

## Area Efficient Fft/Ifft Processor for Wireless Communication

Rekha Masanam<sup>(1)</sup>, B.Ramarao<sup>(2)</sup>

*M.Tech, Associate Professor,  
Amrita Sai Institute Of Science And Technology, Paritala ,Vijayawada<sup>(1)(2)</sup>*

---

**Abstract:** Fast Fourier Transform (FFT) processing is one of the key procedures in popular Orthogonal Frequency Division Multiplexing (OFDM) communication systems. Structured pipeline architectures, low power consumption, high speed and reduced chip area are the main concerns in this VLSI implementation. In this paper, the efficient implementation of FFT/IFFT processor for OFDM applications is presented. The processor can be used in various OFDM-based communication systems, such as worldwide interoperability for Microwave access (Wi-Max), Digital Audio Broadcasting (DAB), Digital Video Broadcasting-Terrestrial (DVB-T). We adopt single-path delay feedback architecture. To eliminate the Read Only Memories (ROM's) used to store the twiddle factors, this proposed architecture applies a reconfigurable complex multiplier to achieve a ROM-less FFT/IFFT processor and to reduce the truncation error we adopt the fixed width modified booth multiplier. The three Processing Elements (PE's), Delay-Line (DL) buffers are used for computing IFFT. Thus we consume the low power, lower hardware cost, and high efficiency and reduced chip size.

**Keywords:** FFT, IFFT, OFDM, Modified Booth Multiplier.

---

### I. Introduction

The Fast Fourier Transform (FFT) and its Inverse Fast Fourier Transform (IFFT) are essential in the field of digital signal processing (DSP), widely used in communication systems, especially in orthogonal frequency division multiplexing (OFDM) systems, wireless-LAN, ADSL, VDSL systems and Wi-MAX. Apart from the applications, the system demands high speed of operation, low power consumption, reduced truncation error and reduced chip size. By considering these facts, we proposed the ROM-less processor with single path delay feedback (SDF) pipeline architecture and modified booth width multiplier. The SDF pipelined architecture is used for the high-throughput in FFT processor. There are three types of pipeline structures; they are single-path delay feedback (SDF), single-path delay commutator (SDC) and multi-path delay commutator. The advantages of single-path delay feedback (SDF) are This SDF architecture is very simple to implement the different length FFT. The required registers in SDF architecture is less than MDC and SDC architectures. The control unit of SDF architecture is easier. We implement the processor in SDF architecture with radix-4 algorithm. There are various algorithms to implement FFT, such as radix-2, radix-4 and split-radix with arbitrary sizes. Radix-2 algorithm is the simplest one, but its calculation of addition and multiplication is more than radix-4's. Though being more efficient than radix-2, radix-4 only can process 4npoint FFT. The radix-4 FFT equation essentially combines two stages of a radix-2 FFT into one, so that half as many stages are required. Since the radix-4 FFT requires fewer stages and butterflies than the radix 2 FFT, the computations of FFT can be further improved. In order to speed up the FFT computation we increase the radix, for reducing the chip size we use ROM-less architecture and for further low power consumption we implement the reconfigurable complex multiplier and delay line buffers, error compensation is carried out using fixed width modified booth multiplier. Several error compensation methods have been proposed here, the fixed width modified booth multipliers achieve better error performance in terms of absolute error and meansquare error. Using radix-4 algorithm, we propose a 64-point FFT/IFFT processor with ROM less architecture. Finally, this paper is organized as follows. Section II describes the FFT/IFFT processor with radix-4 algorithm. Proposed architecture is discussed in Section III. The performance evaluation and results is then discussed in Section IV. Finally conclusion and future work will see in Section V.

### II. Fft/Ifft Algorithm

In last three decades, various FFT architectures such as single-memory architecture, dual memory architecture, pipelined architecture, array architecture and cache memory architecture have been proposed. In order to improve the power reduction, we propose a radix-4 64-point pipeline FFT/IFFT processor. In order to speed up the FFT computations, more advanced solutions have been proposed using an increase of the radix. The radix-4 FFT algorithm is most popular and has the potential to satisfy the current need. The radix-4 FFT equation essentially combines two stages of a radix-2 FFT into one, so that half as many stages are required. To calculate 16-point FFT, the radix-2 takes  $\log_2 16 = 4$  stages but the radix-4 takes only  $\log_4 16 = 2$  stages. A 16-

point, radix-4 decimation-in-frequency FFT algorithm is shown in Figure 1. Its input is in normal order and its output is in digit-reversed order. It has exactly the same computational complexity as the decimation-in-time radix-4 FFT algorithm.

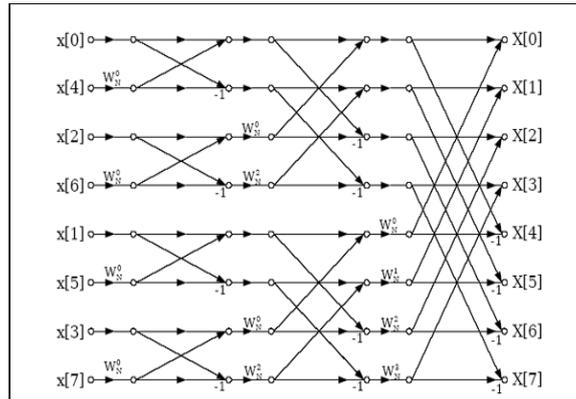


Fig:Flow graph of a 16-point radix-4 FFT algorithm

When the number of data points  $N$  in the DFT is a power of 4, then is more efficient computationally to employ a radix-4 algorithm instead of radix-2 algorithm. A radix-4 decimation in-time FFT algorithm is obtained by splitting the  $N$  point input sequence  $x(n)$  into four sub sequences  $x(4n)$ ,  $x(4n + 1)$ ,  $x(4n + 2)$  and  $x(4n + 3)$ . The radix-4 decimation in frequency butterfly is constructed by merging 4-point DFT with associated coefficients between DFT stages. The four outputs of the radix-4 butterfly namely  $X(4n)$ ,  $X(4n+1)$ ,  $X(4N+2)$  and  $X(4N+3)$  are expressed in terms of its inputs  $x(n)$ ,  $x(n)+N/4$ ,  $x(n)+N/2$  and  $x(n)+3N/4$ .

### III. Proposed Architecture

In this paper, low power techniques are employed for power consumption using reconfigurable complex multiplier. Using radix-4 algorithm, increase the computational speed, further reduce the chip area by three different processing elements (PE's) were proposed in this radix-4 64-point FFT/IFFT processor. Our proposed architecture uses a low complexity reconfigurable complex multiplier instead of ROM tables to generate twiddle factors and fixed width modified booth multiplier to reduce the truncation error.

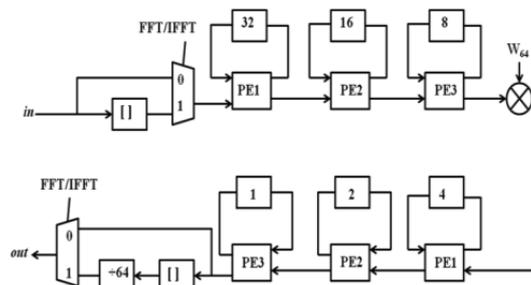


Fig:Proposed radix-4 64-point pipeline FFT/IFFT processor

This proposed architecture consists of three different types of processing elements (PEs), reconfigurable constant complex multiplier, delay line buffers (as shown by a rectangle with a number inside) and some extra processing units for IFFT. Here, the conjugate operation is easy to implement, where we have to generate the 2's complement of the imaginary part of a complex value. This new multiplication structure becomes the key component in reducing the chip area and power consumption. Based on the radix-4 FFT algorithm, the three types of processing elements (PE3, PE2, PE1) proposed in our design. Illustrated in fig.3, fig.4, and fig.5 respectively.

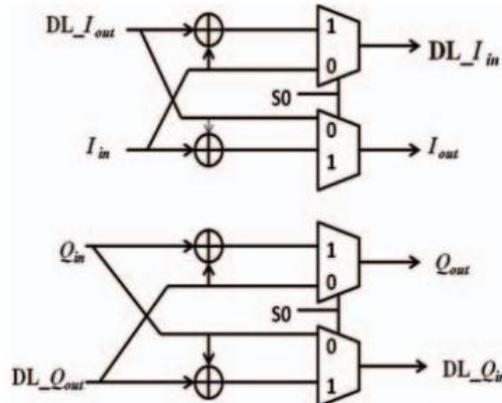


Fig:Circuit diagram of our proposed PE3 stage.

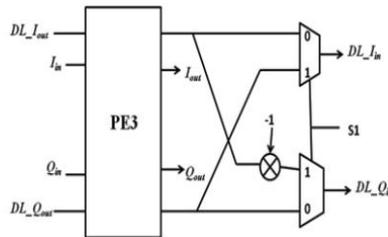


Fig:Circuit diagram of our proposed PE2 stage.

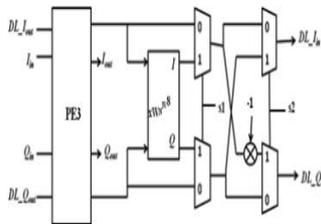


Fig:Circuit diagram of our proposed PE1 stage.

The PE3 stage is used to implement the radix-4 butterfly structure, and serves as sub-modules of the PE2 and PE1 stages. In the PE2 stage, the calculation of multiplication by  $-j$  or  $1$  uses the outcome of the PE3 module. Note that a multiplication by  $-1$  is practically to take the 2's complement of the input value. The PE1 stage is responsible for computing the multiplications by  $-j$ ,  $W_N^{N/8}$ , and  $WN^{3N/8}$ , respectively. Since  $W_N^{N/8} = -j$ ,  $W_N^{N/8}$ , it can be done with a multiplication by  $W_N$  first and then a multiplication by  $-j$ . Hence, our designed hardware utilizes this kind of cascaded calculation and Multiplexers to realize all the calculations in the PE1 stage. The realization of multiplication by  $W_N^{N/8}$  using radix-4 butterfly structure with its both outputs commonly multiplied by  $1/\sqrt{2}$ .

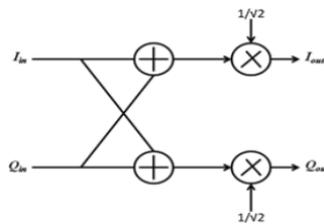


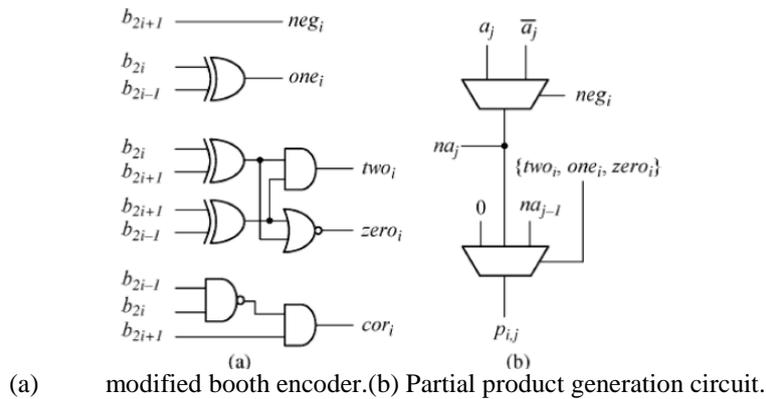
Fig:Circuit diagram of the multiplication by  $W_N^{N/8}$

Here the multiplication operation of modified booth multipliers, the multiplication operation of  $A = a_{n-1}a_{n-2} \dots a_0$  (multiplicand) and  $B = b_{n-1}b_{n-2} \dots b_0$  (multiplier) can be expressed as follows:

$$A = -a_{n-1}2^{n-1} + \sum_{i=0}^{n-2} a_i 2^i, \quad B = -b_{n-1}2^{n-1} + \sum_{i=0}^{n-2} b_i 2^i.$$

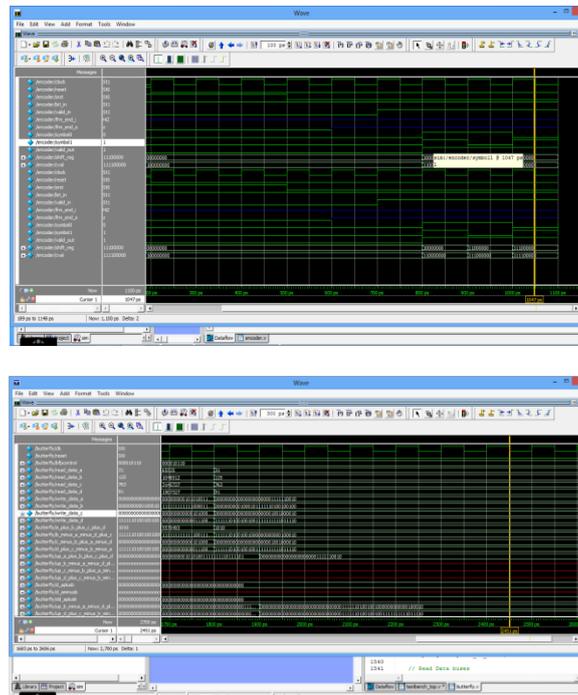
$$B = \sum_{i=0}^{n/2-1} M_i 2^{2i} = \sum_{i=0}^{n/2-1} (-2b_{2i+1} + b_{2i} + b_{2i-1}) 2^{2i}$$

**Booth Encoder Technique:**



Here the partial product matrix of booth multiplication was slightly modified and effective error compensation was derived. The output quality in terms of peak signal to noise ratio (PSNR) for different fixed-width booth multipliers are used in different applications.

**Simulation Result:**



**IV. Conclusion**

A low power pipelined 64-point FFT/IFFT processor for OFDM applications has been described in this paper. Our designed hardware requires about 33.6k gates, and has a working frequency up to 80 MHz synthesized by using 0.18µm CMOS technology. Since our design requires low-cost and consumes low power, as well as reduced SQNR and highly efficient. Hence it can be applied as a powerful FFT/IFFT processor in wireless communication systems.

### **References**

- [1]. Chu yu, Mao-Hsu Yen “ A Low power 64-point
- [2]. FFT/IFFT Processor for OFDM Applications” IEEE Transactions on Consumer Electronics, Vol. 57, Feb 2011
- [3]. N.Kirubanandasarathy, Dr.K.Karthikeyan,” VLSI Design of Mixed Radix FFT Processor for MIMOOFDM in Wireless Communication”, 2011 IEEE Proceedings.
- [4]. FahadQureshi and Oscar Gustafson,”TwiddleFactor Memory Switching Activity Analysis of Radix-22 and Equivalent FFT Algorithms”, IEEE Proceedings, April 2010.
- [5]. Jianing Su, Zhenghao Lu, “Low cost VLSI design of a flexible FFT processor”, IEEE Proceedings, April 2010.
- [6]. He Jing, Ma Lanjaun, XuXinyu, “A Configurable FFT Processor”, IEEE proceeding 2010.
- [7]. M.Merlyn, “FPGA Implementation of FFT Processor with OFDM Transceiver”, 2010 IEEE proceeding.
- [8]. “A Radix based Parallel pipelined FFT processor for MB- OFDM UWB system”, “Nuo Li and N.P.vanderMeijs, IEEE Proceedings, 2009.