International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 9, September 2014)

A New Approach to Design and Implement FFT / IFFT Processor Based on Radix-4^2 Algorithm

Shravankumar Parunandula[1]
Dept. of Electronics and Communication Engg
Aurora's Scientific Technological & Research Academy
JNTU, Hyderabad,Telangana, India.
shravankumar147@gmail.com

Srujan Gaddam[2]
Dept. of Electronics and Communicaton Engg
Aurora's Scientific Technological & Research Academy
JNTU, Hyderabad,Telangana, India.
sujjireddy@gmail.com

Sanathkumar G[3]
Dept. of Low Cost Automation Engg
Central Institute of Tool Design,
Balanagar, Hyderabad
Telangana, India.
Sanathk4@gmail.com

***Abstract*— In this Paper, we propose a new approach to design and implement Fast Fourier Transform(FFT) using Radix-4^2 algorithm, and how the multidimensional index mapping reduces the complexity of FFT computation. Using mathematical analysis on radix-4 DFT(Discrete Fourier Transform) kernel the formal radix-4 butterfly structure is remodeled. This makes the design perspective so simple to implement the mathematical algorithm into the hardware realization model. To further reduce cost of constant multiplier, we reduced the phase factor storage for the entire range of N-point to increase the FFT Computation efficiency.**

***Index Terms*—Fast Fourier Transform, Radix-4^2, Algorithm, multidimensional, index mapping.**

I. INTRODUCTION

The Discrete Fourier Transform (DFT)plays an Important Role in analysis, design and implementation of discrete- time signal processing algorithms and systems. Fast Fourier Transform(FFT)is a Fast and Efficient algorithm to compute DFT of an N-point equally spaced time-domain samples.FFT is an important Digital Signal Processing (DSP) technique to analyze the Phase and Frequency components of a time-domain signal. The Next portable devices such as smart phone, tablet, personal digital assistant demand high trans-mission bandwidth and high communication quality[l].FFT processors have been extensively used in various applications such as communications, image, and bio-medical signal pro-cessing.For example, high performance and low power FFT processing are imperative in Orthogonal Frequency Division Multiplexing (OFDM) based Communication systems, as a programmable base band processor for multiple radio stan-dards, including the wireless LAN standards 802.lla and 802.llb. 802.lla is based on OFDM and uses a 64-point FFT. The WiMAX also baseband is constructed around OFDDM technology requiring high processing throughput. The fixed, IEEE 802.l6e version of WiMAX also needs a 256-point FFT computation. [2]

Many researchers have recently concentrated on designing a reconfigurable FFT processors to achieve a high processing rate and low power consumption on next generation portable devices. He *et al*.[3] has Presented several reliable architectures and the detailed comparisons of the corresponding hardware cost for efflcient pipeline FFT processor. The results of the comparison of these architectures indicate that the Radix-2^2 single path delay feedback (SDF) has the highest butterfly utilization and lowest hardware resource usage in the pipeline FFT/IFFT architecture. Lin *et al*.[4] presented noval Radix-4^2architecture and provided detailed comparisons between Radix-$4^2$and Radix-2^2 SDF architectures. Yang *et al*. [5] presented design methodology for power and area minimization of flexible FFT processors. Also, discussed Radix-2 butterfly based architectures, butterfly structures of Radix - $2/2^2/2^3/2^4$ re-configurable architectures.

We will propose a new approach to compute the FFT, when the number of data points N in the DFT is a power of 4 (i.e., N $=4^v$). We also discuss the usefulness of multidimensional index mapping, as well as how the linear index mapping used to make the computation of the FFT more easy and efficient. The butterfly structure is remolded for easy implementation of hardware using mathematical analysis on DFT kernel of Radix-4^2.And also discussed how a single Processing Element (PE) used to compute the FFT when N $=4^v$

This paper is organized as follows. Section II provides the basics of FFT/[FFT algorithm (II-A), multidimensional index mapping (II-B) and generalized index mapping (II-C). Radix-4^2 FFT/IFFT algorithm is presented in Section III. Section IV demonstrates the Implementation of the Processing Element(PE).

II. FFT ALGORITHM AND ARCHITECTURE

A. *FFT/IFFT ALGORITHMS*

The Discrete Fourier Transform (DFT) of N-point input is defined as [6]

$$X(k)=\sum_{n=0}^{N-1} x[n]W_N^{nk} \tag{1}$$

Inverse Discrete Fourier Transform(IDFT) is given by

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

n=0,l,2,3,...,N-l;

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 9, September 2014)

k=0,l,2,3,...,N-l;

n is the time sequence index of input data ,k is frequency component index of DFT.

Where $W_N = e^{-j2\pi/N}$ is the principle N^{th} root of Unity where is the data sequence of length N. A straight forward computation of the DFT using (1) require $O(N^2)$ operations.[7]

A powerful approach to develop efficient algorithms is pigeonholing the large problem into small groups of problems.One method to do this is to change the original one-dimensional input data into multidimensional input. It is often easy to translate an algorithm using index mapping into an efficient program.

B. *Multidimensional Index Mapping*

Index mapping is a technique to reduce the required arithmetic to compute DFT of a N-point input[8].

We can write a 2-D array on a page of a notebook.Think of the 3-Dimension as the different pages of the notebook.Once we have out of a page (i.e., 2-Dimension array) we do not have limitations. 4-Dimension assumed to be as several notebooks,5-Dimension could be several bookcases full of such notebooks,6-Dimension as several rooms full of such bookcases,and so forth. [9]

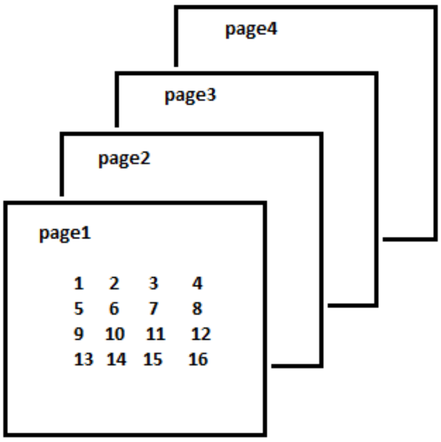

Fig. l. Multidimensional array structure

C. *Index Mapping*

For a N-point sequence,the time index takes on the values

n = 1,2,3, ...,N

where $N=4^v$, so that the index mapping for the N-point of 1-dimensional array to the v - dimensional array is given by

$$n = \frac{N}{4^1}n_1 + \frac{N}{4^2}n_2 + ... + \frac{N}{4^{v-1}}n_{v-1} + \frac{N}{4^v}n_v$$

similarly k is also mapped from l-dimensional array to v-dimensional array as

$$k = \frac{N}{4^v}k_1 + \frac{N}{4^{v-1}}k_2 + ... + \frac{N}{4^2}k_{v-1} + \frac{N}{4^1}k_v$$

where n_1,k_1, n_2,k_2, ….n_v, k_v=0,l,2,3

Therefore (1) can be written as

$$X\left\langle k_1+4k_2+...+4^{v-1}k_v\right\rangle = \sum_{n_v=0}^{3}\sum_{n_{v-1}=0}^{3}\sum_{n_1=0}^{3} x\left(\frac{N}{4^1}n_1 + \frac{N}{4^2}n_2 + ... + \frac{N}{4^v}n_v\right) W_N^{\left(\frac{N}{4^1}n_1+\frac{N}{4^2}n_2+....+\frac{N}{4^v}n_v\right)*\left(k_1+4k_2+...+4^{v-1}k_v\right)} \quad (3)$$

III. RADIX-4^2 FFT / IFFT ALGORITHM

For N=l6 (i.e., $N=4^2$), To perform index mapping on the16-point input, Eq. 3 can be recast as

$$X[k_1+4k_2] = \sum_{n_2=0}^{3}\sum_{n_1=0}^{3} x(4n_1+n_2)W_N^{(4n_1+n_2)*(k_1+4k_2)} \quad (4)$$

here the twiddle factor $W_{16}^{(4n_1+n_2)*(k_1+4k_2)}$ can be decomposed as[10]

$$= W_{16}^{4n_1k_1}.W_{16}^{16n_1k_2}.W_{16}^{n_1k_2}.W_{16}^{4n_2k_2}$$

Where $W_{16}^{16n_1k_2} = 1$. Therefore Eq. 4 can be recast as

$$X[k_1+4k_2] = \sum_{n_2=0}^{3}\left\{\left[\sum_{n_1=0}^{3} x(4n_1+n_2)W_4^{n_1k_1}\right].W_{16}^{n_1k_2}\right\}W_4^{n_2k_2} \quad (5)$$

here $W_4^{n_1k_1}, W_4^{n_2k_2}$ are DFT kernels and both are equal, $W_{16}^{n_1k_2}$ are the twiddle factors, the complex multiplications required are $W_{16}^{k_1}$, $W_{16}^{-k_1}, W_{16}^{2k_1}, W_{16}^{-2k_1}, W_{16}^{3k_1}, W_{16}^{-3k_1}$ in the N-point FFT/IFFT mode.

However, we can achieve Inverse Fast Fourier Transform (IFFT) with a little modification to the FFT algorithm,i.e., sign inversion on the twiddle factors and Normalizing by dividing N. Therefore IFFT formula is given by

$$x[4n_1+n_2]$$

International Journal of Ethics in Engineering & Management Education
Website: www.ijeee.in (ISSN: 2348-4748, Volume 1, Issue 9, September 2014)

$$= \frac{1}{16}\sum_{k_2=0}^{3}\sum_{k_1=0}^{3} X(k_1+4k_2)W_{16}^{-(k_1+4k_2)*(4n_1+n_2)} \tag{6}$$

$$x[4n_1+n_2] = \frac{1}{16}\sum_{k_2=0}^{3}\left\{\left[\sum_{k_1-0}^{3} X(k_1+4k_2)W_4^{-n_1k_1}\right].W_{16}^{-n_1k_2}\right\}W_4^{-n_2k_2} \tag{7}$$

From Eq. 7, it is clear that they are same as Eq. 5 except the negative sign on the twiddle factors and a normalize factor N in computation. So we can compute IFFT using FFT processor with specific changes.

IV. IMPLEMENTATION OF THE PROCESSING ELEMENT

So that the FFT computation takes three steps namely,

1) Previous Computation
The butterfly structure of the first stage of the (4) takes the form of

$$B_4^1 = [x]_{4\times4} * [W_4]_{4\times4} \tag{8}$$

2) Complex Multiplication

$$C_4^1 = [W_4]_{4\times4} \cdot * [B_4^1]_{4\times4} \tag{9}$$

3) Post computation

The butterfly structure of the second stage of the (4) takes the form of

$$B_4^2 = [W_4]_{4\times4} * [C_4^1]_{4\times4} \tag{10}$$

Based on the Eqs. 8, 10 the Operation performed on Previous and Post computation are same. Here the W_4 is the Radix-4^2 kernel. So, we can use a single Processing Element to perform these computations. The input order is given in a special order for the Processing Element to achieve this.

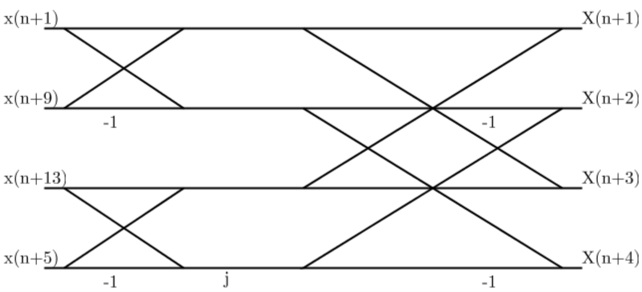

Fig. 2. Modified radix-4^2 butterfly structure

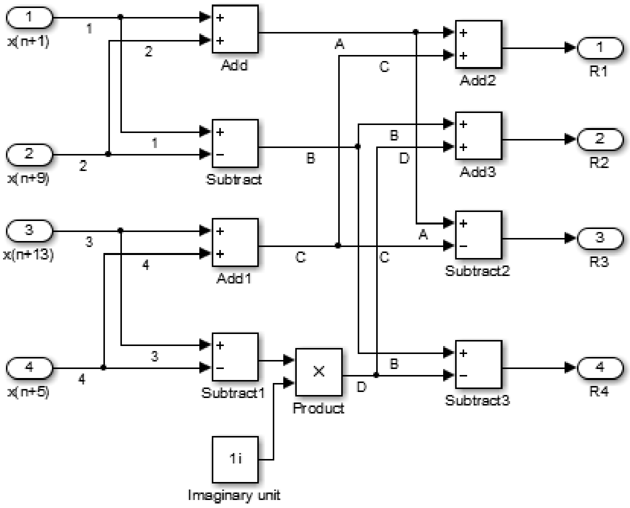

Fig. 3. Block diagram of proposed Processing Element

for n = (0, 1, 2,3), the Processing Element will take the input as

x(1,9,13,5),

x (2, 10, 14,6) ,

x(3, 11, 15,7),

x(4, 12, 16,8).

respectively, and performs the first step.i.e.,Previous computation. Then the complex multiplication takes the place, It is clear that $W_{16}^0 = 1$, therefore the first four outputs of stage one does not need to be multiplied by the Twiddle factors,they pass directly to the butterfly stage II as inputs for post computation, remaining 12 outputs of the stage I undergo the complex multiplication,even though this complex multiplication can be further reduced to 9 by using the same property $W_{16}^0 = 1$ and produce intermediate results for post computation as

R1(1, 2, 3, 4),

R2(5, 6, 7, 8),

R3(9,10,11,12),

R4(13,14,15,16) .

now to compute the final result,these intermediate results are given input to the Processing Element in the following order

R1 (1, 9,13, 5),

R2(2,10,14, 6),
R3(3,11, 15, 7),
R4(4,12,16,8).

for (n = 0, 1, 2,3), the PE computes Rl,R2,R3,R4 respectively and produces the output

X(1,9, 13, 5),

X (2, 10, 14, 6),

X (3, 11, 15, 7),

X (4, 12, 16, 8) .

The Final Output is obtained by applying index mapping on X. i.e., $X\left[k_1 + 4k_2\right]$ for (k_1,k_2 = 0, 1,2, 3), in other words the $\left[X\right]_{4\times4}$ is to be transposed.

V.CONCLUSION

In this paper, we have proposed a new approach to design and implement FFT processor based on a Radix-4^2 algorithm and also given the multidimensional index mapping formula to implement any N-point FFT/IFFT using Radix-4^2 based FFT/IFFT algorithm. Figure 1 represents a 3-Dimensional array, where every page represents a 2-Dimensional array,so that we can perform the proposed algorithm on each page individually,with introducing one constant multiplier. Then conquer all the outputs of the individual pages using Eq. 3 to get the final output.i.e., FFT of N-point where $N=4^v$.

ACKNOWLEDGMENT

The authors would like to express deep gratitude to Prof. Dr.K.V.Rangarao,Dept. of Electronics and Communication Engineering at JNTU Hyderabad, for his valuable and constructive suggestions during the planning and development of this research work.His willingness to give his time so generously has been very much appreciated.Also, the Central Institute of tool design,Hyderabad.

REFERENCES

[1]. Y.-C. Yu and Y.-T. Yu, “Design of a high efficiency reconfigurable pipeline processor on next generation portable device,” in Digital Signal Processing and Signal Processing Education Meeting (DSP/SPE), 2013 IEEE, Aug 2013, pp. 4247.

[2]. E. Tell, O. Seger, and D. Liu, “A converged hardware solution for fft, dct and walsh transform,” in Signal Processing arui Its Applications, 2003.Proceedings. Seventh International Symposium on, vol. 1, July 2003, pp. 609-612 vol.l.

[3]. S. He and M. Torkelson, “A new approach to pipeline fft processor,” in Parallel Processing Symposium, 1996., Proceedings of IPPS ’96, The 10th International, Apr 1996, pp. 766-770.

[4]. C.-T. Lin, Y.-C. Yu, and L.-D. Van, “Cost-effective triple-mode reconfigurable pipeline fft/ifft/2-d dct processor,” Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, vol. 16, no. 8, pp. 1058-1071,Aug 2008.

[5]. C.-H. Yang, T.-H. Yu, and D. Markovic, “Power and area minimization of reconfigurable fft processors: A 3gpp-lte example,” Solid-State Circuits, IEEE Journal ofi vol. 47, no. 3, pp. 757-768, March 2012.

[6]. R. Alan V. Oppenheim, Schafer and J. R. Buck, Discrete-Time Signal Processing (2nd Edition). Pearson Prentice Hall, 2013, ch. Computation of the Discrete Fourier Transform, pp. 655-695

[7]. J. W. Cooley and J. W. Tukey, “An algorithm for the machine calculation of complex fourier series,” Math. Comp.,, vol. 19, pp. 297-301, 1965.

[8]. C. S. Burrus, “Multidimensional index mapping,” May 2012.[Online]. Available: http://cnx.org/contents/3c48e4b5-0786-4d1f-bd30-a0cd860be3ab@ 12

[9]. R. Pratap, Getting Started with MATLAB 7: A Quick Introduction for Scientists and Engineers. Oxford University Press, 2006, ch.Programming in MATLAB:Scripts and Functions, pp. 87-115.

[10]. D. G. M. John G. Proakis, Digital Signal Processing. Pearson Prentice Hall, 2007, 2007, ch. Efficient Computation of the DFT: Fast Fourier Transform Algorithms, pp. 511-536.

Shravankumar Parunandula received B.Tech degree in Electronics and communications Engineering from Jawaharlal Nehru Technological University, Hyderabad. He is currently pursuing the M.Tech degree with the Electronics and Communication Engineering at Aurora’s Scientific Technological & Research Acadamy, Affiliated to JNTU, Hyderabad, Telangana, India.

His research interests are Digital Signal Processing, Communication Systems and Image Processing.

Srujan Gaddam received B.E in Electronics and communications Engineering from Karnatak University, Dharwad, Karnataka and M.Tech in VLSI Design from Bharath University, Chennai, Tamilnadu. Presently, he is working as Associate Professor at Aurora’s Scientific Technological & Research Acadamy, Affiliated to JNTU, Hyderabad, Telangana, India.

His research interests are VLSI system design, Digital signal processing, and Digital Design.He has pulished over 10 journal and conference papers in these areas.

Sanathkumar G received B.Tech degree in Electrical and Electronics Engineering from KL University, Vijayawada, M.Tech degree in Control Systems from BITS Pilani, M.S. in Micro Electronics from IISC, Bangalore. PhD in Medical Applications from IIIT, Hyderabad. He is currently working as Deputy Director of Training at Central Institute of Tool Design, Balanagar, Hyderabad, Telangana, India.

His research interests are VLSI System Design, SoC, low power system design, ASIC Design, and Control Systems. He has published over 20 journal and conference papers in these areas.