Design of 256/64 point FFT for MIMO system

Yun-Ching Tang, Shi-Jie Zhou

Abstract

The IEEE 802.11n protocol had the preliminary mutual recognition in two big camps' coordinations. To reach high effective transmission rates, there are two approaches, OFDM modulation/demodulation and MIMO transmitter/receiver technologies, have been proposed by many authors.

For MIMO-OFDM topology, due to synchronous transmission of the multiple data, it requires high throughput of FFT processors. Reduction of chip area and power consumption of FFT circuits will become an important issue. The proposed radix-2^2 algorithm by integrating the conventional four paralleled R2²SDF structure provides two modes (256 and 64 points) with configurations of 2x2 and 4x4.

By reconfiguring its operation structure, we can use the same amount of registers similar to parallel processes of four groups of the traditional R2²SDF structures, while generate up to 4x throughputs. The architecture makes the hardware utilization ratio reaching 100% after the pipelined process. It increases hardware utilization rate and reduces the number of multipliers in comparison with the traditional R2²SDF structure. Moreover, the improved register access method reduces power consumption.

Keywords: OFDM, MIMO, FFT.

Yun-Ching Tang, Instructor, Department of Electronic Engineering, HIT.
Shi-Jie Zhou, Student, Department of Electrical Engineering, NCHU.

適用於多重輸入輸出/正交分頻多工系統之快速傅立葉轉換晶片設計

湯雲欽、周詩傑

摘 要

IEEE 802.11n 的協定在兩大陣營的協調中有了初步的共識，而為了提高傳輸率所運用的技術仍 有 2 個共通點，就是正交分頻多工系統調變／解調技術，及多重輸入輸出發送／接收技術。本文主要著眼於 MIMO-OFDM 的系統中，多筆資料的同時傳輸，將致使 FFT 電路的需求提高。因此，如何降低由於大量 FFT 電路所造成的大量面積及功率消耗，即是本論文所訴求的目標。

本文依循 MIMO-OFDM 的協定，基於 Radix-2^2 演算法設計一個整合傳統四路並行的 Radix-2^2 Single-path Delay Feedback(R2^{2}SDF)架構並提供 64 點和 256 點兩種模式以及 2x2 與 4x4 兩種組態。利用相同於平行四組傳統 R2^{2}SDF 架構所需的暫存器使用量，改變其運算結構，使得內部電路在管線化之後，乘法硬體使用率到達 100%且維持相同的 4x 產出量(throughput)，相較於 R2^{2}SDF 結構，提升其使用率，也降低了所需的乘法器數量；此外，改良的暫存器存取方式減少非必要的功率消耗。

關鍵詞：多重輸入輸出系統、正交分頻多工系統、快速傅立葉轉換。

湯雲欽：修平技術學院電子工程系講師
周詩傑：中興大學電機所學生

1. Introduction

The new generation of wire/wireless communication systems usually employs OFDM signal process architecture combining with multiple inputs and multiple outputs (MIMO). In order to enhance the FFT/IFFT signal processing capability, an efficient high throughput FFT/IFFT processor will be one of the key issues. The proposed structure can be applied to any multiple-input wireless LAN system based on the quad-rate OFDM, such as the IEEE 802.11 MIMO OFDM system shown in Fig. 1 [1]. It is appropriate not only for IEEE 802.11 OFDM system with 256/64-point FFT/IFFT, but also for any

multiple-input 2^n-point FFT/IFFT with the similar design technique. [2]

With the concept of hardware sharing, it reduces the operation time for each data and keeps the same number of registers. Moreover, the number of multipliers is also reduced.

This approach can be applied to various FFT/IFFT computation algorithms [3]. It is a very efficient operation architecture. This paper describes the conventional FFT/IFFT algorithm in the next section. In Section 3, the 4-path FFT processor architecture is presented. It shows the simulation and compares with the conventional FFT/IFFT structure. Section 4 gives the comparison of the
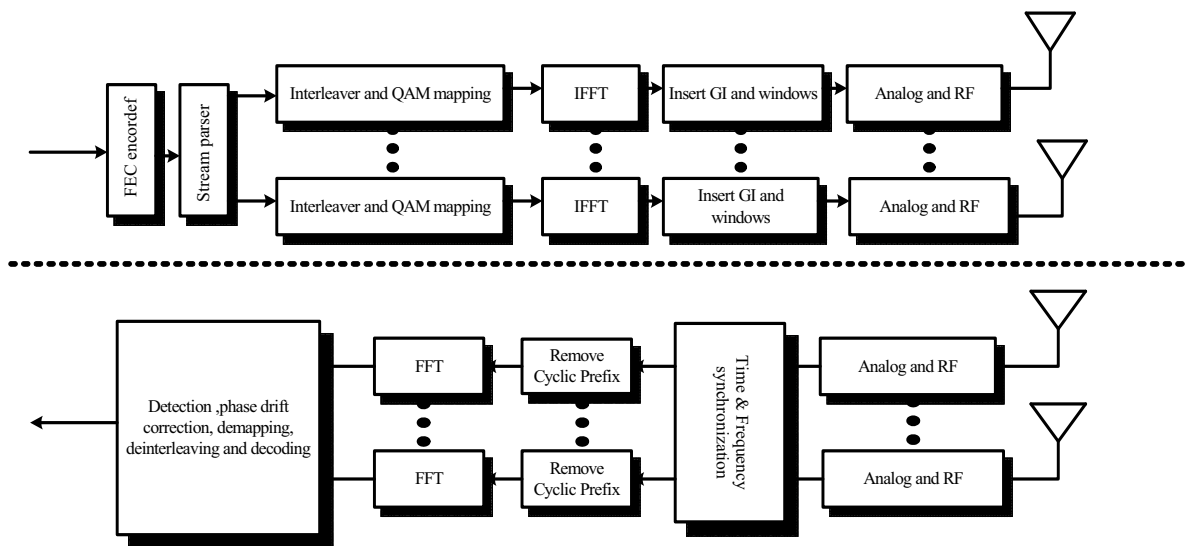

Figure 1. block diagram of the MIMO OFDM system

proposed architecture with the conventional design.

2. The pipelined FFT/IFFT Algorithm

The N-point FFT derived from the digital signal process (DFT) can be expressed as

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{nk} \qquad \text{for k=0,1,.....,N-1}$$

(1)

The N-point IFFT is

$$x[k] = \frac{1}{N} \sum_{n=0}^{N-1} X[n] W_N^{-nk} \qquad \text{for k=0,1,.....,N-1}$$

(2)

Where $\quad W_N^{nk} = e^{-j(2\pi/N)nk}$

The computation of a N-point FFT/ IFFT requires $O(N^2)$

arithmetic operations. It occupies a lot of chip area. An alternative approach is the radix-r fast Fourier formulation which reduces to $O(N\log_2 N)$ arithmetic operations in $\log_r N$ stages. The value of r could be 2, 4 or 8. [4] Since N is an integer of power of two, the radix-2 algorithm is obtained

by decomposing Eq. (1). The even terms can be expressed as

$$X[2r] = \sum_{n=0}^{N/2-1} (x[n] + x[n+(N/2)]) W_N^{2nr}$$

(3)

While the odd ones are

$$x[2r+1] = \sum_{n=0}^{N/2-1} (x[n] - x[n+(N/2)]) W_N^n W_N^{2nr}$$

(4)

We can decompose the Eq(3), Eq(4) to get the radix-4 algorithm. The four terms of radix-4 are expressed as Eq(5)~(8).

$$X[4s] = \sum_{n=0}^{(N/4)-1} \{(x[n] + x[n+N/2]) + (x[n+N/4] + x[n+3N/4])\} W_{N/4}^m$$

(5)

$$X[4s+2] = \sum_{n=0}^{(N/4)-1} \{(x[n] + x[n+N/2]) - (x[n+N/4] + x[n+3N/4])\} W_N^{2n} W_{N/4}^m$$

(6)

$$X[4s+1] = \sum_{n=0}^{(N/4)-1} \{(x[n] - x[n+N/2]) - j(x[n+N/4] - x[n+3N/4])\} W_N^n W_{N/4}^m$$

(7)

$$X[4s+3] = \sum_{n=0}^{(N/4)-1} \{(x[n] - x[n+N/2]) + j(x[n+N/4] - x[n+3N/4])\} W_N^{3n} W_{N/4}^m$$

(8)

Even though the stage number of radix-4 is less than radix-2, the butterfly unit of radix-4 is more complicated than that of radix-2. If the numbers of stages are doubled for radix-4, the complexity of butterfly units can be reduced. Which method is called radix-2^2 algorithm. The proposed architecture was based on radix-2^2 algorithm, it has shorter propagation delay and fewer register
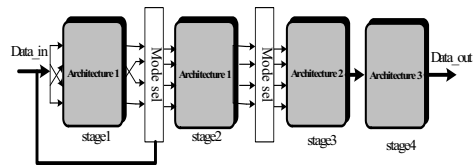

Figure 2. architecture diagram

access.

3. THE PROPOSED ARCHITECTURE

To develop higher data rate OFDM system, we the conventional FFT operation

and the hardware architecture that processing four individual data sequences has been used. [6][7] Here, the architecture of our design in Fig.2 employing 3 kinds of architectures to reduce multiplier numbers and register access. We will give the individual explanation below.

3.1. Register access method:

A good deal of power consumption is the main issue in convention pipeline structure. In order to reduce the redundant power consumption of shift operation, we use the register access method to replace FIFO (first in first out).[5] The key issue of the mechanism are

(1). Reading after writing in the same address must be fellowed.

(2). The addresses of reading and writing must be next to each other.

By using the dual-port register file of Artisan, the complete operation can be performed.   Fig. 3 is the behavior diagram of register access method. A simple control unit is needed to complete the operation of our architecture.

3.2. Architecture 1

Fig. 4 is the detail diagram of architecture 1. The operation can be divided into 2 steps. When individual data sequence was received serially from the receiver at 4x4 mode, architecture 1 does
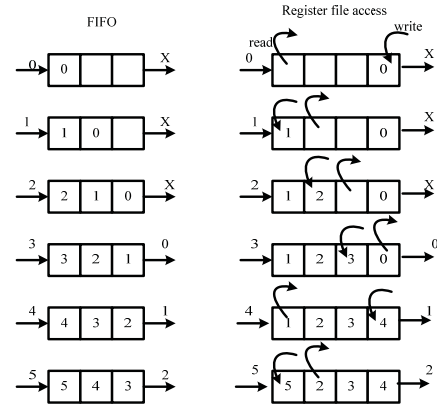
the following works,

Figure3. The behavior of register access method
Step 1:
(a)Odd data sequence:(data_1, data_3)

When x[0]~x[N/2-1] are loaded, data are stored in ff1_1, ff1_3. When x[N/2]~x[3N/4-1] are loaded, data_1, data_3 are switched to net c1, c2 and net b1, b2 is the data from ff1_1, ff1_3. Data_1, data_3 complete the butterfly operation.

a[k]=x[k]+x[N/2+k]

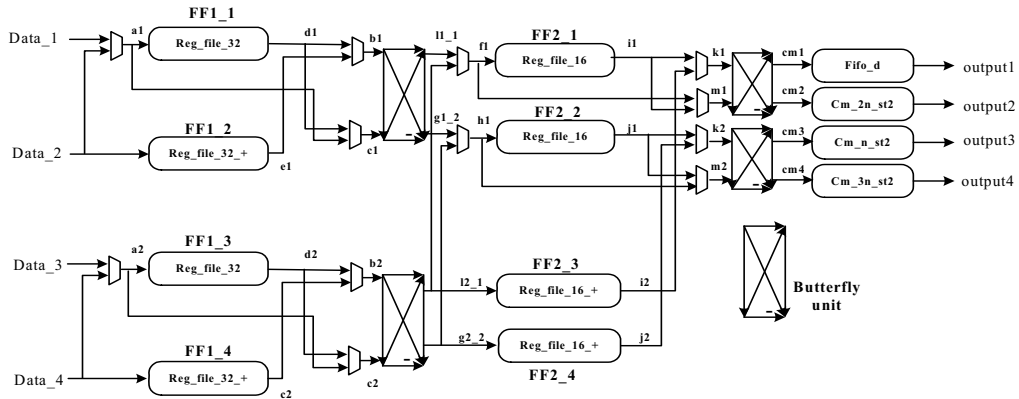Figure 4. The diagram of Architecture 1

b[k]=x[k]-x[N/2+k]

k=0,1,2…..N/2-1

a[k] are stored in ff2_1, ff2_3 and b[k] are stored in ff2_2, ff2_4. When x[3N/4]~x[N-1] are loaded, the data in net k1, k2 are from ff2_1, ff2_2(a[k], b[k], k=0,1,…N/4-1) and m1, m2 are from l1_1, g1_1 (a[k], b[k], k= N/4, …N/2-1) . At the same time, ff2_3, ff2_4 are hold on and l2_1, g2_1 are switched to f1, h1 and stored in ff2_1, ff2_2. The second butterfly operation is performed by (k1, m1) and (k2, m2).

They are processed by complex number multipliers with appropriate twiddle factors. The utilization of complex number multipliers in our design is up to 100%. After FFT/IFFT is processed, the output feeds to the next stage.

(b)Even data sequence:(data_2, data_4)

When x[0]~x[N/2-1] are loaded, data are stored in ff1_2, ff1_4. When x[N/2]~x[N-1] are loaded, data_2, data_4 are switched to net a1, a2 and are stored in ff1_1, ff1_3. At the same time, ff1_2, ff1_4 are hold on.

Step 2:

When the next x[0]~x[N/4-1] are loaded, the data that stored in ff2_1~ff2_4 are pushed to do the second butterfly operation and are processed by complex number multipliers to finish the computation of data_3. At the same time, the first butterfly operation of data_2, data_4 stored in ff1_1~ff1_4 are performed and stored in ff2_1~ff2_4. When x[N/4-1] goes to x[2N/4-1] and x[2N/4-1] goes to x[3N/4-1], the second butterfly operation of data_2 and data_4 is performed. This the complete data flow of architecture 1 in our design.

3.3. Architecture 2

We take the architecture 2 as stage 3. The utilization of complex multiplier in the conventional pipeline structure based on radix-2^2 are always up to 75%. We propose the architecture and enhance the utilization up to 100% by adjusting the order of twiddle factors for different paths. Fig. 5 shows the butterfly operation that pushes data into the corresponding multiplier at the same time.

The symbol cmx in the figure is the corresponding twiddle factor W_N^{xn}. In Fig. 5-(a), each data sequence is pushed into 4 different multipliers with the same twiddle factor at the same timing slot. However,the proposed method only needs multipliers with different twiddle factors at the same time as shown in Fig. 5-(b).

In the conventional version, 4 multipliers are needed and the utilization rate is 75%. On the other hand, 3 multiplier are needed and the utilization of each multiplier is up to 100% in the proposed method.

Fig.6-(a) is the general structure based on Fig. 5-(b). The difference of (a), (b), (c), (d) are the location of butterfly unit. These changes are the key point to make our design more efficient.

Because fewer memory units are required, the register based FIFO was used in placed of register files.
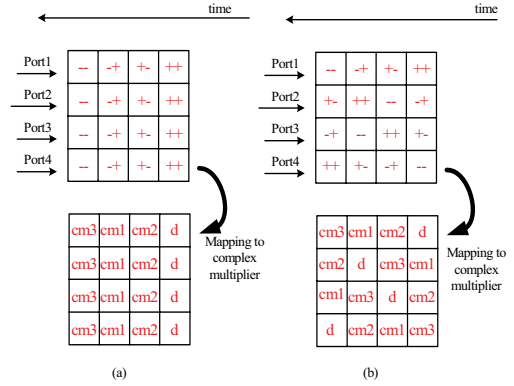

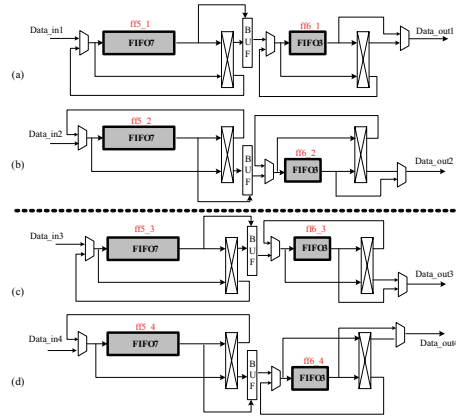Figure 5. Timing diagram of corresponding twiddle factors

Figure 6. The diagram of Architecture 2

3.4. Architecture III:

Because the last stage is not processed by complex number multipliers, it only requires butterfly operation . We can use the conventional structure for the last stage.

3.5. Data out:

Fig. 7 is the output diagram of FFT/IFFT. We show that the operation cycle time for each data sequence is just a quarter of the conventional method. Due to the proposed paralled algorithm.

4. Comparison

4 groups of conventional structures are compared with our proposed architecture and list in Table 1. (FIFO structures in our design are all replaced by register access method.)   The amount of multipliers, adders and memory units in the proposed architecture is less than the conventional approaches.

Table I Comparison of the conventional and proposed architectures with 4 parallel paths

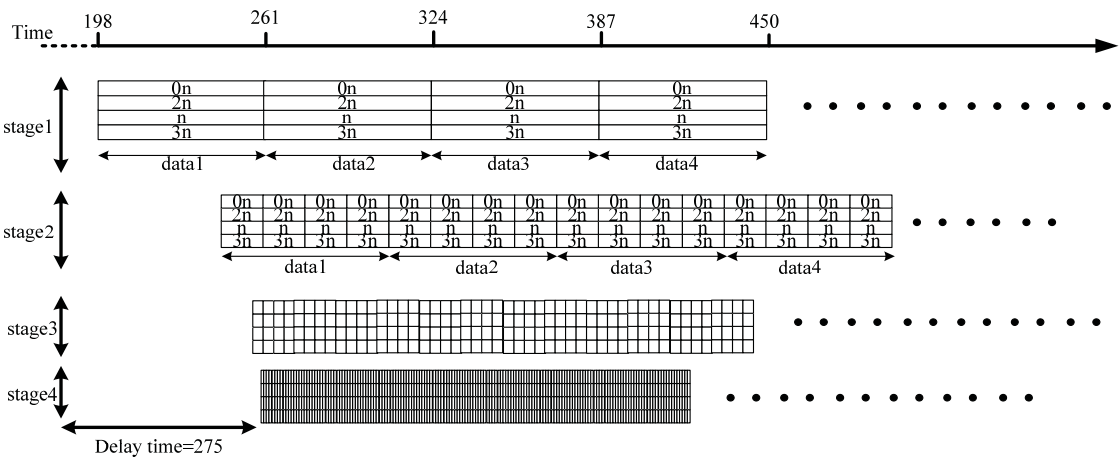| Architecture	Multiplier	Adder	Memory size
R2SDF	24	64	1020
R4SDF	12	128	1020
R2MDC	24	64	1528
R4MDC	36	64	2544
R 2^2 SDF	12	64	1020
R 2^2 MDC	24	64	1528
Proposed	9	32	1020

Figure 7. The timing diagram of 256 point FFT/IFFT

5. Conclusion

Our design of FFT/IFFT is proposed for MIMO-OFDM system. Due to multi paths, architecture duplication will increase the area cost. In addition, large amount of data processing is the key issue of larges power consumption. In this paper, the parallel architecture with more hardware sharing concept is proposed to reduce area cost and power consumption. Register access method mechanism is the key issue to reduce power consumption. We also use fewer multipliers in comparison to the conventional version.

References

[1] Allert van Zelst,Tim C.W. Schenk,"Implementation of a MIMO OFDM-based wireless LAN system" , Signal Processing, IEEE Transactions on Volume 52, Issue 2, Feb. 2004 Page(s):483 – 494

[2] Gordon L. Stuber , John R. Barry, Steve W. Mclaughlin, Ye Li, Mary Ann Ingram, Thomas G. Pratt, "Broadband MIMO-OFDM wireless communications", Proceedings of the IEEE Volume 92, Issue 2, Feb, 2004 Page(s):271 – 294

[3] S. He. and M. Torkelson. "Design and Implementation of a 1024-point Pipeline FFT Processor." in Proc. IEEE Custom Integrated Circuits Conference, pp.131-134, 1998

[4] W. Eberle, M. Badaroglu, V. Derudder, S. Thoen, P. Vandenameele, L. Van der Perre, M. Vergara, B. Gyselinckx, M. Engels, and I. Bolsens, "A digital 80Mb/s OFDM transceiver IC for wireless LAN in the 5GHz band," IEEE International Solid-State Circuits Conference, pp. 74 - 75, February 2000.

[5] H.-L. Lin, H. Lin, Y.-C. Chen and R. C. Chang, "A Novel Pipelined Fast Fourier Transform Architecture for Double Rate OFDM Systems", Proceedings of IEEE Workshop on Signal Processing Systems, Oct. 2004,pp. 7-11

[6] Yu-Wei Lin and Chen-Yi Lee, "Design of an FFT/IFFT processor for MIMO-OFDM Systems" ,IEEE transactions on circuits and systems,April, 2007 Page(s):807 – 815

[7] Yu-Wei Lin,Hsuan-Yu Liu, and Chen-Yi Lee "A 1 GS/S FFT/IFFT processor for UWB application", Solid-State Circuits, IEEE Journal of Volume 40, Issue 8, Aug. 2005 Page(s):1726 – 1735