A High-Speed Low-Complexity Two-Parallel Radix-2^4 FFT/IFFT Processor for UWB Applications

Hanhoo Lee and Minhyeok Shin
School of Information and Communication Engineering
Inha University, Incheon, 402-751, Korea
hhlee@inha.ac.kr, zoomuni@hanmail.net

Abstract—This paper presents a high-speed, low-complexity two data-path 128-point radix-2^4 FFT/IFFT processor for MB-OFDM ultrawideband (UWB) systems. The proposed FFT processor uses a method for compensating the truncation error of fixed-width Booth multipliers with Dadda reduction network, which keep the input and output the 8-bit width. This method leads to reduction of truncation errors compared with direct-truncated multipliers. It provides lower hardware complexity and high throughput with almost same SQNR compared with direct-truncated Booth multipliers. The proposed FFT/IFFT processor has been designed and implemented with 0.18-μm CMOS technology in a supply voltage of 1.8 V. The proposed two-parallel FFT/IFFT processor has a throughput rate of up to 900 Msample/s at 450 MHz while requiring much smaller hardware complexity.

I. INTRODUCTION

Ultrawideband (UWB) communication systems, which enable the delivery of data from a rate of 110 Mb/s at a distance of 10 m to a rate of 480 Mb/s at a distance of 2 m in a realistic multipath environment, are ideally suited to application in short range wireless communications because they can share a frequency band with existing narrowband systems and offer a higher data rate than 802.11 or Bluetooth [1]. One of the communication methods for IEEE 802.15.3a standard is Multiband Orthogonal Frequency Division Multiplexing (MB-OFDM), which offers 528 MHz bandwidth [2][3]. To minimize power consumption and provide multiple Simultaneous Operating Piconet(SOP) satisfying the FCC regulatory, a MB-OFDM UWB system has been proposed. This system transmits OFDM symbols using a different carrier frequency from symbol to symbol according to Time Frequency (TF) codes [2][3]. MB-OFDM-based UWB not only has reliably high-data-rate transmission in time-dispersive or frequency-selective channels without having complex time-domain channel equalizers but also can provide high-spectral efficiency.

The FFT/IFFT processor is one of the modules having high computational complexity in the physical layer of the MB-OFDM UWB system. This paper proposes a high-speed, low-complexity two-parallel FFT/IFFT processor, which uses a method for compensating the truncation error of fixed-width Booth multipliers with Dadda reduction network, which keep the input and output the 8-bit width.

This paper is organized as follows. Section II describes the design issues of MB-OFDM UWB communication systems. Section III describes the proposed 128-point radix-2^4 FFT/IFFT architecture. In Section IV, the hardware cost and throughput rate of the proposed FFT/IFFT architecture is compared with that of the existing 128-point FFT/IFFT architecture for MB-OFDM UWB applications. Conclusions are presented in Section V.

II. DESIGN ISSUE OF THE FFT PROCESSOR FOR THE MB-OFDM UWB SYSTEMS

A block diagram of the proposed physical layer of MB-OFDM UWB systems is shown in [2][3]. In the MB-OFDM UWB system, the data rate is from 53.3 to 480 Mb/s with code rates of 1/3, 11/32, 1/2, 5/8, and 3/4. In order to implement the physical layer of the MB-OFDM UWB system more efficiently, the four data-path approach has been adopted to reduce the data sampling rate from the analog-digital converter (ADC) such that, after the serial-to-parallel (S/P) converter, the data sampling rate of each path can generally be reduced to 132 Msamples/s [4][5]. 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, two-parallel data-path radix-2^4 single-path delay feedback (SDF) FFT/IFFT architecture has been proposed to offer high throughput, low hardware complexity and low power consumption [6].

The main motivation of this paper is to reduce the bit-width of two-parallel data-path pipelined radix-2^4 FFT/IFFT architecture using Booth multipliers with error compensation circuit and minimize the critical path delay using Dadda reduction network. Therefore, the proposed radix-2^4 FFT/IFFT architecture offers low hardware complexity and low power consumption.
III. PROPOSED RADIX-2^4 FFT ARCHITECTURE

Radix-2^4 algorithm is described in detail in [6][7]. The algorithm can take complex constant multiplier instead of programmable complex multiplier. The Canonic Signed Digit (CSD) constant multiplier contains the fewest number of non-zero bits, so it can be reduce the area and power consumption [8].

A block diagram of the proposed two-parallel data-path 128-point R^2 SDF FFT/IFFT processor is shown in Fig. 1. The proposed architecture consists of a memory block, butterfly units (BF1, BF2), complex Booth multipliers, CSD complex constant multipliers, register files, and some multiplexers. The operation of the FFT/IFFT processor is controlled by the control signal “IFFT/FFT_sel” as shown in Fig. 1. 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=128, as shown in Fig. 2. The BF1 stores all of N/2-th input data in RAM. When (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 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]) data stored in RAM at previous cycle are read from the RAM. The BF2 architecture is almost same with the BF1 architecture except the operation of 3N/4-th input data multiplying of -j.

Module 1 operation is almost similar with the BF2 operation but has two-parallel data path operation with memory. First, for the two-parallel data path operation, the two of the complex inputs are stored in the RAM. When N/2-th input data are stored in the RAM, the previously stored input data x[2n-1] are read from the RAM and generates the outputs, which are added and subtracted by x[2n-1] and the new input data x[N/2+2n] simultaneously. And the input data x[N/2+2n] fed to module 1 is stored in the location in which the previous x[2n-1] data was stored. After x[N] data is stored in the RAM, data x[2n] and x[N/2+2n], which are read from the RAM, are added and subtracted in the module 1 at the next cycle simultaneously.

The last BF1 in Fig. 1 is 64-point butterfly unit because it needs to wait the input data of even time. Meanwhile other BF units need just half point of butterfly, because the input data of odd times and even times operate separately. Two complex Booth multipliers are needed in the two-parallel approach to implement the radix-2^4 FFT algorithm. The Booth multiplier needs a ROM to store the multiplicand, which is twiddle factor. Since only 1/8 period of cosine and sine waveforms are needed, 16 kinds of the twiddle factors, which is 1/8 out of 128 points, are stored in ROM [9]. Thus, the ROM stores all 32 bytes (16×8×2 bits) twiddle coefficients. To reduce the truncation error in the fixed-width Dadda multiplier, the error compensation method proposed in [10] was used.

The radix-2^4 FFT algorithm based two-parallel
The twiddle factors, \( W(8), W(16), W(24), \) and \( W(48) \) correspond to the trigonometrical functions of \( \cos(\pi/8), \sin(\pi/8) \) and \( \cos(\pi/4) \), respectively. Table I shows the twiddle factor, which represents the 8-bits coefficients in the decimal format, the 2’s complement, and the CSD format. Fig. 4 shows the structure of the CSD complex constant multipliers for \( \cos(\pi/4) \). To efficiently compensate for the quantization error, the truncated bits are divided into two groups (major group and minor group) depending upon their effects on the quantization error. The error compensation bias is first expressed in terms of the truncated bits in the major group. The effects of the other truncated bits in the minor group are then handled by a probabilistic estimation [8]. The total compensation bias, \( C \), circuit is shown in Fig. 4. The CSD complex multiplier block consists of six CSD constant multipliers, 2’s complement logics, and multiplexers as shown in Fig. 5. Each real value and imaginary value of the output data are outputted by six CSD constant multipliers. When the real and imaginary values of twiddle factors are same, the two CSD constant multipliers are used and theirs two outputs are added to generate the output of the CSD complex multiplier. Otherwise, when the real and imaginary values are not same, the four CSD constant multipliers are used for the multiplication of input and twiddle factors. If inputs don’t need to multiply with twiddle factor (in case of ‘X’ in Table II), the output results are generated from the input directly.

IV. IMPLEMENTATION AND PERFORMANCE EVALUATION

The appropriate word length in the proposed 128-point two-parallel pipelined R2SDF FFT/IFFT processor is determined by a fixed-point simulation before hardware implementation. Fig. 6 shows the simulation results for the relation of SNR with the internal word length of the FFT/IFFT processor. The detailed explanation is described in our previous paper [6]. Based on the simulation results, 8 word lengths of the proposed FFT/IFFT architecture were determined in both real and imaginary parts. In addition, the SQNR of the proposed 128-point R2SDF FFT/IFFT processor is about 27 dB. After the appropriate word length was chosen, the FFT/IFFT processor was implemented using a standard-cell based design methodology and the 0.18-\mu m MagnaChip Components library plus full-custom memory and register file blocks. Fig. 7 shows the layout view of proposed 128-point FFT architecture. The core size excluding the memories is 0.8 \times 0.8 \text{ mm}^2.

The performance and hardware cost of the pipelined FFT/IFFT processor are increased as a result of using the multiple data-path approach. In general, conventional FFT architectures have used a four-parallel data-path approach [4][5], which requires higher hardware cost. However, the proposed two-parallel data-path pipelined R2SDF FFT/IFFT processor provides higher throughput rate with higher clock frequency while the hardware cost is reduced significantly.
Table II shows performance comparisons between the proposed 2-parallel R24SDF FFT/IFFT processor and the existing 128-point FFT/IFFT processors [5][6]. The two-parallel R24SDF FFT/IFFT processor consists of 58,000 gates excluding memories, and the operating clock frequency is about 450 MHz. Although the number of registers in our design is greater than previous four-parallel architecture, it is implemented by two 190×8 bits RAM, which requires small area cost. The result shows that the total gate count of the proposed FFT architecture requires 32% smaller gate count compared to this in the conventional two-parallel R24SDF processor [6] because of the reduced word length. Also, it not only has a significantly reduced number of complex multiplication and complex addition but also can provide the highest clock frequency 450 MHz due to two-parallel data-path and pipelined complex Booth multiplier. The highest throughput rate of our proposed architecture is up to 900 Msamples/s at 450 MHz.

V. CONCLUSION

In this paper, a two-parallel data-path pipelined 128-point radix-24 SDF FFT/IFFT processor for a MB-OFDM UWB system has been proposed. In the proposed architecture, high-speed data processing and low hardware complexity can be achieved due to fixed-width Booth multiplier with Dadda reduction network and the reduced word length. Furthermore, the number of complex Booth multipliers is effectively reduced by using a radix-2 SDF FFT algorithm. The performance results show that the data processing rate is as high as 900 Msamples/s while requiring small hardware complexity. The proposed architecture is expected to be incorporated in high-speed, low-complexity MB-OFDM UWB systems.

ACKNOWLEDGMENT

This research was supported by the MIC (Ministry of Information and Communication), Korea, under the ITRC (Information Technology Research Center) support program supervised by the IITA (IITA-2006-C1090-0603-0019).

REFERENCES


Fig. 7. Layout view of proposed 128-point FFT architecture.