African Journal of Basic & Applied Sciences 9 (2): 81-85, 2017
ISSN 2079-2034
© IDOSI Publications, 2017
DOI: 10.5829/idosi.ajbas.2017.81.85

Vhdl Design and Analysis of 2-Dimensional Discrete Wavelet Transform

A. Santhosh Kumar, K. Boopathiraja, M. Jagadeeswaran and S. Srikanth

Department of ECE, SNS College of Technology, Coimbatore, Tamilnadu, India

Abstract: Discrete Wavelet Transformation (DWT) finds application in fields like image compression, watermarking etc., In IC (Integrated Circuits) like FPGA(Field Programmable Gate Array) using VHDL (VHSIC (Very High Speed Integrated Circuits) Hardware Description Language) improving the processing speed and reducing memory space is a challenging task. The existing design requires more memory space and design complexity on IC. The proposed design with lifting scheme provides reduction in critical path size and number of steps involved pipelining stages. Here Radix-8 Booth multiplier is used for the reduction of critical path size and number of steps is involved in the pipelining stages. The simulation results with device utilization summary, area, delay and schematic output are discussed.

Key words: DWT · FPGA · VHDL · Multiplier · Lifting Scheme · Flipping

INTRODUCTION

Wavelet is time-frequency transformation techniques based on square integral function. Compared with other wavelet transforms, DWT has ah advantages over Fourier transform i.e it captures both frequency and location [1,2]. In this transformation the pair up input values it stores the difference and passing the sum input represented by a list of 2n numbers. This is repeated recursively and pairing up the sums to provide next scale and finally resulting in 2n-1 difference and one sum. In 1996, Ingrid Daubechies and Sweldens introduced the discrete wavelet transformation based on the use of recurrence relations to generate progressively finer discrete samplings of an implicit mother [3,4]. Later, this transformation technique is modified by Dual-Tree Complex Wavelet Transform (CWT) with d-dimensional signals [5].

To increase processing speed, control the increase of hardware cost and reduce the computation time fast algorithm based parallel FIR structures in 2-D DWT are used [6-8]. Parallel-based Lifting Scheme (PLS), Transposition Buffer (TB) and line based architecture for 2-D DWT (LBAQDDWT), Flipping structures (fine stage pipeline) and critical path can be also reduced to one multiplier and delay of adder (Ta). In proposed architecture, only 3 stage of pipelining is used and multiplier is replaced with radix-8 both multiplier for reducing critical path delay.

Proposed Method: Numbers of basic functions are present which can be used as mother wavelet for wavelet transformation. The proposed 2-D DWT core consumes two input data and produces two output coefficients per cycle and its critical path takes one multiplier delay only. The design contains input wait of preprocessing data with circuit and reset signals. It consists of two filter units one is column filter and other is row filter. Column filter has D-flip-flop radix-8 booth multiplier and RAM memory unit. It receives the data from column filter and flow required by the row filter. For final transformation scaling module is used to finish the computation. Memory space is utilized in IC when the blocks are increased. It is used to improve the speed.

Lifting Scheme: It is used for both designing wavelets and discrete wavelets and it is called as second generation wavelet transform (Shown in Fig.1). The transform of the filter or operation in a wavelet is called lifting step. It consists of alternating lifts. Low pass is fixed, High pass will changed and in next step High pass is fixed Low pass will be changed. Due to time delay there will be a transfer of data from to the unit. From input node to Computation node we will eliminate the multipliers. This is achieved by computing the inverse of multiplier coefficients. For reducing the computation delay flipping is replaced by radix-8 booth multiplier. The whole lifting scheme has 917 filter has two lifting steps and one scaling

Corresponding Author: A. Santhosh Kumar, Department of ECE, SNS College of Technology, Coimbatore, Tamilnadu, India step. It is based on hardware implementation. Hence, to reduce the size of the transposing buffer between the column and row filters and to improve the processing speed, proposed design use two-input/two-output with Radix-8 Booth multiplier architecture[6-8].

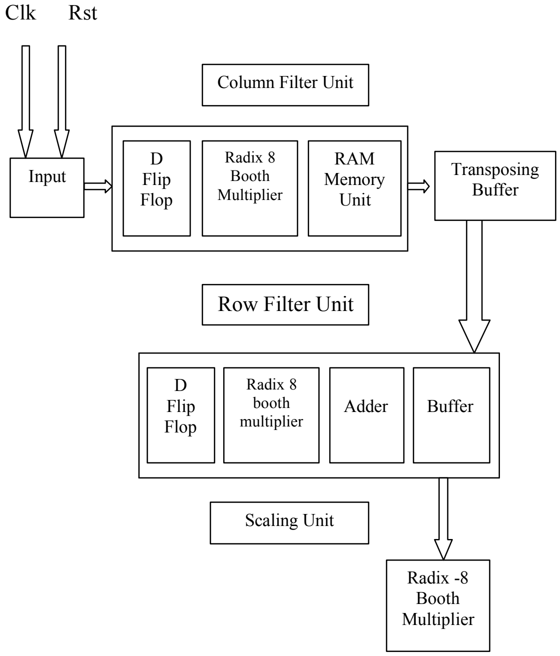

Fig. 1: Architecture of Proposed design

$$\frac{1}{\alpha} y(2n+1) = \frac{1}{\alpha} x(2n+1) + x(2n) + x(2n+2) \tag{1}$$

$$\frac{1}{\beta} y(2n) = \frac{1}{\beta} x(2n) + y(2n-1) + y(2n+1) \tag{2}$$

$$\frac{1}{\gamma} H(2n+1) = \frac{1}{\gamma} y(2n+1) + y(2n) + y(2n+2) \tag{3}$$

$$\frac{1}{\delta} L(2n) = \frac{1}{\delta} y(2n) + H(2n-1) + H(2n+1) \tag{4}$$

Based on eqn. (1)–(4), the flipping structure can achieve only one multiplier delay by pipelining and the above flipping structure based on algorithm it also shows obvious limitations. It needs that the temporal buffer with the size of 11N and to cache the intermediate data.

Substituting (1) into (2) and reordering the expression with the associative law, we can get

$$\frac{1}{\alpha\beta} y(2n) = \frac{1}{\alpha\beta} x(2n) + \frac{1}{\alpha} y(2n-1) + \frac{1}{\alpha} y(2n+1) = \tag{5}$$

$$\left[\left(\frac{1}{\alpha\beta}+1\right) x(2n) + \frac{1}{\alpha} x(2n-1) + x(2n-2)\right] + \left[\frac{1}{\alpha} x(2n+1) + x(2n) + x(2n+2)\right]$$

Four intermediate variables, namely, D_k^1(n), D_k^2 (n), D_k^3 (n) and D_k^4(n) are defined as below, where, 'k' denotes for different values in the row and column transforms. In the row transform, 'k' stands for the number of rows in progress, whereas in the column transform, 'k' stands for the number of scans and therefore one scan means finishing parallel scan of two adjacent rows in the column transform. Thus

$$D_1^k(n) = \frac{1}{\alpha} x(2n+1) + x(2n) \tag{6}$$

$$D_2^k(n) = \left(\frac{1}{\alpha\beta}+1\right) x(2n) + \frac{1}{\alpha} x(2n-1) + x(2n-2) \tag{7}$$

$$D_3^k(n) = \frac{1}{\gamma} y(2n+1) + y(2n) \tag{8}$$

$$D_4^k(n) = \left(\frac{1}{\delta\gamma}+1\right) y(2n) + \frac{1}{\gamma} y(2n-1) + y(2n-2) \tag{9}$$

When Comparing with the flipping-based lifting scheme the modified algorithm suggests a way that in data combination with different coefficients in even data and simplifies the computation process. Then the proposed algorithm combines the predictor with the updater. Then the high-pass and the low-pass signal can be calculated by parallel through the two-input/two-output architecture and at the same time the coefficients of even items are changed by inversion of the factors.

Simulation Results: The design reconstructs the flipping structure by substituting the traditional 5-stage flipping structure with 2-stage structure. However there are more accumulated operations in second stage. In addition, the multiplications on the same path can be merged together to reduce the number of multipliers. The computation nodes can be split into two parts: One is summation of the multiplication results from register node and other one is adder on the accumulation path. The RTL schematic view of proposed design is shown in Fig.2.

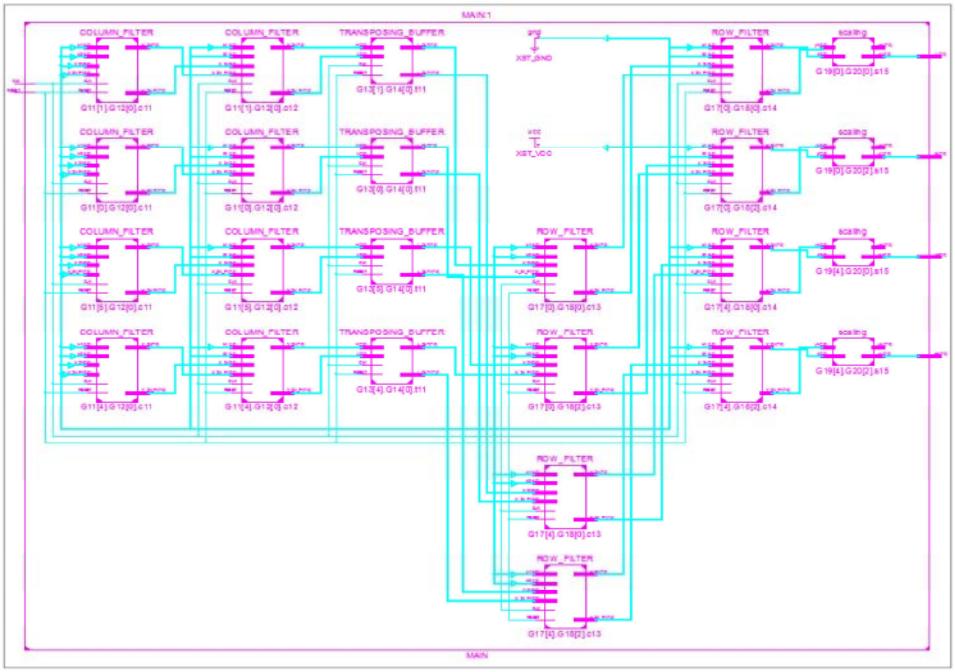

Fig. 2: RTL Schematic View of Proposed Architecture Design

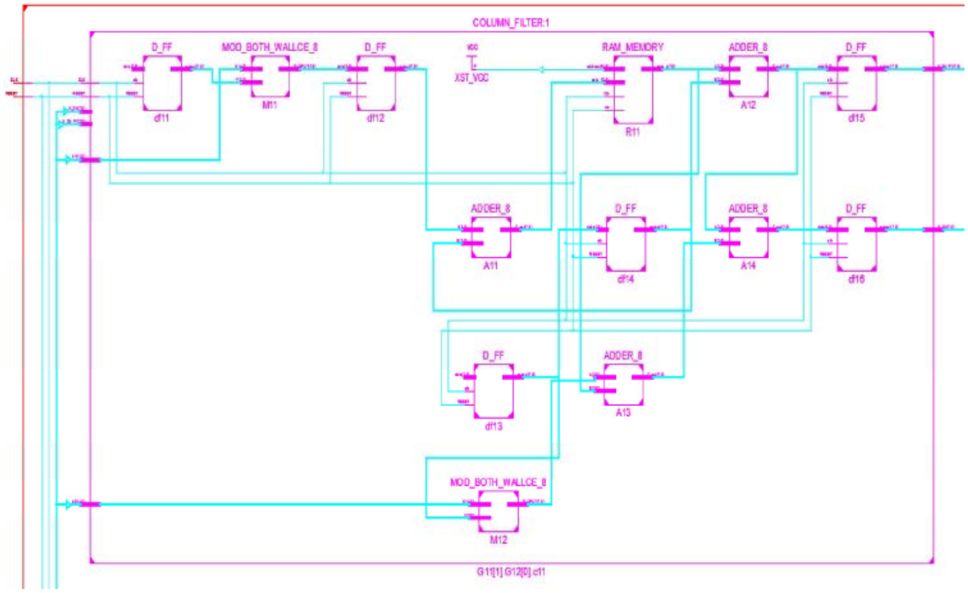

Fig. 3: Column Filter Unit

Advantage is that there is no need of additional multipliers. Three register and two multiplexers are used to make output data meet the order of data flow required by the row filter (Shown in Fig.3 and 4). The total time required for simulation is 3.930ns (2.059ns logic,1.871ns route)(52.4% logic,47.6% route)Total 8.757ns(6.409ns logic,2.348ns route)(73.2% Logic,26.8% Route). Radix-8 algorithm is used to reduce the number of partial products to n/3 where n denotes the number of multiplexer's bits and it allow the time gain in the partial product summation. The proposed design total memory usage is 850056 kilobytes only. In Table 1 the timing reports have clearly proved that there is an overall improvement in performance of the design. It is clearly proved from the result that the proposed design achieved the optimum solution (Shown in Table 2 and 3).

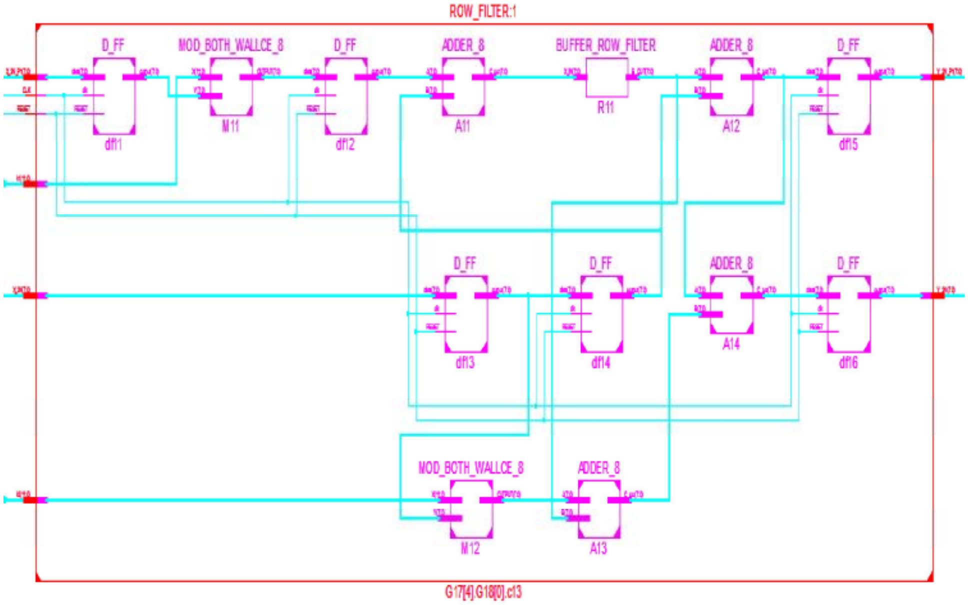

Fig. 4: Design of Row filter Unit

Table 1: Timing Summary

Parameters	Value
Minimum period	11.908ns
Maximum Frequency	80.245MHz
Minimum input arrival time before clock	4.1ns
Maximum output required time after clock	9.23ns
Maximum combinational path delay	No path found

Table 2: Timing Constraints

		Worst	Best case
Constraint	Check	case slack	achievable
Auto time spec constraint for clock net CLK_BUFGP	SETUP HOLD	1.112ns	9.81ns

Table 3: Comparison Table

Parameters	Total no. of Gates	Power (Megawatt)	Delay (nanosecond)
Radix-4	20134	86	46.889
Radix-8	16435	80	18.230

CONCLUSION

The paper proposes a lifting scheme which uses only three pipelining stages with Radix-8 booth multiplexer's their by reducing circuital path delay. Analysis of 2-D DWT architecture is compared with Radix-4 multiplexer architecture in terms of hardware complexity, are power competition time and throughout. Accuracy can be increased by increasing the level of transformation which is taken as future enhancement. It finds its applications in Image processing, Signal analyzer etc.,

REFERENCES

1. Parhi, K.K. and T. Nishitani, 1993. VLSI architecture for discrete wavelet transforms, IEEE Trans. Very Large Scale Integr. (VLSI) Syst., 1(2): 191-202.
2. Jou, J.M., Y.H. Shiau and C.C. Liu, 2001. Efficient VLSI architectures for the biorthogonal wavelet transform by filter bank and lifting scheme, in Proc. IEEE ISCAS, 2: 529-532.
3. Shi, G., W. Liu and L. Zhang, 2009. An efficient folded architecture for lifting-based discrete wavelet transform, IEEE Trans. Circuits Syst. II, Exp. Briefs, 56(4): 290-294.
4. Lai, Y.K., L.F. Chen and Y.C. Shih, 2009. A high-performance and memory-efficient VLSI architecture with parallel scanning method for 2-D lifting-based discrete wavelet transform, IEEE Trans. Consum. Electron, 55(2): 400-407.
5. Huang, C.T., P.C. Tseng and L.G. Chen, 2004. Flipping structure: An efficient VLSI architecture for lifting-based discrete wavelet transform, IEEE Trans. Signal Process., 52(4): 1080-1089.

6. Wu, B.F. and C.F. Lin, 2005. A high-performance and memory-efficient pipeline architecture for the 5/3 and 9/7 discrete wavelet transform of JPEG2000 codec, IEEE Trans. Circuits Syst. Video Technol., 15(12): 1615-1628.
7. Tseng, P.C., C.T. Huang and L.G. Chen, 2002. Generic RAM-based architecture for two dimensional discrete wavelet transform with line- based method, in Proc. Asia-Paci?c Conf. Circuits Syst., 2: 363-366.
8. Xiong, C., J. Tian and J. Liu, 2007. Efficient architectures for two-dimensional discrete wavelet transform using lifting scheme, IEEE Trans. Image Process., 16(3): 607-614.