ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

High Speed Area Efficient Radix-3 and Radix-4 Fast Fourier Transforms

Muneshwar Kumar, Jitendra Kumar Mishra

M. Tech. Scholar, Dept. of Electronics and Communication, Patel Institute of Engineering & Science, Bhopal, India

Associate Professor & HOD, Dept. of Electronics and Communication, Patel Institute of Engineering & Science, Bhopal, India

ABSTRACT: with the advent of new technology in the fields of VLSI and communication, there is also an ever growing demand for high speed processing and low area design. It is also a well-known fact that the chip area and maximum combinational path delay (MCPD) unit forms an integral part of processor design. Due to this regard, high speed and low area architectures become the need of the day. A fast Fourier transform (FFT) is any fast algorithm for computing the DFT. The development of FFT algorithms had a tremendous impact on computational aspects of signal processing and applied science. The decimation-in-time (DIT) fast Fourier transform **(**FFT) very often has advantage over the decimation-in-frequency (DIF) FFT for most real-valued applications, like speech/image/video processing, biomedical signal processing, and time-series analysis, etc., since it does not require any output reordering.

KEYWORDS: FFT, Decimation in Time, Decimation in Frequency, real Value data

I. INTRODUCTION

The discrete Fourier transform (DFT) is an important tool in many branches of science and engineering [1] and has been studied extensively [2]. For many practical applications, it is important to have an implementation of the DFT that is as fast as possible. In the past, speed was the direct consequence of clever algorithms [2] that minimized the number of arithmetic operations. On presentday general-purpose microprocessors, however, the performance of a program is mostly determined by complicated interactions of the code with the processor pipeline, and by the structure of the memory. Designing for performance under these conditions requires an intimate knowledge of the computer architecture. In this paper, we address this problem by means of a novel adaptive approach, where the program itself adapts the computation to the details of the hardware. We developed FFTW, an adaptive, high performance implementation of the Cooley-Tukey fast Fourier transform (FFT) algorithm [3], written in C. We have compared many C and Fortran implementations of the DFT on several machines, and our experiments show that FFTW typically yields significantly better performance than all other publicly available DFT software.
The FFT (Fast Fourier Transform) and its inverse (IFFT) are the key components of OFDM (Orthogonal Frequency Division Multiplexing) systems. Recently, the demand for long length, high-speed and low-power FFT has increased in the OFDM applications. There are three kinds of main design architectures for implementing a FFT processor. One is the single-memory architecture. It has one processing element and one main memory. Hence, it occupies a small area. The second is the dual memory architecture, which has two memories. This architecture has a higher throughput than the single-memory architecture because it can store butterfly outputs and read butterfly inputs at the same time. The fast Fourier transform plays an important role in many digital signal processing (DSP) systems. Recent advances in semiconductor processing technology have enabled the deployment of dedicated FFT processors in applications such as telecommunications, speech and image processing. Specifically, in the OFDM communication systems, FFT and inverse FFT (IFFT) play a very important role. The OFDM technique, due to its effectiveness in overcoming adverse channel effects [1, 2] as well as spectrum utilization, has become widely adopted in wire line and wireless communication standards. The OFDM technique has been adopted in several standards like digital audio broadcasting (DAB) [3], digital video broadcasting terrestrial (DVB-T) [4], asymmetrical digital subscriber line (ADSL) [5] and

very-high-speed digital subscriber line (VDSL) [6]. Therefore, efficient and low-power VLSI implementation of FFT processors is essential for successful deployment of these OFDM-based systems. According to the standards of DAB,

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

DVB-T, ADSL and VDSL, various FFT sizes are required, as shown in Table 1. From this Table, it is clear that variable-length FFT hardware is a crucial module in the low-cost solution of the above communication systems. The Cooley – Tukey N-point FFT algorithm requires O(Nlog N) computations, which is a huge saving over direct computation of the discrete Fourier transform (DFT). However, hardware implementation of the algorithm is both computational intensive, in terms of arithmetic operations, and communication intensive, in terms of data swapping. For real-time processing of FFT, O(log N) arithmetic operations are required per sample cycle. High speed real-time processing can be accomplished in two different ways. In the conventional general-purpose digital signal processor (DSP) approach, the computation is carried out by a single processor driven to a high clock frequency, which is O(log N) times the data sample frequency. In the application specific parallel or pipelined processor approach, the required operations are performed at the clock frequency equivalent to the sample frequency, and this approach usually consumes less power.

II. FAST FOURIER TRANSFORM

Before going further to discuss on the FFT and IFFT design, it is good to explain a bit on the fast Fourier transform and inverse fast Fourier transform operation. The fast Fourier transform (FFT) and inverse fast Fourier transform (IFFT) are derived from the main function which is called Discrete Fourier Transform (DFT). The idea of using FFT/IFFT instead of DFT is that the computation of the function can be made faster where this is the main criteria for implementation in the digital signal processing. In DFT the computation for N-point of the DFT will calculate one by one for each point. While for FFT/IFFT, the computation is done simultaneously and this method saves quite a lot of time. Below is the equation (2.2) showing the DFT and from here the equation is derived to get FFT/IFFT function.

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

X (k) represent the DFT frequency output at the *k*-the spectral point where *k* ranges from 0 to *N-1*. The quantity *N* represents the number of sample points in the DFT data frame.
The quantity x (n) represents the *n*th time sample, where *n* also ranges from 0 to *N-1*. In general equation, x (n) can be real or complex.
The DFT equation can be re-written equation (2) into:

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

The quantity W_N^{nk} is defined as in equation (3)

$$W_N^{nk} = e^{-j2\pi k/N} \quad (3)$$

Here is where the secret lies between DFT and FFT/IFFT where the equation (2.4) function above is called Twiddle Factor. This factor is calculated and put in a table in order to make the computation easier and can run simultaneously. The Twiddle Factor table is depending on the number of point use. During the computation of IFFT, the factor does not to recalculate since it can refer to the Twiddle factor table thus it save time since calculation is done concurrently. Same FFT algorithm can be used to find IFFT function with the changes in certain properties. The changes that implement is by adding a scaling factor of 1/N and replacing twiddle factor value ( ) with the complex conjugate ( ) to the above equation. With these changes, the same FFT flow graph also can be used for the inverse fast Fourier transform.

III. PROPOSED METHOD

The flow chart of the proposed methodology is shown in figure 3. In this paper we are used three techniques i.e. unsigned multiplier, signed multiplier and complex multiplier. In this figure the two signed and unsigned bit multiplier (i.e. multiplier n-bit, multiplicand m-bit) and final output of the multiplier is n+m bits. All the multiplier is design into two parts i.e. partial product generator and multi operand addition.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

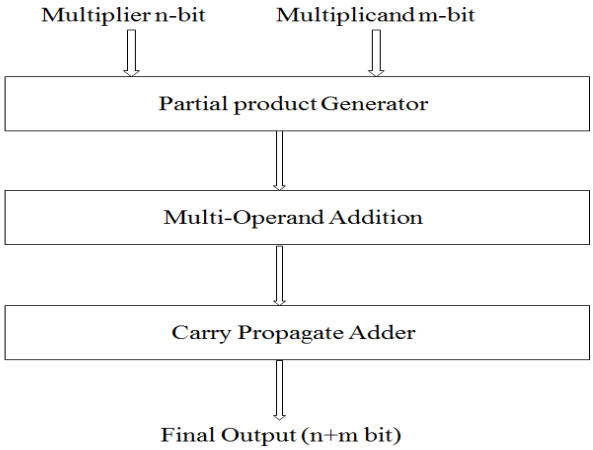
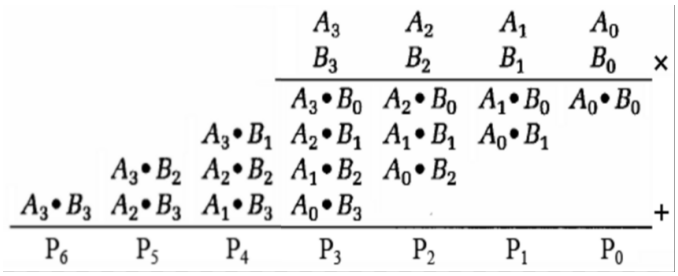
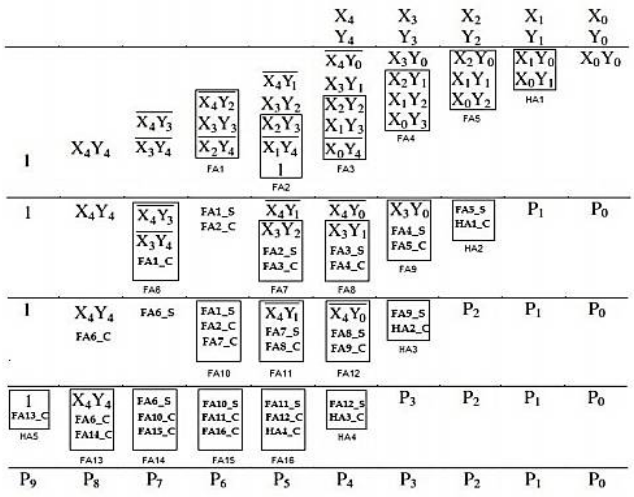


Figure 3: Flow Chart of the proposed Methodology

Array Multiplier:-



Signed Multiplier:-



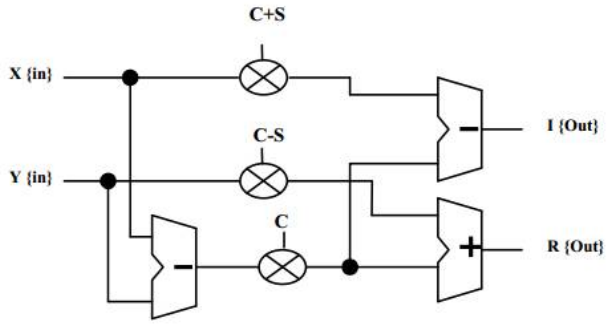
ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

Complex Multiplier:-



The FFT algorithms are classified into two broad categories, namely, the decimation-in-time (DIT) and the decimation-infrequency (DIF) algorithms. The proposed radix-2 DIT algorithm is shown in figure 3.

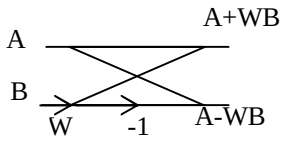


Figure 3: Radix-2 DIT algorithm

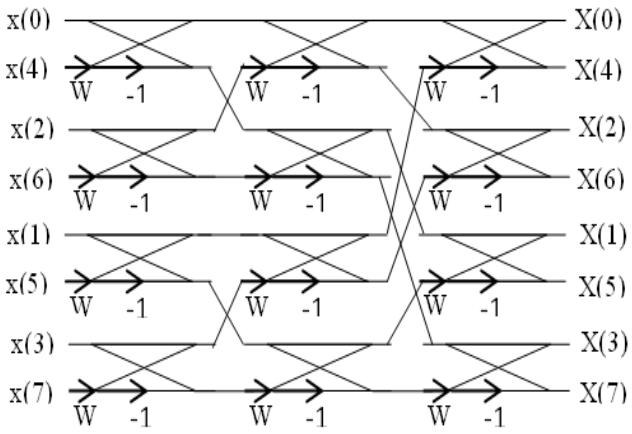


Figure 4: Proposed 8-point Radix-2 DIT Algorithm

DIT butterfly involves a multiplication followed by additions. As shown in Table I the computation time of fixed-point multiplication followed by an addition is less than that of addition followed by a multiplication. The DIT-based FT butterfly thus involves less propagation delay than that of DIF-based RFFT butterfly although both these butterflies involve the same number of multipliers and adders. Therefore, the choice of DIT algorithm to derive FT structure has an advantage over DIF algorithm. In this paper, we present efficient architecture for the DIT radix-2 RFFT algorithm.
This algorithm decomposes a sequence of DFT into four small DFTs of 1/4 lengths in a recursive manner and their outputs are employed to manipulate several other outputs by which the cost of computation will be reduced. The input data is disintegrated into four small sequences of x (4n + i) where n = 0, 1... N/4-1 and i = 0, 1, 2, 3.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

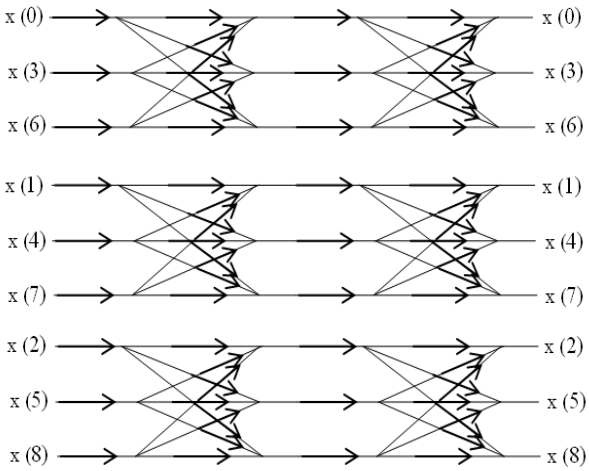


Figure 7: Proposed 9-point Radix-3 DIT Algorithm

This algorithm decomposes a sequence of DFT into four small DFTs of 1/3 lengths in a recursive manner and their outputs are employed to manipulate several other outputs by which the cost of computation will be reduced. The input data is disintegrated into four small sequences of x (3n + i) where n = 0, 1... N/3-1 and i = 0, 1, 2.

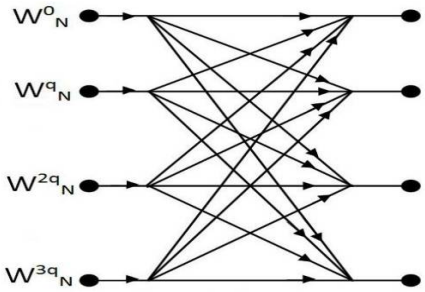


Figure 8: Butterfly element for Radix- 4

IV. SIMULATION RESULT

All the designing and experiment regarding algorithm that we have mentioned in this paper is being developed on Xilinx 14.2i updated version. Xilinx 14.2i has couple of the striking features such as low memory requirement, fast debugging, and low cost. The latest release of ISE™ (Integrated Software Environment) design tool provides the low memory requirement approximate 27 percentage low. ISE 6.1i that provides advanced tools like smart compile technology with better usage of their computing hardware provides faster timing closure and higher quality of results for a better time to designing solution. ISE 14.2i Xilinx tools permits greater flexibility for designs which leverage embedded processors. Also included is the newest release of the chip scope Pro Serial IO Tool kit, providing simplified debugging of high-speed serial IO designs for Virtex-7 FX and Virtex-7 LXT and SXT FPGAs. With the help of this tool we can develop in the area of communication as well as in the area of signal processing and VLSI low power designing.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

Table I: Result for DIT and DIF algorithm for radix-3 algorithm

Parameter	Number of Slice	Number of LUTs	MCPD (nsec)
DIT algorithm for N=9	144	288	15.109
DIF algorithm for N=9	120	240	15.640

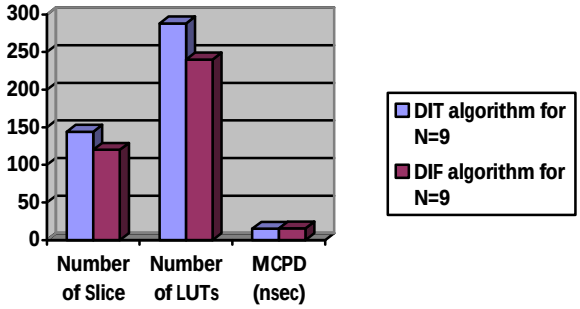


Figure 9: Bar graph of the radix-3 algorithm for N=9

Table II: Result for DIT and DIF algorithm for radix-4 algorithm

Parameter	Number of Slice	Number of LUTs	MCPD (nsec)
DIT algorithm for N=16	320	640	18.526
DIF algorithm for N=16	289	568	19.095

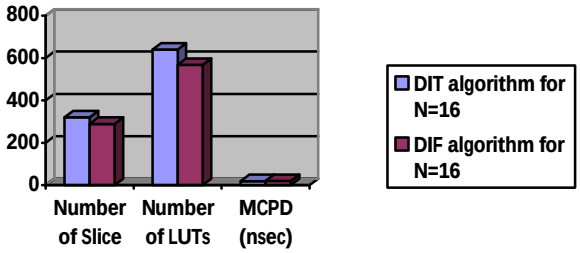


Figure 10: Bar graph of the radix-4 algorithm for N=16

VI. CONCLUSION

The prime objective is to construct a FFT in order to have low power consumption and lesser area. The parameters (i) power consumption (ii) Area occupancy were given due consideration for comparing the proposed circuit with other FFTs. The circuits were simulated using Model-Sim 6.3c and synthesized with Xilinx ISE 14.1.The performance of various 64 point FFT such as Radix-2, Radix-4, split Radix, mixed -radix 4-2, R2MDC and the proposed modified R2MDC were carried out and their performance were analyzed with respect to the number of CLB slices, utilization factor and Power consumption.

REFERENCES

[1] Charles. Roth Jr., "Digital Systems Design using VHDL", Thomson Brooks/Cole, 7th reprint, 2005.
[2] S. S. Kerur, Prakash Narchi, Jayashree C N, Harish M Kittur and Girish V A, "Implementation of Vedic multiplier for Digital Signal Processing", International Conference on VLSI, Communication & Instrumentation (ICVCI) 2011, Proceedings published by International Joural of Computer Applications® (IJCA), pp.1-6.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 10, October 2016

[3] Himanshu Thapaliyal and M.B Srinivas, "VLSI Implementation of RSA Encryption System Using Ancient Indian Vedic Mathematics", Center for VLSI and Embedded System Technologies, International Institute of Information Technology Hyderabad, India.
[4] Jagadguru Swami Sri Bharati Krishna Tirthaji Maharaja, "Vedic Mathematics: Sixteen simple Mathematical Formulae from the Veda", Delhi (2011).
[5] Harpreet Singh Dhillon and Abhijit Mitra, "A Reduced-bit Multiplication Algorithm for Digital Arithmetic", International Journal of Computational and Mathematical Sciences, Febrauary 2008, pp.64-69.
[6] Sumit Vaidya and Depak Dandekar. "Delay-power performance comparison of multipliers in VLSI circuit design". International Journal of Computer Networks & Communications (IJCNC), Vol.2, No.4, July 2010.
[7] Pramod Kumar Mehe, Basant Kumar Mohanty, Sujit Kumar Patel, Soumya Ganguly, and Thambipillai Srikanthan, "Efficient VLSI Architecture for Decimation-in-Time Fast Fourier Transform of Real-Valued Data", IEEE Transactions on Circuits And Systems—I: Regular Papers, Vol. 62, No. 12, December 2015.
[8] M. Ayinala, Y. Lao, and K. K. Parhi, "An in-place FFT architecture for real-valued signals," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 60, no. 10, pp. 652–656, Oct. 2013.
[9] Shashank Mittal, Md. Zafar Ali Khan and M.B. Srinivas, "Area Efficient High Speed Architecture of Bruun's FFT for Software Defined Radio", 1930-529X/07/$25.00 © 2007 IEEE.
[10] B. G. Jo and M. H. Sunwoo, "New continuous-flow mixed-radix (CFMR) FFT processor using novel in-place strategy," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 52, no. 5, pp. 911–919, May 2005.

Muneshwar kumar received the BE degree in electronics and communication from patel college of science and technology, Bhopal India. He is pursuing master of technology degree in digital communication from patel institute of engineering & science, Bhopal India. His research interests are digital signal processing and image processing.

Jitendra Kumar Mishra received the BE degree in electronics and communication and master of technology degree in digital communication. Currently he is an associate professor with head of department electronics and communication from patel institute of engineering & science, Bhopal India. His research interests are digital signal processing, wireless communication and image processing.