Efficient Fault Tolerant Parallel FFTs using Vedic Urdhva Triyakbhyam Sutra and ECC

Apeksha Jaiswal[1], Prof. Pankaj Kumar Vyas[2]

[1]M.Tech. Scholar, [2]Guide

Department of Electronics and Communication Engg., SISTech, Bhopal

Abstract - Modern digital circuits are prone to errors and these are the part of system. So elimination or reduction is the primary need of this research field. Some of the areas which are affected by the errors are wireless communication and signal processing applications etc. Few of them adopting fault tolerant mechanisms to get rid of it or to detect and correct errors. This approach quite suitable for signal processing and wireless communication kind of systems. Signal processing mainly involves fast Fourier transforms (FFTs) which forms large signal processing system. If we are successful to achieve error correction for FFTs then large problems of various communication and signal processing system will be eliminated. In this work the error correction codes and the fault tolerant algorithm is proposed which is quite efficient in terms of resource usage with the integration of Vedic methodologies, an urdhva triyakbhyam (UT) approach. The synthesis outcomes are clearly shows the merits of proposed methodology.

Keywords - Vedic Sutra, Urdhva Triakbhyam, ECC, Fault Tolerant, FFTs.

I. INTRODUCTION

In mathematics, the Fourier Transform is an operation that transforms one function of a real variable into another. The new function, often called the frequency domain representation of the original function, describes which frequencies are present in the original function. This is in a similar spirit to the way that a chord of music that we hear can be described by notes that are being played. In effect, the Fourier Transform decomposes a function into oscillatory functions. The Fourier transform is similar to many other operations in mathematics which make up the subject of Fourier Analysis. In this specific case, both the domains of the original function and its frequency domain representation are continuous and unbounded. The term Fourier Transform can refer to both the frequency domain representation of a function or to the process that transforms one function into the other.

Currently, the FFT is used in many areas, from the identification of characteristic mechanical vibration frequencies to image enhancement. Standard routines are available to perform the FFT by computer in programming languages such as Pascal, Fortran a C, and many spreadsheet and other software packages for the analysis of numerical data allow the FFT of a set of data values to be determined readily.

- Discrete Fourier Transform

The Fourier transform operates on continuous functions, i.e., functions which are defined at all values of the time t. Such a function might, for example, represent a continually-varying analog voltage signal produced by a microphone or other type of transducer. Digital signal processing involves discrete signals, which are sampled at regular intervals of time rather than continuous signals. A modified form of the Fourier Transform, known as the Discrete Fourier Transform or DFT, is used in the case of discrete signals.

$$X_k = \sum_{n=0}^{N-1} x_n e^{\frac{-2\pi i}{N} kn}, \qquad k = 0, \dots\dots\dots\dots\dots\dots N-1$$

where $\frac{-2\pi i}{N}$ is a primitive N'th root of unity.

The Discrete Fourier Transform is one of the most fundamental operations. In mathematics, the Discrete Fourier Transform is one of the specific forms of the Fourier Series Analysis. It transforms one function into another, which is called the frequency domain representation, or simply the DFT, of the original function which is often a function in the time domain. But the DFT requires an input function that is discrete and whose non-zero values have a limited or finite duration. Such inputs are often created by sampling a continuous function. Unlike the discrete-time Fourier Transform, it only evaluates enough frequency components to reconstruct the finite segment that was analyzed. Its inverse transform cannot reproduce the entire time domain, unless the input happens to be periodic. Therefore it is often said that the DFT is a transform for Fourier analysis of finite-domain discrete-time functions. The sinusoidal basis functions of the decomposition have the same properties. Since the input function is a finite sequence of real or complex numbers, the DFT is ideal for processing information stored in computers. In particular, the DFT is widely employed in signal processing and related fields to analyze the frequencies contained in a sampled signal, to solve

partial differential equations, and to perform other operations such as convolutions. The DFT can be computed efficiently in practice using a Fast Fourier Transform algorithm.

When the DFT is applied to a discrete signal, the result is a set of sine and cosine coefficients. When sine and cosine waves of appropriate frequencies are multiplied by these coefficients and then added together, the original signal waveform is exactly reconstructed. The sine and cosine waves are the frequency components of the original signal, in the sense that the signal can be built up from these components. The coefficients determined by the DFT represent the amplitudes of each of these components.

II. SYSTEM MODEL

Vedic Mathematics is the name given to a set of rules derived from Ancient Indian Scriptures, elucidating different mathematical results and procedures in simple and un- derstandable forms. The word Vedic is derived from the word Veda which means the store–house of all knowledge.

It is claimed to be a part of the Sthapatya Veda , a book on civil engineering and architecture, which is an Upaveda (supplement) of the Atharva Veda . It covers explanations of several modern mathematical terms including arithmetic, geometry (plane, co-ordinate), trigonometry, quadratic equations, factorization and even calculus.

The beauty of Vedic mathematics lies in the fact that it reduces the otherwise cumbersome- looking calculations in conventional mathematics to very simple ones. This is because the Vedic formulae are claimed to be based on the natural principles on which the human mind works.

His Holiness Jagadguru Shankaracharya Bharati Krishna Teerthaji Maharaja (1884-1960) comprised all this work together and gave its mathematical explanation while discussing it for various applications.16 sutras (formulae) and 16 Upa sutras (sub formulae) was constructed by Swamiji after extensive research in Atharvana Veda. these formulae were constructed by Swamiji himself obviously these formulae are not to be found in present text of Atharvana Veda since Vedic mathematics which couldn‟t be disapproved because it is not only a mathematical wonder. Vedic Mathematics has already crossed the boundaries of India and has become a leading topic of research abroad due these phenomenal characteristic. Several basic as well as complex mathematical operations are deal with vedic mathematics. The various branches of mathematics like geometry, algebra, arithmetic are mainly included in Vedic mathematics based on 16 Sutras (or aphorisms) dealing with etc. These Sutras are enlisted below alphabetically along with their brief meanings

1. *(Anurupye) Shunyamanyat –If one is in ratio, the other is zero.*
2. *Chalana-Kalanabyham –Differences and Similarities.*
3. *Ekadhikina Purvena –By one more than the previous One.*
4. *Ekanyunena Purvena –By one less than the previous one.*
5. *Gunakasamuchyah –The factors of the sum is equal to the sum of the factors.*
6. *Gunitasamuchyah –The product of the sum is equal to the sum of the product.*
7. *Nikhilam Navatashcaramam Dashatah –All from 9 and last from 10.*
8. *Paraavartya Yojayet –Transpose and adjust.*
9. *Puranapuranabyham –By the completion or no completion.*
10. *Sankalana- vyavakalanabhyam –By addition and by subtraction.*
11. *Shesanyankena Charamena –The remainders by the last digit.*
12. *Shunyam Saamyasamuccaye –When the sum is the same that sum is zero.*
13. *Sopaantyadvayamantyam –The ultimate and twice the penultimate.*
14. *Urdhva-tiryakbhyam –Vertically and crosswise.*
15. *Vyashtisamanstih –Part and Whole.*
16. *Yaavadunam –Whatever the extent of its deficiency.*

These methods of architectures and ideas may be applied to applied mathematics of various kinds and spherical geometry, plain and conics, trigonometry, calculus (both differential and integral) directly. All these Sutras were reconstructed from ancient Vedic texts early in the last century s mentioned earlier. There are so many Sub-sutras were also discovered at the same time but are not discussed here.

The reduction of cumbersome-looking calculations in conventional mathematics to a very simple one is the beauty of Vedic mathematics lies in that. The natural principles on which the human mind works are the basic building blocks of the Vedic formulae . Various branches of engineering such as computing and digital signal processing.

The multiplier architecture can be generally classified into three categories. First is the serial multiplier which emphasizes on hardware and minimum amount of chip area. Second is parallel multiplier (array and tree) which carries out high speed mathematical operations. But the drawback is the relatively larger chip area

consumption. Third is serial- parallel multiplier which serves as a good trade-off between the times consuming

serial multiplier and the area consuming parallel multipliers.

Vedic mathematics is mainly based on 16 Sutras (or aphorisms) dealing with various branches of mathematics like arithmetic, algebra, geometry, etc. Of these, there are two Vedic Sutras meant for quicker multiplication. They have been traditionally used for the multiplication of two numbers in the decimal number system. They are –

1. *Nikhilam Navatashcaramam Dashatah : All from 9 and last from 10*
2. *Urdhva Tiryakbhyam : Vertically and crosswise*

III. PROBLEM STATEMENT

The complexity of communications and signal processing circuits increases every year. This is made possible by the CMOS technology scaling that enables the integration of more and more transistors on a single device. This increased complexity makes the circuits more vulnerable to errors. At the same time, the scaling means that transistors operate with lower voltages and are more susceptible to errors caused by noise and manufacturing variations. The importance of radiation-induced soft errors also increases as technology scales. Soft errors can change the logical value of a circuit node creating a temporary error that can affect the system operation. To ensure that soft errors do not affect the operation of a given circuit, a wide variety of techniques can be used. These include the use of special manufacturing processes for the integrated circuits like, for example, the silicon on insulator. Another option is to design basic circuit blocks or complete design libraries to minimize the probability of soft errors. Finally, it is also possible to add redundancy at the system level to detect and correct errors. The detection and location of the errors can be done using check per FFT or alternatively using a set of checks that form an ECC [1].

IV. PROPOSED METHODOLOGY

This work proposes an efficient algorithm fault tolerant parallel FFTs using Vedic Urdhva Triyakbhyam Sutra and ECC. There are two schemes are used in proposed work fist scheme has been shown figure 4.1 RTL Schematic Top Module First Technique. Shows proposed error detection and error correction scheme. RTL schematic of internal architecture of proposed top module has been shown in figure 4.2. In this work the error correction codes and the fault tolerant algorithm is proposed which is quite efficient in terms of resource usage with the integration of Vedic methodologies, an urdhva triyakbhyam(UT) approach.

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. The Discrete Fourier Transform is a continuous Fourier transform for the use of discrete functions.

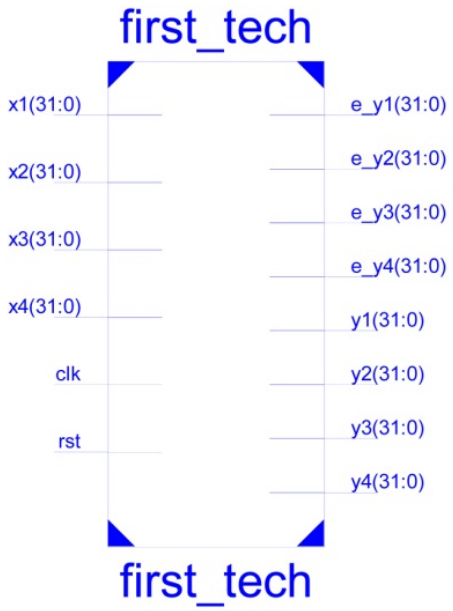

Figure: 4.1 RTL Schematic Top Module First Technique.

In figure 4.2 there are 8 sub block are used to design proposed scheme fist to design a fault tolerant FFTs to detect and correct error during either in communication application or in a computation application

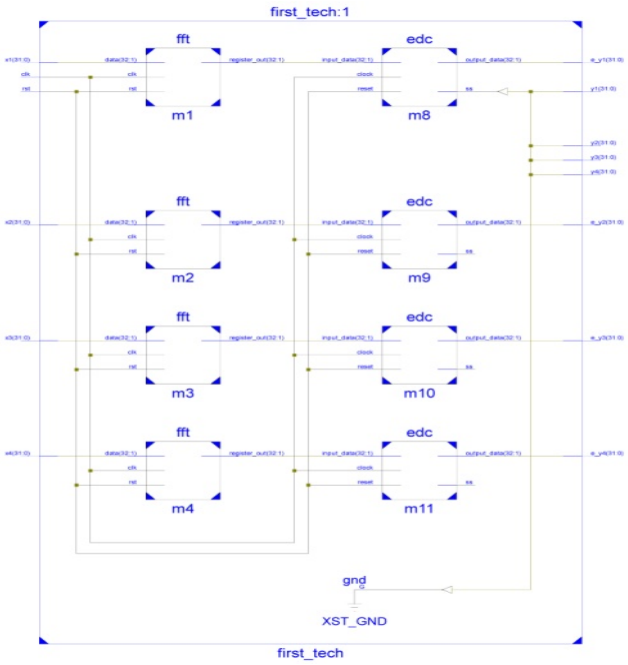

Figure: 4.2 RTL Schematic of Internal circuit of Top Module first technique.

Figure shows the RTL schematic inside FFT and EDC Module fist technique.

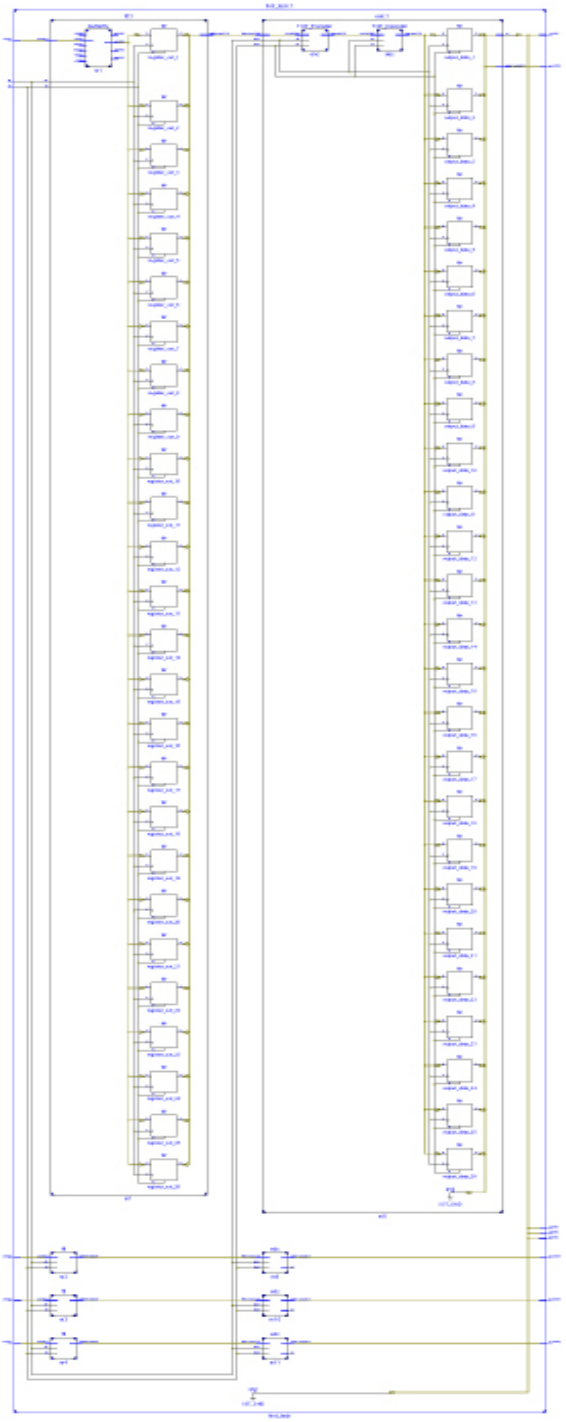

Figure: 4.3 RTL Schematic Inside FFT and EDC Module First Technique.

RTL schematic of second technique based FFT design for fault tolerant error detection and error corection code using Vedic Urdhva Tiryabham has been shown in figure 4.4.

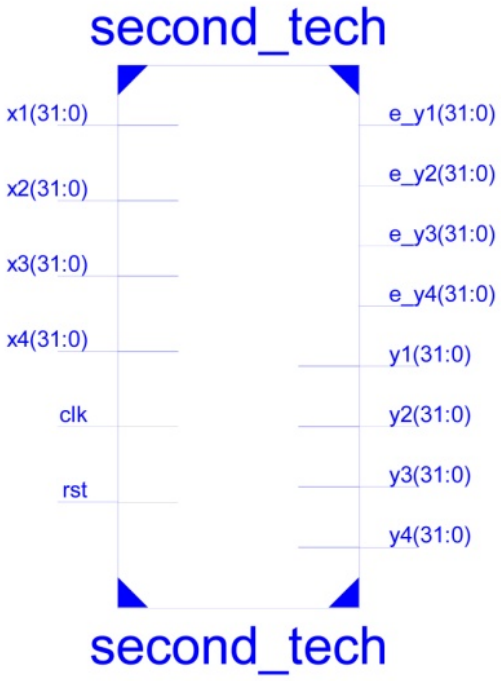

Figure: 4.4 RTL Schematic Top Module Second Technique.

RTL Schematic Top Module internal architecture of Second Technique in figure 4.5. and RTL Schematic Inside FFT and EDC Module Second Technique has been shown in figure 4.6.

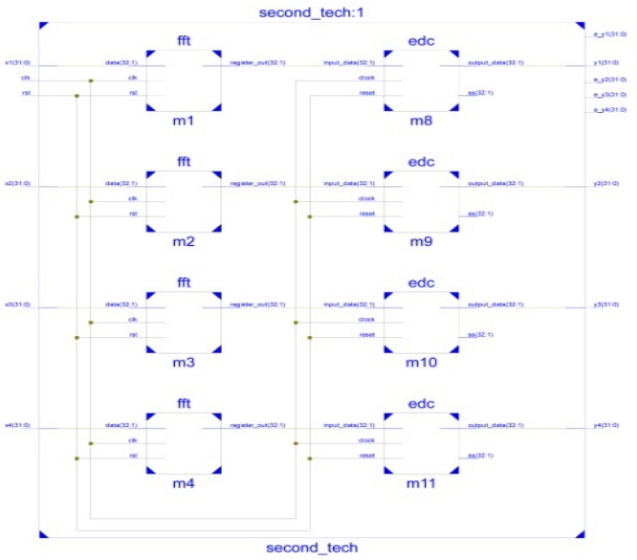

Figure: 4.5 RTL Schematic Inside Top Module Second Technique.

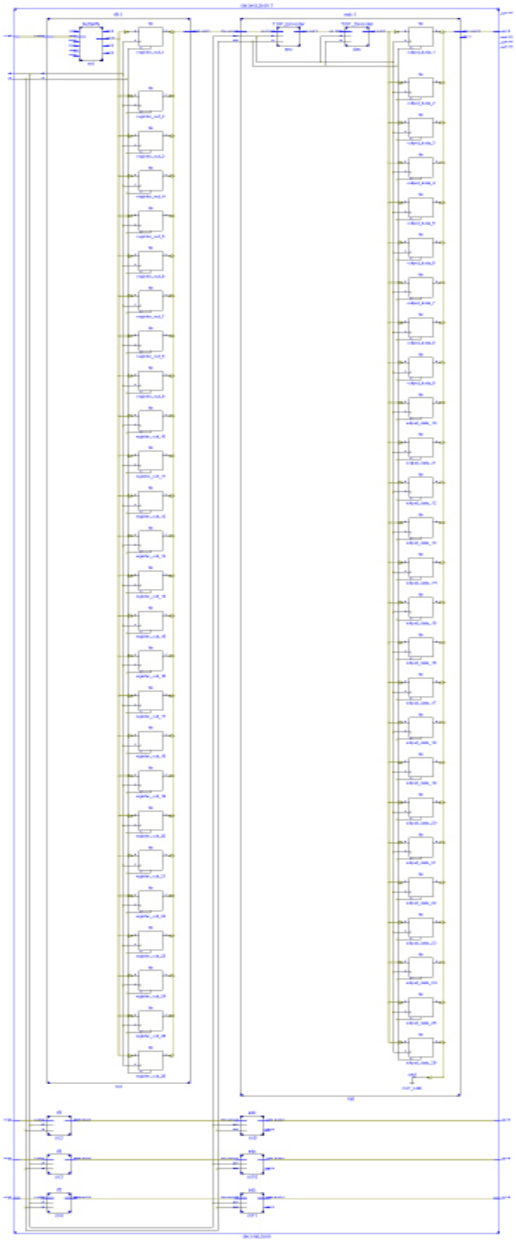

Figure: 4.6 RTL Schematic Inside FFT and EDC Module Second Technique.

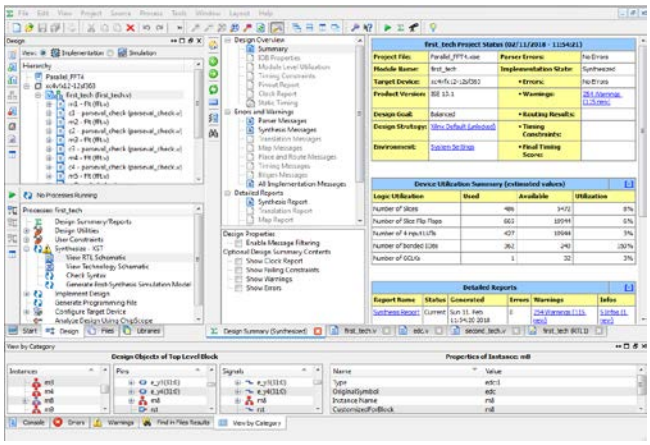

Figure: 5.1 Simulation Screen of Proposed Work fist techniques.

Simulation of propose fault tolerant fast furrier transform FFT based on Vedic multiplier and error correction code ECC has been implemented and simulated on Xilinx 1.3.1 ISE design suite. Simulation screen of proposed work has been shown in figure 5.1.

Device utilization summary of proposed work using first technique has been shown in figure 5.2.

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	486	5472	8%
Number of Slice Flip Flops	665	10944	6%
Number of 4 input LUTs	427	10944	3%
Number of bonded IOBs	362	240	150%
Number of GCLKs	1	32	3%

Figure: 5.2 Device Utilization summary fist technique.

In device utilization summary performance of proposed scheme has been observed based on number of slices 486 out of 5472, number of slice flip flops 665 out of 10944. Number of 4 input LUTs 427 out of 10944. number of bonded IOBs 362 out of 240 are utilized.

Further performance of proposed approach has been compared with existing approach based on above parameters.

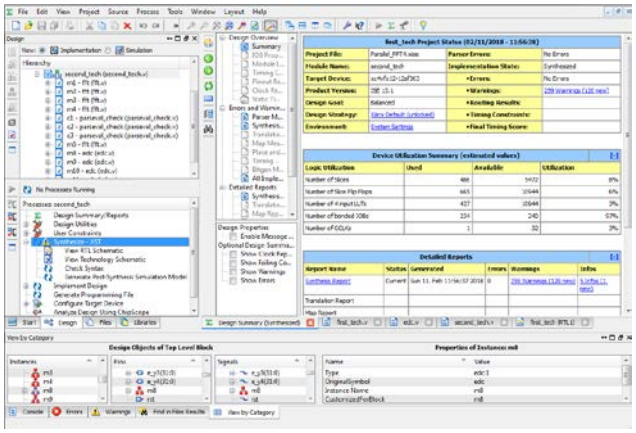

Figure: 5.3 Screen Shots of second technique.

Figure 5.3 shows the simulation screen of second technique based on Vedic Multiplier and error correction code. and figure 5.4 show the device utilization summary second approach.

Device Utilization Summary (estimated values)			[-]
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	486	5472	8%
Number of Slice Flip Flops	665	10944	6%
Number of 4 input LUTs	427	10944	3%
Number of bonded IOBs	234	240	97%
Number of GCLKs	1	32	3%

Figure: 5.4 Device Utilization Second Technique.

a comprehensive analysis of proposed work with existing work has been shown in table 5.1 based on device utilization summary report.

Table: 5.1 Resource comparison with previous design for four parallel FFTs.

Parameters	Previous Design	Proposed Design	
		First Technique	Second Technique
Slices	5468	486	486
Flip-Flops	4148	665	665
LUT-4	10120	427	427

Graphical representation of table 5.1 has been given in figure 5.5. Resource comparison chart with previous design for four parallel FFTs.

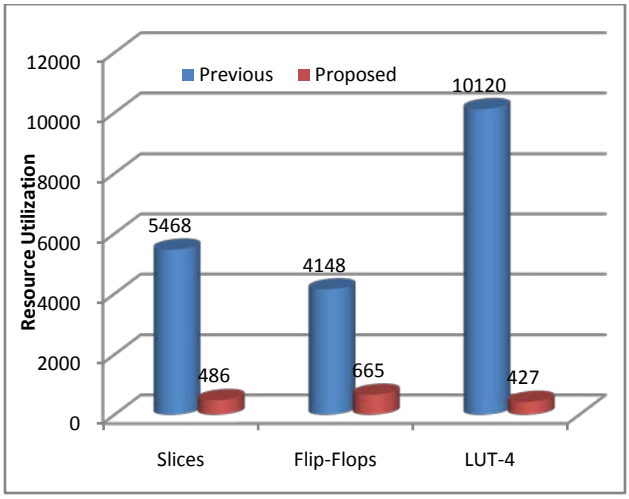

Figure: 5.5 Resource comparison chart with previous design for four parallel FFTs.

V. CONCLUSION AND FUTURE SCOPE

This work presents an efficient fault tolerant parallel FFTs using Vedic Urdhva Triyakbhyam Sutra and ECC to overcome performance issues of modern communication systems. Today Communication Engineering has become the vital field of engineering. Over the past few decades the whole world has been enrolled by communication gadgets whether it is television, radio, atm or any other communication system. Communication basically involves transfers and reception of information from one place to another place or from one point of time to another point of time. In doing so there may be situations that error may be encountered in the channel due to various factors like Electromagnetic Interferences, Cross talk and Bandwidth limitation etc. Reliability and Efficiency are the two important goals which are to be achieved by all the advancement in the field of communication theory. Proposed fault tolerant parallel FFTs provide efficient performance based on device utilization as compared to existing work. The field of communication is highly growing area in research. The goal is to reduce the space and increase the efficiency. The development of highly effective algorithms for the implementation is the area where the significant amount of research work can be done and FFTs can be made more suitable in highly reactive system where the data transfer rate is very fast and error free.

REFERENCES

[1] Z. Gao et al., "Fault Tolerant Parallel FFTs Using Error Correction Codes and Parseval Checks," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 24, no. 2, pp. 769-773, Feb. 2016.

[2] M. I. Momtaz, S. Banerjee and A. Chatterjee, "Probabilistic error detection and correction in switched capacitor circuits using checksum codes," 2017 IEEE 23rd International Symposium on On-Line Testing and Robust System Design (IOLTS), Thessaloniki, 2017, pp. 271-276.

[3] S. K. Mali and M. C. Lakkannavar, "Parallel pipelined FFT architecture for real valued signals using radix-2," 2016 IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), Bangalore, 2016, pp. 1277-1281.

[4] Y. Ji-yang, H. Dan, L. Xin, X. Ke and W. Lu-yuan, "Conflict-free architecture for multi-butterfly parallel processing in-place Radix-r FFT," 2016 IEEE 13th International Conference on Signal Processing (ICSP), Chengdu, 2016, pp. 496-501.

[5] T. T. B. Nguyen and H. Lee, "Shared CSD complex constant multiplier for parallel FFT processors," 2015 International SoC Design Conference (ISOCC), Gyungju, 2015, pp. 27-28.

[6] R. Shirbhate, T. Panse and C. Ralekar, "Design of parallel FFT architecture using Cooley Tukey algorithm," 2015 International Conference on Communications and Signal Processing (ICCSP), Melmaruvathur, 2015, pp. 0574-0578.

[7] J. Park, J. Park and S. Bhunia, "VL-ECC: Variable Data-Length Error Correction Code for Embedded Memory in DSP Applications," in IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 61, no. 2, pp. 120-124, Feb. 2014.

[8] G. Prasad, H. A. Latchman, Y. Lee and W. A. Finamore, "A comparative performance study of LDPC and Turbo codes for realistic PLC channels," 18th IEEE International Symposium on Power Line Communications and Its Applications, Glasgow, 2014, pp. 202-207.