

An Efficient Architecture Design of Low Complexity in Quantization of H.264/AVC

Ramesh Kumar Lama[†], Jung-Hyun Yun^{††}, Goo-Rak Kwon^{†††}

ABSTRACT

An efficient architecture for the reduction of complexity in forward quantization of H.264/AVC is presented in this paper. Since the multiplication operation in forward quantization plays crucial role in complexity of algorithm. More efficient quantization architecture with simplified high speed multiplier is proposed. It uses the modification of the quantization operation and the high speed multiplier is applied for simplification of quantization process.

Key words: H.264/AVC, Integer transform, Low-complexity.

1. INTRODUCTION

H.264/AVC is the latest international video coding standard developed jointly by ITU-T Video Coding Expert Group and ISO/IEC Motion Picture Expert Group. The H.264 architecture shows superiority among H.263 + and MPEG-4 Part 2 because of improved prediction methods and coding efficiency. Variable block sized motion estimation, Quarter pixel accuracy motion estimation; motion vectors over picture boundary, integer transform are some of the enhanced feature of H.264/AVC [1-3]. This transform employs the 16 bit integer arithmetic without multiplication between transformed coefficients and scaling factors. A quanti-

zation parameter (QP), calculated by the rate control algorithm, is used for determining the quantization step size of transform coefficients in H.264. There are 52 quantization parameter values. These values are arranged so that an increase of 1 in quantization parameter means an increase of quantization step size by approximately 12%. An increase of quantization step size by approximately 12% means roughly a reduction of bit rate by approximately 12%.

1.1 Integer transform and quantization in H.264 Codec

$$Y = (C_j X C_j^T) \otimes E = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix} [X] \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix} \otimes \begin{bmatrix} a^2 & ab/2 & a^2 & ab/2 \\ ab/2 & b^2/4 & ab/2 & b^2/4 \\ a^2 & ab/2 & a^2 & ab/2 \\ ab/2 & b^2/4 & ab/2 & b^2/4 \end{bmatrix} \quad (1)$$

Where, $a = \frac{1}{2}, b = \sqrt{2/5}$, X is the input block and Y output arrays of forward transform. The quantizer operation is defined by,

$$Z_{ij} = \text{round}(Y_{ij}/Q_{step}) \quad (2)$$

Where, Y_{ij} is the transformed coefficient, Q_{step} is the quantization step size and Z_{ij} is the quantized coefficient. The input block X is transformed into unscaled coefficients $W = CXCT^T$. Then, each coefficient W_{ij} is quantized and scaled in a single operation.

* Corresponding Author: Goo-Rak Kwon, Address: (501-759) 375 Seosuk-Dong, Dong-Gu, Gwangju, Korea, Phone: +82-62-230-6268, E-mail: grkwon@chosun.ac.kr
Receipt date: May 14, 2011, Revision date: July 1, 2011
Approval date: Aug. 26, 2011

[†] Dept. of Info. & Comm. Engr., Chosun University (E-mail: pakhrin51@yahoo.com)
^{††} Dept. of Photoelectronics Information, Chosun College of Science & Technology (E-mail: frogcop2@naver.com)
^{†††} Dept. of Info. & Comm. Engr., Chosun university

* This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology(2011-0005164).

$$Z_{ij} = \text{round}(W_{ij} \cdot \frac{PF}{Q_{step}}), \quad (3)$$

$$Z_{ij} = \text{round}((W_{ij} \cdot \frac{MF(Q_{m,ij})}{2^{qbits}})). \quad (4)$$

For the simplicity of hardware implementation, the quantization procedure is carried out as follows,

$$|Z_{ij}| = \text{sign}(W_{ij})(|W_{ij}| \cdot MF_{Q_{m,ij}} + f) \gg qbits, \quad (5)$$

$$\text{sign}(Z_{ij}) = \text{sign}(W_{ij}).$$

The DCT transform is obtained in two steps, core DCT transform and the post scaling multiplication denoted by $\otimes E$. The core 2D transform stage is represented by $W = C_f X C_f^T$ and can be obtained only with addition and shift operations without any use of multiplication operations. However, the post scaling complexity and also need more space for the hardware implementation compared to that with core transform.

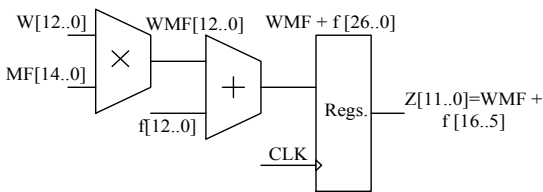


Fig. 1. H.264 Quantization Unit Architecture with common implementation.

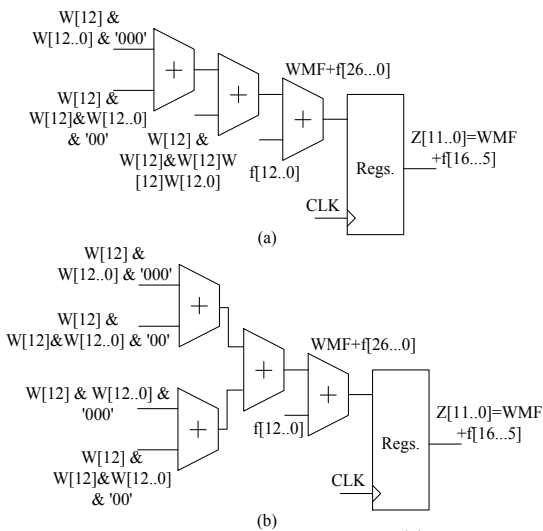


Fig. 2. The design of architecture. (a) Y. Zhang architecture [4]. (b) Proposed architecture.

Table 1. Multiplication factor (MF) of H.264/AVC

QP	Positions (0,0), (2,0), (2,2), (0,2)	Positions (1,1), (1,3), (3,1), (3,3)	Other positions
0	13,107	5,243	8,066
1	11,916	4,660	7,490
2	10,082	4,194	6,545
3	9,362	3,647	5,825
4	8,192	3,366	5,243
5	7,282	2,893	4,559

2. PROPOSED ALGORITHM

2.1 Quantizer optimization

Numerous research works have been conducted for the optimization of quantization process. Zhang et al [4] proposed the new architecture which replaces the multiplier by adders and shifters. This scheme reduces the complexity by cutting down the bit width of multiplication factor MF_{ij} and 2qbits into MF_{ij}' and 2qbits'. The criterion for the reduction of bit size is specified as δ . That represents the percentage modification of magnification factor. Here δ is specified within the range of $\pm 7\%$. The modified magnification factors are shown in Table 2.

The modified scheme gives the significant reduction in bit width with the reduction of hardware design complexity however the modification percentage δ is in the range of $\pm 7\%$. The increase in modification percentage δ gives the increase in MSE error. Michael N. M. et al [5] proposed modified architecture based on [4].

Modification percentage δ is reduced. However it is still in the range of $\pm 2.5\%$. Both of the above mentioned schemes reduce the complexity by replacing the multiplication operation by addition and shifting. Another method is proposed by G. A. Ruiz et al [5]. This method is more focused on reducing the quantization cost by modifying the quantization operation and applying a truncated Booth multiplier based on adaptive statistical

Table 2. Modified multiplication factor of H.264/AVC

QP		Positions (0,0), (2,0), (2,2), (0,2)	Positions (1,1), (1,3), (3,1), (3,3)	Other positions
Top part	0	$102/2^6$	$5/2^5$	$1/2^2$
	1	$90/2^6$	$9/2^6$	$29/2^7$
	2	$81/2^8$	$1/2^3$	$13/2^6$
	3	$9/2^5$	$29/2^8$	$23/2^7$
	4	$1/2^2$	$13/2^7$	$5/2^5$
	5	$7/2^5$	$23/2^8$	$9/2^6$
bottom part	0	-0.49 %	2.4 %	1.54 %
	1	1.14 %	-1.13 %	1.92 %
	2	-0.11 %	2.39 %	1.53 %
	3	1.58 %	0.80 %	2.28 %
	4	0 %	1.65 %	2.4 %
	5	1.59 %	1.17 %	1.06 %

approach. This method gives significant improvement in PSNR and at the wide range of QP values, however the multiplier used in this technique is more complex and the hardware implementation with this technique demands more cost. Hence, an efficient means of designing the quantization architecture is proposed in this paper with the consideration of reduction in hardware cost while maintaining the PSNR value at satisfactory level.

2.2 Proposed Quantization Architecture

Proposed quantization scheme is described in two sections.

2.2.1 Modification in quantization operation

The quantization process is modified as according to [6] prior to multiplication operation. Which makes our algorithm free from creating $|W_{ij}|$ and subsequent sign conversion.

2.2.2 Simplification of multiplication.

MBE (Modified Booth Encoding) is one of the widely used techniques for the multiplication. There is no doubt that the MBE is efficient as it comes reducing the partial products. That is, the

number of partial product row is reduced to $n/2$ from n . However, because of sign extension prevention and negative encoding so, one extra partial product row is added and of course this partial product row requires not only additional hardware space but the time will also increased. In order to remove the extra product the last negative signal needs to be removed, and thus the time of addition carry save adding stage and hardware required for additional carry save adding will be saved.

In conventional methods, negative number is represented as complement of binary number and adds 1 to complemented number. However, this method produces the propagation delay of the carry, which increases linearly with the increase in word size and it would be much greater than the delay to generate the partial products. Therefore, this procedure will not be suitable for our design. The fastest method to represent the two's complement is complementing all the bits before the rightmost "1" in the given word while the remaining bits after this "1" is kept unchanged. For example the two's complement of binary number "00101100₂" is "11010100₂". In this example the rightmost 1 is supposed to be the bit 2 from left. Now the only the bits located on the left side of the changing bit are complemented while rightmost bits are kept unchanged. The searching of rightmost one however may be more complicated because the previous bits information must be transferred to the MSB. Finally, we should find the most convenient way to find the rightmost bit "1".

The search of rightmost "1" can be achieved in the most efficient way by a binary search tree like structure. First find the conversion signal for 2-bit group by grouping two consecutive bits and finding the conversion signal in each group. Then we find the conversion signal for 4-bit and then 8-bit and so on. 4 bit signal conversion can be done as in fig. 3.

Applying the 2's complement method above mentioned, the partial product row is correctly re-

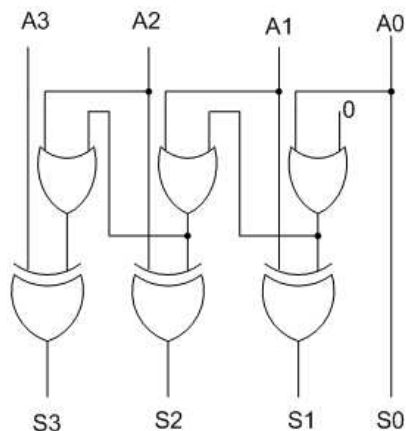


Fig. 3 schematic diagram for 2's complement method for 4bit input.

placed without negative signal. Now the multiplication can have a smaller critical path. This avoids having to include one extra carry saving the adding stage.

3. SIMULATION AND RESULTS

Experiments are performed by using the Model sim simulator by Mentor Graphics and Xilinx ISE. First of all the circuit is designed using Verilog HDL[7,8] and simulated in Modelsim simulator then designed circuit is synthesized with Xilinx Project Navigator 13.1 for Xilinx Virtex5 (xc5v1x30). Furthermore, the power consumption by the circuit is estimated using logical picture analysis. Design of the proposed system is tested on 0.35µm with 3.3v supply. In [5] the design is implemented using the truncated booth multiplier. In truncated booth multiplier products of parallel multipliers are rounded to a shorter word size and the least-sig-

nificant columns of the multiplication matrix are not used. Experimental results show that, by implementing the proposed algorithm gives reduction in both the power consumption[9] as well as the delay of circuit in Table 3.

4. CONCLUSIONS

This Paper presented a new architecture for quantization of H.264/AVC. By using the efficient multiplication technique in the quantization process, the proposed algorithm achieves efficient increase in speed and reduction in power consumption while maintaining the multiplication factor of H.264/AVC is kept unchanged.

REFERENCE

- [1] Wiegand, T., Sullivan, G. J., Bjontegaard, and G. Luthra, A., "Overview of H.264/AVC Video Coding Standard," *IEEE Trans. Circuits Syst. Video Techno*, 13(7), pp. 560-576, 2003.
- [2] Iain E.G. Richardson, "H.264 and MPEG -4 Video Compression," John Wiley & Sons, 2003.
- [3] Yong-Hwan Kim, Je-Woo Kim, Tae-Wan Kim, and Byeongho Choi, "Optimization of H.264 Encoder using SIMD Instructions," *Proceeding of KMMS Conference*, pp.175-178, Nov. 2003.
- [4] Zhang Y., Jiang G., Yi W., Li. F., Jiang Z., and Liu W., "Low-Complexity Quantization for H.264/AVC," *J. Real-Time Image Proc.*, 4, pp. 3-12, 2009.
- [5] G.A. Ruiz1 and J.A. Michell, "Low-Cost VLSI Architecture Design for Forward Quantization of H.264/AVC" *Proc. of SPIE Vol. 6590*, pp. 1-12, 2007.
- [6] Y. W. Huang, B. Y. Hsieh, T. C. Chen, and L. G. Chen, "Analysis, Fast Algorithm, and VLSI Architecture Design for H.264/AVC Intra Frame Code," *IEEE trans. on circuits*

Table 3. Synthesized results

	Multiplier	Power (mW/Mhz)	Critical delay(ns)
Zhang Y. [3]	1	0.41	7.007 ns
Michael [4]	1	0.42	7.167 ns
Proposed method	1	0.39	6.81 ns

and systems for video technology, Vol.15, No.3, 2005.

- [7] S. M. Kang, "Accurate Simulation of Power Dissipation in VLSI Circuits," *IEEE Journal of Solid-State Circuits*, Vol.21, No.5, pp. 889-891, 1986.
- [8] T. Kuroda, "Low-Power High-Speed CMOS VLSI Design," *Proc. of IEEE International Conf. on Computer Design*, pp. 310-315, 2002.
- [9] R. Burch, F.N. Najm, P. Yang, and T.N. Trick, "A Monte Carlo Approach for Power Estimation," *IEEE Transactions on VLSI Systems*, Vol.1(1), pp. 63-71, 1993.
- [10] Michael N. Michael, and Kenneth W. Hsu, "A Low Power Design of Quantization for H.264 Video Coding Standard," *IEEE International SOC Conference 2008*, issue 17-20, pp. 201-204, 2008.



Ramesh Kumar Lama

received the B.S. degree in PulbanChal University and the M.S. degree in Chosun university, in 2010. Currently, he is pursuing a Ph.D. course in Chosun university. His interest research fields are Image and Video

Coding and Image enhancement.



Jung-Hyun Yun

received the B.S., M.S., and Ph. D. degrees in the Dept. of Electronic Engineering from Chosun University, GwangJu, in 1993, 1995, and 1999, respectively.

In 2006, he was a Visiting Professor with the Chosun University, GwangJu. In 2007, he joined the Dept. of Photoelectronics, Chosun College of Science & Technology, GwangJu, where he is currently an Assistant Professor. His research interests include Optical Integrated Circuit, Optical Communication System, and Dye-Sensitized Solar Cell.



Goo-Rak Kwon

received the M.S. degree from SungKyunKwan University, in Electronic Engineering, in 1999.

He received a Ph.D. degree from Korea University in Mechatronic Engineering in 2007. He has also served as Chief Executive Officer and Director of Dalitech Co. Ltd. from May 2005 to Feb. 2007. In March 1st, 2008, he joined the Department of Information & Communication Engineering at Chosun University, Gwangju, Korea, where he is currently a assistant professor. His interest research fields are A/V signal processing, multimedia communication, and applications.