High-Speed Four-Parallel 64-Point Radix-2^4 MDF FFT/IFFT Processor for MIMO-OFDM Systems

Hang Liu[1] and Hanho Lee[2]
[1,2] School of Information and Communication Engineering, Inha University
253 Yonghyun-dong, Nam-gu,Incheon, 402-751, Republic of Korea
E-mail: [1]mailiuhang@hotmail.com, [2]hhlee@inha.ac.kr

Abstract: This paper presents a design and implementation result of a high-speed, low-complexity four data-path 64-point radix-2^4 FFT/IFFT processor for high-throughput MIMO-OFDM wireless LAN system. The proposed FFT/IFFT processor can provide a higher throughput rate and low hardware complexity by using a four-parallel data-path scheme and a multipath delay feed-back (MDF) structure. The proposed four data-path 64-point MDF FFT/IFFT processor has a throughput rate of up to 600 M sample/s at 150 MHz while requiring much smaller hardware complexity satisfying IEEE 802.11n standard requirements.

1. Introduction

The growing demand of data, multi-media and communication has requested the need for generating many disparate devices into a high speed and efficiency bandwidth network capacity. The multiple-input multiple-output (MIMO) schemes have been widely studied and received great attention by both academy and industry in wireless communication systems. The application of multiple antennas at both transmitter and receiver provides enhanced performance over diversity system. This technique can significantly increase the data rates of wireless system without increasing system power or bandwidth. And the combination MIMO-OFDM is a very natural and beneficial choice since OFDM enables support of more channels and larger bandwidths, because it is simplifies equalization greatly in MIMO systems. The general transceiver structure of MIMO-OFDM system is depicted in Fig. 1. It mainly consists of channel encoding and modulator, IFFT, MUX, RF/IF unit, antennas, DEMUX, FFT, demodulator, and channel decoding.

In the MIMO-OFDM systems physical layer, FFT/IFFT processor is one of the kernel modules having high computational complexity. Since the MIMO-OFDM system needs more independent channel operators and processors, the system complexity and hardware cost dramatically increase. For example, a MIMO-OFDM system with N transceiver inputs and outputs require N basebands to be operated, and therefore N FFT/IFFT processors are necessary, also the system complexity increase N times. Thus, a scheme of high-speed, low-complexity FFT/IFFT processor can obviously reduces the complexity of MIMO-OFDM systems. This paper proposes a high-speed, low-complexity four-parallel radix-2^4 FFT/IFFT processor with a multipath delay feed-back (MDF).

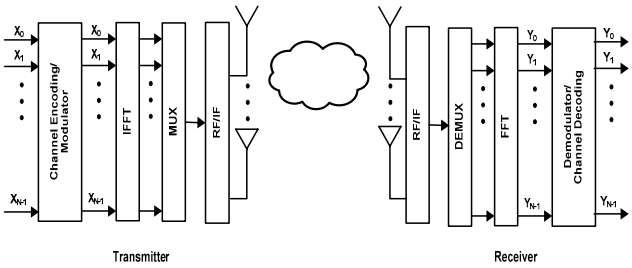


Figure 1. General transceiver of MIMO-OFDM system.

2. Design issue

In order to implement the physical layer of the MIMO-OFDM system more efficiently and low complexity, the four data-path approach has been adopted to reduce the data sampling rate from the analog-digital converter (ADC), after the serial-to-parallel (S/P) converter, the data sampling rate of each path can generally be reduced to 132 M sample/s [1]-[3]. However, the hardware cost is also increased significantly, because more memory and complex multipliers are needed to allow multiple data to be operated simultaneously. Therefore, radix-2^4 SDF FFT/IFFT architecture has been proposed to offers high throughput, low hardware complexity and low power consumption by reducing the number of complex multiplications. In pipelined FFT hardware scheme, the multipath delay commutator (MDC) scheme can achieve higher throughput rate by using multiple data paths, while the single-path delay feed-back (SDF) scheme needs less memory and hardware complexity with the delay feed-back scheme [2]-[4]. The proposed MDF architecture can provide higher throughput rate with minimal hardware cost by combining the features of MDC and SDF.

3. Proposed architecture

The radix-2^4 algorithm can take complex constant multiplier instead of programmable complex multiplier, so it can reduce the area and power consumption. Fig. 2 shows the proposed four-parallel data-path 64-point radix-2^4 MDF (R2^4MDF) FFT/IFFT processor. It consists of RAM units, butterfly units (BF1, BF2), complex Booth multipliers, CSD complex constant multipliers, MUXs, multiplexers and adders.

Butterfly units are the kernel of a FFT processor. For the multiply factors j in different butterfly steps, we design two kinds of similar butterfly units. The BF1 unit stores all of N/2-th input data in RAM. Meanwhile, the RAM keeps the swapping buffer space for the next input data. When

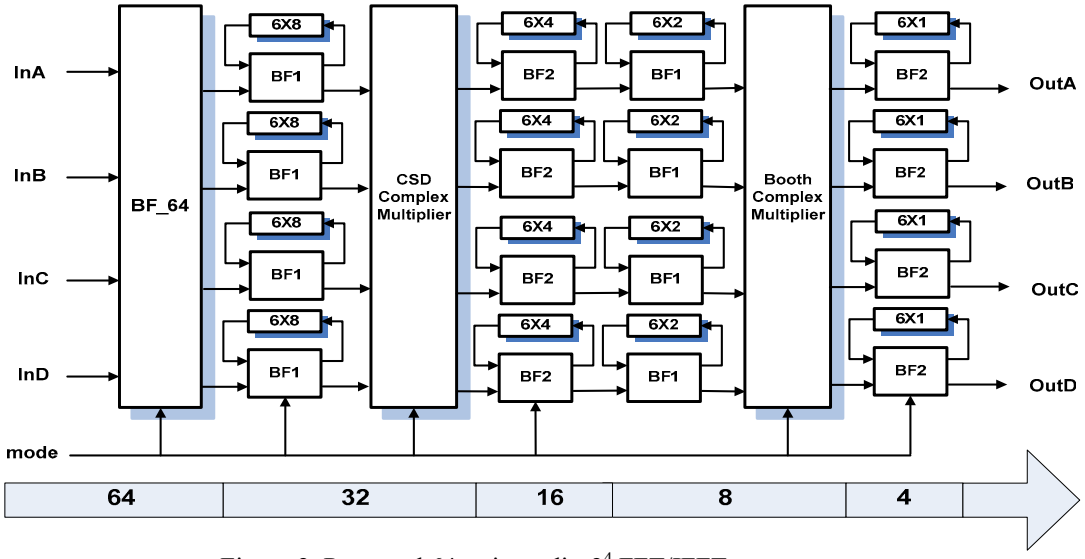


Figure 2. Proposed 64-point radix-2^4 FFT/IFFT processor.

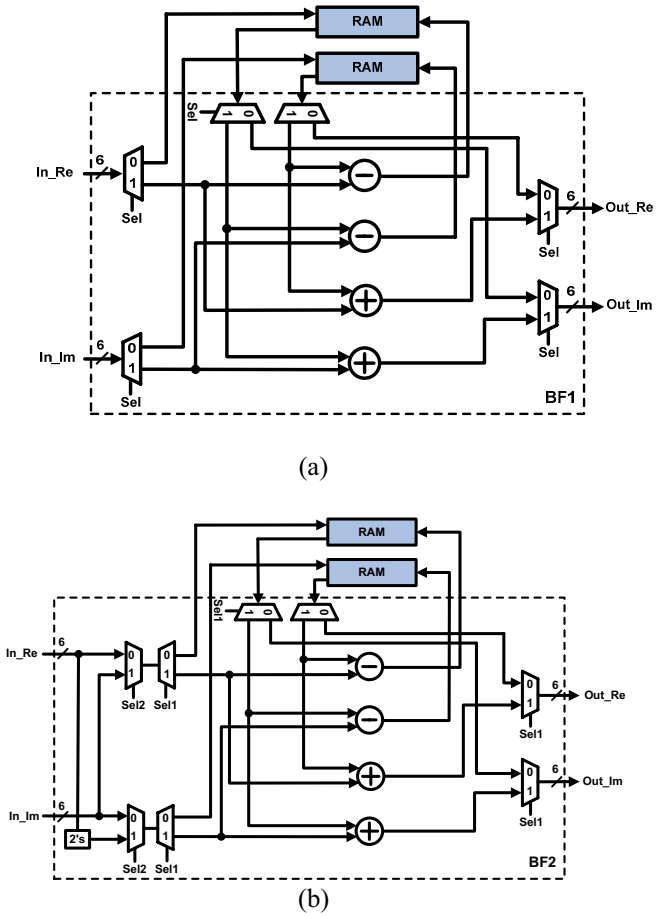


Figure 3. Radix-2^4 butterfly units: (a) BF1, (b) BF2.

(N/2+n)-th input data are fed to BF1, the input data x[n] stored in RAM are read and are added by new input data. And then the subtracted output data (x[n]-x[N/2+n]) are stored in the location of previous input data x[n]. When the

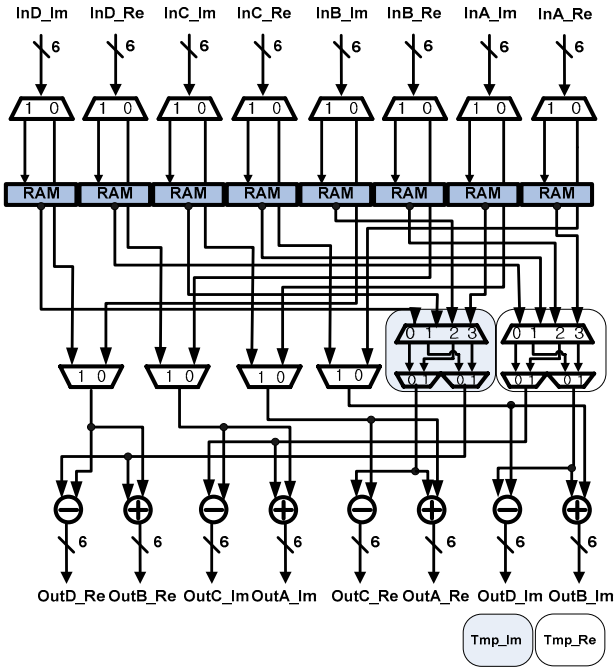
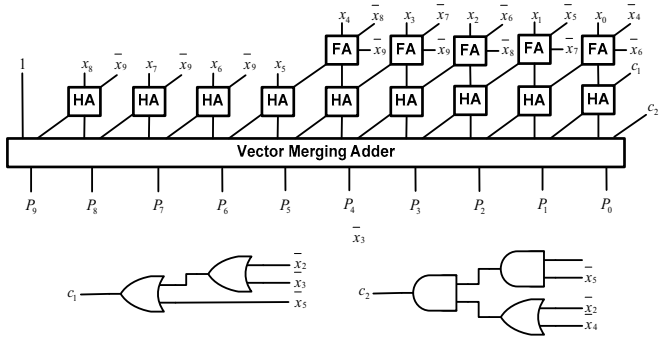


Figure 4. BF_64 module.

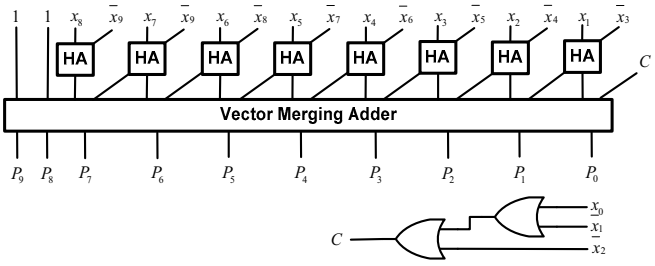
new set of input data are fed to BF1 after the N-th input operation, the new input data are stored in the RAM and (x[n]-x[N/2+n]) RAM. The BF2 unit is almost same with the BF1 unit except the operation of 3N/4-th input data multiplying of *–j*. The output results of butterfly units are complex addition and complex subtraction of two input data x[n] and x[N/2+n], where N=64, as shown in Fig. 3. The BF_64 module used the MDC hardware scheme. It consists of RAMs, and each RAM can store 8 complex data, adders, subtracters and multiplexers, as shown in Fig. 4. Four-parallel data is processed through the RAMs. It operates as following steps: at the beginning 16 cycles, the first 32 data are stored in the register file. When the next N/2-th input data are stored in the register file, the

Table 1. CSD Representation of Twiddle Factor.

Coefficients	Decimal	2'S COMP.	CSD
$\sin(\frac{\pi}{8})$	0.3827	001100	$010\bar{1}00$
$\cos(\frac{\pi}{8})$	0.9239	011101	$1000\bar{1}0$



(a)



(b)

Figure 5. (a) cos(π/8) = 0.9239, (b) sin(π/8) = 0.3827.

previously stored input data x[2n-1] are read from the register file and generates the outputs. Then these outputs are added and subtracted by x[2n-1] and the new input data x[N/2+2n-1] simultaneously. And the input data x[N/2+2n] written into BF_64 is stored in the location where the previous x[2n-1] data was. After x[N] data is stored in the register file, data x[2n] and x[N/2+2n], which are read from the register file, are added and subtracted in the BF_64 module at the next cycle simultaneously.

The radix-2^4 FFT algorithm with four-parallel data-path architectures has fewer multipliers than other schemes of lower radix FFT algorithm. The twiddle factors, W(8), W(16), W(24), and W(48) correspond to the trigonometrical functions of cos(π/8) and sin(π/8), respectively. Table 1 shows the twiddle factors, which illustrates the 6-bits coefficients in the decimal representation, the 2's complement representation, and the CSD representation. Fig. 5 depicts the structure of the multiplier twiddle factors for cos(π/8) and sin(π/8). The total CSD complex multiplier block consists of eight CSD constant multipliers, 2's complement logics, and multiplexers as shown in Fig. 6. When the real and imaginary values of twiddle factors are same, the two CSD constant multipliers are used and their two outputs are added to generate the output of the CSD complex multiplier.

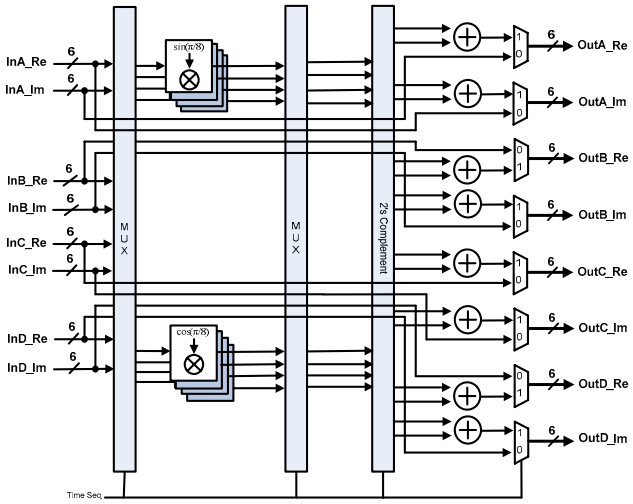


Figure 6. CSD complex constant multiplier.

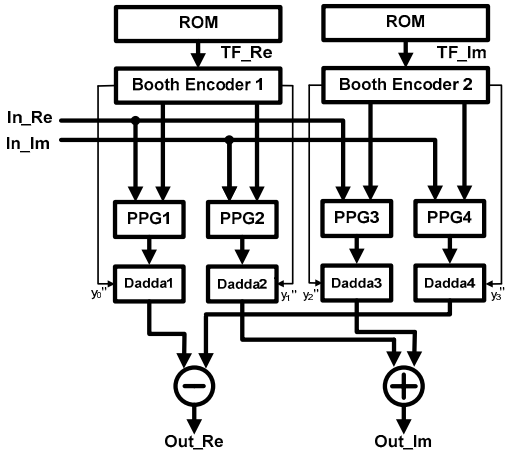


Figure 7. Booth complex multiplier.

Otherwise, the CSD constant multipliers are used for the multiplication of input and twiddle factors. If inputs don't need to multiply with twiddle factor, the output results are generated from the input directly.

For the four-parallel approach to implement the radix-2^4 FFT algorithm, we design the four-parallel data path complex Booth multipliers module. This kind of Booth multiplier needs ROMs to store the twiddle factors. To reduce the truncation error in the fixed-width multiplier, the Dadda reduction network with error-compensation circuit was used, as shown in Fig. 7. This method can compensate the truncation error of fixed-with Booth multipliers.

4. Implementation and performance

The appropriate word length in the proposed processor is determined by a fixed-point simulation before hardware implementation. Based on the simulation results, 6 word lengths of the proposed FFT/IFFT architecture were determined in both real and imaginary parts. In addition, the SQNR of the proposed FFT/IFFT processor is about 30 dB. After the appropriate word length was chosen, the

Table 2. Performance of The 64-point pipelined FFT/IFFT architectures.

	4-parallel $R2^4MDF$ [Proposed]	4-parallel MRMDC [5]	1-parallel $R2^2$ SDF [6]
Technology	Virtex-4 FPGA	0.18μm CMOS	Virtex-E FPGA
No. of Complex Multipliers	4	4	-
No. of Adders and Subtractors	163	120	-
No. of Slice	3,470	-	3,017
Logic Gate Count	43k	52k	-
Word Length	6 bit	10 bit	8 bit
Parallel format	4 data-path	4 data-path	1 data-path
Radix	Radix-2^4	Mixed Radix (Radix 2, Radix 4)	Radix-2^2
Max Clock rate	150 MHz	-	88 MHz
Throughput rate (R: clock rate)	4R (600 Msample/s)	4R	R (88 Msample/s)
Standard	802.11n	802.11n	-
SQNR	30 dB	-	-

FFT/IFFT processor was implemented using Xilinx Virtex-4 FPGA and functionally verified using Mentor Graphics' ModelSim simulator. The processor has four separated input and output data paths (A, B, C and D), and the operation of the FFT/IFFT processor is switched by the control signal "mode". The logic 0 state of the "mode" signal sets the processor to perform FFT operation while its logic 1 state enables the processor to perform IFFT operation.

Table 2 shows performance comparisons between the proposed processor and the existing 64-point FFT/IFFT processors in terms of technology, number of complex multipliers, adders and subtractors, slices, logic gate count, word length and so on. From the comparison of the implementation results, the total logic gate count of the proposed four-parallel $R2^4$MDF FFT/IFFT processor is about 43k, which is about 83% of MRMDC FFT/IFFT processor gate count[5]. The four-parallel $R2^4$MDF FFT/IFFT processor consists of 3,470 slices, and the operating clock frequency is about 150 MHz. The highest throughput rate of the proposed architecture is as high as 600 Msample/s at 150 MHz.

5. Conclusion

In this paper, we presented a four-parallel data-path pipelined 64-point radix-2^4 MDF FFT/IFFT processor. This design can operate the 64 point FFT/IFFT processing. Besides, the four-parallel data-path scheme structure enables a high performance at a comparatively low clock frequency of 150 MHz, thus it saves mass of power dissipation without the limitation of the signal processing ability. In the proposed architecture, high-speed data processing and low hardware complexity can be achieved due to radix-2^4 algorithm, CSD constant multiplier and fixed-width Booth multiplier. Furthermore, the number of memories is effectively reduced by using the radix-2^4 MDF FFT architecture. All these schemes can accomplish a high throughput performance by using low device expenditure. The proposed architecture is expected to be incorporated in MIMO-OFDM systems such as IEEE 802.11n WLAN, IEEE 802.16e mobile WiMAX and 4G.

Acknowledgement

This research was supported by the Ministry of Knowledge Economy, Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA(Institute of Information Technology Assessment)" (IITA-2008-C1090-0801-0019).

References

[1] C-H. Shin, S. Choi, H. Lee, and J-K. Pack, "Design and Performance of 4-Parallel MB-OFDM UWB Receiver," *IEICE Trans. Communications*, vol.E90-B, no. 3, pp. 672-675, March 2007.

[2] Y-W. Lin, H-Y. Liu, and C-Y. Lee, "A 1-GS/s FFT/IFFT processor for UWB applications," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 8, pp. 1726-1735, August 2005.

[3] J. Lee and H. Lee, "A High-Speed 2-Parallel Radix-2^4 FFT/IFFT Processor for MB-OFDM UWB Systems," *IEICE Transactions on Fundamentals of Electronics, Communications, and Computer Sciences,* vol. E91-A, no. 4, pp. 1206-1211, April 2008

[4] W-C. Yeh and C-W. Jen, "High-speed and low-power split-radix FFT," *IEEE Transactions on Acoustics, Speech, and Signal Processing,* vol. 51, no. 3, pp.864–874, March 2003.

[5] S. Lee, Y. Jung, J. Kim, "Low complexity pipelin FFT processor for MIMO-OFDM systems," *IEICE Electronics Express*, vol.4, no.23, pp.750–754, October 2007.

[6] S. Sukhsawas, K. Benkrid, "A high-level implementation of a high performance pipeline FFT on Virtex-E FPGAs," *Proceedings of the IEEE Computer Society Annual Symposium on VLSI Emerging Trends in VLSI System Design*, vol.2, no.4 pp. 229-232, February 2004