JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue

JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

VLSI Implementation of Low Delay and High Speed Butterfly DIF-FFT For DSP Application

Jeetendra Singh Chauhan

M.E. Scholar, Department of Electronics and Communication Engineering
UIT, RGPV, Bhopal, India

Abstract : A fast Fourier transform (FFT) is a calculation that figures the discrete Fourier transform(DFT) of a grouping, or its backwards (IDFT). Fourier examination changes over a sign from its unique space (frequently time or space) to a portrayal in the frequency area as well as the other way around. In this Paper we have done VLSI implementation of the decimation in frequency-Fast Fourier Transform algorithm. Previous it is designed using the vedic multiplier. The proposed DIF-FFT is design using the booth multiplier. The Xilinx platform is used to simulate and implementation of the work and to calculate and improve various parameters values like area, power, delay and power delay product (PDP) is prime objective of this research work.

IndexTerms **– DIF FFT, Booth Multiplier, PDP, Xilinx, DSP**

I. INTRODUCTION

The DFT is obtained by deteriorating a grouping of values into parts of various frequencies.[1] This activity is helpful in many fields, however figuring it straightforwardly from the definition is frequently too delayed to be in any way down to earth. A FFT quickly processes such transformations by factorizing the DFT lattice into a result of inadequate (for the most part zero) factors. Thus, it figures out how to lessen the intricacy of processing the DFT from O(N^{2}, which emerges in the event that one essentially applies the meaning of DFT, to O(N\log N)}, where N is the information size.

The most popular FFT calculations rely on the factorization of N, however there are FFTs with O(N log N) intricacy for all N, in any event, for prime N. Numerous FFT calculations rest assured e^{-2\pi I/N} is a N-th crude base of solidarity, and hence can be applied to comparable to transforms over any limited field, for example, number-hypothetical transforms. Since the converse DFT is equivalent to the DFT, yet with the contrary sign in the example and a 1/N factor, any FFT calculation can undoubtedly be adjusted for it.

II. DSP ARCHITECTURE

The structure of approximate multiplier is actualized into a custom DSP. The engineering displayed in figure 1.1, it is a specially crafted DSP structure with least control rationale and center significance is given to arithmetic unit planned with an accumulator, barrel shifter, approximate multiplier and an area effective convey select adder.

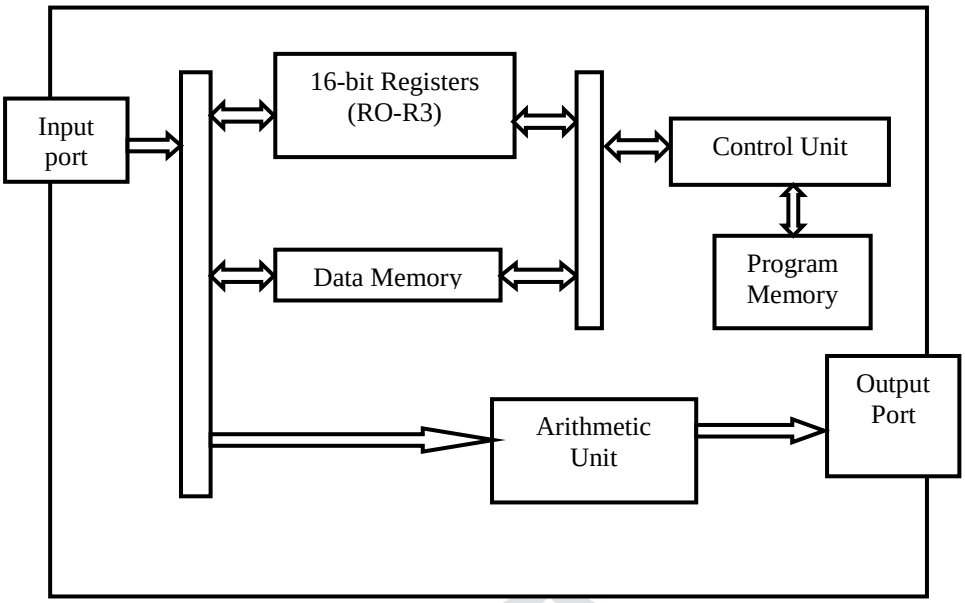

Figure 1: DSP Architecture

III. PROPOSE METHODOLOGY

The goal of proposed work is to VLSI implementation of the decimation in frequency-Fast Fourier Transform algorithm. Previous it is designed using the vedic multiplier. The proposed DIF-FFT is design using the booth multiplier. The Xilinx platform is used to simulate and implementation of the work and to calculate and improve various parameters values like area, power, delay and power delay product (PDP) is prime objective of this research work.

- To VLSI implementation of the Decimation in frequency Fast Fourier Transform (DIF-FFT) algorithm.
- To use booth multiplier for the operation of DIF-FFT.
- To use verilog coding on Xilinx platform for implementation of work
- To calculate various parameters values like area, power, delay and power delay product (PDP).
- To compare proposed implementation with existing DIF-FFT using vedic multiplier.

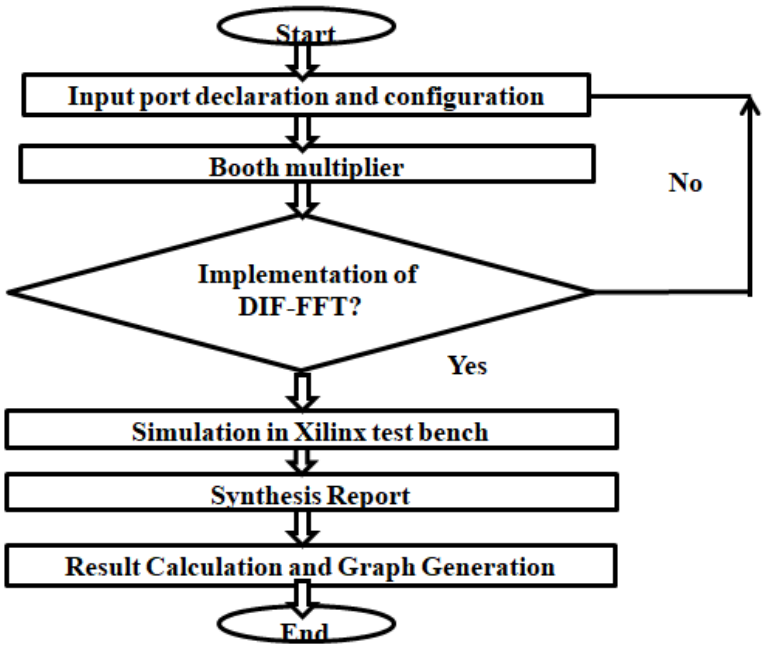

Figure 2: Flow Chart

Radix-2 FFT algorithm is called the decimation-in-frequency algorithm, is obtained by using the divide-and-conquer approach. To derive the algorithm, we begin by splitting the DFT formula into two summations, one of the involves the sum over the first *N*/2 data points and the second sum involves the last N/2 data points. Thus we obtain

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

$$= \sum_{n=0}^{\left(\frac{N}{2}\right)-1} x(n)W_N^{kn} + W_N^{Nk/2} \sum_{N}^{\left(\frac{N}{2}\right)-1} x\left(n + \frac{N}{2}\right) W_N^{kn}$$

Since $W_N^{Nk/2} = (-1)^k$

$$X(k) = \sum_{n=0}^{\left(\frac{N}{2}\right)-1} \left[x(n) + (-1)^k x\left(n + \frac{N}{2}\right)\right] W_N^{kn}$$

Now, let us spilt (decimate) X(k) into the even- and odd-numbered samples. Thus we obtain

$X(2k) = \sum_{n=0}^{\left(\frac{N}{2}\right)-1} \left[x(n) + x\left(n + \frac{N}{2}\right)\right], \quad k = 0,1,\cdots,\frac{N}{2} - 1\, X(2k+1) = \sum_{n=0}^{\left(\frac{N}{2}\right)-1} \left\{\left[x(n) - x\left(n + \frac{N}{2}\right)\right]\right\}, \quad k = 0,1,\cdots,\frac{N}{2} -$ 1Where we have used the fact that $W_N^2 = W_{N/2}$

The computational procedure above can be repeated through decimation of the *N*/2-point DFTs *X*(2*k*) and *X*(2*k*+1). The entire process involves v = $\log_2 N$ stages of decimation, where each stage involves *N*/2 butterflies of the type. Consequently, the computation of the N-point DFT via the decimation-in-frequency FFT requires (*N*/2)$\log_2$N complex multiplications and N$\log_2$N complex additions, just as in the decimation-in-time algorithm.

IV. Simulation Result

Framework level testing may be performed with ISIM or the ModelSim method of reasoning test framework, and such test projects ought to likewise be formed in HDL vernaculars. Test seat tasks may incorporate mirrored input signal waveforms, or screens which watch and affirm the yields of the contraption under test. ModelSim or ISIM may be used to play out the going with sorts of diversions.

Along these lines, one can perform the increase activity utilizing three move and two expansion/subtraction tasks. In this methodology, the closest qualities for An and B as 2n ought to be resolved. At the point when the estimation of An (or B) is equivalent the 3 × 2p−2 (where p is an arbitrary positive whole number bigger than one), it has two closest qualities as 2n with equivalent total contrasts that are 2p and 2p−1. While the two qualities lead to a similar impact on the exactness of the proposed multiplier, selecting the bigger one (with the exception of the instance of p = 2) prompts a littler equipment execution for deciding the closest adjusted worth, and subsequently, it is considered in this venture. It originates from the way that the numbers as 3 × 2p−2 are considered as couldn't care less in both gathering together and down streamlining the procedure, and littler logic articulations might be accomplished on the off chance that they are utilized in the gathering together.

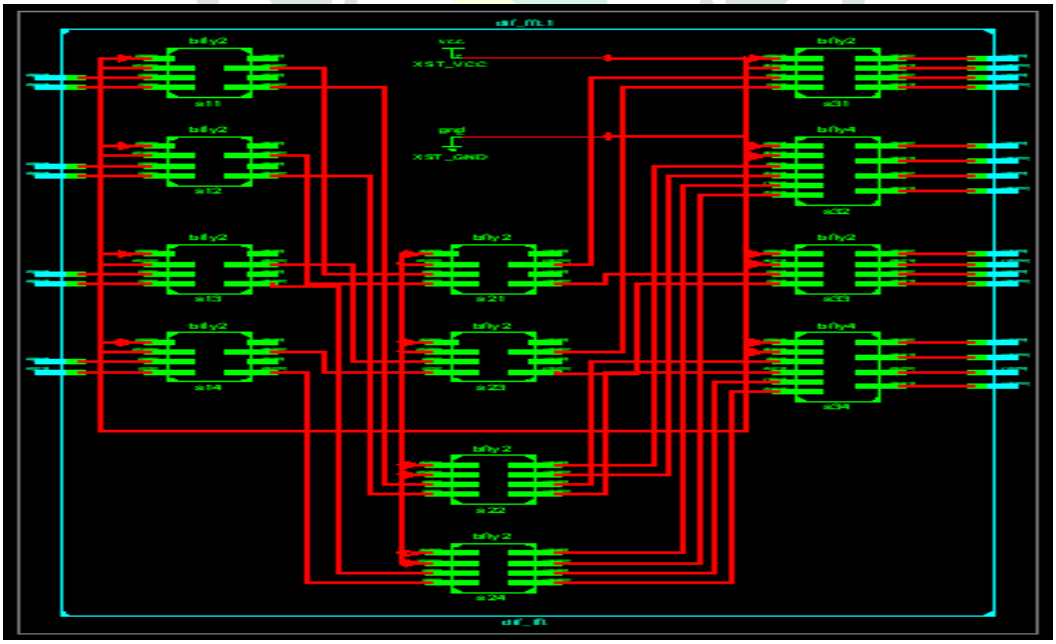

Figure 3: Complete register transfer level (RTL) View

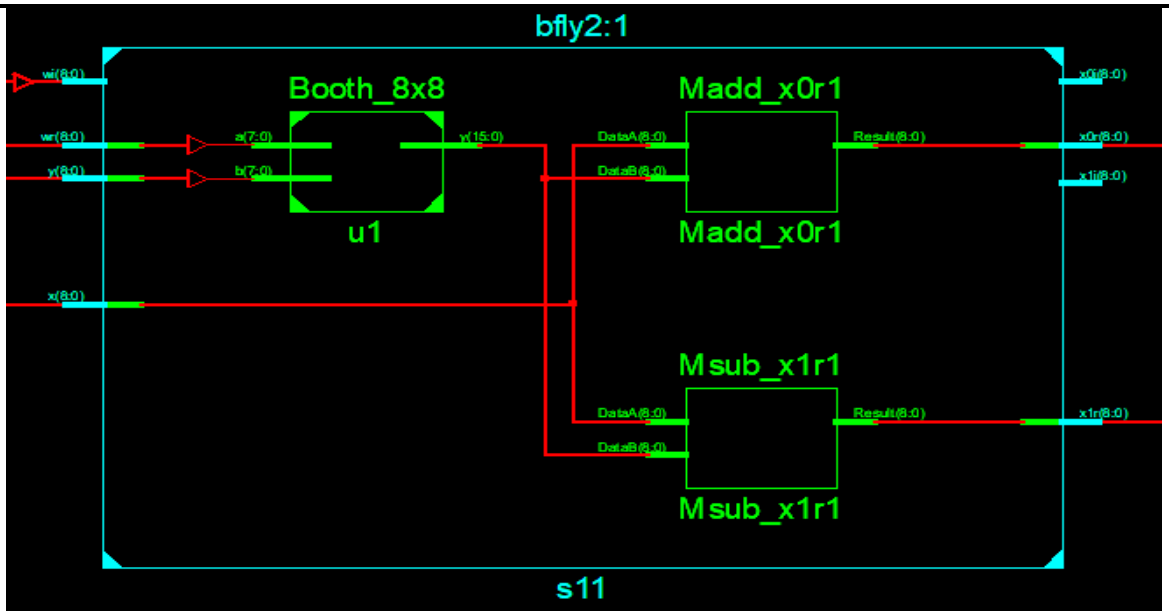

Figure 4: RTL view of the booth multiplier component

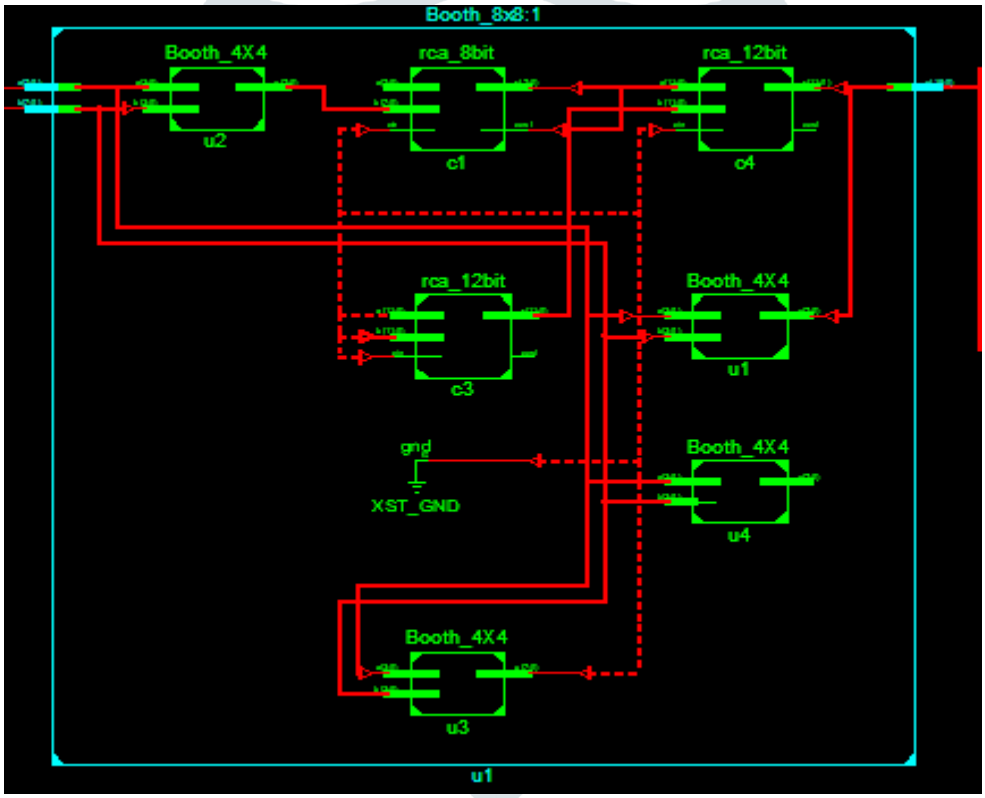

Figure 5: RTL view of the 4X4 booth multiplier

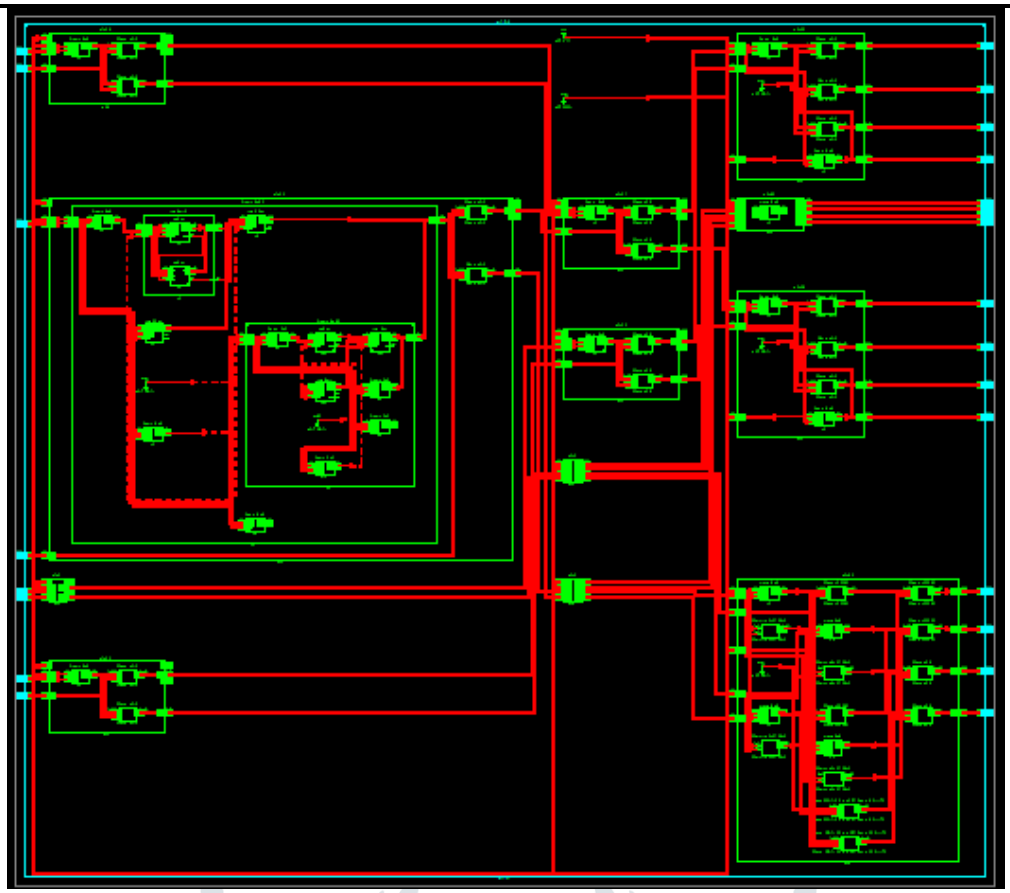

Figure 6 : Complete RTL view of internal circuit

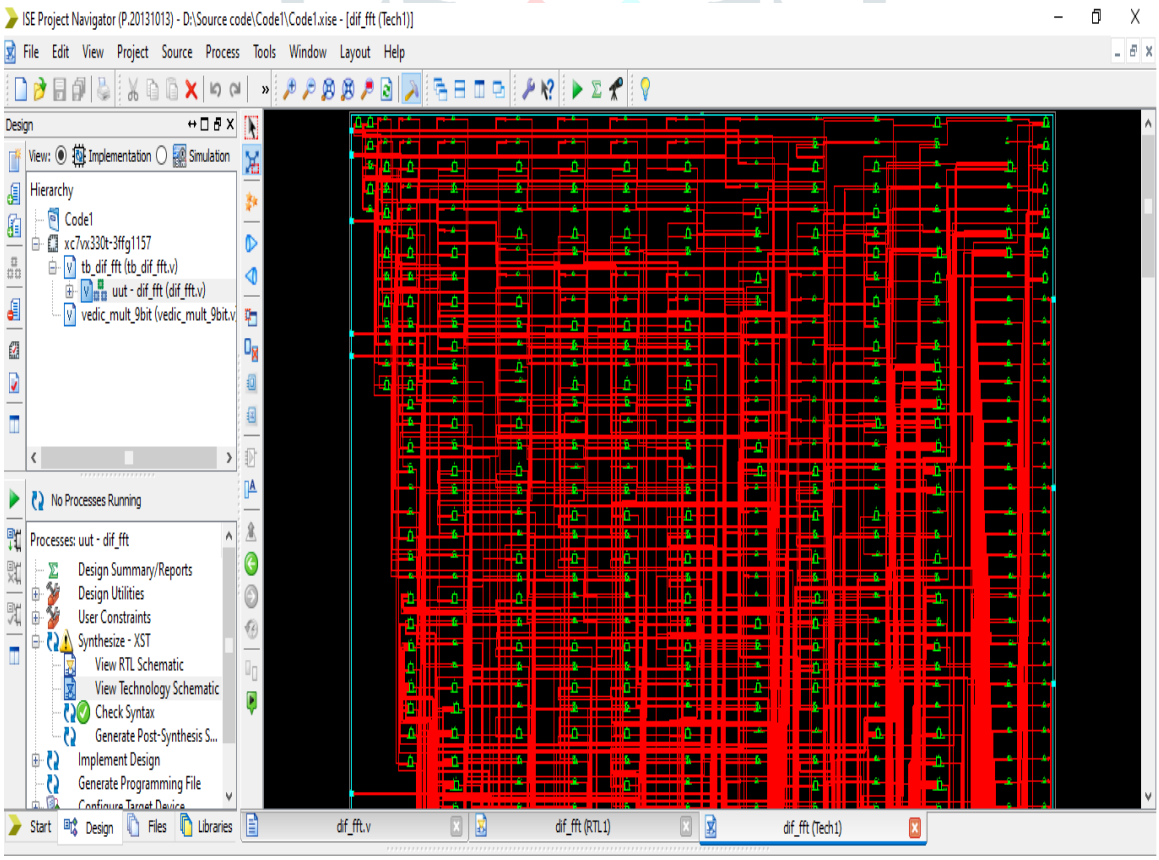

Figure 7: Technological RTL view

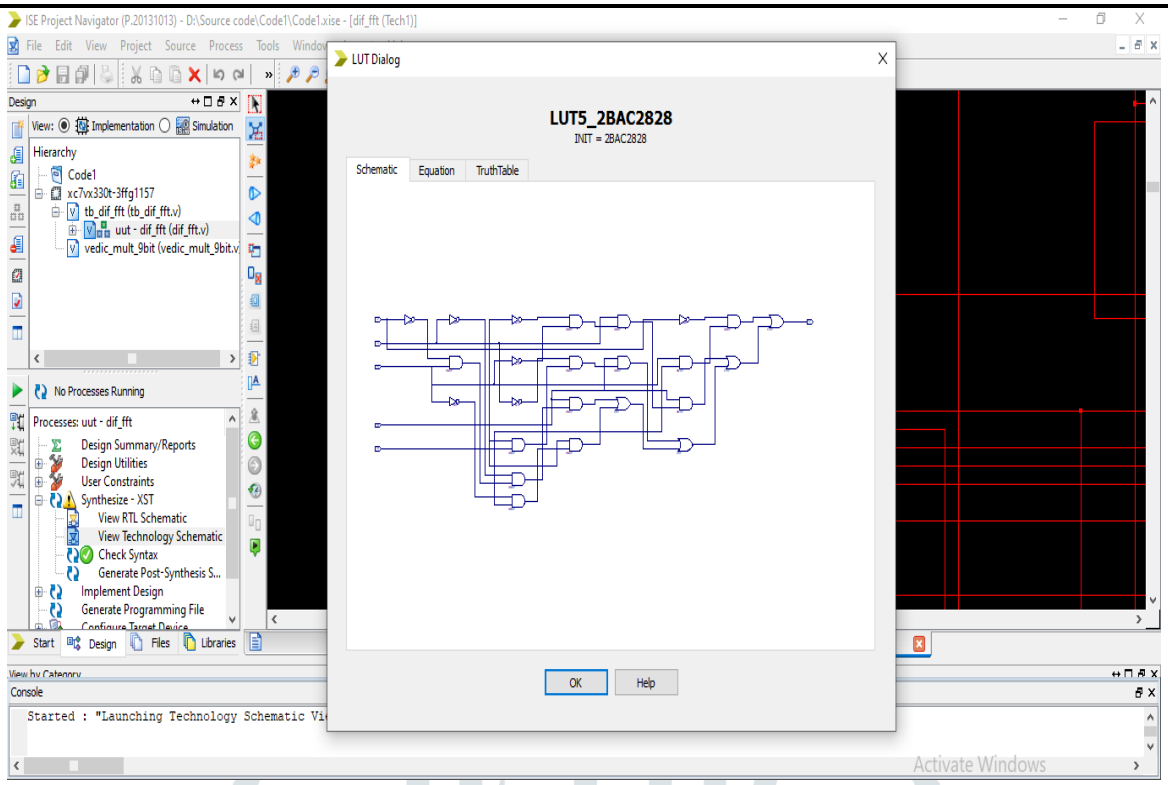

Figure 8: Loot up table 2- schematic

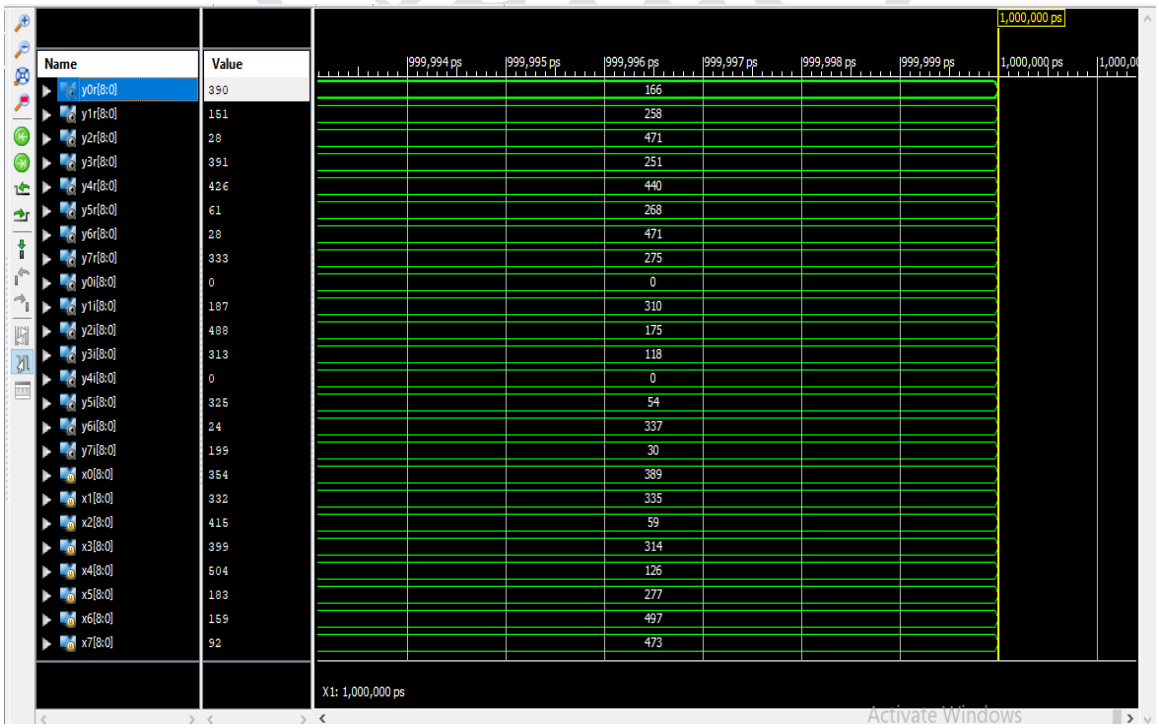

Figure 9: Result in decimal

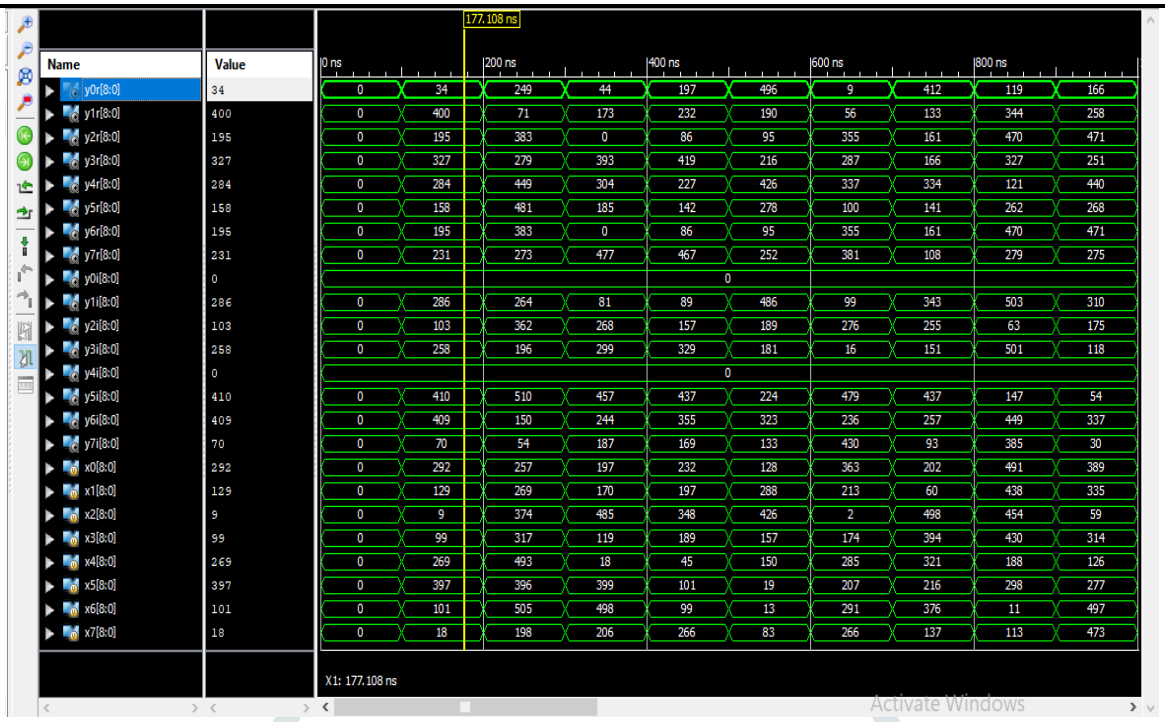

Figure 10: Various input and output in decimal

Table 5.1: Simulation Parameter

Sr No.	**Parameters**	**Proposed Work**
1	Type of FFT	DIF-FFT
2	Area	11.66%
3	Delay	1.718 ns
4	Power	128 microwatt
5	PDP (Power delay product)	219.90
6	Frequency	582 MHz
7	Throughput	4.6 Gbps

Table 5.2: Result Comparison

Sr No.	Parameter	Previous Work [1]	Proposed Work
1	Method	DIF-FFT using Vedic Multiplier	DIF-FFT using booth Multiplier
2	Total Component count	1572	766
3	Delay	4.35 ns	1.718 ns
4	Power	210 micro watt	128 micro watt
5	Frequency	436.94 MHz	582 MHz

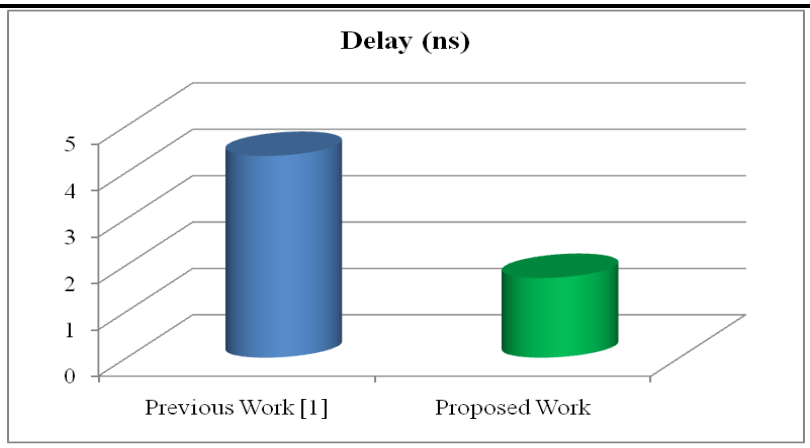

Figure 6: Comparison of delay

V. CONCLUSION

The "Fast Fourier Transform" (FFT) is an important measurement method in the science of audio and acoustics measurement. It converts a signal into individual spectral components and thereby provides frequency information about the signal. As the name implies, the Fast Fourier Transform (FFT) is an algorithm that determines Discrete Fourier Transform of an input significantly faster than computing it directly. In computer science lingo, the FFT reduces the number of computations needed for a problem of size N from O(N^2) to O(NlogN) This dissertation proposed VLSI implementation of the decimation-in-frequency- Fast Fourier Transform using the booth multiplier algorithm. Simulation is performed using the Xilinx ISE 14.7 software. Simulation results show that total component of the proposed work is 766 while the previous algorithm uses 1572 components. The delay is optimized by the proposed work is 1.718 ns while previous it is 4.35 ns. The power is 128 microwatt in the current design while previous it is 210 microwatt

.

REFERENCES

[1] U. Kumar Malviya, "Design and Verification of High-Speed Radix-2 Butterfly FFT Module for DSP Applications," 2020 4th International Conference on Trends in Electronics and Informatics (ICOEI)(48184), 2020, pp. 37-42, doi: 10.1109/ICOEI48184.2020.9143051.

[2]D. Massicotte, M. A. Jaber, C. Neili and M. A. Ouameur, "FPGA Implementation for the Multiplexed and Pipelined Building Blocks of Higher Radix-2k FFT," 2020 IEEE 11th Latin American Symposium on Circuits & Systems (LASCAS), 2020, pp. 1-4, doi: 10.1109/LASCAS45839.2020.9069029.

[3]L. Pang, Y. Huang, C. Wang, C. Yang, Y. Xie and H. Chen, "Design and Implementation of High Performance FFT Processor with Radix-2kAlgorithm," 2019 IEEE International Conference on Signal, Information and Data Processing (ICSIDP), 2019, pp. 1-4, doi: 10.1109/ICSIDP47821.2019.9173055.

[4]A. Li, L. Pang, Y. Zhou, C. Yang, Y. Xie and H. Chen, "Word length Optimization Method for Radix-2k Fixed-Point Pipeline FFT Processors," 2019 IEEE International Conference on Signal, Information and Data Processing (ICSIDP), 2019, pp. 1-4, doi: 10.1109/ICSIDP47821.2019.9173398.

[5]Q. Yuan, H. Zhang, Y. Song, C. Li, X. Liu and Z. Yan, "The Design and Implementation of High Speed Hybrid Radices Reconfigurable FFT Processor," 2019 IEEE 13th International Conference on ASIC (ASICON), 2019, pp. 1-4, doi: 10.1109/ASICON47005.2019.8983511.

[6]V. Harish and K. S., "Comparative Performance Analysis of Karatsuba Vedic Multiplier with Butterfly Unit," 2019 3rd International conference on Electronics, Communication and Aerospace Technology (ICECA), 2019, pp. 234-239, doi: 10.1109/ICECA.2019.8821955.

[7]M. Nazmy, O. Nasr and H. Fahmy, "A Novel Generic Low Latency Hybrid Architecture for Parallel Pipelined Radix-2k Feed Forward FFT," 2019 IEEE International Symposium on Circuits and Systems (ISCAS), 2019, pp. 1-5, doi: 10.1109/ISCAS.2019.8702144.

[8]A. K. Y. Reddy and S. P. Kumar, "Performance Analysis of 8-Point FFT using Approximate Radix-8 Booth Multiplier," 2018 3rd International Conference on Communication and Electronics Systems (ICCES), 2018, pp. 42-45, doi: 10.1109/CESYS.2018.8724107.