nm & 180nm as shown in belowTable.

Table 10: Cadence Report for the 90nm and 180nm Technology

Technology	Design Technique	Method	Total Number of Gates	Total Power(nw)	Area(μm²)
90nm	Parity-SOS-ECC	Existing	96770	9727815.429	96770
90nm	Parity-CE-VD	Extension	15092	359167.380	15093
180nm	Parity-SOS-ECC	Existing	302552	6463945.394	302553
180nm	Parity-CE-VD	Extension	47580	1214050.515	47581

In this Table represents the performance analysis of Parity-SOS-ECC and Parity-CE-VD, which reduces the area, power and gates.

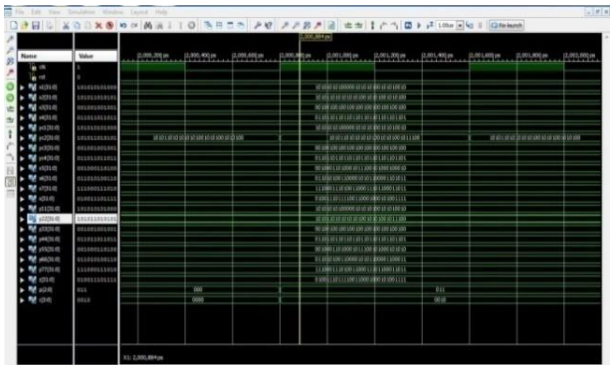

Fig. 13: Simulation Results for Parity-SOS-ECC Fault-Tolerant Parallel Ffts Fault and Corrected Output.

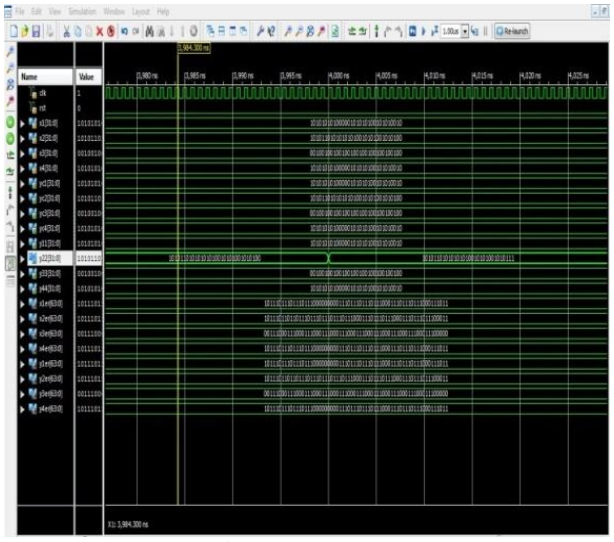

Fig. 14: Simulation Results for Parity-CE-VD Fault-Tolerant Parallel FFTS Fault and Corrected Output.

Atlastly, we can observe that ECC scheme can identifyand adjustment all the faults as shown in above Fig.13.Parity-CE-VD schemeis used to detect and correctmultiple number offault bits which reduces area,power,delay.Comparisons of both the methods are executed for fault injection experiments are completed to identify the percentage of mistakes are detected and adjusted. The execution and analysis have been completed in terms of overhead (power & area) fault coverage.

6. Conclusion

The protection of parallel FFTs has been studied. Detection and correction of errors have been implemented. To reduce the complexity of the FFT, it is used in parallel which increases the speed of circuit operation. Detection and correction of errors have been executed. In the existing method, use the three schemes for the protection of FFT which is able to detect up to two faults and correct only single fault. The 4-point FFT with the input bit length 32 is used and protected. Now proposal Scheme is depends on the combination of Parity-CE-VD (trellis code) called as Convolutional Encoder-Viterbi Decoder. This method is used to detect and correct multi fault bits.Convolutional encoder is used to transmit the message sequence and produce errors and the path metric (PM) and branch metric (BM) are used to find errors and it can be corrected by using Viterbi Decoding. To compare both the techniques parity-SOS-ECC and parity-CE-VD which reduce its area, power, delay in results compared to existing techniques.Cadence report is analyzed for both the Parity-SOS-ECC and Parity-CE-VD techniques for 90nm & 180nm technology. Simulation and Synthesis for all the techniques are obtained by using Xilinx ISE 14.5v. Verilog HDL language is used.

References

[1] N.Kanekawa,E.H.Ibe,T.Suga, and Y.Uematsu, Dependabilitu in Electronic Systems: Mitigation of Hardware Failures, Soft Errors, and Electro-Magnetic Disturbances. NewYork, NY, USA: Springer-Verlag, 2010.

[2] R.Baumann, “soft errors in advanced computer systems,” IEEE Des.Test Computers., Vol.22,No.3,pp.258-266,May/Jun.2005.https://doi.org/10.1109/MDT.2005.69.

[3] M.Nicolaidis, “Design for sofr error mitigation,”IEEE Trans.Device Mater.Rel., Vol.5, No.3, pp.405-418, sep.2005.https://doi.org/10.1109/TDMR.2005.855790.

[4] A.L.N.Reddy and P.Banerjee, “Algorithm-based fault detection for signal processing applications,IEEETrans.Computrs” Vol.39,No.10,pp.1304-1308,Oct.1990.

[5] T.Hitana and A.K.Deb, “Bridging concurrent and non-concurrent error detection in FIR filters,”in proc.Norchip Conf., Nov.2004, pp.75-78.https://doi.org/10.1109/NORCHP.2004.1423826.

[6] S.Ponta’relli,G.C.Cardarilli,M.Re,and .Salsano,”Totally fault tolerant RNS based FIR filters,”in ‘Proc.14th IEEE Int.On-Line Test Symp. (IOLTS), jul.2008, pp.192-194.

[7] B.Shim and N.R.Shanbhag, “Energy-efficient soft error-tolerant digital signal processing,”IEEE Trans.Very Large ScaleInteger. (VLSI)Syst., Vol.14, No.4, pp.336-348, Apr.2006.

[8] J.Y.Jou and J.A.Abraham, “Fault-tolerant FFT networks, IEEETrans.Computer, Vol.37, No.5, pp.548-561, May1998. https://doi.org/10.1109/12.4606.

[9] 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.https://doi.org/10.1109/JPROC.2003.821912.

[10] P.Reviriego,S.Pontarelli,C.J.Bleakley,andJ.A.Maestro,”Area efficient concurrent error detection and correction for parallel filters,”IET Electron.Lett.,Vol.48,No.20,pp.1258-1260,sep.2012.https://doi.org/10.1049/el.2012.2237.

[11] 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.

[12] R.W.Hamming,”Error detecting and error correctingcodes, BellSyst.Tech.J” Vol.29, No.2, pp.147-160, apr.1950.https://doi.org/10.1002/j.1538-7305.1950.tb00463.x.

[13] Zhen Gao, Pedro Reviriego,Zhan Xu,Xin Su, Ming Zhao, Jing Wang, and Juan Antonio Maestro, “ Fault-Tolerant Parallel FFTs using Error Correction Codes and Parseval Checks,” IEEE Trans.Very Large Scale Integer.(VLSI) Syst., Vol.24,No.2,pp.769-773,Feb.2016.