

Regular Paper

A Low Bandwidth Integer Motion Estimation Module for MPEG-2 to H.264 Transcoding

XIANGHUI WEI,^{†1} TAKESHI IKENAGA^{†1}
and SATOSHI GOTO^{†1}

A low-bandwidth Integer Motion Estimation (IME) module is proposed for MPEG-2 to H.264 transcoding. Based on bandwidth reduction method proposed in Ref. 1), a ping-pang memory control scheme combined with Partial Sum of Absolute Differences (SAD) Variable Block Size Motion Estimation (VBSME) architecture are realized. Experiment results show bandwidth of the proposed architecture is 70.6% of H.264 regular IME (Level C+ scheme, 2 Macro Block (MB) stitched vertically), while the on-chip memory size is 11.7% of that.

1. Introduction

Video transcoding performs operations to transform one compressed video stream to another²⁾. These operations include bit rate, frame rate, spatial resolution, coding syntax and content transform, etc. MPEG-2 is presently dominating video coding standard used in most products ranging from digital TV to DVD³⁾. H.264/AVC is the latest video coding standard developed by Joint Video Team (JVT)⁴⁾. It introduces several advanced technologies to improve the compression efficiency and will be deployed in recent future. Therefore there exist emergent needs to transcode existing MPEG-2 video stream to H.264/AVC in many applications, for example, video storage.

Many works have been contributed to develop an efficient MPEG-2 to H.264/AVC transcoding algorithm. MPEG-2 Motion Vector (MV) reuse is a commonly employed method to reduce the computation burden of Motion Estimation (ME). In Ref. 5), decoded MPEG-2 motion vector is used as one of Motion Vector Predictors (MVP) in the Enhanced Predictive Zonal Search (EPZS) algorithm. In Ref. 6), MPEG-2 MV is reused as MVP. Then MVP selection, motion vector refinement and a top-down splitting strategy for sub-block motion

vector re-estimation are applied. In MVP selection, motion costs of MPEG-2 MV and MV from neighboring blocks are compared. The motion vector with the smallest cost is chosen as search center.

A critical issue in hardware design of video coding system is bandwidth reduction because external bandwidth is a limited resource. Several methods are proposed for Full Search Block Matching Algorithm (FSBMA). In Ref. 7), four search region data reuse methods ranged from Level A to Level D are discussed. In Ref. 8), a Level C+ algorithm and its associated *n*-stitched zig-zag scan are introduced. This method utilizes horizontal and partially vertical overlapping area of search window to further reduce bandwidth.

The related works on search window reuse method are based on regular reference pixel accessing order inherited from FSBMA. In MPEG-2 to H.264/AVC transcoding, reusing MPEG-2 motion vector as search center breaks up this regularity. Therefore it is difficult to directly apply related methods in transcoding. A Level C search window reuse scheme for transcoding is proposed in Ref. 1), which utilizes the similarity between successive MVs to regularize center of search window.

In this paper, an IME hardware architecture combined with Level C scheme is presented for MPEG-2 to H.264/AVC transcoding. In Section 2, the MPEG-2 to H.264/AVC transcoder architecture, data reuse methods for video coding systems and the Level C search window reuse scheme for transcoding are briefly introduced. The proposed IME hardware architecture is presented in Section 3. Experiment results are shown in Section 4. In Section 5, we will draw some conclusions.

2. Level C Search Window Reuse Scheme for MPEG-2 to H.264/AVC Transcoding

2.1 MPEG-2 to H.264/AVC Transcoder Architecture

The video transcoding functions are shown in **Fig. 1**. Transcoding functions are classified as: homogeneous, heterogenous and additional functions. The homogeneous transcoding performs conversion between video bitstreams from the same standard; The heterogenous transcoding provides conversions between different video coding standards, in which a syntax conversion between standards

^{†1} Graduate School of Information, Production and Systems, Waseda University

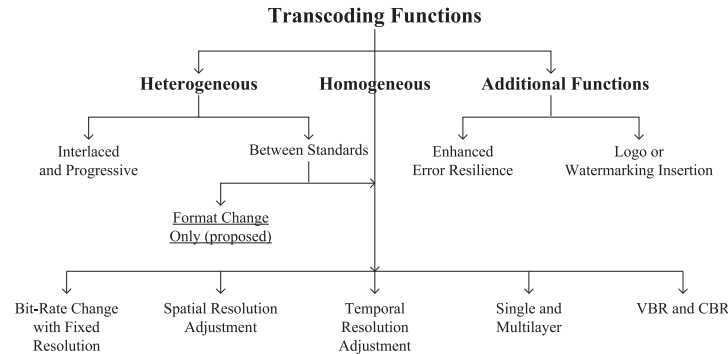


Fig. 1 The transcoding classification and functions.

is necessary. The shared functions in transcoding include bit-rate change, spatial and temporal resolution conversion, etc. Additional functions include error resilience and logo/watermarking insertion. This work focuses on heterogenous transcoding between MPEG-2 to H.264/AVC, in which only format conversion is taken into consideration.

Video transcoding architectures are categorized as: open-loop, closed-loop, spatial domain and frequency domain⁹⁾. The closed-loop architecture directly cascades the decoder and encoder. The source video stream (V_S) is fully decoded and then re-encoded into the target video stream (V_T). Therefore no video quality degradation is involved. The open-loop architecture eliminates feed-back loop in decoder and encoder end, which leads to drift-error. The frequency domain architecture performs some of the transcoding functions in frequency domain as shown in Fig. 2. Therefore drift-error is also unavoidable. Considering the big syntax gap between MPEG-2 and H.264/AVC as well as video quality, the closed-loop architecture combined with motion information reuse is realized in the proposed architecture. Benefitting from accurate MPEG-2 MV, a small search window ($[-16, 16]$) is achieved in H.264/AVC encoder end. Therefore computation burden of ME in H.264/AVC encoder end is greatly reduced.

2.2 Data Reuse Scheme for Video Coding System

Data reuse scheme plays a critical role in video coding system design to reduce bandwidth. It exploits data locality to reduce redundant data access during ME.

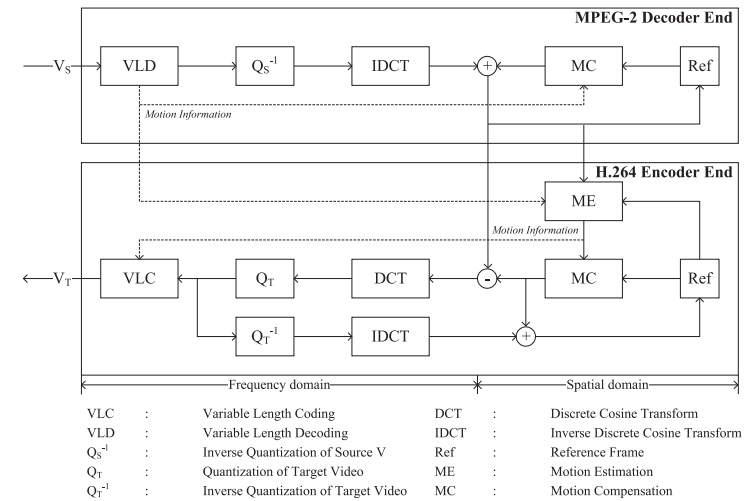


Fig. 2 The applied transcoding architecture.

Reference 7) analyzes data locality in ME. It is categorized as following

- (1) Locality in current frame. This kind of locality refers to invariability of MB data during its ME processing.
- (2) Locality in previous frame. It refers to overlapping between neighboring search areas in reference frame. There exist four kinds of locality
 - (a) Level A—local overlapping of successive reference blocks.
 - (b) Level B—local overlapping of reference strips.
 - (c) Level C—global overlapping between successive search windows.
 - (d) Level D—global overlapping of search area strips.

In hardware implementation, the current MB buffer exploits locality in current frame to reduce bandwidth. The Level A and Level B locality are embedded in Level C implementation in existing hardware design. This is achieved by parallelizing several Processing Element Group (PEG) and reference data scheduling. The Level D locality requires relatively large on-chip memory, which makes it impractical. The Level C locality is commonly utilized because it strikes a balance between bandwidth and on-chip memory.

Reference 8) proposes another method, called Level C+ scheme. It achieves

lower bandwidth compared with Level C scheme by loading search window for MB group, which is composed of several adjacent MBs in vertical direction. Therefore both horizontal and partially vertical overlapping area of search windows are utilized. A reference data scheduling called zig-zag scan is also proposed for this method.

2.3 Level C Search Window Reuse Scheme for Transcoding

The Level C search window reuse scheme proposed in Ref. 1) is introduced in this section. In following discussion, search range is defined as $[-p_h, p_h]$ in horizontal direction and $[-p_v, p_v]$ in vertical direction; MB size is defined as $N \times N$. Accordingly, search window size is $(sr_h + N - 1)(sr_v + N - 1)$, where $sr_h = 2p_h$ and $sr_v = 2p_v$. The raster-scan is assumed to be the processing order for H.264/AVC ME.

2.3.1 Overall Algorithm

The Level C scheme for transcoding is based on the fact that neighboring MPEG-2 MV often have similar value. If MV difference between successive MVs is less than a threshold, it is defined as the smooth MV field. The successive MVs are assumed to have fixed interval. That is successive two MBs have 16

```

for each  $MB_i$  in sequence
  begin
    if  $|\Delta \vec{mv}| = |\vec{mv}_i - \vec{mv}_{i-1}| < t$  then
      begin
         $x_i = x_{i-1} + 16$ 
         $y_i = y_{i-1}$ 
         $SW_{reuse} = \text{SearchWin}(\vec{mv}_i) \cap \text{SearchWin}(\vec{mv}_{i-1})$ 
         $SW_{loaded} = \text{SearchWin}(\vec{mv}_i) - SW_{reuse}$ 
         $SW_i = SW_{loaded} \cup SW_{reuse}$ 
      end
    else
       $SW_i = SW_{loaded} = \text{SearchWin}(\vec{mv}_i)$ 
       $\text{MotionEstimation}(MB_i, SW_i)$ 
    end
  end

```

Fig. 3 The Level C scheme for transcoding.

pixel difference in x -coordinate. As shown in **Fig. 3**, successive two MBs can share part of search window (SW_{reuse}) when MV field is smooth; otherwise search window should be flushed.

The Level C scheme for transcoding is shown in Fig. 3. $\vec{mv}_i$ and $\vec{mv}_{i-1}$ are current and previous MPEG-2 MV; (x_i, y_i) and (x_{i-1}, y_{i-1}) are their coordinates; t is a predefined threshold which is set to 6; MotionEstimation represents the function of ME performed within SW_i ; SearchWin is the function to determine search window based on $\vec{mv}_i$.

2.3.2 Performance Evaluation

Two factors must be taken into consideration to evaluate performance of data reuse scheme: on-chip memory size for reference data and redundancy access factor⁸⁾. The on-chip memory size represent the required buffer of reference data. The redundancy access factor R_α evaluates external bandwidth and is defined as the number of reference pixel be loaded for each MB pixel. R_α of the proposed Level C method is calculated as the expectation of R_α of all MB, which is shown as following

$$R_\alpha = p_{reuse} \cdot R_{\alpha(\text{Level C})} + (1 - p_{reuse}) \cdot R_{\alpha(\text{No Reuse})} \quad (1)$$

where p_{reuse} is the probability that a MB can reuse search window. $R_{\alpha(\text{Level C})}$ and $R_{\alpha(\text{No Reuse})}$ are calculated as following

$$R_{\alpha(\text{Level C})} = \frac{N(sr_v + N - 1)}{N^2} \quad (2)$$

$$R_{\alpha(\text{No Reuse})} = \frac{(sr_h + N - 1)(sr_v + N - 1)}{N^2} \quad (3)$$

Based on the hardware design result, the on-chip memory is equal to 4/3 times of the size of search window, which is $(sr_h + N - 1)(sr_v + N - 1)$. This is because reference pixel buffer must be flushed when the coordinate difference is larger than the threshold; and the next MB employs different control schedule to load reference pixel.

In Ref. 1), it is shown that more than 90% MB in HDTV720p sequence can reuse search window, while the average reuse rate is 84.2% in CIF sequence. It is also shown that video quality is almost not degraded although there exists MV modification.

Table 1 Bandwidth and on-chip memory comparison.

		R_α	On-chip Memory (Kbyte)
H.264/AVC (Level C)		8.94	20.0
H.264/AVC (Level C+)	2 MB	4.97	24.7
	4 MB	2.98	35.6
Transcoder (No Reuse)		8.63	2.2
Transcoder (Level C, Proposed)		3.51	2.9

2.3.3 Bandwidth and On-chip Memory Comparison

The bandwidth and on-chip memory of four IME modules are compared based on Refs. 1), 7) and 8). The type of test sequence is HDTV720p. The search range is set to $[-64, 64]$ for regular H.264/AVC and $[-16, 16]$ for H.264/AVC encoder end in transcoder.

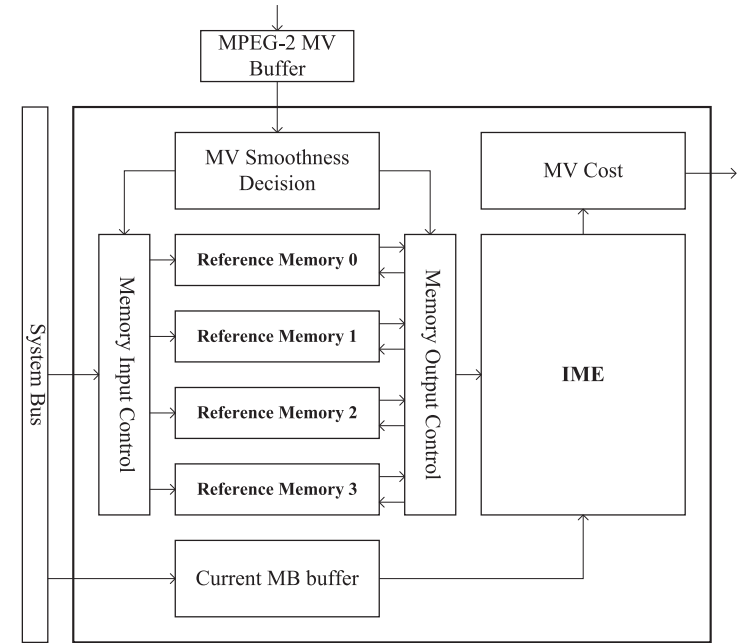
- H.264/AVC (Level C)—a regular H.264/AVC IME module with Level C scheme.
- H.264/AVC (Level C+)—a regular H.264/AVC IME module with Level C+ scheme (2 and 4 MB stitched vertically).
- Transcoder (No Reuse)—an IME module in H.264/AVC encoder end of transcoder, no data reuse scheme.
- Transcoder (Level C, Proposed)—a transcoding IME module in H.264/AVC encoder end of transcoder, Level C scheme. p_{reuse} is set to 0.9¹⁾.

It is observed from **Table 1** that the redundancy access factor R_α of the proposed Level C scheme for transcoding is at the same level of regular H.264/AVC IME with Level C+ scheme. But the on-chip is smaller (11.7% of 2 MB stitched and 8.1% of 4 MB stitched). The proposed method achieves 40.6% bandwidth of transcoder without any data reuse scheme, while the on-chip memory is 4/3 times of that.

3. Hardware Architecture for Level C Scheme

3.1 Top-Level Architecture

Figure 4 shows top-level architecture of the proposed transcoder IME module for HDTV720p application. There exist four reference pixel memories, each of which is a 47×16 bytes single-port SRAM. Actually the search window is 47×47 bytes. The applied memories size is for an easy hardware implementation. The

**Fig. 4** The schema of transcoder IME architecture.

memory update and output is controlled by memory input and output control unit, which is controlled by MV smoothness decision unit. The IME module is implemented with Partial SAD architecture.

3.2 Performance Analysis

Given the working frequency (f) of IME module, extra frequency (f_{extra}) for initialization latency and number of PEG (m), the processing ability of IME is expressed as $(f - f_{extra}) \times m$. The number of reference pixel to be processed is expressed as $(r \times n_{ref} \times H \times W \times sr_v \times sr_h)/256$, while r is frame rate, n_{ref} is number of reference frame, W and H is frame width and height. Therefore Eq. (4) must be satisfied to process specific video sequence.

$$(f - f_{extra}) \times m \geq \frac{r \times n_{ref} \times H \times W \times sr_v \times sr_h}{256} \quad (4)$$

In the discussed transcoding application of HDTV720p, f_{extra} is set to 16

clocks because this number of latency clocks are needed to produced SAD of the first reference position. If one PEG is used to process HDTV720p video stream (1 reference frame, search range $[-16, 16]$), working frequency f must satisfy Eq. (5). Therefore the longest critical path delay must be less than 9.04 ns.

$$f \geq \frac{30 \times 1 \times 720 \times 1,280 \times 32 \times 32}{256 \times 1} + 16 \simeq 110.6 \text{ MHz} \quad (5)$$

3.3 IME Architecture

In H.264/AVC, VBSME is adopted. For one MB, it has 4 block modes (16×16 , 16×8 , 8×16 , 8×8). For 8×8 mode, it is further divided into 4 modes, namely 8×8 , 8×4 , 4×8 , 4×4 . For each MB, coding costs of all 7 block mode are calculated, and the block mode with the smallest cost is chosen as the MB mode. Compared with fixed block size ME algorithm, VBSME provides higher compression ratio, but it also puts heavy burden on the ME module. In hardware design, partial SAD reuse methodology is adopted to reduce computation complexity, which means the SAD of smaller blocks are stored and accumulated to get the SAD of bigger ones.

The SAD Tree and Partial SAD architecture are proposed by Ref. 10) to realize VBSME. SAD Tree architecture is suitable for highly parallelized application and can share reference buffer between parallel PEG. But it has long critical path delay, which is 14.1 ns based on our implementation. This delay cannot meet the performance requirement according to Section 3.2. To reduce delay, 16 12-bit registers can be inserted between $\text{SAD}4 \times 4$ and larger block's SAD addition to form a 2-stage pipeline. When applying snake-scan processing order, one PEG (256 PE) of SAD Tree architecture needs $16 \times 12 + 16 \times 17 \times 8 = 2,368$ bit register.

Partial SAD architecture has smaller gate count and suitable for medium and small resolution videos. Another advantage is that it has shorter critical path delay compared with SAD Tree because partial SAD is stored and propagated by propagation and delay register. If one PEG (256 PE) of Partial SAD is used, it needs 1,872 bit register. Therefore 496 bit register can be saved compared to SAD Tree architecture. In this paper, the Partial SAD architecture is chosen to implement IME architecture as shown in Fig. 5.

3.4 Memory Input and Output Control Unit

The memory input and output control units must achieve two primary goals:

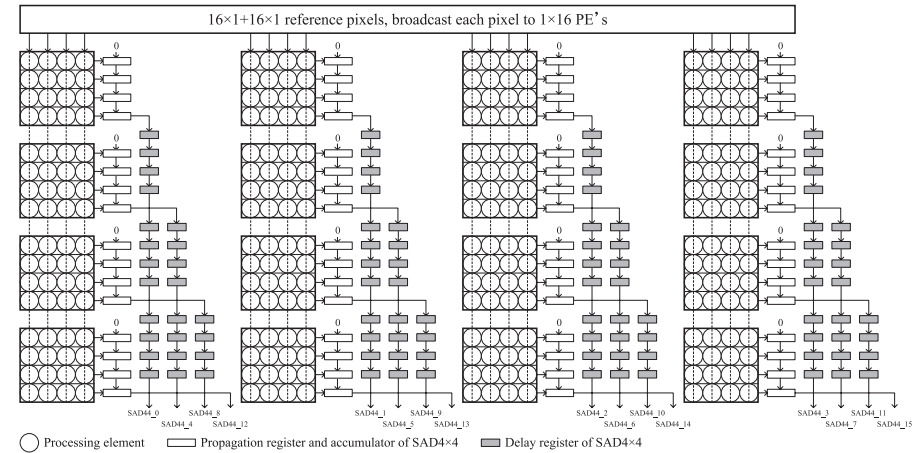


Fig. 5 The partial SAD architecture of VBSME.

1) avoid memory input and output confliction; 2) keep IME module to be fully utilized, which means the ME operation must has no stall. These objects are achieved by apply ping-pong strategy. That is when one SRAM is used, the other one is updated.

The proposed architecture contains four memory banks (Mem 0–3) for storage of reference pixel. Two memories are involved to perform ME of 47×16 reference pixels. Reference pixel for ME operation of each MB is stored in three memory banks. In our design, Mem 0–2 are circularly accessed when MV field is smooth; Mem 3 is used when MV field is non-smooth.

An example of memory transition is presented in Fig. 6 to show the generation of memory control signal. The associate Finite State Machine (FSM) is shown in Fig. 7. The FSM is composed of nine states, whose state labels indicate which memory to output and the output order. The decision about whether MV field is smooth is made in S01, S12 and S20, based on which next state is decided. For example, if MV field is smooth in S20, the next state is S20a. This means that Mem 2 and Mem 0 can be reused between search windows of successive two MBs. In S20a, Mem 1 must be updated since it will be accessed in next state S01. If MV field is non-smooth in S20, the next two states are S31 and S12. Mem 3 and

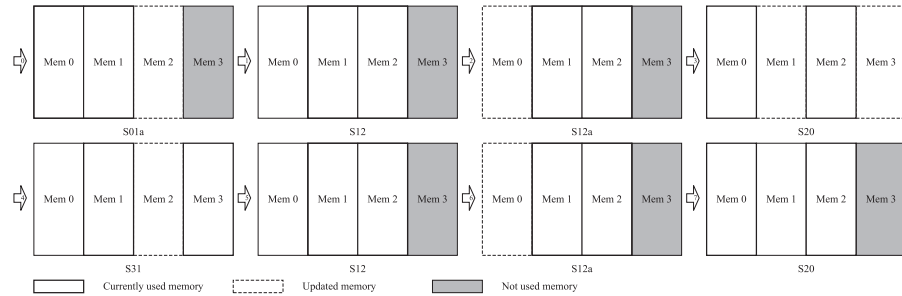


Fig. 6 An example of memory transition.

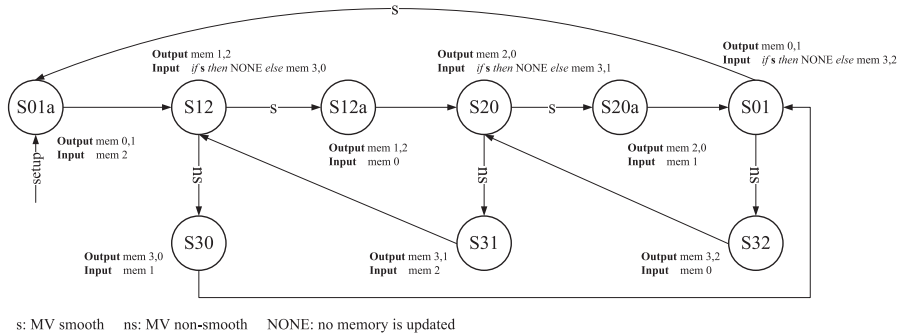


Fig. 7 The finite state machine of memory control unit.

Mem 1 is updated in S20; Mem 2 is updated in S31.

Table 2 shows the memory output, update and motion estimation sequence following the example shown in Fig. 6. In the Partial SAD architecture, $32 \times 16 + 15 = 527$ clocks are needed to process 47×16 reference pixels. 47 clocks are needed to update one memory bank. Therefore concurrent memory input and output do not lead to any memory accessing confliction. For example in state S01a, Mem 2 is updated from clock 0 to clock 46 because it will be used in next state S12, while ME is performed by using Mem 0 and Mem 1 ranged from clock 0 to clock 526. Two memory banks should be updated when MV filed is non-smooth, such as in state S20 ranged from clock 1,581 to 2,107.

In HDTV720p application, more than 90% MB¹⁾ just uses Mem 0–2. Therefore Mem 3 can be disabled to save power in this situation.

Table 2 Memory update and motion estimation sequence.

Clock	State	Δmv	Mem 0	Mem 1	Mem 2	Mem 3	ME
0	S01a		Output	Output	Update		ME(0,1)
46			Output	Output	Update		ME(0,1)
526			Output	Output			ME(0,1)
527	S12	S		Output	Output		ME(1,2)
1,053				Output	Output		ME(1,2)
1,054	S12a		Update	Output	Output		ME(1,2)
1,100			Update	Output	Output		ME(1,2)
1,580			Output	Output	Output		ME(1,2)
1,581	S20	NS	Output	Update	Output	Update	ME(2,0)
1,627			Output	Update	Output	Update	ME(2,0)
2,107			Output	Output	Output		ME(2,0)
2,108	S31			Output	Update	Output	ME(3,1)
2,154				Output	Update	Output	ME(3,1)
2,634				Output	Update	Output	ME(3,1)
2,635	S12	S		Output	Output		ME(1,2)
3,161				Output	Output		ME(1,2)
3,162	S12a		Update	Output	Output		ME(1,2)
3,208			Update	Output	Output		ME(1,2)
3,688			Output	Output	Output		ME(1,2)
3,689	S20	S	Output		Output		ME(2,0)
4,215			Output		Output		ME(2,0)

S: smooth NS: non-smooth ME (Memory number 1, Memory number 2)

4. Experiment Result

The proposed architecture is realized with Verilog HDL and synthesized by Synopsys Designer Compiler with TSMC 0.18 μm technology. Working condition is set to the worst (1.62 V, 125°C). In the following discussion, experiment data is the simulation result from Synopsys Designer Compiler.

The implementation conditions and hardware cost of the proposed transcoding architecture are summarized in **Table 3**. The critical path delay in worst condition is 4.67 ns, which meets the frequency requirement discussed in Section 3.2. It is observed that the proposed design can achieve 110.6 MHz with 98.2K NAND gate. In Table 3, “Current MB” is the buffer needed to storage MB data; “Partial SAD” is the Partial SAD architecture; “Min SAD” is the hardware module to calculate minimum SAD and MV; “Control” is the the control unit; “Total” is the hardware cost of the whole architecture.

Table 4 shows hardware cost comparison with a regular IME module. It is

Table 3 VLSI implementation result.

Technology	TSMC 0.18 μ m 1P6M
Operating Conditions	1.62 V, 125°C
Frequency	110.6 MHz
Current MB	15.7K
Partial SAD	68.4K
Min SAD	13.1K
Control	1,699
Total	98.2K

Table 4 Performance comparison.

	Ref. 11)	Proposed
# of PE	128×8	256
Process	0.18 μ m	0.18 μ m
Gate Count	305K	98.2K
Frequency	81/108 MHz	110.6 MHz
Application	SDTV/HDTV720p	HDTV720p
Search Window	128×64	16×16
# of Ref.	4/1	1

observed that the number of PE and search window size is reduced, which benefits from precise search center indicated by MPEG-2 MV.

5. Conclusion

An IME architecture for MPEG-2 to H.264/AVC transcoding is proposed for HDTV720p application. The IME architecture and ping-pang memory control logic are discussed in this paper. Combined with the Level C search window reuse method, the proposed architecture can reach the bandwidth level of regular H.264/AVC IME module with Level C+ scheme, while the on-chip memory is at most 11.7% of that.

Acknowledgments This research was supported by “Ambient SoC Global COE Program of Waseda University” of the Ministry of Education, Culture, Sports, Science and Technology, Japan.

References

- 1) Wei, X.H., Li, S., Song, Y. and Goto, S.: An Irregular Search Window Reuse Scheme for MPEG-2 to H.264 Transcoding, *IEICE Trans. on Fundamentals of*

- Electronics, Communications and Computer*, Vol.E91-A, No.3, pp.749–755 (Mar. 2008).
- 2) Xin, J., Lin, C.-W. and Sun, M.-T.: Digital Video Transcoding, *Proc. IEEE*, Vol.93, No.1, pp.84–97 (Jan. 2005).
- 3) Information Technology—Generic Coding of Moving Pictures and Associated Audio Information: Video, Int. Standard 13818-2, ITU-T Recommendation H.262, 2nd ed. (Feb. 2000).
- 4) Advanced Video Coding for Generic Audiovisual Services, Int. Standard 14496-10, ITU-T Recommendation H.264. (Mar. 2005).
- 5) Lu, X., Tourapis, A., Yin, P. and Boyce, J.: Fast mode decision and motion estimation for H.264 with a focus on MPEG-2/H.264 transcoding, *IEEE International Symposium on Circuits and Systems, 2005 (ISCAS 2005)*, Vol.2, pp.1246–1249 (23–26 May 2005).
- 6) Zhou, Z., Sun, S., Lei, S. and Sun, M.-T.: Motion information and coding mode reuse for MPEG-2 to H.264 transcoding, *IEEE International Symposium on Circuits and Systems, 2005 (ISCAS 2005)*, Vol.2, pp.1230–1233 (23–26 May 2005).
- 7) Tuan, J.-C., Chang, T.-S. and Jen, C.-W.: On the data reuse and memory bandwidth analysis for full-search block-matching VLSI architecture, *IEEE Trans. on Circuits and Systems for Video Technology*, Vol.12, No.1, pp.61–72 (Jan. 2002).
- 8) Chen, C.-Y., Huang, C.-T., Chen, Y.-H. and Chen, L.-G.: Level C+ data reuse scheme for motion estimation with corresponding coding orders, *IEEE Trans. on Circuits and Systems for Video Technology*, Vol.16, No.4, pp.553–558 (Apr. 2006).
- 9) Ahmad, I., Wei, X., Sun, Y. and Zhang, Y.-Q.: Video transcoding: An overview of various techniques and research issues, *IEEE Trans. on Multimedia*, Vol.7, No.5, pp.793–804 (Oct. 2005).
- 10) Chen, C.-Y., Chien, S.-Y., Huang, Y.-W., Chen, T.-C., Wang, T.-C. and Chen, L.-G.: Analysis and architecture design of variable block-size motion estimation for H.264/AVC, *IEEE Trans. on Circuits and Systems I: Regular Papers*, Vol.53, No.3, pp.578–593 (Mar. 2006).
- 11) Chen, T.-C., Chien, S.-Y., Huang, Y.-W., Tsai, C.-H., Chen, C.-Y., Chen, T.-W. and Chen, L.-G.: Analysis and architecture design of an HDTV720p 30 frames/s H.264/AVC encoder, *IEEE Trans. on Circuits and Systems for Video Technology*, Vol.16, No.6, pp.673–688 (June 2006).

(Received May 23, 2008)

(Revised August 22, 2008)

(Accepted October 9, 2008)

(Released February 17, 2009)

(Recommended by Associate Editor: Masaya Yoshikawa)



Xianghui Wei received the B.S. in Information Engineering from Beijing Institute of Graphical Communication in 1997. He is currently a Ph.D. candidate in Graduate School of Information, Production and Systems, Waseda University, Japan. His research interests include video coding technology, Optical Character Recognition (OCR) and artificial intelligence.



Takeshi Ikenaga received his B.E. and M.E. degrees in electrical engineering and the Ph.D. degree in information & computer science from Waseda University, Tokyo, Japan, in 1988, 1990 and 2002 respectively. He joined LSI Laboratories, Nippon Telegraph and Telephone Corporation (NTT) in 1990, where he has been undertaking research on the design and test methodologies for high performance ASICs, a real-time MPEG2 encoder chip set, and a highly parallel LSI & system design for image understanding processing. He is presently an associate professor in the system LSI field of the Graduate School of Information, Production and systems, Waseda University. His current interests are application SoCs for image, security and network processing. Dr. Ikenaga is a member of the IPSJ and the IEEE. He received the IEICE Research Encouragement Award in 1992.



Satoshi Goto was born on January 3rd, 1945 in Hiroshima, Japan. He received the B.E. degree and the M.E. degree in Electronics and Communication Engineering from Waseda University in 1968 and 1970, respectively. He also received the Dr. of Engineering from the same university in 1981. He is IEEE fellow, Member of Academy Engineering Society of Japan and professor of Waseda University. His research interests include LSI System and Multimedia System.