

Design of Low-Complexity FFT Processor for MIMO-OFDM Based SDR Systems

Gijung Yang, Yunho Jung

*School of Electronics, Telecommunications, and Computer Engineering, Korea Aerospace University,
100 Hanggongdae-gil Hwajeon-dong Deokyang-gu Goyang-city Gyeonggi-do 412-791 Korea
{ykj1985, yjung}@kau.ac.kr*

Abstract— In this paper, an area-efficient FFT processor is proposed for MIMO-OFDM based SDR systems. The proposed scalable FFT processor can support the variable lengths of 64, 128, 512, 1024 and 2048. By reducing the required number of non-trivial multipliers with mixed-radix (MR) and multi-path delay commutator (MDC) architecture, the complexity of the proposed FFT processor is dramatically decreased without sacrificing system throughput. The proposed FFT processor was designed in hardware description language (HDL) and implemented with Xilinx Virtex-4 FPGA. With the proposed architecture, the number of slices for the processor is 1784, the number of DSP48s (dedicated multiplier) is 48, and the size of memory is 90Kbits, which are reduced by 41.9%, 62.5% and 39%, respectively, compared with those of the 4-channel radix-2 single-path delay feedback (R2SDF) FFT processor. Also, compared with 4-channel radix-2 MDC (R2MDC) FFT processor, it is confirmed that the number of slices, the number of DSP48s and memory size are reduced by 26.2%, 25% and 26.8%, respectively.

Keywords— MIMO-OFDM, MDC, MR, Scalable FFT, SDR

I. INTRODUCTION

Recently, the multiple-input multiple-output (MIMO) schemes have been proposed as an efficient solution for the high-speed wireless systems. MIMO schemes can achieve a significant increase in the bandwidth efficiency as well as an improvement of the transmission reliability over flat fading channels. However, the large delay spreads in non-flat fading channels such as frequency-selective channels destroy the function of MIMO system. Using orthogonal frequency-division multiplexing (OFDM), the channel impulse response can be considered to be flat within each sub-carrier. Therefore, MIMO with OFDM can be effectively used in non-flat fading channels [1]-[4]. However, since the MIMO-OFDM system transmits multiple data streams, it requires several independent baseband processors and its hardware complexity seriously increases compared with single-input single-output OFDM (SISO-OFDM) systems. Particularly, since the fast Fourier transformation (FFT) processor is one of the largest modules in the OFDM baseband processor, it is very important to design the low-complexity FFT/IFFT processor for MIMO-OFDM based communication systems [5].

Software defined radio (SDR) system has become a topic of great interest due to the needs for the reconfigurable and converged wireless communication systems [6]-[8]. In particular, the SDR system which can support both IEEE 802.11n WLAN mode in indoor and IEEE 802.16e mobile WiMax mode in outdoor has attracted increasing attention for seamless wireless communications. The FFT processor for that system should perform a 64/128-point FFT for IEEE 802.11n WLAN system, and a 128/512/1024/2048-point FFT for IEEE 802.16e mobile WiMax system [9]-[10].

The most common way is to design FFT processors is to use as many FFT processors as the number of data streams [11]. Since the single-path delay feedback (SDF) pipeline FFT architecture includes the smallest number of non-trivial multiplications, which are the most complex operations in the FFT processor, it is widely used in a SISO-OFDM system [12]-[13]. However, the hardware complexity of FFT in the MIMO-OFDM systems increases linearly as the number of data streams. In order to solve this problem, multi-path delay commutator (MDC)[14] architecture, which can handle four input data streams simultaneously by a single FFT processor, is presented in [15]. The hardware complexity of FFT processor for MIMO-OFDM systems are analyzed, and it is confirmed that for k input data stream, radix-4 MDC (Rk MDC) architecture is more area efficient than k radix-2 SDF (R2SDF) architecture. In [16]-[17], mixed-radix MDC (MRMDC) architecture is proposed for MIMO-OFDM system with four transmit and four receive antennas, and MRMDC architecture can reduce the number of non-trivial multiplications more than Rk MDC architecture. However, MRMDC architecture of [16]-[17] cannot support the variable length of FFT due to the repetitive architecture of simple radix-4/2 operations.

In this paper, a scalable FFT processor is proposed for MIMO-OFDM based SDR systems. The proposed FFT processor is based on mixed-radix algorithm and MDC architecture to support 4-channel 64, 128, 512, 1024 and 2048-point FFT operation. The rest of the paper is organized as follows: Section II presents the proposed FFT algorithm and the hardware architecture of the proposed FFT processor is described in Section III. Section IV presents the design and

implementation results of the proposed FFF processor, and finally, Section V concludes the paper.

II. PROPOSED FFT ALGORITHM

Since IEEE 802.11n WLAN system requires 64/128-point FFT operation and IEEE 802.16e mobile WiMax system does 128, 512, 1024 and 2048-point FFT, the SDR system that can support both modes should adaptively provide 64, 128, 512, 1024 and 2048-point FFT operation. In this paper, we propose the MR-2/2/4/2/4/2/4/2 index decomposition method for that FFT processor, which makes it possible to support various length of FFT and to minimize the required number of non-trivial multiplications.

An N -point discrete Fourier transform (DFT) can be expressed as

$$X(k) = \sum_{n=0}^{N-1} x(n) \cdot W_N^{nk}, (0 \leq n < N, 0 \leq k < N) \quad (1)$$

Replacing n and k with (2), (1) can be rewritten as (3) for $N=2048$, where x_{2048} , A_1 , A_2 , A_3 , B_1 and B_2 are given by (4), respectively. From (3), it is shown that 2048-point FFT can be calculated by computing 2-point FFT and 1024-point FFT. Similarly, 1024-point FFT can be computed by 2-point FFT and 512-point FFT, and 512-point FFT can be calculated by 4-point FFT and 128-point FFT. Also, 128-point FFT is composed of 2-point FFT and 64-point FFT. Namely, by the proposed decomposition scheme, 1024-point FFT can be easily implemented by skipping the first radix-2 operation and the multiplication by $W_4^{n_2 k_1}$ from 2048-point FFT. Also, 512-point FFT can be performed by skipping the second 2-point FFT and the multiplication by $W_{256}^{B_1 A_1}$, and 128-point FFT does by omitting the third 4-point FFT and the multiplication

by $W_8^{n_4 k_3}$. It means that the FFT processor with the proposed decomposition can support 64, 128, 512, 1024 and 2048-point FFT adaptively. In order to minimize the required number of non-trivial multiplication, the indexes for twiddle factors (TF), $W_{256}^{B_1 A_1}$, $W_{64}^{B_2 A_2}$, are also determined adaptively as FFT-length: For 2048-point FFT, $B_1 = 16n_3 + 8n_4 + 2n_5 + n_6$ and $A_1 = k_1 + 2k_2$, For 1024-point FFT, k_1 becomes zero, so, A_1 is $2k_2$. Similarly, in case of 512-point FFT, B_2 and A_2 of $W_{64}^{B_2 A_2}$ are given by $2n_5 + n_6$ and $k_3 + 4k_4$, respectively, and in case of 128-point FFT, k_3 is zero, then A_2 is $4k_4$.

Also, the TF $W_{2048}^{n_7 A_3}$ in (3) can be expressed as (5). By comparing the two expressions in (5), $W_{2048}^{n_7 A_3}$ and $W_{1024}^{n_7 A_3}$ have the same value, when k_1 is equal to zero. It means that the TF of a 1024-point FFT are equal to the first 1024 TF of a 2048-point FFT. Similarly, $W_{2048}^{n_7 A_3}$ and $W_{512}^{n_7 A_3}$ have the same value, when k_1, k_2 are equal to zero. Also, $W_{2048}^{n_7 A_3}$ and $W_{128}^{n_7 A_3}$ have the same value, when k_1, k_2, k_3 are equal to zero, and $W_{2048}^{n_7 A_3}$ and $W_{64}^{n_7 A_3}$ have the same value, when k_1, k_2, k_3, k_4 are equal to zero. Therefore, one 2048-point FFT twiddle factor ROM (TFR) can generate all 64, 128, 512, 1024 and 2048-point TFs, which can reduce the hardware complexity.

The proposed MR-2/2/4/2/4/2/4/2 decomposition scheme supports 1024 and 2048-point FFT by using two radix-2 operations, which require four non-trivial multiplications. Also, it supports 64, 128 and 512-point FFT by the repeated mixed-radix (radix-4/2) operations, which include eight non-trivial multiplications, which are reduced by 20 compared with four-channel R2SDF FFT processor.

$$\left\{ \begin{array}{l} n = \frac{N}{2}n_1 + \frac{N}{4}n_2 + \frac{N}{16}n_3 + \frac{N}{32}n_4 + \frac{N}{128}n_5 + \frac{N}{256}n_6 + n_7 \\ n_1 \in \{0,1\}, n_2 \in \{0,1\}, n_3 \in \{0,1,2,3\}, n_4 \in \{0,1\}, n_5 \in \{0,1,2,3\}, n_6 \in \{0,1\}, n_7 \in \left\{0, \dots, \frac{N}{256} - 1\right\} \\ k = k_1 + 2k_2 + 4k_3 + 16k_4 + 32k_5 + 128k_6 + 256k_7 \\ k_1 \in \{0,1\}, k_2 \in \{0,1\}, k_3 \in \{0,1,2,3\}, k_4 \in \{0,1\}, k_5 \in \{0,1,2,3\}, k_6 \in \{0,1\}, k_7 \in \left\{0, \dots, \frac{N}{256} - 1\right\} \end{array} \right. \quad (2)$$

$$\begin{aligned} & \sum_{n=0}^{2047} x(n) W_{2048}^{nk} \\ &= \sum_{n_7=0}^7 \left(\sum_{n_6=0}^1 \left(\sum_{n_5=0}^3 \left(\sum_{n_4=0}^1 \left(\sum_{n_3=0}^3 \left(\sum_{n_2=0}^1 \left(\sum_{n_1=0}^1 x_{2048} \cdot W_2^{n_1 k_1} \right) W_4^{n_2 k_1} \right) W_2^{n_2 k_2} \right) W_{256}^{B_1 A_1} \right) W_4^{n_3 k_3} \right) W_8^{n_4 k_3} \right) W_2^{n_4 k_4} \right) W_{64}^{B_2 A_2} \right) W_4^{n_5 k_5} \right) W_8^{n_6 k_5} \right) W_2^{n_6 k_6} \right) W_{2048}^{n_7 A_3} \right) W_8^{n_7 k_7} \end{aligned} \quad (3)$$

$$\left\{ \begin{array}{l} x_{2048} = x[1024n_1 + 512n_2 + 128n_3 + 64n_4 + 16n_5 + 8n_6 + n_7] \\ A_1 = k_1 + 2k_2, A_2 = k_3 + 4k_4, A_3 = k_1 + 2k_2 + 4k_3 + 16k_4 + 32k_5 + 128k_6 \\ B_1 = 16n_3 + 8n_4 + 2n_5 + n_6, B_2 = 2n_5 + n_6 \end{array} \right. \quad (4)$$

$$\begin{cases}
W_{2048}^{n_7 A_3} = W_{2048}^{n_7 (k_1 + 2k_2 + 4k_3 + 16k_4 + 32k_5 + 128k_6)} \\
W_{1024}^{n_7 A_3} = W_{1024}^{n_7 (k_2 + 2k_3 + 8k_4 + 16k_5 + 64k_6)} \\
= W_{2048}^{2n_7 (k_2 + 2k_3 + 8k_4 + 16k_5 + 64k_6)} \\
W_{512}^{n_7 A_3} = W_{512}^{n_7 (k_3 + 4k_4 + 8k_5 + 32k_6)} \\
= W_{2048}^{4n_7 (k_3 + 4k_4 + 8k_5 + 32k_6)} \\
W_{128}^{n_7 A_3} = W_{128}^{n_7 (k_4 + 2k_5 + 8k_6)} = W_{2048}^{16n_7 (k_4 + 2k_5 + 8k_6)} \\
W_{64}^{n_7 A_3} = W_{64}^{n_7 (k_5 + 4k_6)} = W_{2048}^{32n_7 (k_5 + 4k_6)}
\end{cases} \quad (5)$$

III. HARDWARE ARCHITECTURE OF THE PROPOSED FFT PROCESSOR

Figure 1 shows the hardware block diagram of the proposed scalable FFT processor, which consists of data mapping module (DMM), radix-2 butterfly module1 (R2BM1), R2BM2, radix-4 butterfly module1 (R4BM1), R4BM2 and data reordering module (DRM). At first, 1-4 input data streams are reconstructed to be sized as FFT-length by DMM. If $N=2048$, the reconstructed data are transferred to R2BM1. However, if $N=1024$, the data enter the first R2BM2 by switch (multiplexer). Similarly, the reconstructed data are transferred to R4BM1, the second R2BM2, or R4BM1 when $N=512$, 128, or 64, respectively. Each butterfly module consists of butterfly unit (BU), delay commutator, and (trivial or non-trivial) multipliers as depicted in Figure 2. The hardware architecture of R2BM1 is shown in Figure 2(a), which is composed of two BUs, one trivial multiplier, and one delay commutator with delay factor of 256. Figure 2(b) depicts the architecture of R2BM2. R2BM2 consists of two BUs, four non-trivial multipliers, TFR, one delay commutator with delay factor of $2d$. d is 64, 8, or 1 for the first, second, or third R2BM2, respectively. Also, TFR in R2BM2 generates the TF

adaptively; $W_{256}^{B_1 A_1}$ is for the first R2BM2, $W_{64}^{B_2 A_2}$ is for the second R2BM2, and $W_{2048}^{n_7 A_3}$ is for the third R2BM2. The structure of R4BM1 is shown in Figure 2(c), which is composed of one radix-4 BU, three trivial multipliers, and one delay commutator with delay factors of j , $2j$, and $3j$. j is given by 32 for the first R4BM1 and by 4 for the second R4BM1. R4BM2 includes one radix-4 BU, three trivial multipliers, and one delay commutator with delay factor of 1.

IV. DESIGN AND IMPLEMENTATION RESULTS

The proposed FFT processor was designed in hardware description language (HDL) and implemented with Xilinx Virtex-4 FPGA. With the proposed architecture, the number of slices is 1784, the number of DSP48s (dedicated multiplier) is 48, and the size of the required memory is 90Kbits. In order to verify the efficiency of the proposed architecture, FFT processors with 4-channel 2048-point R2SDF and R2MDC architectures are also designed. Table 1 shows the comparison results for the logic slices, DSP48s, and memory size. R2SDF FFT processor includes the logic slices of 3072, the DSP48s of 128, and the memory of 147Kbits, and R2MDC FFT processor does the logic slices of 2416, the DSP48s of 64, and the memory of 123Kbits. Comparison results show that MDC architecture is more area-efficient than SDF architecture, and the proposed architecture is more area-efficient than R2MDC architecture.

Table 1. Comparison results for the logic slices, DSP48s, and memory size of the scalable FFT/IFFT processors (2048-point)

	No. of slices	No. of DSP48s	Memory (bits)
4*R2SDF	3072 (100%)	128 (100%)	147K (100%)
2*R2MDC	2416 (78.6%)	64 (50%)	123K (84%)
Proposed	1784 (58.1%)	48 (37.5%)	90K (61%)

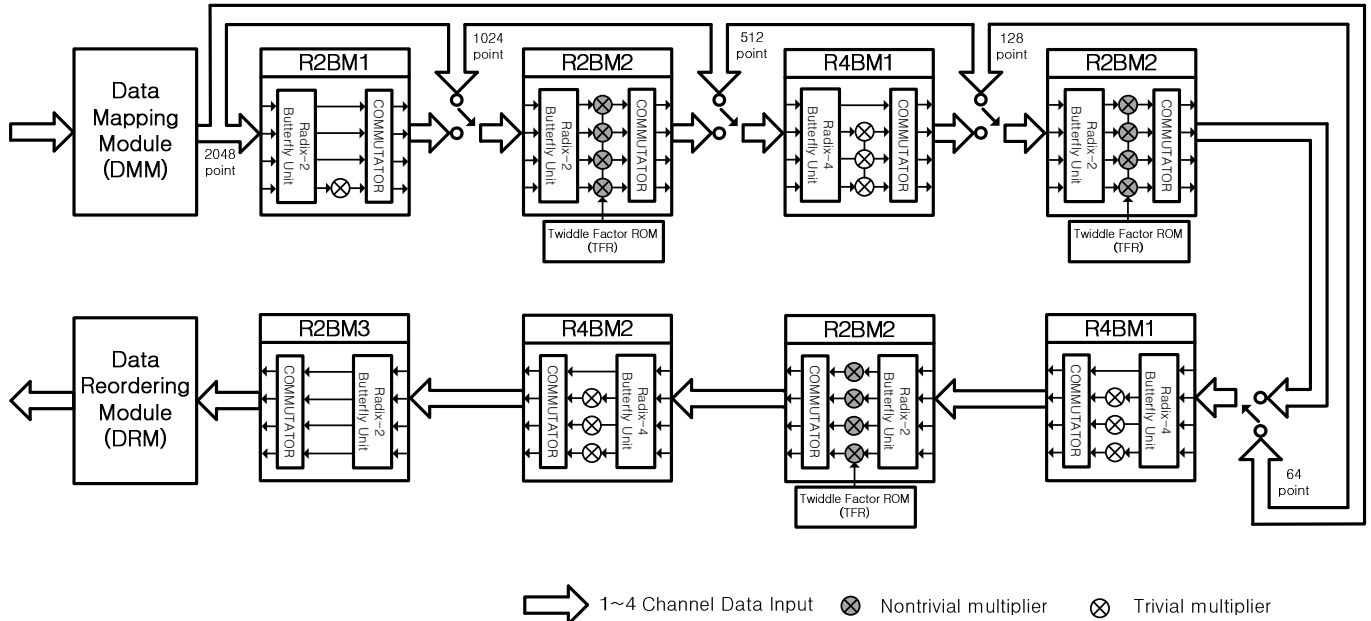


Figure 1. Block diagram of the proposed for MIMO-OFDM Based SDR systems

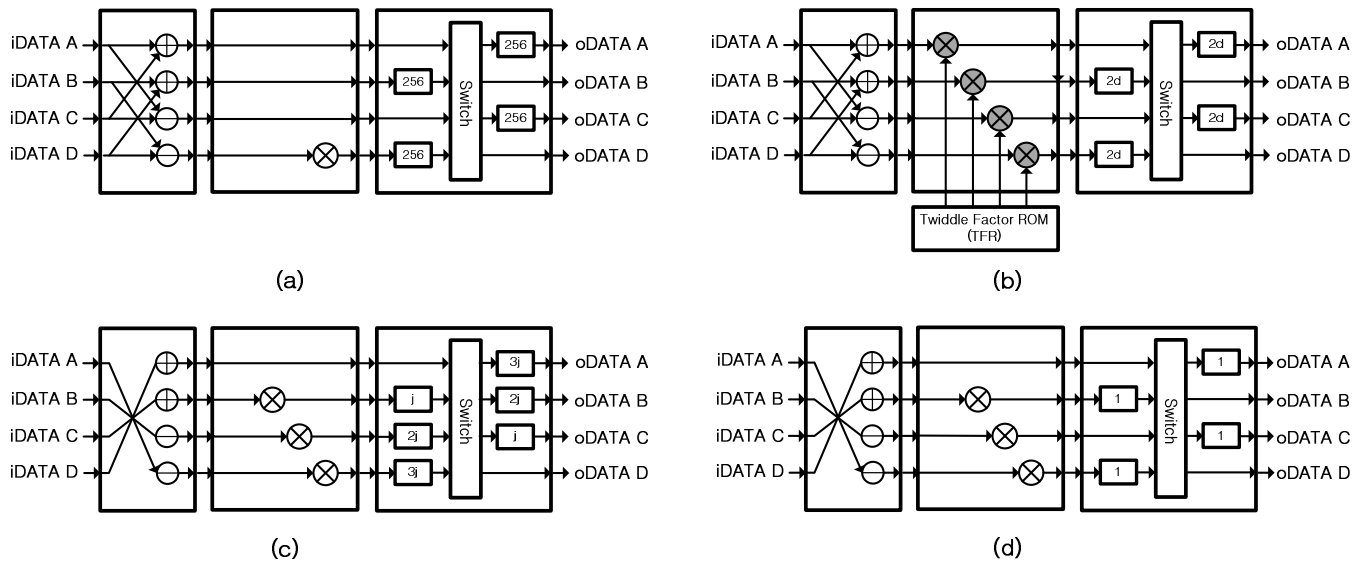


Figure 2. Block diagram of the proposed (a) R2BM1, (b) R2BM2, (c) R4BM1 and (d) R4BM2

V. CONCLUSION

In this paper, an area-efficient FFT processor is proposed for MIMO-OFDM based SDR systems. With the proposed MR-2/2/4/2/4/2/4/2 decomposition method and pipelined MDC architecture, the proposed FFT processor can support 4-channel 64/128/512/1024/2048-point FFT, which is needed for SDR system that can support both WLAN and WiMax modes. By using the proposed mixed radix algorithm, the required number of non-trivial multiplications is also minimized. FPGA-based implementation results show that the proposed FFT processor saves the logic slices, DSP48 (dedicated multiplier), and memory size by 41.9%, 62.5%, and 39%, respectively, compared with the conventional 4-channel R2SDF FFT processor. Compared with 4-channel R2MDC FFT processor, it can also save them by 26.2%, 25%, and 26.8%, respectively. Since FFT processor is one of the largest modules in MIMO-OFDM systems, the proposed FFT processor would greatly contribute to low-complexity implementation of MIMO-OFDM based SDR systems.

ACKNOWLEDGMENT

This work was supported by National Research Foundation of Korea Grant funded by the Korean Government (2009-0071983) and CAD tool was supported by IDEC.

REFERENCES

- [1] H. Sampath, S. Talwar, J. Tellado, V. Erceg, and A. Paulraj, "A fourth-generation MIMO-OFDM: broadband wireless system: Design, performance, and field trial results," *IEEE Commun. Mag.*, vol. 40, no. 9, pp. 143-149, Sept. 2002.
- [2] A. van Zelst, Tiim C. W. Schenk, "Implementation of a MIMO OFDM-Based wireless LAN system," *IEEE Trans. on Signal Processing*, vol. 52, no. 2, pp. 483-494, Feb. 2004.
- [3] G. L. Stuber, J. R. Barry, S. W. McLaughlin, Y. Li, M. A. Ingram, and T. H. Pratt, "Broadband MIMO-OFDM wireless communications," in *Proc. IEEE*, vol. 92, no. 2, pp. 271-297, Feb. 2004.
- [4] H. Bolcskei and E. Zurich, "MIMO-OFDM wireless systems: basics, perspectives, and challenges," *IEEE Trans. Wireless Commun.*, vol. 13, no. 4, pp. 31-37, Aug. 2006.
- [5] Y. W. Lin, C. Y. Lee, "design of an FFT/IFFT processor for MIMO OFDM systems," *IEEE Trans. on Circuits and Systems*, vol. 54, no. 4, pp. 807-815, Apr. 2007.
- [6] J. Mitola, "The Software Radio Architecture," *IEEE Commun. Mag.*, vol. 33, no. 5, pp. 26-38, May 1995.
- [7] C. K. Rauwerda, P. M. Heysters, and C. J. M. Smit, "Towards software defined radios using coarse-grained reconfigurable hardware," *IEEE Trans. on VLSI Systems*, vol. 16, no. 1, pp. 3-13, Jan. 2008.
- [8] T. Shono, Y. Shirato, H. Shiba, K. Uehara, K. Araki, and M. Umehira, "IEEE 802.11 wireless LAN implemented on software defined radio with hybrid programmable architecture," *IEEE Trans. on Wireless Commun.*, vol. 4, no. 5, pp. 2299-2308, Sep. 2005.
- [9] IEEE Std. 802.11n, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications," 2005.
- [10] IEEE Std. 802.11e, "Local and Metropolitan Area Networks-Part 16: Air Interface for Fixed Broadband Wireless Access System," Oct. 2004.
- [11] Petrus, P. et al. "An integrated draft 802.11n compliant MIMO baseband and MAC processor," in *Proc. of ISSCC'07*, pp. 266-268, Feb. 2007.
- [12] S. He and M. Torkelson, "A New Approach to Pipeline FFT Processor," *Parallel Processing Symposium*, pp. 766-770, Apr. 1996.
- [13] S. He and M. Torkelson, "Designing pipeline FFT processor for OFDM (de)modulation," in *Proc. IEEE URSI Int. Signals, Systems, and Electron.*, vol. 2, pp. 257-262, Oct. 1998.
- [14] L. R. Rabiner and B. Gold, "Theory and Application of Digital Signal Processing," *Prentice-Hall*, 1975.
- [15] T. Sansaloni, A. Perex-Pascual, V. Torres and J. Valls, "Efficient pipeline FFT processors for WLAN MIMO-OFDM systems," *Electronics Letters*, vol. 41, no. 19, pp. 1043-1044, Sep. 2005.
- [16] Y. Jung, H. Yoon, and J. Kim, "New efficient FFT algorithm and pipeline implementation results for OFDM/DMT applications," *IEEE Trans. Consum. Electron.*, vol. 49, no. 1, pp. 14-20, Feb. 2003.
- [17] S. Lee, Y. Jung, and J. Kim, "Low complexity pipeline FFT processor for MIMO-OFDM systems," *IEICE Electronics Express*, vol. 4, no. 23, pp. 750-754, Dec. 2007.