INTERNATIONAL JOURNAL FOR TRENDS IN ENGINEERING & TECHNOLOGY
VOLUME 34 ISSUE 1 – OCTOBER 2018 - ISSN: 2349 – 9303

VLSI Implementation of Alternate DIT-FFT Algorithm for OFDM Applications

Sasikala.C[1]
1PG Scholar Dept of Electronics and Communication Engineering,
Dr. Mahalingam College of Engineering and Technology
sasikalacsk777@gmail.com

Dr.K.N.Vijeya Kumar[2]
2Asst Professor Dept of Electronics and Communication Engineering,
Dr. Mahalingam College of Engineering and Technology
vijey.tn@gmail.com

Abstract—This project directs on the development of architecture for the Fast Fourier Transform (FFT), based on Decimation-in-Time (DIT) domain, radix-2 algorithm. The noteworthiness of this project lies in the way that both the time domain and frequency domain values are in natural order. No bit reverse ling is involved. The proposed architecture is implemented in VHDL and its functionality is verified. Further, the architecture is processed using Xilinx System Generator for real time processing applications.

Index Terms—Decimation in Time(DIT), Fast Fourier Transform(FFT), Orthogonal Frequency Division Multiplexing(OFDM), Bit Reversal, Xilinx, Radix-2, Transform.

1 INTRODUCTION

Orthogonal Frequency Division Multiplexing (OFDM) was introduced by change of bell labs in 1996. It is a modulation format that is being used for many of the existing wireless, broadcast and telecommunication standards. It is an integration of both multiplexing and modulation techniques [1]. OFDM is a form of signal modulation that divides a high data rate modulating stream placing them onto a many slowly modulated narrowband closely-spaced subcarriers and in this way is less sensitive to frequency selective fading. OFDM is a multi-carrier modulation technique used in MIMO [2]. Greg Raleigh (1996) invented Multiple Input Multiple Output (MIMO) technology, which multiplies capacity by transmitting different signals over multiple antennas. MIMO-OFDM is the underlying basis for most advanced wireless local area network (wireless LAN)and mobile broadband network standards because it attains the greatest spectral efficiency and therefore, deliver the highest capacity and data throughput. The combination of both MIMO and OFDM is most practical at higher data rates [3]. The block diagram of OFDM is shown in Fig 1.

At the transmitter side, the input bit stream is baseband modulated. The bit stream is parallelized in N different sub streams. Each sub stream will modulate a separate carrier through the IFFT block. The data symbols are back serial converted. A cyclic prefix is inserted in order to eliminate the inter-symbol interference (ISI). At the receiver side, the reverse operation is performed. OFDM transmits a large number of narrowband channels. The frequency range between carriers is carefully chosen in order to make them orthogonal to one another.

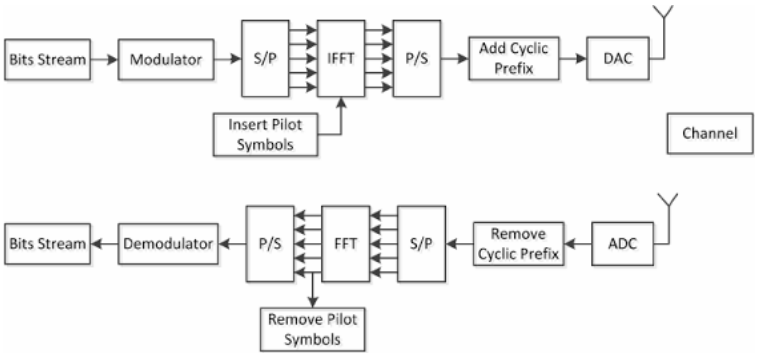

Fig 1: Block Diagram of OFDM

The ability to generate and to demodulate the signal using a software implementation of FFT algorithm is the key of OFDM current popularity [4]. The block diagram shows that the IFFT and FFT are used in transmitter and receiver of the OFDM systems respectively.

Fast Fourier Transform (FFT) is proposed by Cooley and Turkey (1965) [5] .The Fast Fourier Transform is one of the predominant operations in the field of digital signal and image processing applications [6]. It includes signal analysis, sound filtering, data compression, partial differential equations, multiplication of large integers, image filtering etc. It computes the Discrete Fourier

Transform (DFT) of a finite series and requires less number of computations than that of direct evaluation of DFT. DFT is obtained by sampling one period of the Fourier transform at a finite number of frequency points. DFT suffers from the disadvantage that it requires many numbers of complex additions and complex multiplications.

FFT reduces the computations by taking advantage of the fact that the calculation of the coefficients of the DFT can be carried out iteratively [7] [8]. FFT algorithms are based on the fundamental principle of decomposing the computation of discrete fourier transform of a sequence of length N into successively smaller N/2 discrete fourier transforms[9].The FFT is subdivided into two types: Decimation-in-Time FFT and Decimation-in-Frequency FFT. The algorithm is known as radix-2 algorithm which means the number of output points N can be expressed as a power of 2, that is, N= , where M is an integer [10].

The rest of this paper is organized as: Section 2 provides the types of FFT and the proposed architecture is discussed. Section 3 provides the simulation results. Conclusions are given in Section 4.

2 OVERVIEW OF FFT ALGORITHM

2.1 DIT-FFT

In decimation-in-time, the sequence for which we need the DFT is successively divided into smaller sequences and the DFTs of these sub sequences are combined in a certain pattern to obtain the required DFT of the entire sequence [11] as shown in Fig 2. The time domain occurs in bit reversed order and the frequency domain occurs in natural order [12] [13].

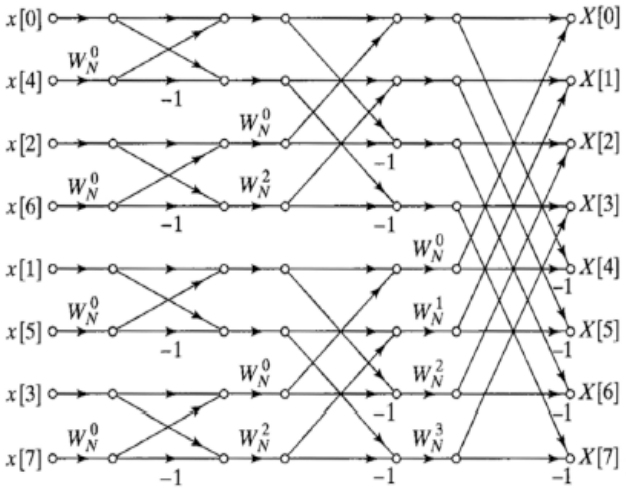

Fig 2: Architecture of DIT-FFT Algorithm

2.2 DIF-FFT

In the decimation-in-frequency approach, the frequency samples of the DFT are decomposed into smaller and smaller sub sequences as shown in Fig 3.The time domain occurs in natural order and the frequency domain occurs in bit reversed order [14] [15].

2.3 Proposed Architecture

The architectures so far described require bit reverse ling process. It is possible to calculate the bit reversed indices in software. But bit reversals become trivial when implemented in hardware. To overcome this disadvantage, we propose an architecture in which bit reversal is not required. Both the time domain and frequency domain values are in natural order. The architecture is shown in Fig 4.

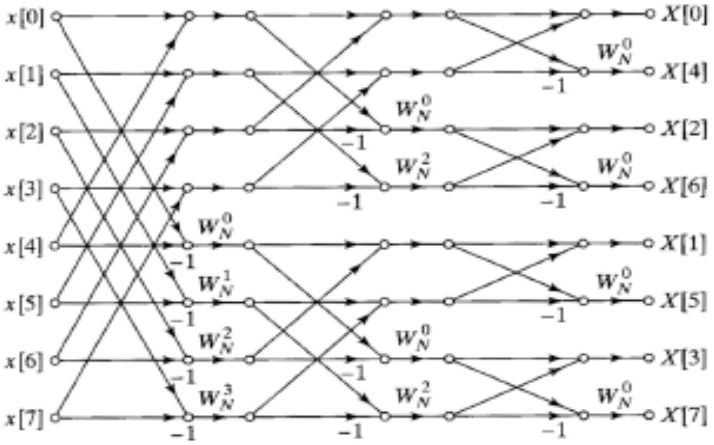

Fig 3: Architecture of DIF-FFT Algorithm

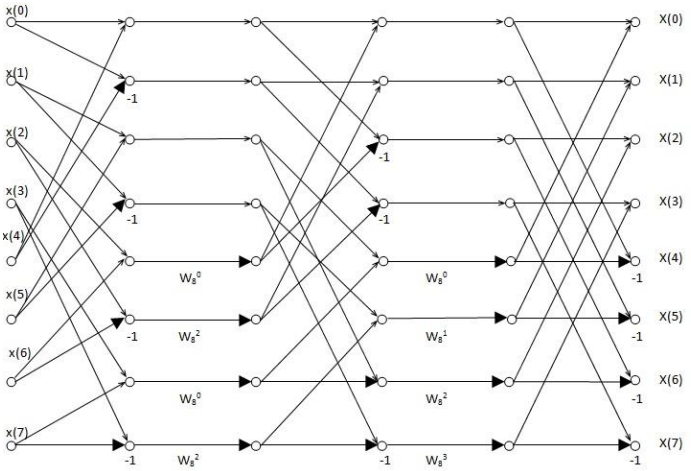

Fig 4: Proposed Architecture for DIT-FFT algorithm

The main functional unit of DIT-FFT architecture is butterfly unit shown in Fig 5. In the context of fast fourier transform algorithms, a butterfly is a portion of the computation that combines the results of smaller DFTs into a larger DFT, or vice versa. The basic computation block in the FFT is butterfly in which the two inputs are combined to give two outputs. The name "butterfly" comes from the shape of the data-flow diagram in the radix-2 case. The figure shows that each butterfly stage consists of multipliers and adders, which are designed using VHDL. There is a presence of complex number in twiddle factor, which shows that complex additions and complex multiplications are involved in the design of FFT.

There are several types of adders [16] available in now-a-days. This proposed architecture makes use of a Ripple Carry Adder (RCA) which is a logic circuit in which carry out of each full adder is the carry in of the succeeding next most significant full adder. It is called a ripple carry adder because each carry bit gets rippled into the next stage.

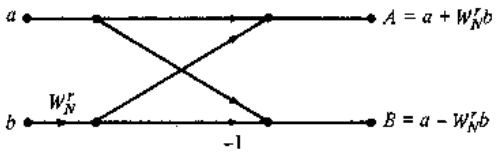

Fig 5: Radix-2 Butterfly Diagram

Multipliers play an important role in today's digital signal processing, image processing and computing architectures[17]. With advances in technology, many researchers have tried and are trying to design multipliers which offer either of the following design targets – high speed, low power consumption, less area, less delay etc [18]. The proposed design makes use of Vedic multiplier which is one of the fastest and low power multiplier over traditional array and booth multipliers which is shown in Fig 6. It deals with a total of sixteen sutras or algorithms for predominately logical operations. Urdhvatiryagbhyam is the most generalized sutra for implementation of vedic multiplier. The beauty of vedic multiplier lies in the fact that they can be used to solve cumbersome mathematical operations orally thereby improving speed.

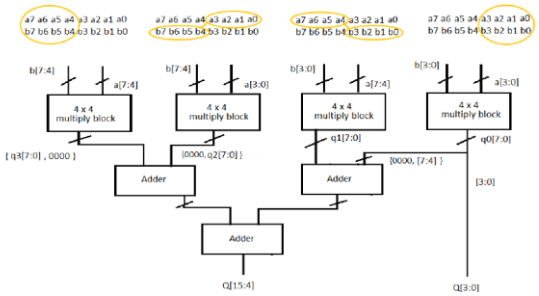

Fig 6: 8 bit Vedic Multiplier

3 SIMULATION RESULTS

System generator is a design tool from Xilinx created to implement real time image processing and signal processing applications on FPGA. System generator supports a black box block that allows RTL to be imported to same link and co-simulated with either modelsim or Xilinx ISE simulator. Hence, simulations are carried out for FFT using Xilinx software. The RTL Schematic view is shown in Fig 7.

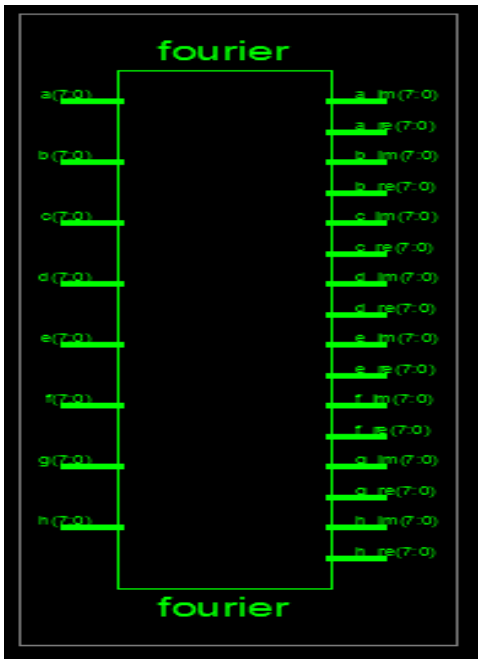

Fig 7: RTL Schematic view

The inputs are decomposed into odd and even numbered sequences and they are joined to form the total transform. The N length sequence is subdivided into N/2 length sequence. Again, N/2 length sequence is subdivided into N/4 length sequence and further, N/4 length sequence is subdivided into N/8 length sequence. For N number of inputs, there must be M=log2 N number of stages. Each stage consists of N/2 butterflies. Twiddle factor is multiplied at each stage. The periodicity and symmetric properties of the twiddle factor are preserved.

The proposed architecture is implemented in VHDL. 8 bit inputs are given. They are processed and outputs are formulated. The real and imaginary parts are carried out separately. The inputs and the outputs are shown in Fig 8 and Fig 9 respectively.

4 CONCLUSION:

The functionality of the architecture is tested and their simulation and synthesis are finished in VHDL using Xilinx ISE tool. The output produced by the proposed architecture is same that of the existing DIT-FFT algorithm. In future, this can be implemented in system generator for real time image processing and signal processing applications. Resource utilization factor can also be estimated.

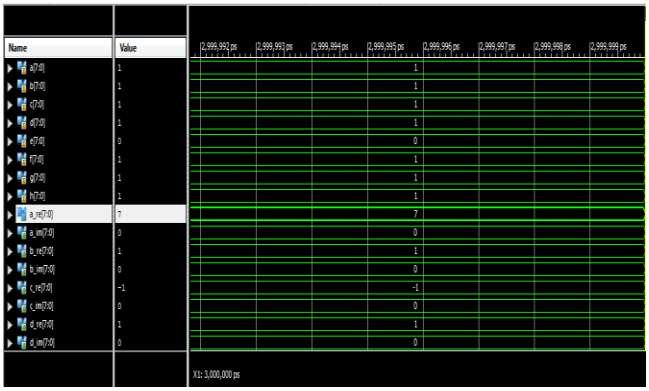

Fig 8: FFT Input

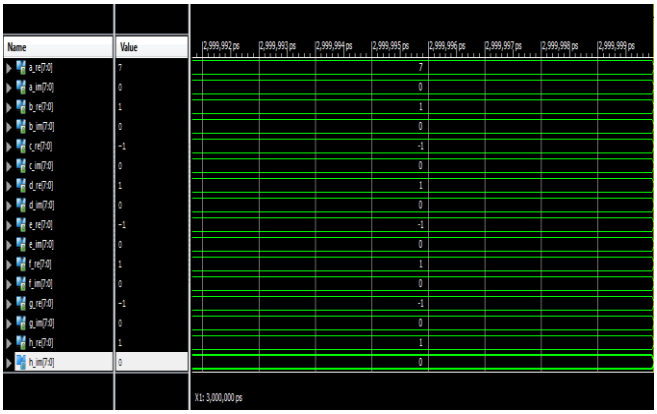

Fig 9: FFT output

REFERENCES:

[1] S. Manoj Prabhakar, K. Kalyani and S. Rajaram, "FPGA Implementation of Efficient FFT Architecture for OFDM Applications", International journal of control theory and applications, ISSN:0974-5572, vol.10, no.6, 2017.

[2] Kausar Ali, Nashrah Fatima, Paresh Rawat, "Review Paper on Radix-2 DIT and DIF Fast Fourier Transform", International Journal of Computer Applications (0975 – 8887), Vol.151, No.6, October 2016.

[3] R. Sai Brunda, M.V.R. Vittal, "Design and Implementation of Variable Length FFT Processor for OFDMA System Using FPGA", International Journal of Engineering Research and Applications, Vol. 3, Issue 2, March -April 2013, pp.1250-1253.

[4] Deepenti Dawande, Geetesh Wagadre, Jitendra Ku. Mishra, "Review Paper on Radix-2 Fast Fourier Transform using Real Value Data", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 6, Issue 4, April 2016.

[5] James W.Cooley, William T.Cochran, David L.Favin, Howard D.Helms, Reg A.Kaenel, "what is the fast fourier transform?" , proceedings of the IEEE, vol.55, no.10, october 1967.

[6] James W.Cooley, Peter A.W. Lewis, Peter D.Welch, "historical notes on the fast fourier transform", proceedings of the IEEE, vol.55, no.10, october 1967.

[7] James W.Cooley, Peter A.W.Lewis, Peter D.Welch, "The fast fourier transform and its applications", IEEE transactions on education, vol.12, no.1, march 1969.

[8] James W. Cooley, Peter A. W. Lewis, and Peter D. Welch, "Application of the Fast Fourier Transform to computation of fourier integrals, fourier series and convolution integrals", IEEE Transactions on audio and electroacoustics, vol.15, no.2, June 1967.

[9] J. W.Cooley, E'. A. W. Lewis, P. D. Welch, "The Finite Fourier Transform", IEEE Transactions on audio and electroacoustics, vol.17, no.2, June 1969.

[10] Mayura Patrikar, Prof.Vaishali Tehre, "Design and Power Measurement of 2 And 8 Point FFT Using Radix-2 Algorithm for FPGA Implementation", IOSR Journal of VLSI and Signal Processing (IOSR-JVSP) Volume 7, Issue 1, Ver. I (Jan. - Feb. 2017), PP 44-48 e-ISSN: 2319 – 4200, p-ISSN No. : 2319 – 4197.

[11] Akshata. U, Gopika.D.K, Hameem shanavas.I, "Hardware Implementation of Decimation in Time FFT", MIT International Journal of Electronics and Communication Engineering, Vol. 3, No. 1, Jan. 2013, pp. 39–42.

[12] Pramod Kumar Meher, Basant Kumar Mohanty, SujitKumar Patel, Soumya Ganguly,and Thambipillai Srikanthan, "EfficientVLSIArchitectureforDecimation-in-Time FastFourierTransformofReal-ValuedData", IEEE transactionsoncircuitsand systems,vol.62,no.12,december2015.

[13] Neha V. Mahajan, Dr. J. S. Chitode, "Simple Computation of DIT FFT", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 5, May 2014.

[14] Kunwar Ved Pratap Singh Parihar, Prof. Monika Kapoor, "Review Paper on Radix-2 Fast Fourier Transform Using Real Value Data", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 6, Issue 6, June 2016.

[15] Ranbeer Rathore, Navneet Kaur, "Comparison Study of DIT and DIF Radix-2 FFT Algorithm", International Journal of Computer Applications (0975 – 8887) Volume 150 – No.7, September 2016.

[16] Jasbir Kaur, Lalit Sood, "Comparison Between Various Types of Adder Topologies", IJCST Vol. 6, Issue 1, Jan - March 2015.

[17] M. Aravind Kumar, O. Ranga Rao, M. Dileep, C V Pradeep Kumar Reddy , K.P. Mani, "Performance Evaluation of Different Multipliers in VLSI using VHDL", International Journal of Advanced Research in Computer and Communication Engineering Vol. 5, Issue 3, March 2016.

[18] Nitesh Kumar Sharma, Mrs. Jigyasa Maru, Dr. M.R. Khan, "VLSI Implementation of a Different Types of Multiplier Unit", International Journal of Scientific & Engineering Research, Volume 7, Issue 4, April-2016.