J Inf Process Syst, Vol.15, No.2, pp.374~385, April 2019
https://doi.org/10.3745/JIPS.01.0041

ISSN 1976-913X (Print)
ISSN 2092-805X (Electronic)

JOURNAL OF INFORMATION PROCESSING SYSTEMS JIPS

Optimization of Pipelined Discrete Wavelet Packet Transform Based on an Efficient Transpose Form and an Advanced Functional Sharing Technique

Hung-Ngoc Nguyen*, Cheol-Hong Kim**, and Jong-Myon Kim*

Abstract

This paper presents an optimal implementation of a Daubechies-based pipelined discrete wavelet packet transform (DWPT) processor using finite impulse response (FIR) filter banks. The feed-forward pipelined (FFP) architecture is exploited for implementation of the DWPT on the field-programmable gate array (FPGA). The proposed DWPT is based on an efficient transpose form structure, thereby reducing its computational complexity by half of the system. Moreover, the efficiency of the design is further improved by using a canonical-signed digit-based binary expression (CSDBE) and advanced functional sharing (AFS) methods. In this work, the AFS technique is proposed to optimize the convolution of FIR filter banks for DWPT decomposition, which reduces the hardware resource utilization by not requiring any embedded digital signal processing (DSP) blocks. The proposed AFS and CSDBE-based DWPT system is embedded on the Virtex-7 FPGA board for testing. The proposed design is implemented as an intellectual property (IP) logic core that can easily be integrated into DSP systems for sub-band analysis. The achieved results conclude that the proposed method is very efficient in improving hardware resource utilization while maintaining accuracy of the result of DWPT.

Keywords

AFS Technique, CSDBE, Daubechies, DWPT, FIR Filter, FPGA, Pipelined Architecture

1. Introduction

The wavelet transform (WT) plays an important role in signal processing applications, especially in decomposing signals into various sub-bands, feature analysis, modeling and reconstruction. The advantages in excellent locality of the WT in the time-frequency domain overcome the conventional transforms, such as the Fourier transform. Ever since the discrete wavelet transform (DWT) is introduced by Mallat [1], it is widely used in many different applications due to its multiresolution ability in signal analysis. The DWT has been successfully and extensively used in most fields, including image processing [2], audio and video compression [3], signal denoising [4], pattern recognition [5], biomedical applications [6,7], and adaptive filtering [8].

The discrete wavelet packet transform (DWPT) is a generalizable version of the DWT, which provides good resolution in the time–frequency domain. Due to its flexibility, the DWPT is also extensively used

Manuscript received June 21, 2017; first revision October 10, 2017; accepted October 31, 2017.
Corresponding Author: Jong-Myon Kim (jmkim07@ulsan.ac.kr)
* School of Electrical, Electronics and Computer Engineering, University of Ulsan, Ulsan, Korea (hungnguyenvldt@gmail.com, jongmyon.kim@gmail.com)
** School of Electronics and Computer Engineering, Chonnam National University, Gwangju, Korea (cheolhong@gmail.com)

in applications of image processing and video coding, digital communication systems, signal analysis and enhancement [9,10], and monitoring applications related to fault diagnosis in machinery [11-13], in which the DWPT is successfully employed to analyze the essential defect information in the non-stationary signals because of its decomposition ability to separate the signals into the low-frequency and high-frequency sub-bands. The DWPT decomposes more effective than the DWT for both high-frequency and low-frequency information from a signal; thereby, it obtains the most efficient and useful information description from the signal. The localization of time–frequency analysis in the short-time Fourier transform (STFT) is overcome via a discrete time–frequency representation of signals in the DWPT [14]. A disadvantage of the STFT is the usage of a fixed width window and hence requires trade-off between time and frequency decomposition. Thus, the WT has been successfully developed for analysis of signals in the time–frequency domain. The width of window function in the wavelet domain is adjusted appropriately so that the low-frequency components are offered by a larger frequency resolution, whereas a larger time resolution provides for the high-frequency components of the signal. However, the DWT only offers non-uniform frequency sub-band decomposition of input signal [4]. This shortcoming can be solved by using the DWPT, which provides an uniform frequency sub-band resolution for signal analysis; therefore, it is more popular used in harmonics estimation applications [10].

As mentioned above, the DWPT performs better than DWT. Many hardware solutions have been performed in order to provide efficient implementations for the DWT [15]. However, there are not many solutions for the DWPT. The development of specific architectures for the DWPT has attracted widespread attention by experts from various fields. Several of the reported hardware methods for the WT claimed satisfactory performance, but their hardware implementations still suffer from many problems of high complexity and extensive hardware resource requirements. Therefore, the design of an efficient DWPT processor is a matter of great significance. With fast advances in the technology, it is possible to implement an integrated circuit for specific applications. The DWPT is mainly implemented using technologies, such as digital signal processors (DSPs), application-specific integrated circuits (ASICs), and field-programmable gate arrays (FPGAs) [4,10,12]. An FPGA is notably useful for implementing DWPT processors. It is a programmable logic device which offers more flexibility, requires less design time; also has cost less than ASICs or DSPs. Moreover, the increasing gate density of FPGAs in recent years has enabled researchers to exploit the inherent parallelism in signal processing algorithms by implementing massively parallel architectures. Furthermore, the computer-aided design software has been significantly increased to develop FPGA-based designs that allow the complicated systems to be implemented with an easy and efficient solution.

In this paper, we propose an efficient pipeline architecture for the Daubechies mother wavelet function-based DWPT processor and present an implementation of this architecture on an FPGA platform. The proposed architecture is designed as a DWPT IP logic core, which can be integrated into DSP systems for time-frequency signal analysis. The proposed DWPT is realized by using finite impulse response (FIR) filter banks that comprise two independent FIR filters of high-pass and low-pass. The signal decomposition in the DWPT is based on the convolution between the filter's coefficients and the input signal. FIR filtering followed by down-sampling is the traditional method of the direct form for implementing the DWPT. Unlike the traditional method, we present an efficient transpose form structure in which down-sampling is performed before filtering operations. Moreover, the feed-forward pipelined (FFP) architecture is exploited for implementation of the DWPT to make all pipelined stages of the architecture occurring simultaneously; thereby, the system performance is significantly improved in a

manner suitable for real-time computations [16,17]. The canonical-signed digit-based binary expression (CSDBE) algorithm is also used to optimize hardware resources for convolution-based FIR filter banks in DWPT decomposition through the shift-add method. In addition, the advanced functional sharing (AFS) technique is proposed for more efficient implementation of the DWPT, allowing for the saving of hardware resources while still ensuring accuracy of the system. The main drawbacks of the existing designs are that the convolution of the filter coefficients is usually handled by embedded, dedicated DSP blocks, and the intermediate coefficients are stored using on-chip memory, which involves extremely memory access during computation of system; these issues lead to a large of resources and area used and significant power dissipation. The proposed DWPT processor for signal decomposition through convolution-based FIR filter pairs uses only shifters and adders instead of embedded dedicated blocks, making it appropriate for further researches as a dedicated ASIC chip or VLSI design. The whole proposed design is verified on the Virtex-7 FPGA using Verilog HDL programming. The performance of the design is validated using an experimental test signal in a hardware simulation environment. This work is an extended version of our work initially presented in [18]. We present an explicit and detailed analysis of the previous work and the extended results of an optimal implementation of the proposed AFS and CSDBE-based pipelined DWPT processor. Efficiency of the proposed design is enhanced via the combination of these two optimization techniques.

The rest of this paper is organized as follows. A brief summary of the DWPT is given in Section 2. Section 3 discusses the proposed Daubechies mother wavelet function-based pipelined DWPT processor and the use of the CSDBE and AFS techniques to optimize convolution operations in the proposed architecture. The achieved results and comparison of the proposed design with traditional designs are presented in Section 4. Finally, Section 5 concludes the paper.

2. Discrete Wavelet Packet Transform

The DWPT offers a time–frequency domain representation for the analysis of signals, which is often implemented by convolution via multistage FIR filter banks. The DWPT is better than DWT due to processing ability for the outputs of the low-pass filter and the high-pass filter simultaneously at the next level. It is more flexibility to generate distinctive decomposition of the input signal on the time-frequency domain.

The general scheme of a DWPT for three-level decomposition is based on FIR filter banks which provide eight frequency bands, as presented in Fig. 1. The filter bank consists of wavelet functions for the low-pass, $H(z)$, and high-pass, $G(z)$. The input signal $x(n)$ is transferred to their parallel low-pass and high-pass filters. The output of the low-pass $y_{low}(n)$ provides approximation coefficients and those of the high-pass $y_{high}(n)$ provides detailed coefficients. Thereby, the detail and approximation coefficients at each resolution level j are analyzed by wavelet filters into new coefficients for the subsequent level. The wavelet coefficients in the next level are obtained from convoluting of the input signal with the filter's coefficients in the previous level and then down-sampling by a factor of two, defined as follows:

$$y_{low}(n) = \sum_{k=0}^{N-1} h(k).x(2n-k) \tag{1}$$

$$y_{high}(n) = \sum_{k=0}^{N-1} g(k).x(2n-k)\,, \tag{2}$$

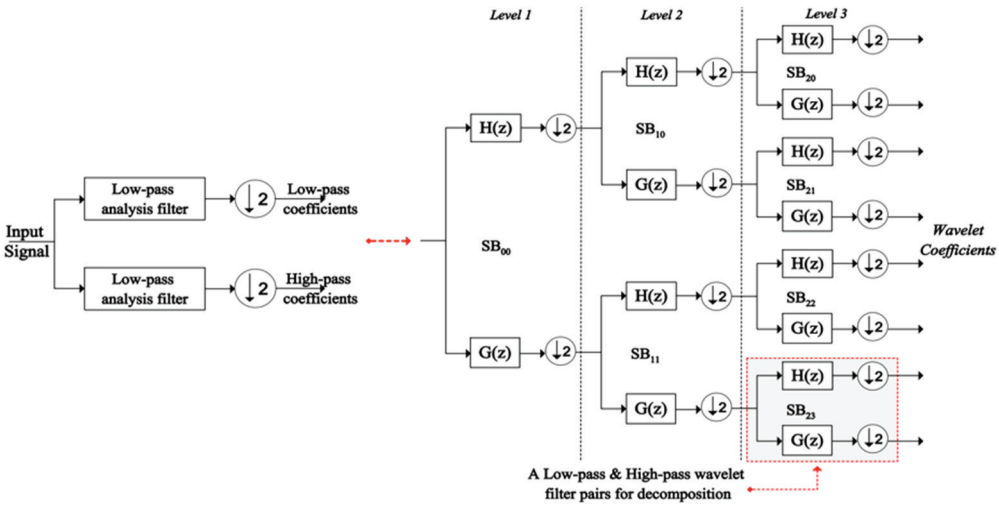


Fig. 1. The DWPT for three-level decomposition based on the FIR filter banks.

where $h[k]$ and $g[k]$ are the impulse responses of the transfer functions $H(z)$ and $G(z)$ of filters, respectively.

At each decomposition level j, there are 2^j ($j = 0, 1, \ldots,$ level−1) wavelet filter pairs of low-pass and high-pass. They decompose the previous input signals into two sub-bands at the next level, the number of sub-bands is indicated by SB_{ji} ($i = 0, 1, \ldots, 2^j - 1$). An assumption is that the transfer function of a FIR filter is presented as follows:

$$H(z) = h_0 + h_1 z^{-1} + h_2 z^{-2} + \ldots + h_{(N-2)} z^{-(N-2)} + h_{(N-1)} z^{-(N-1)} . \tag{3}$$

The Eq. (3) should be arranged as a sum of the coefficients having even and odd indexes, which can be shown as follows:

$$\begin{aligned} H(z) = {} & h_0 + h_2 z^{-2} + \ldots + h_{(N-4)} z^{-(N-4)} + h_{(N-2)} z^{-(N-2)} \\ & + z^{-1} (h_1 + h_3 z^{-2} + \ldots + h_{(N-3)} z^{-(N-4)} + h_{(N-1)} z^{-(N-2)}) \end{aligned} . \tag{4}$$

Thus, Eq. (4) can be rewritten by combining of two even $H_e(z)$ and odd $H_o(z)$ component filters, separately, as follows:

$$H(z) = H_e(z) + z^{-1} . H_o(z) , \tag{5}$$

with,

$$\begin{aligned} H_e(z) &= h_0 + h_2 z^{-1} + \ldots + h_{(N-4)} z^{-(N-4)/2} + h_{(N-2)} z^{-(N-2)/2} \\ H_o(z) &= h_1 + h_3 z^{-1} + \ldots + h_{(N-3)} z^{-(N-4)/2} + h_{(N-1)} z^{-(N-2)/2} \end{aligned} .$$

It is possible to similarly implement of the FIR filter, as shown in Fig. 2. Both filters in a low-pass and high-pass pair for DWPT decomposition concurrently; together, the pair is referred to as a wavelet filter process element (WFPE). The WFPE is fed the same input sample $x(n)$ at the same time. Operation of the parallel filters takes place in the architecture with the similar delay and produces the output data concurrently. The previous output samples are forwarded to the buffer memory of the next stage and the procedure of computation is continued until completing the DWPT decomposition tree.

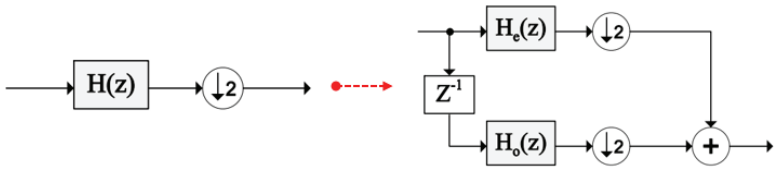


Fig. 2. FIR wavelet filter with the transfer function $H(z)$.

3. Hardware Implementation of the Daubechies-based Pipelined DWPT Processor

The FIR filter-based DWPT processor using the efficient pipelined architecture with five-level full decomposition is considered in this study. The Daubechies-2 (Db2) mother wavelet function is chosen as the primary filter to be implemented in order to verify the efficiency of the proposed design. The FFP architecture is exploited for pipelined DWPT implementation, as shown in Fig. 3. Each pipelined stage corresponds to one decomposition level. Each WFPE in the FFP architecture consists of a wavelet filter pair and a down-sampling operation that down-samples by a factor of two.

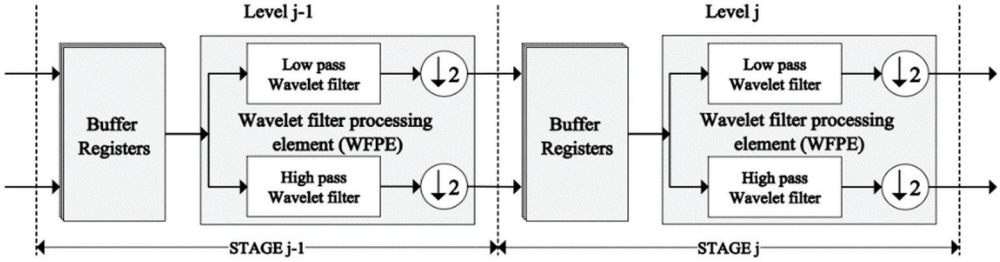


Fig. 3. FFP architecture for DWPT implementation.

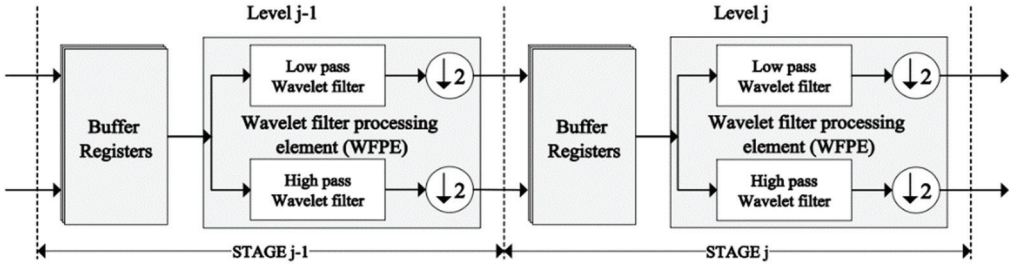
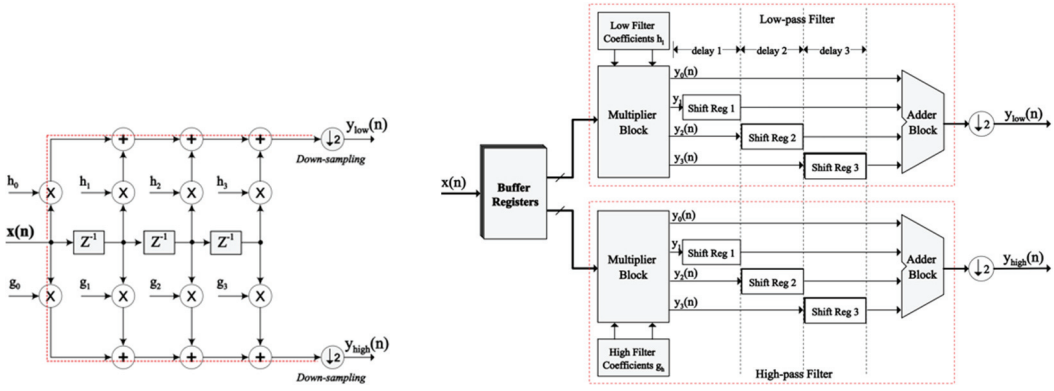


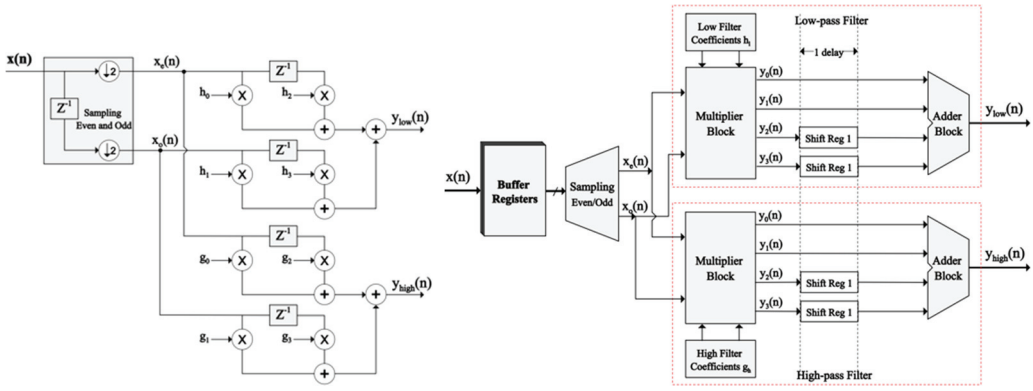
Fig. 4. Proposed pipelined architecture for a five-level DWPT processor.

A general block diagram for the proposed five-level pipelined DWPT processor is shown in Fig. 4. The buffer registers represent memories that store intermediate coefficients, and the WFPEs compute the wavelet coefficients through convolution in FIR filters at each level. The central control unit controls read and write access to the memories and generates controlling signals to manage the function of the multipliers and adders in the WFPE processor. To control the read and write access to the memories, the control unit uses an address generation circuit and a state counter to determine the appropriate periods for data access at each stage in the pipelined architecture.



(a) (b)

Fig. 5. Hardware implementation of DWPT using a Db2-based FIR filter in traditional direct form: (a) convolution of the Db2-based FIR filter in direct form and (b) corresponding hardware architecture.



(a) (b)

Fig. 6. Hardware implementation of DWPT using a Db2-based FIR filter based on transpose form: (a) convolution of the Db2-based FIR filter in this form and (b) corresponding hardware architecture.

3.1 Transpose Form Structure for DWPT

It can be seen that for the traditional direct form of the DWPT implementation, half of the mathematical computations are significantly wasted, as down-sampling in the WFPE cell eliminates a half of the filtered samples. This disadvantage is overcome by using the transpose form structure in which the down-sampling is placed before filtering to avoid the wasted operations in the WFPE. The traditional direct form of the DWPT using the Db2-based FIR filter with four coefficients each for low-pass and high-pass is illustrated in Fig. 5(a), and the hardware architecture of the convolution is shown in Fig. 5(b). In the transpose form, the convolutions of filter's coefficients with input data are carried out right after down-sampling, as shown in Fig. 6(a). The corresponding hardware architecture is presented in Fig. 6(b) and result in reducing computation time by half. For each resolution level, the output is now expressed as follows:

$$y_{low}(n) = \sum_{k=0}^{\frac{N}{2}-1} h_e(k).x_e(n-k) + \sum_{k=0}^{\frac{N}{2}-1} h_o(k).x_o(n-k-1), \quad (6)$$

$$y_{high}(n) = \sum_{k=0}^{\frac{N}{2}-1} g_e(k).x_e(n-k) + \sum_{k=0}^{\frac{N}{2}-1} g_o(k).x_o(n-k-1) \ . \quad (7)$$

Here, the $x_e(n)$ and $x_o(n)$ are the even and odd samples of input signal $x(n)$, respectively. The low-pass filter, $H(z)$, is illustrated by even $h_e(k)$ and odd $h_o(k)$ coefficients and also similar for high-pass filter, $G(z)$, by $g_e(k)$ and $g_o(k)$.

3.2 CSDBE-based Optimization of Convolution in the FIR Filter

The shift-add method is efficiently employed for convolution in the filter. The number of non-zero binary bits in the filter coefficients determines the number of shift and add operations in architecture. The CSDBE is an efficient solution to give a coefficient's binary representation that has the least number of non-consecutive non-zero bits and hence requires the fewest shifters and adders for convolution. The CSDBE algorithm is presented as follows:

Step 1: dem = count of the number of "1" bits in a binary sequence
Step 2: If dem ≥ 2, then replace the sequence with 10…0$\underline{1}$, where 1: +1 and $\underline{1}$: – 1
11 → 10$\underline{1}$; 111 → 100$\underline{1}$; 1101111 → 10$\underline{1}$1000$\underline{1}$
Step 3: Check and replace the corresponding value pairs: 1$\underline{1}$→ 01; $\underline{1}$1→ 0$\underline{1}$; $\underline{11}$→ $\underline{1}$01

As mentioned above, the Db2-based FIR filter is used for the time-frequency analysis in this paper, with h_l = [−0.1294, 0.2241, 0.8365, 0.4830] and g_h = [−0.4830, 0.8356, −0.2241, −0.1294] as filter's coefficients for the low-pass and high-pass decompositions, respectively. The used CSDBE can reduce the number of shifters and adders needed for a convolution operation, thereby reducing its area and enhancing performance. A comparison of the 16 bit 2's complement and a CSDBE representation of the low and high-pass filter coefficients is given in Table 1. The CSDBE-based optimization requires fewer resources than a conventional expression.

Table 1. Representation of the Db2-based wavelet filter coefficients for convolution in DWPT

	Filter coefficient	2's Complement representation	CSDBE representation
$H(z)$ low-pass	-0.1294	1.110111101110000	0.00$\underline{1}$0000$\underline{1}$00$\underline{1}$0000
	0.2241	0.001110010101111	0.0100$\underline{1}$0$\underline{1}$0$\underline{1}$0$\underline{1}$000$\underline{1}$
	0.8365	0.110101100010010	1.00$\underline{1}$0$\underline{1}$0$\underline{1}$00010010
	0.4830	0.011110111010010	0.10000$\underline{1}$00$\underline{1}$010010
$G(z)$ high-pass	-0.4830	1.100001000101110	0.$\underline{1}$0000$\underline{1}$0010$\underline{1}$00$\underline{1}$0
	0.8365	0.110101100010010	1.00$\underline{1}$0$\underline{1}$0$\underline{1}$00010010
	-0.2241	1.110001101010001	0.0$\underline{1}$00$\underline{1}$0$\underline{1}$0$\underline{1}$010001
	-0.1294	1.110111101110000	0.00$\underline{1}$0000$\underline{1}$00$\underline{1}$0000

3.3 AFS Technique for Hardware Implementation of DWPT

In this study, the input signal is convoluted by many coefficients at the same time. An AFS technique is proposed, which allows further savings in hardware resource utilization for the convolution operation.

The AFS reuses the same functional operators (i.e., taking advantage of the shifters and adders similar to the architecture) for saving hardware resources on a chip. A convolution of input samples and h_l coefficients can be expressed in a CSDBE form, as follows:

$Y_0 = X * (-0.1294) = X * 0.00\underline{1}0000\underline{1}00\underline{1}0000_{CSDBE}$

$Y_1 = X * 0.2241 = X * 0.0100\underline{1}010\underline{1}0\underline{1}0001_{CSDBE}$

$Y_2 = X * 0.8365 = X * 1.00\underline{1}0\underline{1}0\underline{1}00010010_{CSDBE}$

$Y_3 = X * 0.4830 = X * 0.10000\underline{1}00\underline{1}010010_{CSDBE.}$

By setting $C_0 = X >> 4$, $C_1 = X >> 3$, $C_2 = X >> 2$, $C_3 = X >> 1$, and $C_4 = X >> 0$, we have $B_0 = X * 0.1001_{CSDBE} = C_0 + C_3$, $B1 = X * 0.101_{CSDBE} = C_1 + C_3$, $B_2 = X * 0.\underline{1}01_{CSDBE} = C_1 - C_3$, and $B_3 = X * 1.00\underline{1}_{CSDBE} = C_4 - C_1$. We can get the following results by combining the above expressions:

$Y_0 = -C1 - B_0 >> 8,$

$Y_1 = C_2 + B_2 >> 5 - B_1 >> 9,$

$Y_2 = B_3 - B_1 >> 5 + B_0 >> 11,$

$Y_3 = C_3 - B_0 >> 6 + B_0 >> 11.$

The output of the convolution using the shift-add method is $Y_{low}(n) = Y_0 + Y_1 >> 1 + Y_2 >> 2 + Y_3 >> 3$, as shown in Eq. (1). A hardware implementation of the AFS technique is presented in Fig. 7. In this case, a hardware implementation of the AFS technique for the FIR filter design can be synthesized with only 11 adders, showing very high resource savings. The proposed design is evaluated by comparing its efficiency to others in the same architecture. The performance of the system is analyzed for a 1024-point DWPT computation in the next section.

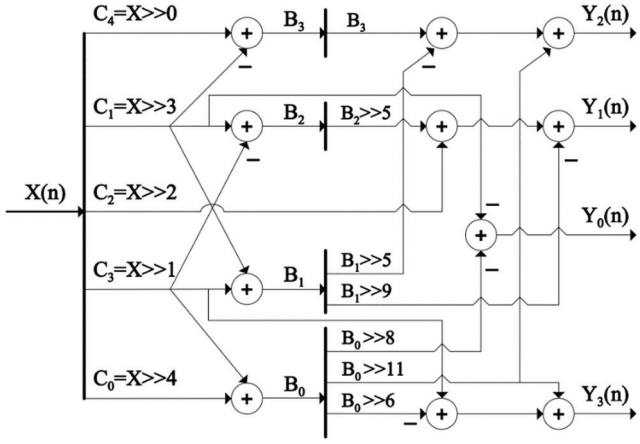


Fig. 7. Implementation of the Db2-based low-pass FIR filter using the AFS technique.

4. The Experimental Results

The proposed pipelined DWPT processor for 1024-point computation with five-level decomposition and the aforementioned designs are implemented on a Virtex 7 XC7VX485T FPGA board using the Xilinx Vivado Design Suite (XVDS) tool for testing function, timing simulation, and design synthesis.

The Verilog HDL code is used for programming the high-level description of the designs. The process of implementation is simulated in a hardware environment and its obtained results are compared with MATLAB to confirm the accuracy of the system. Consider the case of 1024-point data representing a frame of an acoustic emission (AE) signal sampled at 1 MHz. Since five-level DWPT is used, there are 32 sub-band analyses at the output. The accuracy of the design is measured via the mean squared error (MSE) values, and results obtained via MATLAB are used as a baseline for comparison. The average MSE value is about 10^{-5}, which is very small. This indicates that the proposed design can provide high-accuracy of DWPT processing.

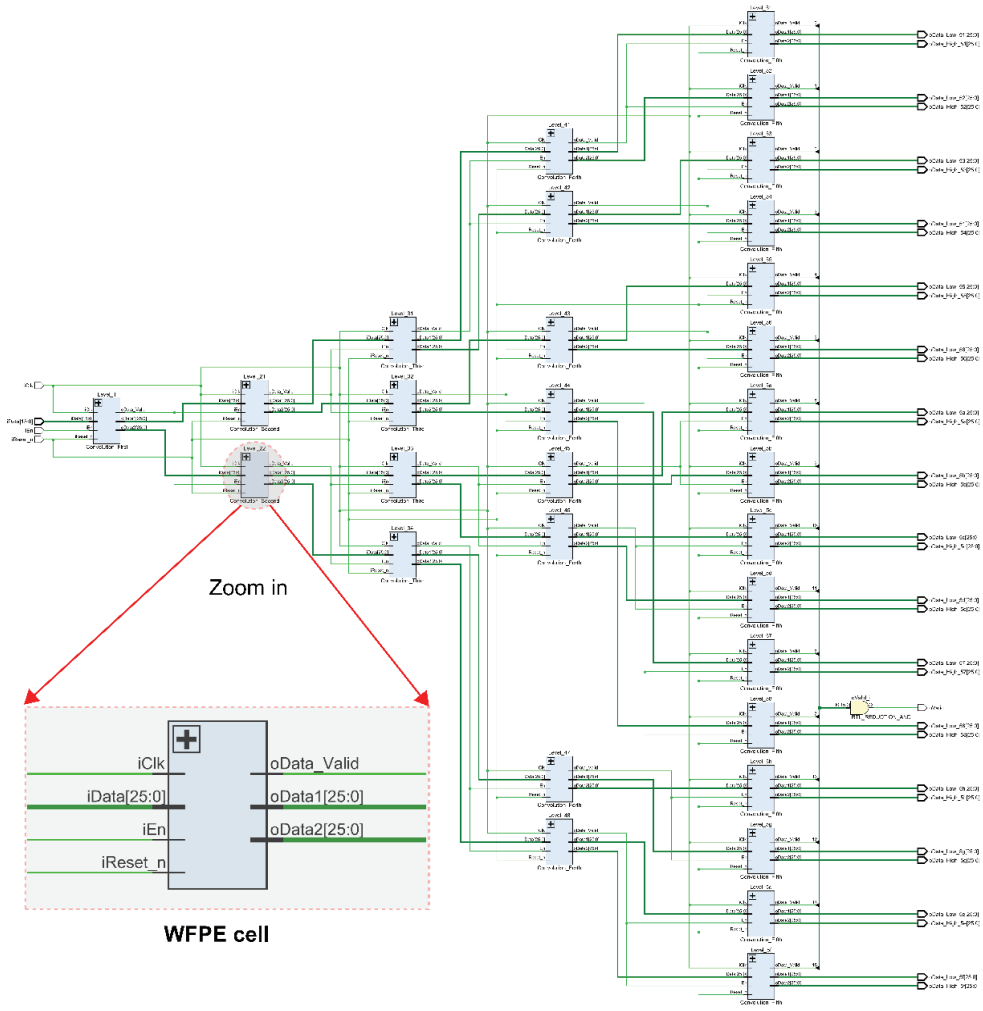


Fig. 8. The schematic RTL in the gate-level of the five-level pipelined DWPT processor.

The proposed Db2-based DWPT core IP is efficiently designed by employing the FFP architecture and advantage of the CSDBE and AFS algorithms. The data streams are represented in a signed fixed-point format using 16-bit word length for the in/out data and 10-bit precision (i.e., 10 fractional bits) used for internal computation process of the system. A schematic RTL in the gate-level of the five-level pipelined DWPT processor using a Db2-based FIR filter is presented in Fig. 8. At each decomposition level j (j = 0,

1, …, level−1), there are 2^j WFPE cells for sub-band analysis. The synthesized result on hardware for a WFPE cell at first decomposition level of DWPT processor is shown in Fig. 9. The WFPE cells are based on the efficient transpose form structure, which improves the memory resource utilization for hardware implementation.

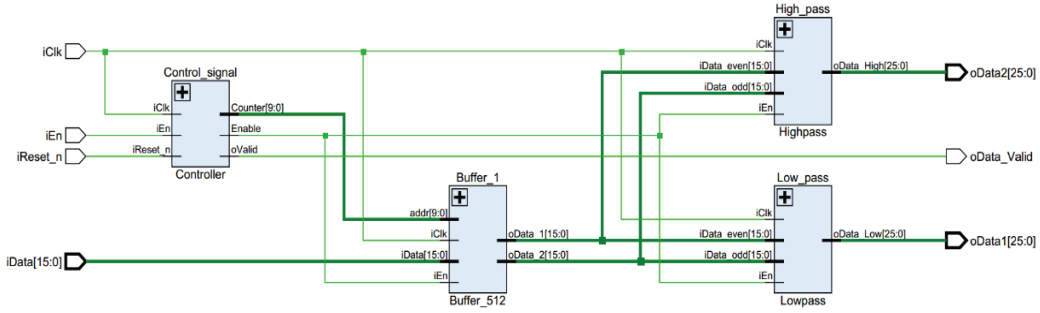


Fig. 9. The schematic RTL of a WFPE cell at the first decomposition level.

Table 2 summarizes the resource utilization of the above architecture and compares its hardware complexity to that of traditional designs. The traditional architecture usually uses more distributed logic resources such as flip-flops (FFs), look-up tables (LUTs), memory LUTs, block RAMs, and block DSPs. The proposed AFS and CSDBE-based DWPT processor uses fewer resources and does not require any embedded dedicated DSP blocks. Convolution in Db2-based FIR filters for DWPT decomposition only requires configurable logic blocks (CLBs) and distributed memory, further reducing the logic resources on the FPGA chip and avoiding the need for DSP blocks. Overall, the proposed design employing a combination of the CSDBE and AFS techniques achieves better hardware resource utilization compared with conventional designs.

Table 2. Hardware synthesis results and evaluation of the proposed DWPT design

Resource utilization	(A)	(B)		(C)		Available
		# of slices	% Savings	# of slices	% Savings	
# of registers (CLB flip-flops)	6,441	3,328	48.33	3,319	48.47	607,200
# of LUTs	42,242	26,866	36.39	18,903	55.25	303,600
# of memory LUTs	7,400	4,840	34.59	4,840	34.59	130,800
# of block RAMs	4	2	50	2	50	1,030
# of block DSPs	8	0	100	0	100	2,800

(A) = the traditional design usually using a lot of distributed logic resources, (B) = the proposed design based on the CSDBE algorithm, (C) = the proposed design based on a combination of the CSDBE and AFS techniques.

5. Conclusion

In this paper, we presented the Db2 mother wavelet function-based efficient implementation of a five-level pipelined DWPT processor using FIR filter banks. The proposed AFS and CSDBE-based DWPT processor was verified on the Virtex-7 FPGA board using the XVDS tool. This optimized design is based on an efficient transpose form structure, thereby reducing its computational complexity by half, while

also achieving significant savings in hardware resources for its FPGA implementation. The proposed design successfully exploited the FFP architecture and enhanced the performance by employing both the CSDBE algorithm and the AFS technique. Experimental results showed that the proposed design achieves better hardware resource utilization compared to the conventional designs, while maintaining high accuracy of the result of DWPT.

Acknowledgement

This research was support by The Leading Human Resource Training Programof Regional Neo Industry through the National Research Foundation of Korea funded by the Ministry of Science, ICT, and future Planning (No. NRF-2016H1D5A1910564). It was alsofunded in part by the Korea Institute of Energy Technology Evaluation and Planning (KETEP) and by the Ministry of Trade, Industry, & Energy (MOTIE) of the Republic of Korea (No. 20172510102130).

References

[1] S. Mallat, *A Wavelet Tour of Signal Processing*. San Diego, CA: Academic Press, 1999.

[2] C. Anibou, M. N. Saidi, and D. Aboutajdine, "Classification of textured images based on discrete wavelet transform and information fusion," *Journal of Information Processing Systems*, vol. 11, no. 3, pp. 421-437, 2015.

[3] D. Wang, F. Yang, and H. Zhang, "Blind color image watermarking based on DWT and LU decomposition," *Journal of Information Processing Systems*, vol. 12, no. 4, pp. 765-778, 2016.

[4] M. Bahoura and H. Ezzaidi, "FPGA-implementation of discrete wavelet transform with application to signal denoising," *Circuits, Systems, and Signal Processing*, vol. 31, no. 3, pp. 987-1015, 2012.

[5] M. Satone and G. K. Kharate, "Face recognition based on PCA on wavelet subband of average-half-face," *Journal of Information Processing Systems*, vol. 8, no. 3, pp. 483-494, 2012.

[6] J. Agarwal and S. S. Bedi, "Implementation of hybrid image fusion technique for feature enhancement in medical diagnosis," *Human-centric Computing and Information Sciences*, vol. 5, article no. 3, 2015.

[7] S. Ardhapurkar, R. Manthalkar, and S. Gajre, "ECG denoising by modeling wavelet sub-band coefficients using kernel density estimation," *Journal of Information Processing Systems*, vol. 8, no. 4, pp. 669-684, 2012.

[8] S. Lalani and D. Doye, "Discrete wavelet transform and a singular value decomposition technique for watermarking based on an adaptive fuzzy inference system," *Journal of Information Processing Systems*, vol. 13, no. 2, pp. 340-347, 2017.

[9] C. Wang, J. Zhou, L. Liao, J. Lan, J. Luo, X. Liu, and M. Je, "Near-threshold energy-and area-efficient reconfigurable DWPT/DWT processor for healthcare-monitoring applications," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 62, no. 1, pp. 70-74, 2015.

[10] V. K. Tiwari and S. K. Jain, "Hardware implementation of polyphase-decomposition-based wavelet filters for power system harmonics estimation," *IEEE Transactions on Instrumentation and Measurement*, vol. 65, no. 7, pp. 1585-1595, 2016.

[11] R. Yan, R. X. Gao, and X. Chen, "Wavelets for fault diagnosis of rotary machines: a review with applications," *Signal Processing*, vol. 96(Part A), pp. 1-15, 2014.

[12] M. Kang, J. Kim, and J. M. Kim, "An FPGA-based multicore system for real-time bearing fault diagnosis using ultrasampling rate AE signals," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 4, pp. 2319-2329, 2015.

[13] J. Uddin, R. Islam, and J. M. Kim, "Texture feature extraction techniques for fault diagnosis of induction motors," *Journal of Convergence*, vol. 5, no. 2, pp. 15-20, 2014.
[14] G. Strang and T. Nguyen, *Wavelets and Filter Banks*. Wellesley, MA: Wellesley-Cambridge Press, 1996.
[15] C. H. Hsia, J. H. Yang, and W. Wang, "An efficient VLSI architecture for discrete wavelet transform," in *Proceedings of 2015 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA)*, Hong Kong, China, 2015, pp. 684-687.
[16] M. Chehaitly, M. Tabaa, F. Monteiro, and A. Dandache, "A fast and configurable architecture for discrete wavelet packet transform," in *Proceedings of 2015 Conference on Design of Circuits and Integrated Systems (DCIS)*, Estoril, Portugal, 2015, pp. 1-6.
[17] M. Bahoura and H. Ezzaidi, "Pipelined architecture for discrete wavelet transform implementation on FPGA," in *Proceedings of 2010 International Conference on Microelectronics*, Cairo, Egypt, 2010, pp. 459-462.
[18] H. N. Nguyen, C. H. Kim, and J. M. Kim, "Efficient Daubechies-based pipelined discrete wavelet package transform for sub-band analysis using advanced functional sharing," in *Proceedings of the 11th International Conference on Multimedia and Ubiquitous Engineering (MUE)*, 2017, Seoul, Korea.

Hung-Ngoc Nguyen https://orcid.org/0000-0001-9416-5137

He received his B.S. and M.Sc. degrees in Electronics, Telecommunications, and Computer Engineering in 2007 and 2011, respectively, from the University of Science in Ho Chi Minh City, Vietnam. He is currently a PhD student of Electrical, Electronics, and Computer Engineering, University of Ulsan, Ulsan, Korea. His research interests include hardware implementation of efficient signal processing algorithms on FPGAs and embedded system design and multi-core architecture.

Cheol-Hong Kim https://orcid.org/0000-0003-1837-6631

He received his B.S., M.S., and Ph.D. degrees in Computer Engineering from Seoul National University, Seoul, Korea, in 1998, 2000, and 2006, respectively. He is an Associate Professor of Electronics and Computer Engineering at Chonnam National University, Gwangju, Korea. His research interests include multi-core architecture and embedded systems.

Jong-Myon Kim https://orcid.org/0000-0002-5185-1062

He received a B.S. in Electrical Engineering from Myongji University in Yongin, Korea, in 1995, an M.S. in Electrical and Computer Engineering from the University of Florida in Gainesville, FL, USA, in 2000, and a Ph.D. in Electrical and Computer Engineering from the Georgia Institute of Technology in Atlanta, GA, USA, in 2005. He is a Professor of IT Convergence at the University of Ulsan, Korea. His research interests include multimedia processing, digital watermarking, multimedia specific processor architecture, parallel processing, and embedded systems. He is a member of IEEE and the IEEE Computer Society.