

A High-Throughput VLSI Architecture for Deblocking Filter in HEVC

Weiwei Shen, Qing Shang, Sha Shen, Yibo Fan, Xiaoyang Zeng
State Key Lab of ASIC and System, Fudan University
Shanghai, China
10110720024@fudan.edu.cn, fanyibo@fudan.edu.cn

Abstract—As the next generation standard of video coding, the High Efficiency Video Coding (HEVC) aims to provide significantly improved compression performance in comparison with all existing video coding standards. We propose a four-stage pipeline hardware architecture on a quarter-LCU basis of deblocking filter in HEVC. Coupled with the novel filter order, a memory interlacing technique is adopted to increase the throughput, which can access the data in the process of both vertical and horizontal filtering efficiently. As a result, our design can support 4Kx2K (4096x2048) at 30 fps applications with merely 28MHz working frequency.

I. INTRODUCTION

ITU-T Video Coding Experts Group (VCEG) and ISO/IEC Moving Picture Experts Group (MPEG) formed a Joint Collaborative Team on Video Coding (JCT-VC) in 2010, and the next generation coding standard, High Efficiency Video Coding (HEVC), is now being developed. HEVC aims to reduce 50% bit rate in comparison with the existing H.264/AVC high profile, under the same visual quality [1].

Alike the previous video coding standard H.264/AVC, HEVC is also adopted the block based hybrid coding framework [2], [3]. A quadtree based coding structure is an important feature of the new standard.

Several new coding structures have been introduced in HEVC: coding unit (CU), prediction unit (PU) and transform unit (TU). The CU is the basic unit of region splitting used for intra/inter coding [4]. It can be split from the largest coding unit (LCU, can be as large as 64x64 pixels) into the smallest coding unit (SCU, 8x8 pixels). Coupled with CU, PU carries the information related to the prediction processes. TU is used for transform and quantization, and it depends on PU partitioning modes.

Blocking is known as one of the most visual and objectionable artifacts of block-based compression methods [5]. For this reason, deblocking filter (DF) is employed to improve the subject and objection quality of the video. Low-pass filters in DF are adaptively applied to every 4x4 block boundaries in H.264/AVC. [6], [7] are the work about hardware implementations on deblocking filter in H.264/AVC.

HEVC also uses an in-loop deblocking filter similar to the one in H.264/AVC. As mentioned above, there are three basic units (CU, PU, and TU) in HEVC. Union of these three units' boundaries is involved in the deblocking filter process. All filtering operations are applied to 8x8 block boundaries, except for 4x4 block boundaries, which are not filtered to reduce the complexity.

The vertical edges in a picture are filtered at first. Then the horizontal edges in a picture are filtered with samples modified by deblocking filter of vertical edges as input [8]. In consideration of the tradeoff between on-chip memory area and high throughput, we present a four-stage pipeline hardware architecture on a quarter-LCU basis; the quarter-LCU is defined as a 32x32 block. Based on the quarter-LCU structure, a novel filter is suggested in order to keep the same result on a picture basis. Meanwhile, we also propose a memory interlacing scheme to arrange the data in on-chip memory, and access the data in the process of both vertical and horizontal filtering efficiently.

The rest of this paper is organized as follows. In section II, the deblocking filter algorithm of HEVC is briefly introduced. In section III, our proposed architecture is described. In section IV, our implementation results are presented. Finally, in section V, some conclusions are drawn.

II. DEBLOCKING FILTER ALGORITHM IN HEVC

Fig.1 illustrates the overall processing flow of deblocking filter processing [4]. Firstly, the decision should be made that whether the current boundary is a boundary of CU, PU or TU. If not, the filtering processing should be not applied to the current boundary. Boundary strength (BS) reflects how strong the filtering is needed for the boundary, and the value of this parameter is an integer ranging from 0 to 2. It is determined by some coding information, such as prediction mode, motion vector (MV) and so on. Threshold values β and t_c which are used for filter on/off decision, strong/weak filter selection and the process of the filtering are derived based on the QP of P block and Q block in Fig 2. As shown in Fig 2, P block and Q block are two adjacent 4x4 blocks across the boundary involved in filtering. The value d is also involved in the

This paper is supported by National High Technology Research and Development Program (863, 2012AA012001), State Key Lab of ASIC & System Project (11MS004).

decision of filter on/off, strong and weak filter. This parameter is derived from the value of twelve pixels in the first and the fourth line. These twelve pixels are labeled as red circles shown in Fig 2.

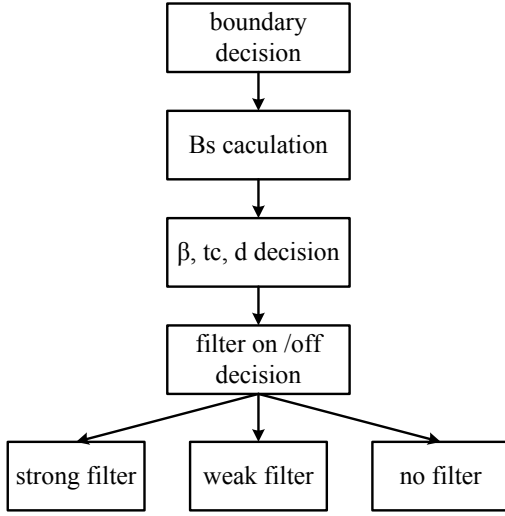


Figure 1. Overall processing flow of deblocking filter

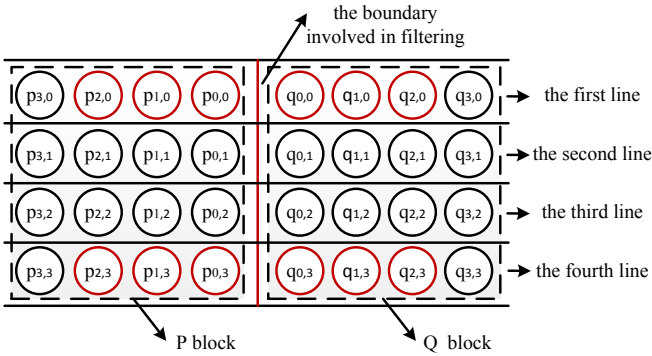


Figure 2. Two adjacent 4x4 blocks

These four lines involved in filtering share some common decisions including β , tc , d . Meanwhile, each line has its respective values, such as dE , dEp and dEq , which are used for the decision of filter on/off, strong and weak filter for each line.

$$dE \neq 0, bs \neq 0 \quad (1)$$

$$bs > 1 \quad (2)$$

The values of eight pixels across the boundary are denoted as $p_{3,0}, p_{2,0}, p_{1,0}, p_{0,0}, q_{0,0}, q_{1,0}, q_{2,0}, q_{3,0}$, which are labeled as the first line in Fig 2. For the luminance component, the first line is filtered only when the condition of equation (1) is satisfied. For the chrominance component, the first line is filtered only when the condition of equation (2) is satisfied. The decisions of strong and weak filter are detailed in [8].

III. PROPOSED DEBLOCKING FILTER ARCHITECTURE

A. Top Architecture

The top architecture of our proposed deblocking filter is shown in Fig 3.

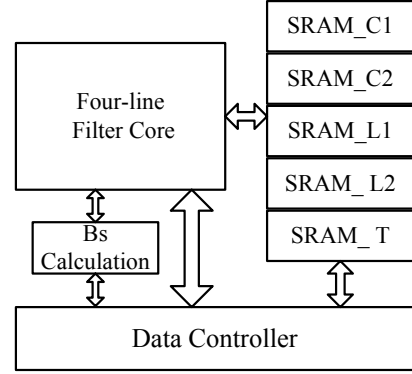


Figure 3. Top architecture of deblocking filter

Our design is mainly composed of four components: data controller, four-line filter core, bs calculation, and five two-port SRAMs.

B. Proposed four-stage pipeline architecture

We present a four-stage pipeline architecture of deblocking filter. It can filter four lines simultaneously. The function of each stage is illustrated as follows:

Stage 1: Calculate the bs value according to the information such as the prediction mode, motion vector and so on. Reading the pixels to be filtered from on-chip SRAM (SRAM_C1, SRAM_C2, SRAM_L1, SRAM_L2, and SRAM_T) is also done in this stage.

Stage 2: Calculate the threshold value β , tc , the condition dE , dEp , dEq used for the decision of filter on/off, strong and weak filter.

Stage 3: According to the bs , tc , dE , dEp , dEq obtained from the above stage, different taps are applied to filter the pixels across the current boundary. And the detailed filtering computation of both the strong filter and the weak filter can be found in HEVC [8].

Stage 4: Write the filtered data back to on-chip SRAM to be further filtered, or to the external memory to be referenced for the other modules of the video codec.

C. Proposed filter order scheme

In this section, three filter order levels named picture level, boundary level and four line level, are defined to describe our proposed filter order.

Picture level filter order: the vertical edges in a picture are filtered first. Then the horizontal edges are filtered with samples modified by deblocking filter of vertical edges as input. The vertical edges are filtered starting with the edge on the left-hand through the edges towards the right-hand in their geometrical order. The horizontal edges are filtered starting with the edge on the top-hand through the edges towards the

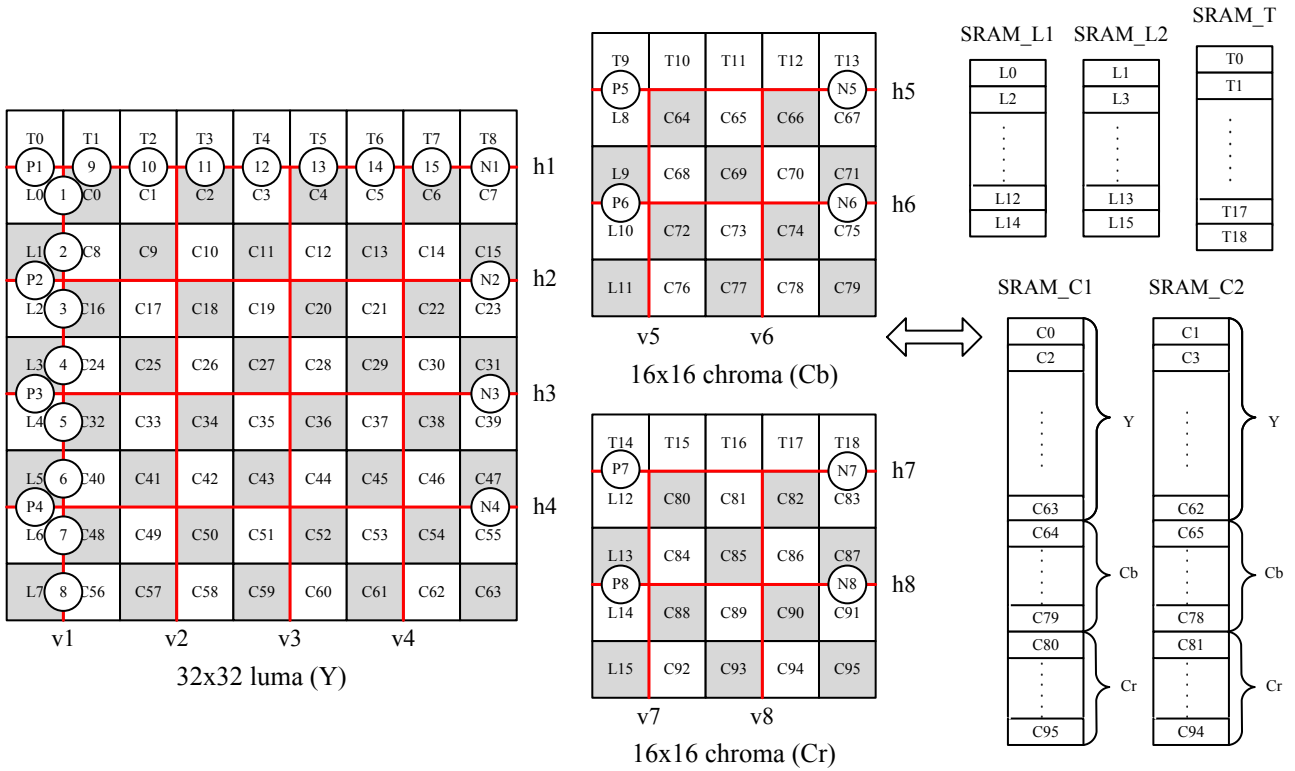


Figure 4. Memory organization of deblocking filter

bottom-hand in their geometrical order. This is the basic filter order principle of deblocking filter in HEVC [8].

Boundary level filter order: the boundaries involved in filtering on a quarter-LCU basis are labeled as red lines in Fig 4. The vertical boundaries including $\{v1, v2, v3, v4, v5, v6, v7, v8\}$ are filtered from left to right (if any), and the horizontal boundaries including $\{h1, h2, h3, h4, h5, h6, h7, h8\}$ are filtered from top to bottom (if any).

Four line level filter order: each boundary consists of several four-line units. The vertical boundaries are filtered from top four-line unit to the bottom four-line unit (e.g., from four-line unit 1 to four-line unit 8 in vertical boundary $v1$). The horizontal boundaries are filtered from left four-line unit to the right four-line unit (e.g., from four-line unit 9 to four-line unit 15 in horizontal boundary $h1$).

Due to our quarter-LCU structure, we adopt the filter order scheme combining the boundary level filter order and the four line level filter order. Firstly, vertical boundaries are filtered from $v1$ to $v8$, then from $h1$ to $h8$. And each boundary obeys the four line level filter order. Note that, four-line unit $\{n1, n2, n3, n4, n5, n6, n7, n8\}$ would not be filtered in current quarter-LCU by explicitly setting the bs value to 0, expect that the current quarter-LCU is on the right boundary of the entire picture. These lines would be filtered in right quarter-LCU which is on the right of current quarter-LCU, corresponding to $\{p1, p2, p3, p4, p5, p6, p7, p8\}$ in the right quarter-LCU. With our proposed filter order scheme, the final filtered data are the same as those specified in HEVC [8].

D. Proposed memory organization

The on-chip memory is used to reduce the I/O bandwidth between the chip and the system. The main challenge is to properly arrange the data in memory modules in order to access the data smoothly from the memory modules to the four-line filter core, on both vertical and horizontal boundaries. We present an interlacing memory organization to access the data effectively in the processing of both vertical and horizontal filtering.

Our approach is to divide the on-chip memory into five modules (SRAM_C1, SRAM_C2, SRAM_L1, SRAM_L2, and SRAM_T), shown in Fig 4. SRAM_C1 and SRAM_C2 store the pixels of the current quarter-LCU, those in SRAM_L1 and SRAM_L2 are from the left neighboring quarter-LCU. Meanwhile, the pixels in SRAM_T come from the top neighboring quarter-LCU and left-top neighboring quarter-LCU. Each square in Fig.4 stands for a 4x4 pixels of luminance or chrominance component. For example, the grey squares from $C0$ to $C95$ belong to SRAM_C1, while the white squares from $C1$ to $C94$ belong to SRAM_C2. A detailed memory structure is shown in Table I. And two 4x4 pixel blocks on both sides of every boundary always come from different memory modules. As a result, all pixels can be easily accessed from the SRAMs on the vertical and horizontal filtering operations.

A two-port SRAM can support one read and one write at the same clock cycle. In our proposed design, we utilize five two-port SRAMs. These SRAMs can also be used to store the

TABLE I. MEMORY STRUCTURE

	SRAM Size (bit)	Memory Structure	SRAM Type
SRAM_C1	48x128	grey squares from C0 to C95	two-port
SRAM_C2	48x128	white squares from C1 to C94	two-port
SRAM_L1	8x128	white squares from L0 to L14	two-port
SRAM_L2	8x128	grey squares from L1 to L15	two-port
SRAM_T	19x128	from T0 to T18	two-port

immediate pixels for the further filter. Therefore, no more on-chip memories are needed.

E. Data Flow

The deblocking filtering process starts as soon as the current quarter-LCU's pixels are received from the external memory or the internal memory from other modules of the video codec.

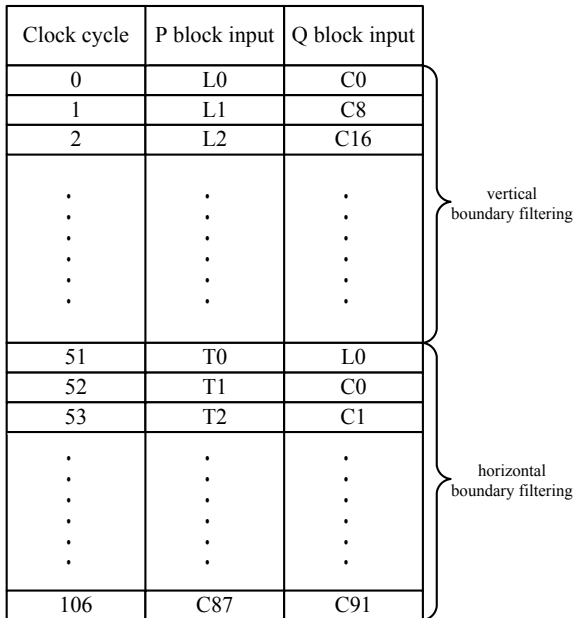


Figure 5. Two adjacent 4x4 blocks

The processing begins with the vertical filtering of the L0 and the C0, filtered by the four-line filter core. The filtered data should be stored back to the on-chip memory for horizontal boundary filter, which is detailed in Fig 5. Meanwhile, the filtered data after horizontal filtering operation can be wrote to the external memory. As a result, a quarter-LCU can be processed in 110 clock cycles.

IV. IMPLMENTION RESULTS

The proposed architecture is implemented in Verilog-HDL and synthesized under a timing constraint of 200MHz with

TABLE II. GATE COUNT SUMMARY

Purpose	Gate count(NAND2)
Four-line filter core	21K
Data controllor (including bs caculation)	10K
SRAM	44K
Total	75K

0.13-um library. Gate count of every module is summarized in Table II.

According to Table II, the cost of filter circuits and that of SRAMs are found to be as 21K and 44K, respectively. It also demonstrates the fact that the proposed architecture achieves a trade-off between high-throughput performance and memory size. In addition, the control logic occupied 10K.

V. CONCLUSION

To the authors' knowledge, this paper firstly reports a hardware implementation on the deblocking filter in HEVC, which features high throughput. With the featured four-stage balanced pipeline, the maximum speed can reach 200MHz. Coupled with a memory interlacing technique, the four-line filter architecture can accomplish a quarter-LCU in 110 clock cycles. As a result, our design can support 4Kx2K at 30fps with merely 28MHz working frequency, and it is capable for higher resolution or low power applications.

REFERENCES

- [1] Chih-Ming Fu, Ching-Yeh Chen, Yu-Wen Huang, Shawmin Lei, "Sample adaptive offset for HEVC," Multimedia Signal Processing (MMSP), 2011 IEEE 13th International Workshop on , pp.1-5, 17-19 Oct. 2011.
- [2] Draft ITU-T recommendation and final draft international standard of joint video specification, ITU-T Rec. H.264/AVC/ISO/IEC 14496-10 AVC, JCT-G050, 2003.
- [3] T. Wiegand, B. Bross, W.-J. Han, J.-R. Ohm, and G. J. Sullivan, "Working Draft 3 of High-Efficiency video Coding," JCTVC-E603, March 2011.
- [4] I.-K. Kim, K. McCann, K. Sugimoto, B. Bross, and W.-J. Han, "High Efficiency Video Coding (HEVC) Test Model 7 (HM 7) Encoder Description," JCTVC-I1002, May 2012.
- [5] Pourazad, M.T., Dautre, C., Azimi, M., Nasiopoulos, P., "HEVC: The New Gold Standard for Video Compression: How Does HEVC Compare with H.264/AVC?," Consumer Electronics Magazine, IEEE , vol.1, no.3, pp.36-46, July 2012.
- [6] Weiwei Shen, Yibo Fan, Xiaoyang Zeng, "A 64 Cycles/MB, Luma-Chroma Parallelized H.264/AVC Deblocking Filter for 4Kx2K Applications," IEICE TRANS.ELECTRON., vol.E95-C, no.4, pp.441-446, April 2012.
- [7] Ke Xu, Chiu-Sing Choy, "A Five-Stage Pipeline, 204 Cycles/MB, Single-Port SRAM-Based Deblocking Filter for H.264/AVC," Circuits and Systems for Video Technology, IEEE Transactions on , vol.18, no.3, pp.363-374, March 2008.
- [8] B. Bross, W.-J. Han ,G. J. Sullivan, J.-R. Ohm , and T. Wiegand, "High Efficiency Video Coding (HEVC) text specification draft 8," JCTVC-J1003, July 2012.