Journal of Southeast University (English Edition) Vol. 31, No. 1, pp. 19 - 24 Mar. 2015 ISSN 1003—7985

A high-throughput VLSI design for JPEG2000 9/7 discrete wavelet transform

Wang Jianxin Zhu En

(School of Information Science and Engineering, Southeast University, Nanjing 210096, China)

Abstract: To achieve high parallel computation of discrete wavelet transform (DWT) in JPEG2000, a high-throughput two-dimensional (2D) 9/7 DWT very large scale integration (VLSI) design is proposed, in which the row processor is based on flipping structure. Due to the difference of the input data flow, the column processor is obtained by adding the input selector and data buffer to the row processor. Normalization steps in row and column DWT are combined to reduce the number of multipliers, and the rationality is verified. By rearranging the output of four-line row DWT with a multiplexer (MUX), the amount of data processed by each column processor becomes half, and the four-input/four-output architecture is implemented. For an image with the size of $N \times N$, the computing time of one-level 2D 9/7 DWT is $0.25N^2 + 1.5N$ clock cycles. The critical path delay is one multiplier delay, and only $5N$ internal memory is required. The results of post-route simulation on FPGA show that clock frequency reaches 136 MHz, and the throughput is 544 Msample/s, which satisfies the requirements of high-speed applications.

Key words: JPEG2000; flipping structure; 2D discrete wavelet transform (DWT); 9/7 DWT; very large scale integration (VLSI)

doi: 10.3969/j.issn.1003-7985.2015.01.004

Discrete wavelet transform (DWT) can decompose signals into dyadic tree-structured subbands with multi-resolution structure, which is widely used in digital signal processing, image analysis, communication, and image compression[1]. The international still image coding standard JPEG2000 adopts DWT as the former data transform, which obtains high compression quality at a low rate. Generally, JPEG2000 adopts two-dimensional (2D) 9/7 DWT for lossy mode.

Convolution-based architectures were first used in DWT VLSI design. Several classic structures[2-5] were proposed for one-dimensional (1D) DWT, and Zervas et al.[6] proposed a 2D structure. However, the main drawbacks of convolution-based architectures are high computation and a large memory cost. To reduce the computation, lifting-based DWT was proposed by Sweldens et al.[7-8] and adopted in JPEG2000 standard. Huang et al.[9] proposed an optimized architecture of lifting-based DWT, called flipping structure, which extracts a proper constant in each computing unit of polyphase matrix of lifting-based DWT. Compared with lifting structure, flipping structure improves peak signal to noise ratio (PSNR) by more than 40% for multilevel DWT, obtains a shorter critical path, and costs less memory. To gain a higher throughput, Tian et al.[10] proposed a multi-input/multi-output (MIMO) structure for common use. However, its basic computing unit adopts the lifting-based structure, which limited the performance. Wu et al.[11] designed a pipeline architecture for JPEG2000 by merging the predictor and updater into one single step. However, it only has one-input/one-output throughput, leading to a high computing time.

We propose a four-input/four-output 2D 9/7 DWT VLSI design for JPEG2000, in which the row and column processors are based on flipping structure. Also, the normalization processes are combined to save multipliers and computing time. Experimental results show that the proposed architecture has a great performance in critical path, computing time, and memory cost.

1 Flipping Structure for 1D 9/7 DWT

The basic 1D 9/7 filter convolution can be expressed as follows:

$$\left.\begin{aligned} y_{\mathrm{l}} &= \sum_{i=0}^{8} h(i)x(2n-i) \\ y_{\mathrm{h}} &= \sum_{i=0}^{6} g(i)x(2n-i) \end{aligned}\right\} \tag{1}$$

where $h(z)$ and $g(z)$ are lowpass and highpass analyzer filters, respectively; $x(n)$ is the original sequence; y_{l} and y_{h} are the outputs of lowpass and highpass filters, respectively.

$$\left.\begin{aligned} h(z) &= h_{\mathrm{e}}(z^2) + z^{-1}h_{\mathrm{o}}(z^2) \\ g(z) &= g_{\mathrm{e}}(z^2) + z^{-1}g_{\mathrm{o}}(z^2) \end{aligned}\right\} \tag{2}$$

$$\boldsymbol{P}(z) = \begin{bmatrix} h_{\mathrm{e}}(z) & g_{\mathrm{e}}(z) \\ h_{\mathrm{o}}(z) & g_{\mathrm{o}}(z) \end{bmatrix} = \begin{bmatrix} 1 & \alpha(1+z^{-1}) \\ 0 & 1 \end{bmatrix}.$$

Received 2014-08-25.

Biographies: Wang Jianxin (1989—), male, graduate; Zhu En (corresponding author), male, doctor, professor, zhuenpro@seu.edu.cn.

Foundation item: The National Science and Technology Major Project of the Ministry of Science and Technology of China (No. 2014ZX03003007-009).

Citation: Wang Jianxin, Zhu En. A high-throughput VLSI design for JPEG2000 9/7 discrete wavelet transform[J]. Journal of Southeast University (English Edition), 2015, 31(1): 19 - 24. [doi: 10.3969/j.issn.1003-7985.2015.01.004]

$$\begin{bmatrix} 1 & 0 \\ \beta(1+z) & 1 \end{bmatrix}\begin{bmatrix} 1 & \gamma(1+z^{-1}) \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 \\ \delta(1+z) & 1 \end{bmatrix}\begin{bmatrix} K & 0 \\ 0 & \dfrac{1}{K} \end{bmatrix} \tag{3}$$

The filter bank $h(z)$ and $g(z)$ can be expressed with a polyphase matrix $\boldsymbol{P}(z)$. Since any filter bank of perfect reconstruction can be implemented by lifting structure[8], $\boldsymbol{P}(z)$ can be decomposed into several computing units, as expressed in Eq. (2) and Eq. (3). Lifting-based DWT is based on Eq. (3). Then Eq. (1) can be expressed as

$$[y_l \quad y_h] = [x_e \quad x_o]\boldsymbol{P}(z) \tag{4}$$

where x_e and x_o are the original data in even and odd positions, respectively. According to Eq. (3) and Eq. (4), the original data are separated into even and odd parts. The rule of the interactive calculation between the two parts is determined by $\boldsymbol{P}(z)$.

Inversing the multiplier coefficient in each computing unit of $\boldsymbol{P}(z)$, we obtain $\boldsymbol{P}(z)$ for flipping-based DWT as follows:

$$\begin{aligned} \boldsymbol{P}(z) = &\begin{bmatrix} \dfrac{1}{\alpha} & 1+z^{-1} \\ 0 & \dfrac{1}{\alpha} \end{bmatrix}\begin{bmatrix} \dfrac{1}{16\beta} & 0 \\ \dfrac{1+z}{16} & \dfrac{1}{16\beta} \end{bmatrix}\begin{bmatrix} \dfrac{1}{2\gamma} & \dfrac{1+z^{-1}}{2} \\ 0 & \dfrac{1}{2\gamma} \end{bmatrix}\cdot \\ &\begin{bmatrix} \dfrac{1}{2\delta} & 0 \\ \dfrac{1+z}{2} & \dfrac{1}{2\delta} \end{bmatrix}\begin{bmatrix} 64\alpha\beta\gamma\delta K & 0 \\ 0 & \dfrac{64\alpha\beta\gamma\delta}{K} \end{bmatrix} = \\ &\begin{bmatrix} \dfrac{1}{\alpha} & 1+z^{-1} \\ 0 & \dfrac{1}{\alpha} \end{bmatrix}\begin{bmatrix} \dfrac{1}{\beta'} & 0 \\ \dfrac{1+z}{16} & \dfrac{1}{\beta'} \end{bmatrix}\begin{bmatrix} \dfrac{1}{\gamma'} & \dfrac{1+z^{-1}}{2} \\ 0 & \dfrac{1}{\gamma'} \end{bmatrix}\cdot \\ &\begin{bmatrix} \dfrac{1}{\delta'} & 0 \\ \dfrac{1+z}{2} & \dfrac{1}{\delta'} \end{bmatrix}\begin{bmatrix} \alpha\beta'\gamma'\delta' K & 0 \\ 0 & \dfrac{\alpha\beta'\gamma'\delta'}{K} \end{bmatrix} \end{aligned} \tag{5}$$

where α, β, γ, δ are the lifting coefficients; K and $1/K$ are the normalization coefficients; $\beta' = 2^4\beta$, $\gamma' = 2\gamma$, $\delta' = 2\delta$. The second, third and fourth units in Eq. (5) have less roundoff noise than that in Eq. (3), which will be discussed in Section 2.

Though the first four computing units of $\boldsymbol{P}(z)$ in Eq. (5) only have a difference in multiplier coefficients from those in Eq. (3), the critical path delay is reduced by one adder delay. For example, the output of the first unit in Eq. (3) is $[x_e(n) \quad \alpha(x_e(n) + x_e(n-1)) + x_o(n)]$. If this unit is computed in one clock cycle, the critical path delay is the sum of one multiplier delay and two adders delay. If two clock cycles are allocated to this unit, the delay will be reduced by one adder delay. In Eq. (5), the output of the first unit is $[x_e(n)/\alpha \quad x_e(n) + x_e(n-1) + x_o(n)/\alpha]$. If this unit is computed in one cycle, $x_e(n) + x_e(n-1)$ and $x_o(n)/\alpha$ will be calculated simultaneously. Thus, the critical path delay is the sum of one multiplier delay and one adder delay. If two cycles are allocated, $x_e(n) + x_e(n-1)$ and $x_o(n)/\alpha$ can be calculated in the first cycle, and the sum can be calculated in the second cycle. Thus, the critical path delay is only one multiplier delay. The other three units have constant coefficients (1/16 or 1/2), which can be easily implemented by dislocation assignment.

According to Eq. (5), 1D 9/7 DWT has two similar flipping steps and a normalization step. The first flipping step contains the first and second units. The second flipping step contains the third and forth units. The normalization step is implemented by the fifth unit. Therefore, two similar processors only being different in multiplier coefficients are designed and linked to be the main structure of 1D DWT. Fig. 1 shows an architecture of the first flipping step with a five-stage pipeline, and the data flow is similar to that of the column processor in Wu's structure[11]. The path between two registers is computed in one

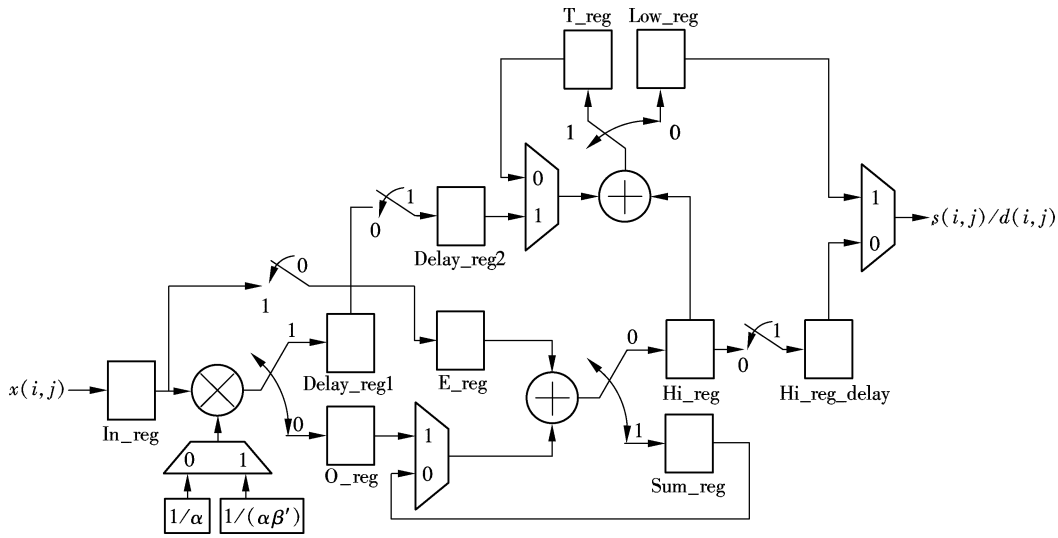

Fig. 1 1D 9/7 DWT basic processor's detailed architecture(first lifting step)

clock cycle, and occupies one stage in the pipeline. In Fig. 1, the first stage contains multiplication, and the other stages only contain assignment or an addition operation. Therefore, the critical path is dominated by the first stage with a delay of one multiplier. Since the multiplying operation cannot be avoided, the critical path delay has reached its limit.

The main drawback of Wu's structure [11] is the large range of the output in each unit. For example, the value of β in Eq. (3) is close to -0.05, so the valid range will be improved 20 times in Eq. (5) if 1/16 is not multiplied, which means that five more bits have to be used to express the output. Therefore, the proposed architecture uses $1/(\alpha\beta)'$ as the second multiply coefficient. As a result, the bit width of registers in the proposed structure is smaller than that in Wu's structure.

2 Proposed 2D 9/7 DWT Architecture

Based on the 1D 9/7 DWT flipping structure, a four-input/four-output 2D 9/7 DWT architecture is proposed, as shown in Fig. 2, in which N is the height and width of the image. For multi-level DWT, the input of the first level is raw image data, and the input of the M-th level is the coefficients of LL subband in the $(M-1)$-th level. The data flows of three key points in Fig. 2 are shown in Fig. 3. To implement high throughput, a multiplexer (MUX) is used to ensure the data of the former $N/2$ columns export in the first two lines, and the data of the latter $N/2$ columns export in the last two lines (see "MUX output and delay" in Fig. 3). Therefore, from the perspective of the column processors, the length of a row is $N/2$ not N. To rearrange the four-line data, the output of the latter two lines in row DWT needs to be delayed $N/2$ cycles (see "MUX input" in Fig. 3). The MUX contains a counter and four basic two-input/one-output multiplexers.

In Fig. 2, row-p1 and row-p2 are row processors for the first and second flipping steps, respectively. Col-p1 and

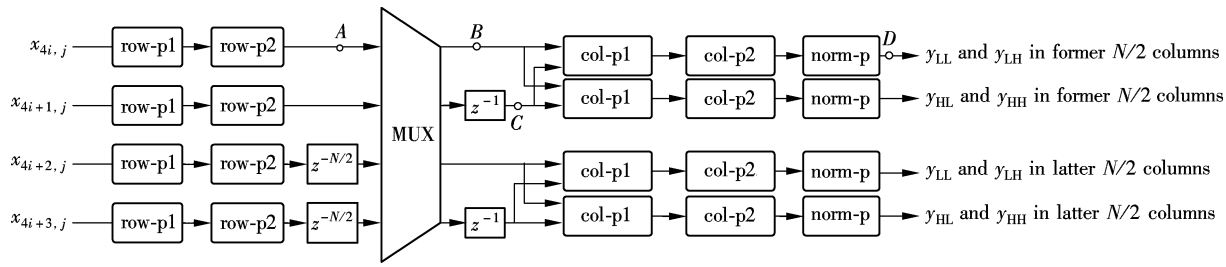

Fig. 2 Four-input/four-output flipping-based DWT architecture

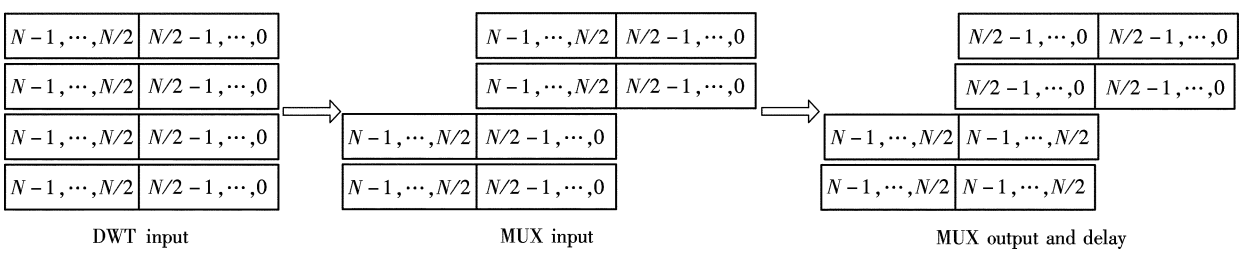

Fig. 3 Data flow of three key points of 2D DWT architecture

col-p2 are the column processors for the first and second flipping steps, respectively. Norm-p represents the combined normalization steps. The structure in Fig. 1 can be directly used as row-p1, because it works at front-ends and obtains original row data in time. Row-p2 can be obtained from row-p1 by changing $1/\alpha$ to $1/(\beta'\gamma')$, $1/(\alpha\beta')$ to $1/(\gamma'\delta')$. The output of row-p2 is the coefficients of L and H subband without the normalization step. However, col-p1 and col-p2 obtain the data from the same column per $N/2$ clock cycles; thus, buffers are needed to store the intermediate data. In addition, the coefficients of L and H subband are interlaced in each line of the MUX output (see Fig. 4(b)); thus, each col-p1 needs to select the coefficients of L (or H) subband from two lines. Therefore, col-p1 is obtained from row-p1 by adding the input selector, which changes T_reg and Sum_reg to T_FIFO and Sum_FIFO with $N/4$ depth, as shown in Fig. 5. Only one of $x(i,2j)$ and $x(i,2j+1)$ is valid at a time, and the switcher is used to choose the valid signal and export the ordered input data. Since col-p2 need not select the line, it can be obtained from col-p1 by discarding the input selector and changing $1/\alpha$ to $1/(\beta'\gamma')$, $1/(\alpha\beta')$ to $1/(\gamma'\delta')$. The precise data flows of three key points marked in Fig. 2 are displayed in Fig. 4. Fig. 4(a) shows that the output of the first line in row DWT contains L and H subband coefficients. Fig. 4(b) shows that the repeated cycles of each column are changed from N to $N/2$. Fig. 4(c) shows that the output of the first line in column DWT contains the coefficients of two subbands, and the repeated cycles are $N/2$. The data flow of point C is the same as that of point B except one clock delay. Therefore, the coefficients of L subband are selected by

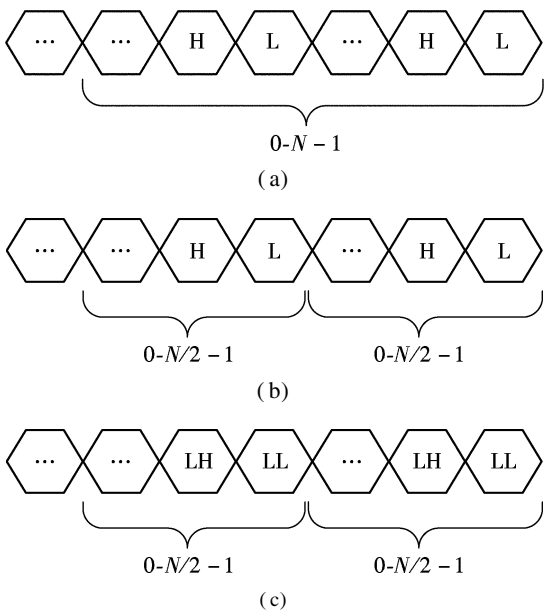

Fig. 4 Data flows of key points marked in Fig. 2. (a) Point *A*; (b) Point *B*; (c) Point *D*

col-p1 in the first line, and the coefficients of H subband are selected by col-p1 in the second line.

Norm-p combines the normalization steps of row and column DWT, which saves one multiplier in each line and one clock cycle, but it decreases the precision of column DWT input and affects the PSNR tightly.

By inversing the multiplier coefficients and multiplying 1/16 or 1/2, flipping-based DWT makes the coefficients in the computing units less than but close to 1, leading to a low roundoff noise and a high PSNR[9]. Tab. 1 shows a PSNR comparison of the lifting-based structure, the flipping-based structures with and without combining normalization steps. Each value in Tab. 1 is the average PSNR of eight gray images with the size of 256 × 256 pixels. Tab. 1 shows that all PSNR values have a difference of less than 0.5 dB between the flipping-based structure with and without combining normalization steps, both of which obtain a much higher PSNR than the lifting-based structure. Therefore, the decrease of PSNR by combining normalization steps can be ignored.

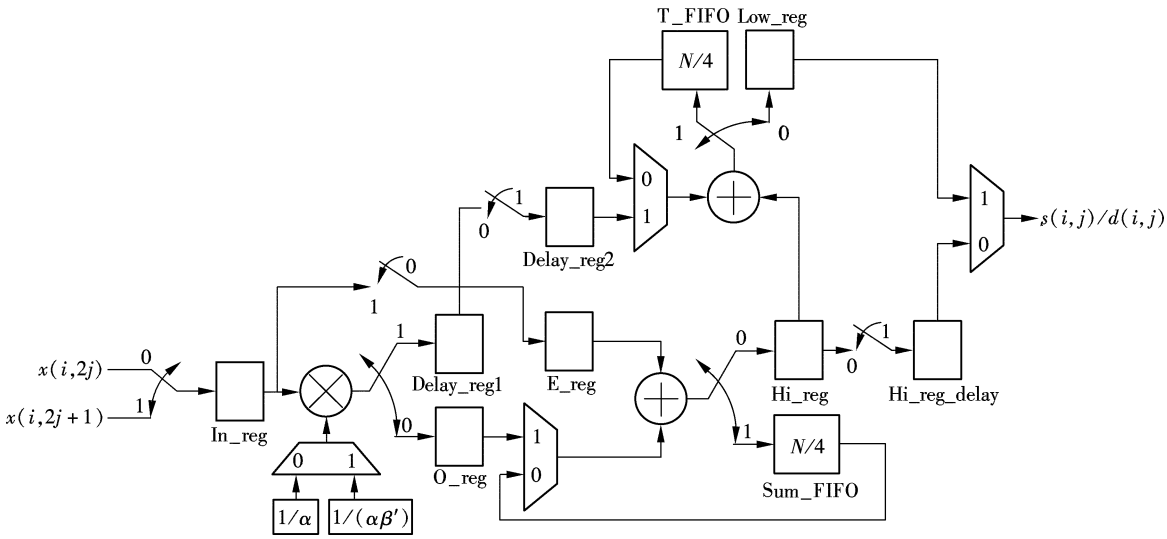

Fig. 5 Detailed architecture of col-p1

Tab. 1 PSNR comparison (12-bit multiplier coefficients) dB

2D 9/7 DWT level	Proposed flipping-based structure	Normal flipping-based structure	Lifting-based structure
1	60.495	60.689	51.357
2	57.425	57.773	44.484
3	55.176	55.577	40.646
4	53.345	53.782	37.599
5	51.979	52.343	35.066

The output of norm-p contains coefficients of two subbands (see Fig. 4(c)). As shown in Fig. 2, the coefficients of LL and LH subbands are interleaved in the first-line and third-line norm-ps, while the coefficients of HL and HH subbands are interleaved in the other two lines. The former two norm-ps export coefficients are transformed from the former $N/2$ columns, while the latter two norm-ps export coefficients are transformed from the latter $N/2$ columns. According to Eq. (5), the normalization coefficient for LL subband is $(\alpha\beta'\gamma'\delta'K)^2$, for HH subband is $(\alpha\beta'\gamma'\delta'/K)^2$, and for the other subbands is $(\alpha\beta'\gamma'\delta')^2$. The detailed structure of norm-p is shown in Fig. 6. The control signal is determined by the subband that the "Input" signal belongs to. In the first and third

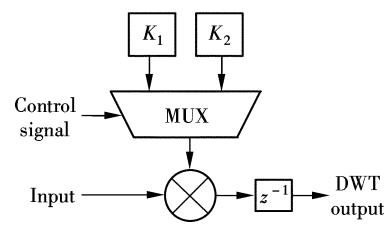

Fig. 6 Detailed structure of norm-p

norm-ps, $K_1 = (\alpha\beta'\gamma'\delta'K)^2$ and $K_2 = (\alpha\beta'\gamma'\delta')^2$. In the second and forth norm-ps, $K_1 = (\alpha\beta'\gamma'\delta')^2$ and $K_2 = (\alpha\beta'\gamma'\delta'/K)^2$.

3 Experimental Results and Comparison

The four-input/four-output architecture for an image with the size of 256×256 pixels has been implemented by Verilog HDL and synthesized on Xilinx Spartan6 lx150t FPGA. The synthesis result is shown in Tab. 2, and the system frequency is as high as 154 MHz. The width of FIFO and delay registers is 12-bit, and memory cost is only 2.375 KB. The results of post-route simulation show that the frequency can reach 136 MHz. Since four coefficients are processed per cycle, the throughput reaches 544 Msample/s on Spartan6 platform.

Tab. 2 Synthesis result of the proposed architecture

Technology Library	Frequency/MHz	LUT number	Memory/KB
Xilinx Spartan6 lx150t	154	4301	2.375

To evaluate the performance of the proposed architecture, different architectures are compared in terms of throughput rate, computing time, etc. The size of the test image is $N \times N$. The computing time is measured by clock cycles. In applying the proposed architecture, the computing time required to transform one-level 2D 9/7 DWT is $0.25N^2 + 1.5N$ clock cycles. As discussed in Section 2, the critical path delay is one multiplier delay.

The memory refers to the registers and the depth of FIFOs. The memory cost of the proposed architecture is $5N$, including $N/2$ memory in each column processor and N registers before the MUX. One processor contains one multiplier and two adders, and a norm-p contains one multiplier. Thus, the proposed architecture costs 20 multipliers and 32 adders.

The performance comparison of several 2D 9/7 DWT architectures is listed in Tab. 3. Compared with most published works, the proposed design gains a great performance in speed and memory cost. The architecture in Ref. [10], even using fewer multipliers, is very close to the proposed architecture. However, its computing time is $0.25N^2$ clock cycles for the four-input/four-output architecture, which does not include the delay before the MUX and the FIFOs delay in the column processors. Thus, the computing time in Ref. [10] is at least $0.25N^2 + 1.5N$ clock cycles. Since the lifting-based structure is adopted in Ref. [10], its PSNR is much lower than that of the proposed architecture. Besides, the architecture in Ref. [10] costs about 20% more memory. The computation cycles of the other three architectures are not available, but their throughputs are much lower than that of the proposed design. Generally, the two-input/two-output structures take twice as much computing time as the proposed design while the one-input/one-output ones take fourfold as much computing time. In addition, the proposed architecture has an advantage in its critical path, which indicates a higher system frequency.

Tab. 3 Performance comparison

Architectures	Throughput rate	Computing time/cycles	Multiplier	Adder	Memory	Critical path delay
Proposed	4 input/output	$0.25N^2 + 1.5N$	20	32	$5N$	T_m
Lifting + 4 stages[10]	4 input/output	$0.25N^2$	16	32	$6N + 40$	$T_m + 2T_a$
Flipping + no pipe[9]	2 input/output		10	16	$4N$	$T_m + 5T_a$
Flipping + 5 stages[9]	2 input/output		10	16	$11N$	T_m
Lifting + 5 stages[11]	1 input/output		6	8	$5.5N$	T_m

Note: T_m means the delay of one multiplier, and T_a means the delay of one adder.

4 Conclusion

A VLSI design for JPEG2000 9/7 DWT is proposed in this paper. It adopts a flipping structure as the basic computing unit and combines the normalization steps in row and column DWT. Besides, it uses a MUX to rearrange the output of the row processors to implement four-line input and four-line output. For a gray image with the size of $N \times N$, the computing time of one-level 9/7 DWT is $0.25N^2 + 1.5N$ clock cycles. The critical path delay is one multiplier delay, and only $5N$ internal memory is required. The results of the post-route simulation on Spartan6 platform show that the frequency is 136 MHz, and the throughput reaches 544 Msample/s. It can be an efficient alternative for high-speed and memory-efficient applications.

References

[1] Taubman D, Marcellin M. *JPEG*2000 *image compression fundamentals, standards and practice* [M]. Norwell, MA, USA: Kluwer Academic Publishers, 2001: 248.

[2] Vishwanath M, Owens R M, Irwin M J. VLSI architectures for the discrete wavelet transform [J]. *IEEE Transactions on Circuits and Systems* Ⅱ: *Analog and Digital Signal Processing*, 1995, **42**(5): 305 - 316.

[3] Chakrabarti C, Vishwanath M. Efficient realizations of the discrete and continuous wavelet transforms: from single chip implementations to mappings on SIMD array computers [J]. *IEEE Transactions on Signal Processing*, 1995, **43**(3): 759 - 771.

[4] Parhi K K, Nishitani T. VLSI architectures for discrete wavelet transforms [J]. *IEEE Transactions on Very Large Scale Integration Systems*, 1993, **1**(2): 191 - 202.

[5] Chakrabarti C, Vishwanath M, Owens R M. Architectures for wavelet transforms: a survey [J]. *Journal of VLSI Signal Processing Systems for Signal, Image and Video Technology*, 1996, **14**(2): 171 - 192.

[6] Zervas N D, Anagnostopoulos G P, Spiliotopoulos V, et al. Evaluation of design alternatives for the 2-D-discrete wavelet transform [J]. *IEEE Transactions on Circuits and Systems for Video Technology*, 2001, **11**(12): 1246 - 1262.
[7] Sweldens W. Lifting scheme: the new philosophy in biorthogonal wavelet constructions [C]//*Wavelet Applications in Signal and Image Processing* Ⅲ. San Diego, CA, USA, 1995, **2569**: 68 - 79.
[8] Daubechies I, Sweldens W. Factoring wavelet transforms into lifting steps [J]. *Journal of Fourier Analysis and Applications*, 1998, **4**(3): 247 - 269.
[9] Huang C T, Tseng P C, Chen L G. Flipping structure: an efficient VLSI architecture for lifting-based discrete wavelet transform [J]. *IEEE Transactions on Signal Processing*, 2004, **52**(4): 1080 - 1089.
[10] Tian X, Wu L, Tan Y H, et al. Efficient multi-input/multi-output VLSI architecture for two-dimensional lifting-based discrete wavelet transform [J]. *IEEE Transactions on Computers*, 2011, **60**(8): 1207 - 1211.
[11] Wu B F, Lin C F. A high-performance and memory-efficient pipeline architecture for the 5/3 and 9/7 discrete wavelet transform of JPEG2000 codec [J]. *IEEE Transactions on Circuits and Systems for Video Technology*, 2005, **15**(12): 1615 - 1628.

一种高吞吐率 JPEG2000 9/7 离散小波变换 VLSI 设计

王建新　朱　恩

（东南大学信息科学与工程学院，南京 210096）

摘要：为提高 JPEG2000 系统中离散小波变换的计算并行度，设计了一种高吞吐率二维 9/7 离散小波变换 VLSI 架构. 其行变换核采用翻转结构，并根据行列变换核输入数据流的差异，在行变换核基础上增加输入选择器和数据缓存模块得到列变换核. 对行列变换的归一化过程进行融合以节省乘法器，并论证了其合理性. 通过多路选择器重排 4 个行变换核的输出，使每个列变换核处理的数据量减半，实现四路输入、四路输出. 对一幅 $N \times N$ 的灰度图像进行一层 9/7 小波变换，计算时间为 $0.25N^2 + 1.5N$ 个周期，关键路径延迟为 1 个乘法器延迟，且只需 $5N$ 存储空间. FPGA 后仿真结果表明，时钟频率可达 136 MHz，吞吐率达到 544 Msample/s，可以满足高速率应用的要求.

关键词：JPEG2000；翻转结构；二维离散小波变换；9/7 离散小波变换；VLSI
中图分类号：TN47