

# Variable Length Floating Point FFT Processor Using Radix-2<sup>2</sup> Butterfly Elements

P.Augusta Sophy<sup>#1</sup>, R.Srinivasan<sup>\*2</sup>, J.Raja<sup>§3</sup>, S.Anand Ganesh<sup>#4</sup>

<sup>#</sup> School of Electronics, VIT University, Chennai, India

<sup>\*</sup> Department of Electronics & Communication Engineering, SSN College of Engineering, Chennai, India

<sup>§</sup> Department of Electronics & Communication Engineering, Sai Ram Engineering College, Chennai, India.

<sup>1</sup>augustasophyt.p@vit.ac.in

<sup>2</sup>srinivasanr@ssn.edu.in

<sup>3</sup>rajakanakiraman@gmail.com

<sup>4</sup>anandganesh250@gmail.com

**Abstract**— A mixed radix, floating point FFT processor is designed using radix-2 and radix-2<sup>2</sup> butterfly elements, adapting a pipelined architecture for a variable length of 128/512/2048. The single-path delay feedback (SDF) architecture is employed to exploit the symmetry in signal flow graph of FFT algorithm. Area minimization has been achieved for the reconfigurable FFT processor by using pipelining and higher radix butterfly structures. (radix-2<sup>2</sup>). Then area power trade off is done with parallel mixed radix processing blocks, to achieve better throughput. A reconfigurable architecture has been achieved by bypassing certain processing blocks while keeping the other blocks functional through control mechanism. The proposed design is implemented in 45nm technology and the synthesis results show a silicon area of 4.7mm<sup>2</sup> and a power consumption of 152mw at 50MHz and 208.5mw at 100MHz.

**Keywords**— Fast Fourier transforms (FFT), mixed radix, reconfigurable architecture, pipelining, single path delay feedback

## I. INTRODUCTION

Fast Fourier Transform (FFT) is one of the most important digital signal processing techniques which is used to examine the phase and frequency components of a time domain signal. It helps in spectral analysis, spectral estimation, interpolation, decimation, convolution, correlation, filtering, etc. FFT and IFFT are also used as demodulation and modulation kernels in the OFDM systems and thus FFT computation is a major part of the base band processing.

The use of dedicated FFT processor core is essential in the field of communications, image and biomedical image processing. The advancements in technology develop applications with diverse features. This needs devices to be produced with varied features to meet the requirement. Therefore in this work the reconfigurable/flexible FFT architecture with features like varying sampling rate and FFT sizes has been investigated. An extensive study has been already done on FFT processor architectures by the researchers over the decades. In this work, after perusing the pipelined architectures and also the FFT architectures that use a radix other than the usual radix-2, a variable length FFT processor based on radix-2<sup>2</sup> is proposed and implemented.

The FFT processor architectures basically fall under the following two categories, pipelined architecture [1] and memory based architecture [2]. Different pipelined architecture approaches like Radix-2 Single-path Delay Feedback (R2SDF), Radix-2 Multi-path Delay Commutator (R2MDC), and Radix-4 Single-path Delay Feedback (R4SDF), Radix-4 Single-path Delay Commutator (R4SDC) and Radix-4 Multi-path Delay Commutator (R4MDC) [1] have been proposed and practiced by the researchers. Combination of pipelining and parallel processing, resulting in a parallel-pipelined architecture is proposed in [3], where as an architecture with more than one processing element for a stage, is proposed in [4] so that the through put is increased and the frequency can be accordingly decreased to result in a lower power consumption. In a parallel pipelined architecture there is no increase in the memory size, but only the hardware cost of the additional processing element (PE) is increased.

For a FFT size of  $N = 2^n$ , 'n' pipeline stages are required, and for bigger value of 'N', the long pipeline occupies more silicon area, so a novel pipelined architecture called as locally pipelined architecture using a modified radix-2 single deep delay feedback (R2SD<sup>2</sup>F) architecture is described in [5]. Various other low power implementations of pipelined FFT processor are explained in [6], [7], [8], [9] and [10]. Radix 2<sup>2</sup> was proposed in [11] and [7], while radix-2<sup>3</sup> based pipelined architecture was proposed in [12]. In [8] the FFT system is designed specifically for 3GPP LTE system using mixed radix algorithm, along with parallelism to get minimum power area product.

Most of the prior work has been done with fixed point data representation. In this project floating point data representation which is used for high precision communication systems is adapted. The proposed FFT processor has used single path delay feedback (SDF) architecture, which is one the most common pipelined architecture along with higher radix structures (radix-2<sup>2</sup>) and mixed radix algorithm to take advantage of factorizations.

**II. FAST FOURIER TRANSFORM AND MIXED RADIX ALGORITHMS**

The *N*-point Discrete Fourier Transform (DFT) of an *N*-point input sequence {*x*(*n*)} is given in equation (1).

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

where *k* varies from 0 to *N*-1 and *W<sub>N</sub>* denotes, exp {−*j*2π/*N*}, the *N*th primitive root of unity with its exponent being evaluated with modulo *N* and ‘*n*’ is the time index and the ‘*k*’ is the frequency index. *W<sub>N</sub>* is also referred to as twiddle factor. In general the inputs becomes complex at any intermediate stage, so the computational complexity is very high as *N*<sup>2</sup> complex multiplications and *N* (*N*-1) additions are required. Cooley-Tukey proposed an efficient algorithm [13] for reducing the computations in the calculation of DFT. This is the most commonly used algorithm for FFT. This algorithm is based on divide and conquer policy where the *N* point sequence is recursively partitioned into smaller DFT sequences.

The radix-2 based Cooley-Tukey algorithm is known for its simple structure and modularity. But using the split-radix FFT, which combines radix-2 and radix-4 factorizations, the resultant algorithm, requires reduced number of adders and multipliers compared to radix-2 FFT. The signal flow structure of split-radix is asymmetrical and various radix factorizations based on split radix have been proposed for efficient implementation of FFTs. By the use of higher radix structures the number of complex multiplications can be reduced, thereby power reduction and area reduction are achieved. But the problem with higher radix structures is the increased architecture complexity. Thus for optimal radix design higher radix structures based on radix-2 are used along with radix-2 structures. This is commonly referred to as mixed radix algorithm/radix-2<sup>i</sup> algorithms.

*A. The Concept of mixed-radix decomposition*

Using smaller size DFTs to obtain the *N* point DFT is the underlying concept of Cooley and Tukey [13] and the same technique helps the architecture to be designed as a reconfigurable one. It is also used in developing the mixed radix algorithms.

The *N* point FFT can be written in the two dimensional form if *N* is factorized as *N* = *r*<sub>1</sub>.*r*<sub>2</sub> [8] or in the three dimensional form if *N*= *r*<sub>1</sub>.*r*<sub>2</sub>.*r*<sub>3</sub> [14] or in an *n*-dimensional form.

The two dimensional decomposition of *N* point DFT, with *N* = *r*<sub>1</sub>.*r*<sub>2</sub> can be done with the new indices for *n* and *k* as given in equation (2) and *X*(*k*) gets transformed as given in equations (3) and (4).

$$\begin{aligned} n &= r_2 n_1 + n_2 & n_1, k_1 &= 0 \dots \dots \dots r_1 - 1 \\ k &= k_1 + r_1 k_2 & n_2, k_2 &= 0 \dots \dots \dots r_2 - 1 \end{aligned} \tag{2}$$

$$\text{Now, } X(k) = \sum_{n_2}^{r_2-1} \left( \sum_{n_1}^{r_1-1} x(n_1 n_2) W_N^{(r_2 n_1 + n_2)k} \right) \tag{3}$$

This can be written as

$$X(k) = \sum_{n_2}^{r_2-1} \left( \sum_{n_1}^{r_1-1} x(n_1 n_2) W_{r_1}^{n_1 k_1} \right) W_N^{n_2 k_1} W_{r_2}^{n_2 k_2} \tag{4}$$

This mixed radix decomposition reduces the number of complex multiplications from *N*<sup>2</sup> to *N* (*r*<sub>1</sub>+*r*<sub>2</sub>+1). In a similar method, the decomposition can be done in three or more dimensions.

Many mixed-radix, variable length FFT processors have been proposed and implemented by many researchers and proved for less computational complexity and reduced chip area. Review of FFT architectures is done in [15]. The appropriate combinations of radix-2, radix-2<sup>2</sup> and radix-2/4/8 butterfly processing elements are used to achieve a variable FFT size of 512/1024/2048/4096/8192 as in [16]. A variable length (64 to 8192) FFT processor suitable for multi-mode, Orthogonal Frequency Division Multiplexing (OFDM) applications, proposed in [2] uses radix-2 and radix-2<sup>2</sup> processing elements. References [17], [18], [19], [20] have proposed prime sized FFT using mixed radix, radix 8-2 mixed-radix algorithm is presented in [21], where as radix-2<sup>i</sup> algorithms and architectures are depicted in [22], [10]. Implementation of high-speed 512 point FFT using

radix-2<sup>5</sup> butterfly is presented in [22], where as [8] has proposed a design using radix-2<sup>2</sup>/2<sup>3</sup>/2<sup>4</sup> butterfly elements for 3GPP-LTE.

**B. Radix-2<sup>i</sup> algorithms and Radix-2<sup>2</sup> algorithm**

The radix-2 algorithm recursively divides an N-point sequence into 2-point DFTs. If the DFT can be decomposed as r-point DFTs while r is a multiple of two then r can be represented as  $r = 2^k$ . This is the basic concept behind radix 2<sup>i</sup> algorithm. Thus higher radix structures such as radix-4 and radix-8 can be represented as radix-2<sup>2</sup> and radix-2<sup>3</sup> respectively. This algorithm takes advantage of the fact that most of the internal multiplication can be carried out using constant multipliers (multiplication by ‘1’ and ‘-j’), thereby reducing the requirement for complex multipliers.

In the previous work presented in [8], higher radix structures up to radix-2<sup>4</sup> have been used by taking advantage of the constant multipliers. But in this work structure up to radix-2<sup>2</sup> have been used because of the use of floating point data format.

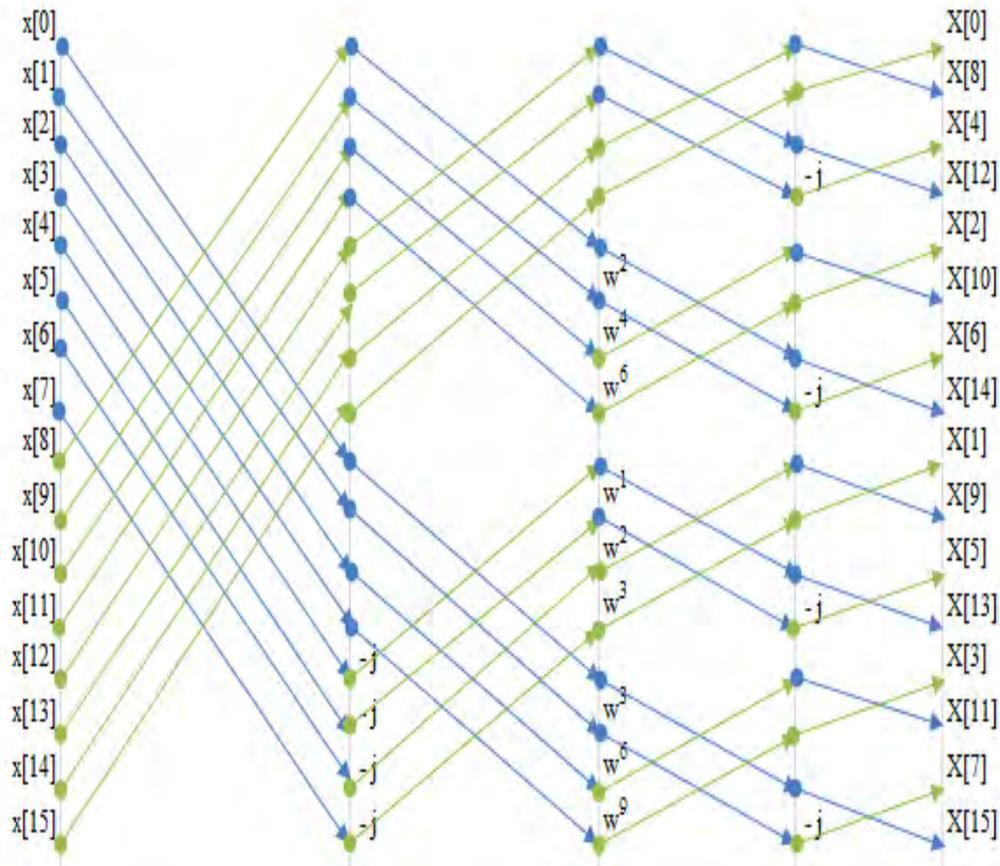


Fig.1. Signal flow graph of the radix-2<sup>2</sup> FFT for N = 16

Radix-2<sup>2</sup> algorithm based FFT processor is implemented in [1], [7], and [10] mainly for the advantage of minimized area and power. The number of non-trivial complex multiplications is reduced in radix-2<sup>2</sup> algorithm compared with radix-2 algorithm.

The efficient utilization of the computational resources and reduced dynamic power of the VLSI implementation of the FFT algorithm depends not only on its computational complexity, but also on the spatial regularity of the data flow graph [1]. In [1] the author proposed the radix-2<sup>2</sup> FFT algorithm and its implementation in 1998. As explained in the references, this algorithm has the computational complexity of radix-4 FFT algorithm, but retains the simplicity and regularity of radix-2 FFT algorithm. The main advantage of this radix-2<sup>2</sup> algorithm is that the number of non-trivial multiplications is reduced. The first stage has got only trivial multiplications by -j which is achieved by swapping the real and imaginary part. In the second stage and from there on, only the alternate stages have the non-trivial multiplications by the twiddle factors. The signal flow graph of the radix-2<sup>2</sup> FFT for N = 16 is shown in Fig. 1.

**III. RADIX-2<sup>2</sup> / RADIX-2 BASED SDF RECONFIGURABLE FFT ARCHITECTURE**

In this work radix-2 and radix-2<sup>2</sup> DIF butterflies are used to implement the reconfigurable FFT processor adapting the single delay feedback (SDF) architecture. This processor can be used for an input sample length of 128/512/2048. The basic building blocks of this architecture explained below.

*A. The single Delay Feedback structure using radix-2<sup>2</sup> butterfly processing element*

The basic processing element in our design is the additive butterfly structure, with two inputs, and two outputs one being the sum and the other the difference of the two input samples, along with an optional ‘-j’ multiplier for the difference output. The ‘-j’ multiplier is present according to the stage of the butterfly. This is illustrated in Fig.2 and Fig.3. In figure 1 the unmarked output points of each stage is the sum of the two samples corresponding to equation (5) and the black dotted outputs are the difference of the two samples as expressed in equation (6).

$$G(k) = x(n) + x(n + N / 2) \tag{5}$$

$$G(k + 2) = x(n) - x(n + N / 2) \tag{6}$$

Two butterfly structures are required to design a radix-2<sup>2</sup> block. Both the butterfly structures are identical in structure except that the second butterfly also contains the logic to implement the twiddle factor multiplication. Thus the non-trivial twiddle multiplication is done external to the common butterfly processing element.

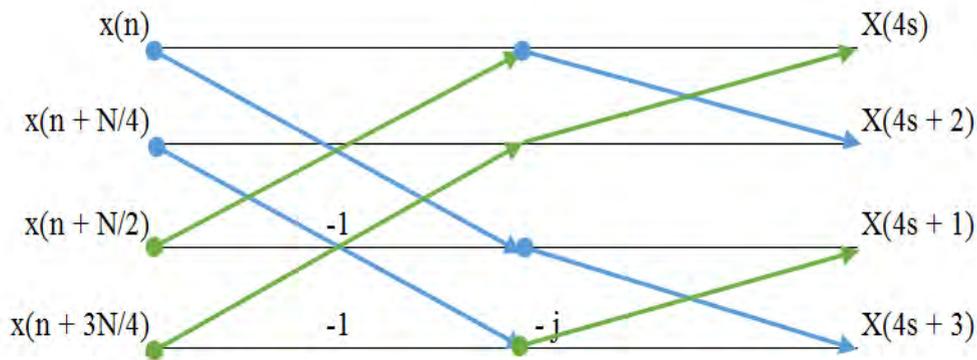


Fig.2. Radix-2<sup>2</sup> butterfly structure

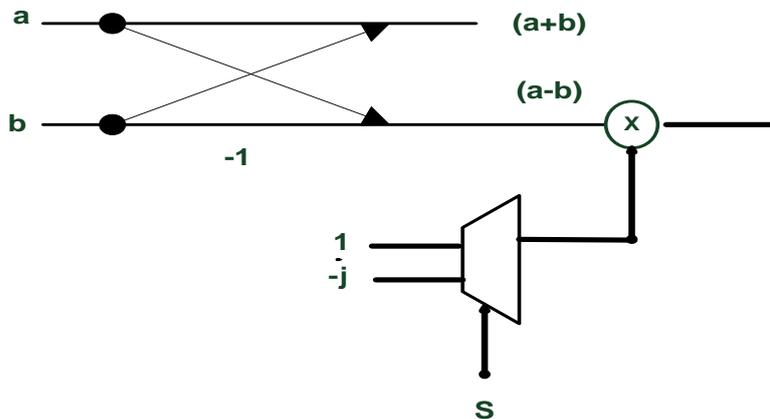


Fig.3. Basic block in the design of radix2/2<sup>2</sup> butterfly

The signal flow in the FFT processor based on radix-2<sup>2</sup> SDF architecture is scheduled in the following order. For an N-point FFT, during the first N/2 clock cycles, the first butterfly module is in idle state until the first N/2 input samples are moved to the shift register. On the next N/2 clock cycles, samples stored in shift register is retrieved one by one and fed to the butterfly processing element along with the current input sample. This data along with the coming input data sample is used to get the output out of the two-point additive butterfly structure.

The second butterfly operates in the same way as the first one except the distance of input sequence in the second butterfly module is just N/4 and it also contains the important twiddle factor multiplication logic along with the butterfly structure. The data thus passes through a complex multiplier working at 75 percentage

utilization efficiency. On Further processing the pattern gets repeated with the distance of the input data decreasing by half at each consecutive stage of butterfly. The first DFT transform output is received after N-1 clock cycles. This output is in bit-reversed form. The next frame of transform can be computed without breaking the input data sequence, due to the pipelined processing of each stage.

**B. Two fifty six point FFT block**

Fig.4 shows the block diagram of a FFT block which can be reconfigured for 16 or 64 or 256 point FFT. The basic butterfly block is same for all stages with the only difference in the associated feedback memory. Thus for 256 point FFT all the blocks will be used. For 64 point FFT, using multiplexer the first two blocks will be bypassed and the remaining blocks will be utilized to get the result. Similarly for 16 point FFT, using multiplexer the first four blocks will be bypassed. The multiplexers are controlled by a stage control block. This block controls multiplexers to bypass the butterfly blocks as per the input stage information received from input line.

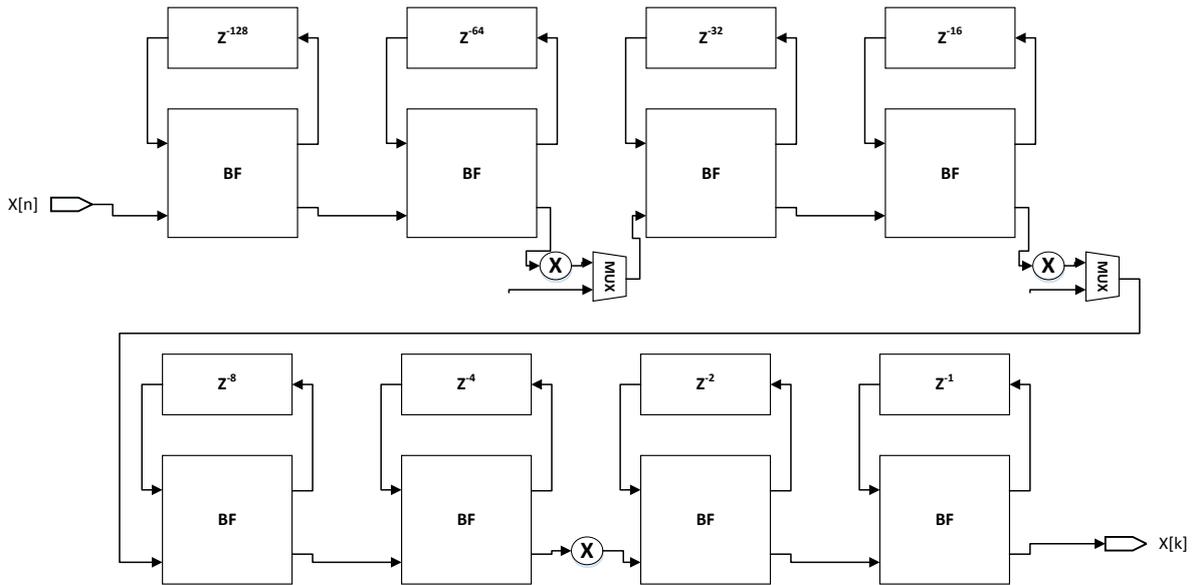


Fig.4. 256-point reconfigurable FFT block

The twiddle factor multiplications in between the butterfly blocks are carried out using complex multipliers. To minimize the hardware cost of the inter-stage complex multipliers, they are implemented by using 3 real multipliers and 5 real adders instead of 4 real multipliers and 2 adders using the sub-expression elimination technique as shown in equation (7).

$$(a + jb)(c + jd) = (c(a - b) + b(c - d)) + j(d(a + b) + b(c - d)) \tag{7}$$

**C. The 2048/512/128 point reconfigurable FFT processor**

Usually the mixed radix processors are designed using the concept of mixed radix FFT decomposition. The decomposition is based on factoring N into a product of smaller positive integers. Many FFT algorithms are often defined in terms of such factorizations of N. The general equation for mixed radix algorithm is obtained when the N point DFT is represented in the two dimensional form by factorizing N = M X L. Thus the N point DFT is obtained using many smaller DFTs of size M and L [8]. Thus the general equation for decomposed DFT is given in equation (8) as specified in [8].

$$X(k) = \sum_{l=0}^{L-1} \left( \sum_{m=0}^{M-1} x(l,m)W_M^{mq}W_N^{lq} \right)W_L^{lp} \tag{8}$$

Where  $k = (Mp + q), p = 0.....L - 1, q = 0.....M - 1$

In this case for N = 2048 , with M = 256 and L = 8,  $k = (256p + q), 0 \leq p < 8, 0 \leq q < 256$ , the DFT decomposition equation is as stated in equation (9)

$$X(k) = \sum_{l=0}^7 \left( \sum_{m=0}^{255} x(l,m)W_M^{mq}W_{2048}^{lq} \right)W_8^{lp} \tag{9}$$

Thus the proposed processor can process up to 2048 samples by using eight 256 point FFT blocks. The block diagram of this reconfigurable processor is shown in Fig.5.

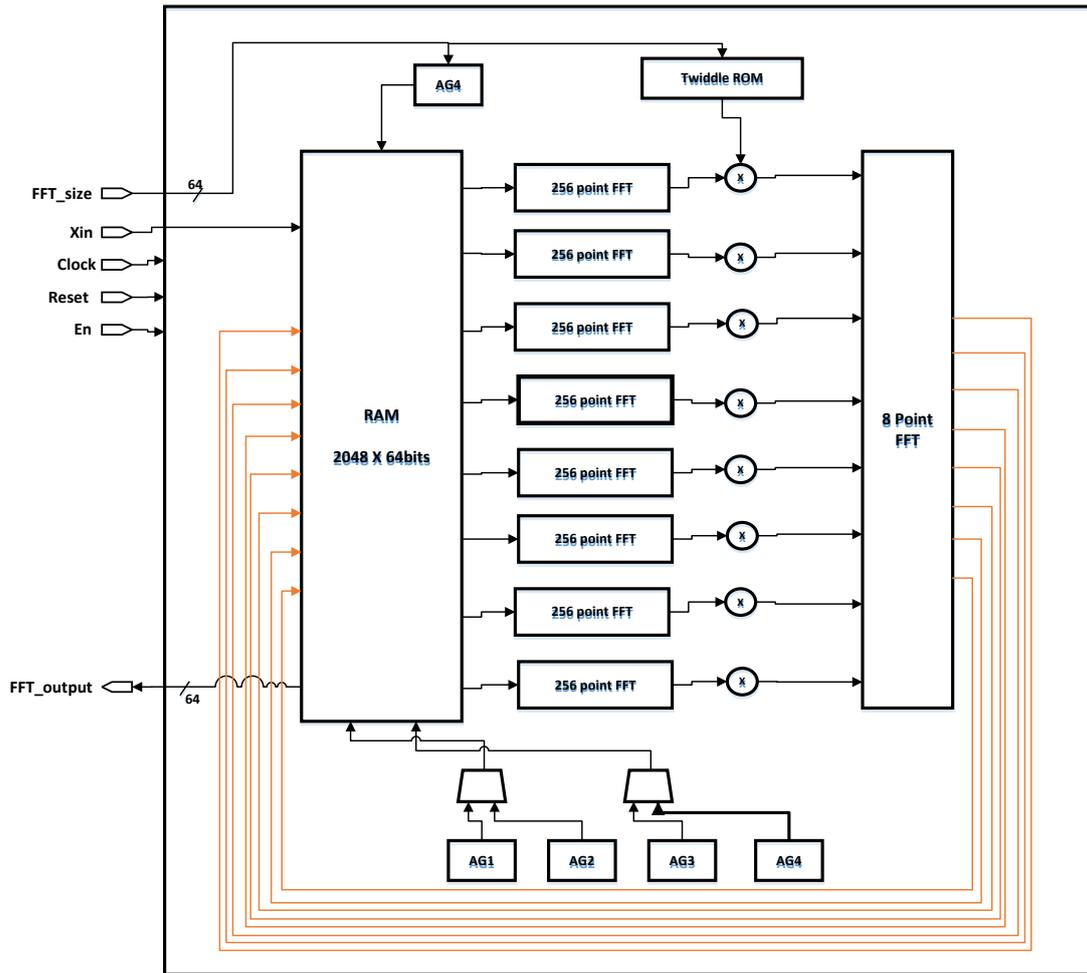


Fig.5. Block Diagram of 2048/512/128 point Reconfigurable Floating Point FFT Processor

The 256 point FFT block, can be configured for 256/64/16 point FFT as explained in the previous section. Thus using those eight blocks, the processor can be configured to do 256 X 8 (2048) or 64 X 8 (512) or 16 X 8 (128) point FFT. The important blocks of this processor are the dual port RAM, eight 256 point reconfigurable FFT Blocks, one 8-point fixed FFT block, five different address generation units, twiddle factor ROMs, inter-stage multipliers and multiplexers. The selection of ROM bank and reconfigurability of 256 point block are programmed by the stage input value.

The first address generation block is used to store the data into the 2K X 64 bits RAM serially. The third address generation unit generates eight different addresses in parallel to fetch data from eight different locations of RAM at a time and these data are fed as an input to the eight different 256 point reconfigurable blocks. The fifth address generation unit is used to generate addresses of twiddle factor ROM and fetch eight twiddles in parallel to give it as input to multiplier block where they are multiplied with the output from the corresponding 256 point FFT blocks. The output of multiplier is given to the 8-point FFT block from which the final output is obtained. The fourth address generation unit is used to write eight simultaneous outputs of the 8 point FFT block into eight separate locations of the 2K RAM. Finally the second address generation unit is used to read the data serially from the RAM and give the data to the output port.

#### IV. SIMULATION AND SYNTHESIS RESULTS

The mixed radix processor, reconfigurable up to 2048 points is implemented using Verilog HDL and simulated in Modelsim for functional verification. For prototyping the design, the major blocks are individually implemented on Xilinx VIRTEX-6 xc6vlx75t-3ff484 device using Xilinx ISE 12.2. Synthesis is done for the 256-point reconfigurable FFT block and also for all the hierarchical blocks individually

The input and output waveform for the mixed radix FFT processor are simulated and shown in Fig.6 and Fig.7. By looking at the output waveform of the FFT processor, it is known that the outputs are available almost at the same time as that of a simple 256 point SDF FFT block. Thus by including parallelism in the design the throughput for 2048 point design is of almost same order of that of 256 point SDF FFT.

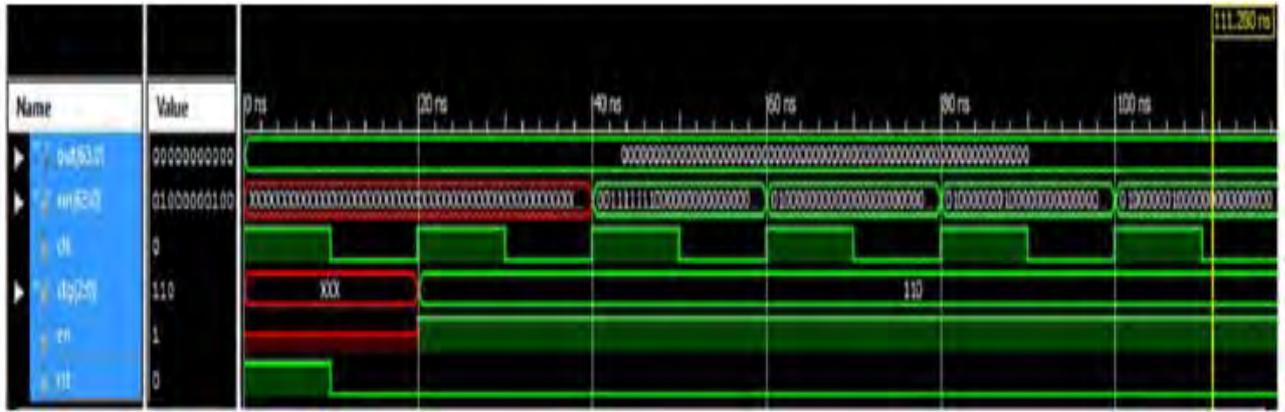


Fig. 6. Input waveform of 2048 point FFT Processor



Fig. 7. Output waveform of 2048 point FFT Processor

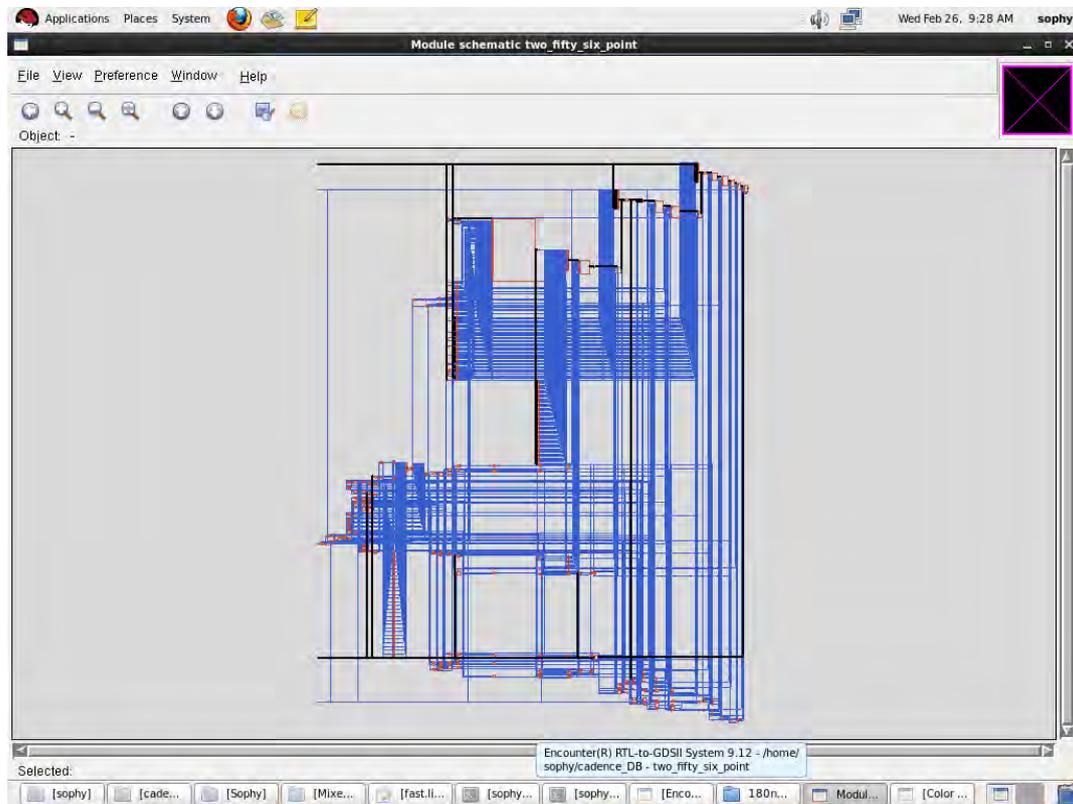


Fig. 8. Schematic of 256 point FFT (Cadence Encounter)

This reconfigurable floating point processor is also synthesized using Cadence RTL Compiler and the layout of the design is done using Cadence Encounter, the Cadence ASIC Design tool. For synthesis 45nm TSMC GPDK technology library is used, along with the defined clock frequencies of 50MHz and 100MHz. The area and power reports from RTL compiler are tabulated in TABLE I.

TABLE I  
IMPLEMENTATION RESULTS OF RADIX-2<sup>2</sup> / RADIX-2  
2048/512/128 POINT FFT PROCESSOR

Technology	TSMC 45nm	TSMC 45nm
Operating Frequency	50MHz	100MHz
Operating Voltage	1.08V	1.08V
No of bits used/sample	64	64
Internal Memory	256 X 64 bits	256 X 64 bits
ROM	512 X 64 bits	512 X 64 bits
Total Area	5.1 mm <sup>2</sup>	4.7 mm <sup>2</sup>
No of cells	2075583	2077314
Total Power	152 mw	208.5mw
Power (2048 point FFT)	112.4 mw	155.5mw
Power (512 point FFT)	104.6 mw	147.5mw
Power (128 point FFT)	136.5 mw	136.5mw

The synthesis results show a very low area of 4.7 mm<sup>2</sup> for a 2K FFT, and a power of 208.5 mw even for a larger data width of 64 bits and a high operating frequency of 100 MHz. At 50 MHz frequency, the power is only 152mw. For an application which can run at a lower frequency, with this proposed FFT processor design, the dynamic power gets further reduced. The design is implemented stage by stage using Cadence Encounter and the layout of 16 to 256 point reconfigurable FFT, is shown in Fig. 9.

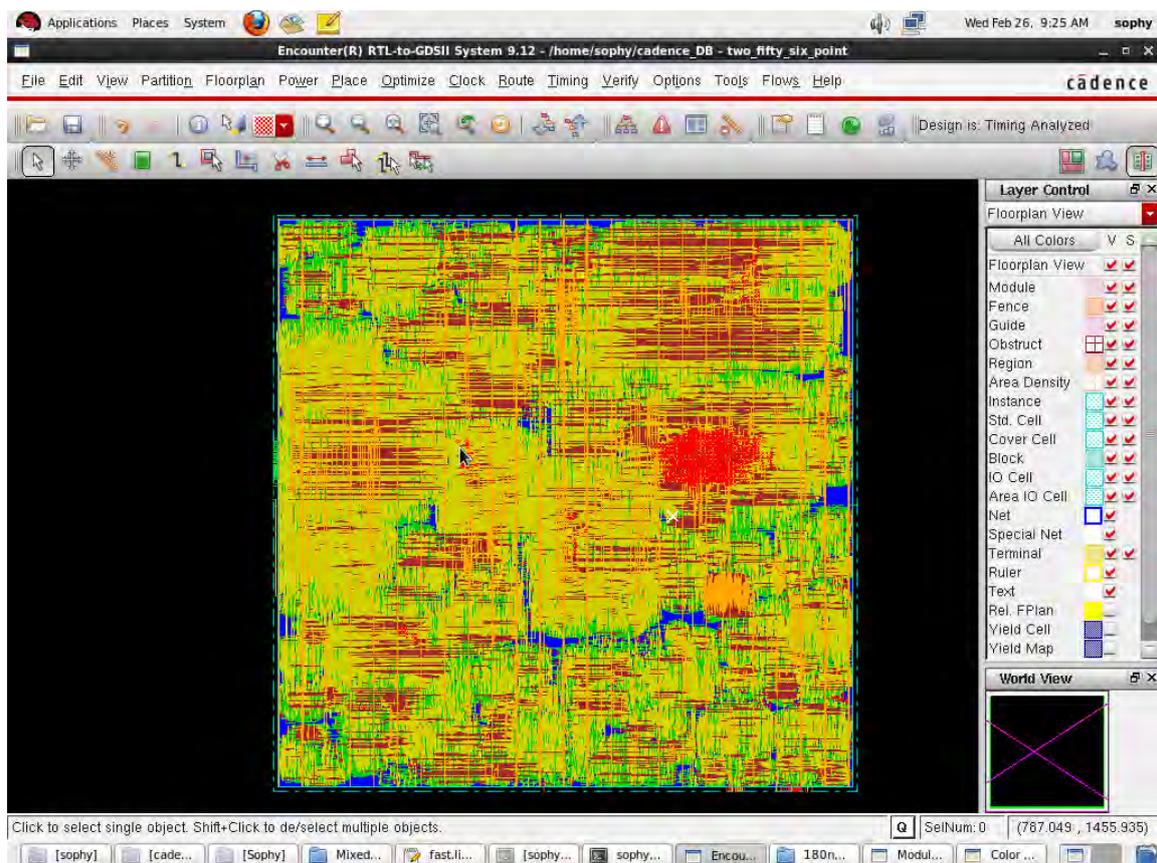


Fig.9. Layout of 16 to 256 point FFT

## V. CONCLUSION

A mixed radix single delay feedback based pipelined FFT architecture has been used for designing a floating point reconfigurable FFT processor (2048/512/128 point). The use of mixed radix (radix-2 and radix  $2^2$ ) has resulted in a system with reduced area and increased throughput as compared to radix-2 SDF architecture. Reconfigurable FFT architecture has been achieved by using the control signals. Area power trade off has been done by introducing parallelism to improve the throughput and to achieve minimum power area product. The results have not been compared with existing work in terms of area and power as the specifications used in the existing work are different from the specifications of this work.

## REFERENCES

- [1] S. He, M. Torkelson, "Design and Implementation of 1024-point FFT Processor", Proc. IEEE Custom Integrated Circuits Conference, 1998.
- [2] Chung-P.Hung, Sau-Gee Chen, Kun-Lung Chen, "Design of An Efficient Variable-Length FFT Processor", Circuits and Systems, 2004. ISCAS'04. Proceeding of the 2004 International Symposium (Volume:2).
- [3] W. Han, T.Arslan, A.T.Erdogan, M.Hasan, "Multiplier-Less Based Parallel-Pipelined FFT Architectures For Wireless Communication Applications", Proc.ICASSP'05.IEEE International Conference on (Volume:5).
- [4] H. Jiang, H. Luo, J. Tian , W. Song, "Design of an efficient FFT Processor for OFDM Systems", Consumer Electronics,IEEE Transactions on (Volume:51, Issue:4), 2005.
- [5] L. Yang, K. Zhang, H. Liu, J. Huang and S. Huang, "An Efficient Locally Pipelined FFT Processor", IEEE Transactions on Circuits and Systems-II:Express Briefs, Vol 53, No.7, JULY 2006
- [6] G. Liu, Q. Feng, "ASIC Design of Low Power Reconfigurable FFT Processor", ASIC,2007.ASICON '07.7<sup>th</sup> Internal Conference, IEEE 2007
- [7] Guoan Bi, Gang Li, "Pipelined structure Based on Radix- $2^2$  FFT Algorithm," ICIEA.2011
- [8] Chia-Hsiang Yang, Tsung-Han Yu, D. Marković, "Power and Area Minimization of Reconfigurable FFT Processors: A 3GPP-LTE Example", IEEE journal of solid-state circuits, vol. 47, no. 3, March 2012.
- [9] P.Chow, Z.G.Vranesic and J.L.Yen,"A pipelined Distributed Arithmetic PFFT processor", IEEE Transactions on Computers, Volume: C-32 , Issue: 12 Publication Year: 1983 , Page(s): 1128 – 1136
- [10] Gin-Der Wu and Yi-Ming Liu, "Radix- $2^2$  Based Low power Reconfigurable FFT Processor", IEEE International Symposium on Industrial Electronics (ISIE 2009), Seoul Olympic Parktel, Seoul, Korea, July 5-8, 2009.
- [11] Chu Yu, Mao-Hsu Yen, Pao-Ann Hsiung, Sao-Jie Chen,"A Low Power 64-point Pipeline FFT/IFFT Processor for OFDM Applications", Consumer Electronics, IEEE Transactions on(Volume:57, Issue: 1),2011.
- [12] Jung-Yeol OH and Myoung-Seob LIM, "New Radix-2 to the 4th Power Pipeline FFT Processor", IEICE Trans. Electron., VOL.E88–C, NO.8 August 2005.
- [13] 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
- [14] G.Zhong, F.Xu, A.N.Willson, "A Power-Scalable Reconfigurable FFT/IFFT IC Based on Multi-Processor Ring", IEEE Journal of Solid State Circuits, Vol 41, No.2, February 2006.
- [15] V.Sarada, T.Vigneswaran, "Reconfigurable FFT Processor – A broader Perspective Survey", International Journal of Engineering and Technology, Vol 5, No 2, May 2013, Pg 949 – 956.
- [16] Shuenn-Shyang Wang , Chien-Sung Li, "An Area – Efficient Design of Variable Length Fast Fourier Transform Processor", Journal of VLSI Signal Processing 2007
- [17] Shen\_Jui Huang, Sau\_Gee Chen, "A Green FFT Processor with 2.5-GS/s for IEEE 802.15.3c (WPANs)", 2010 IEEE
- [18] J. Park, "Design of a radix-8/4/2 FFT Processor for OFDM Systems", CPRE/COMS 583 Project Paper.
- [19] Chen-Fong Hsiao, Y. Chen, Chen-Yi Lee, "A Generalized Mixed-Radix Algorithms for memory Based FFT Processor", IEEE Transactions on Circuits and Systems-II:Express Briefs, Vol 57, No.1, January 2010.
- [20] H. Xiao, A. Pan, Y. Chen, Xiaoyang Zeng, "Low-Cost Reconfigurable VLSI Architecture for Fast Fourier Transform", Consumer Electronics, IEEE Transactions on (Volume:54, Issue: 4).
- [21] M.Mohamed Ismail, M.J.S.Ranggachar,D.V.P. Rao, "VLSI Implementation of OFDM using Efficient Mixed-Radix 8-2 FFT algorithm with bit reversal for the output sequences", International Journal of Electronics and Communication Engineering. ISSN 0974-2166 Volume 5, Number 4(2012), pp. 513-520
- [22] T. Cho, H. Lee, "A High-Speed Low Complexity Modified Radix- $2^5$  FFT Processor for High Rate WPAN Applications", IEEE Transactions on Very Large Scale Integration (VLSI) Systems, Vol.21,No.1, January 2013.