

A 125 μ W, Fully Scalable MPEG-2 and H.264/AVC Video Decoder for Mobile Applications

Tsu-Ming Liu, *Student Member, IEEE*, Ting-An Lin, *Student Member, IEEE*, Sheng-Zen Wang, Wen-Ping Lee, Jiun-Yan Yang, Kang-Cheng Hou, and Chen-Yi Lee, *Member, IEEE*

Abstract—A low-power dual-standard video decoder has been developed for mobile applications. It supports MPEG-2 SP@ML and H.264/AVC BL@L4 video decoding in a single chip and features a scalable architecture to reach area/power efficiency. This chip integrates diverse algorithms of MPEG-2 and H.264/AVC to reduce silicon area. Three low-power techniques are proposed. First, a domain-pipelined scalability (DPS) technique is used to optimize the pipelined structure according to the number of processing cycles. Second, bandwidth scalability is implemented via a line-pixel-lookahead (LPL) scheme to improve the external bandwidth and reduce the internal memory size, leading to 51% of memory power reduction compared to a conventional design. Third, low-power motion compensation and deblocking filter are designed to reduce the operating frequency without degrading system performance. A test chip is fabricated in a 0.18 μ m one-poly six-metal CMOS technology with an area of 15.21 mm². For mobile applications, H.264/AVC and MPEG-2 video decoding of quarter-common intermediate format (QCIF) sequences at 15 frames per second are achieved at 1.15 MHz clock frequency with power dissipation of 125 μ W and 108 μ W, respectively, at 1 V supply voltage.

Index Terms—H.264/AVC, inverse discrete cosine transform (IDCT), mobile communication, motion compensation, MPEG-2, video coding.

I. INTRODUCTION

RECENTLY, portable devices such as cellular phones, video camcorders, personal digital assistants and handheld digital TVs are becoming increasingly popular. The portable requirement implies an important issue on power reduction. However, the increased power consumption generally comes from the sophisticated algorithms and architectural challenges, especially in newly announced H.264/AVC video coding standard [1]. Therefore, a hardware solution, which improves system performance and achieves less power consumption, is demanded when multimedia capabilities are offered in portable systems [2]–[5].

The advent of H.264/AVC provides high compression ratio, but it is not backward compatible to the prevalent MPEG-x and H.26x families of video coding standards. For instance, in the video broadcasting space, DVB-T has paved the way for the introduction of MPEG-2 based digital TV services. Mobile DVB,

presently called DVB-H, allows the transmission of H.264/AVC signals due to its bandwidth-efficiency. However, DVB-H is backward compatible to DVB-T but is transmitted with different video contents (i.e., MPEG-2 versus H.264/AVC). This leads to the design challenge for integrating H.264/AVC and MPEG-2 video standards. Although several high-performance MPEG-2 [6] and H.264/AVC [7]–[10] video processors have been reported of the time, these solutions used separate modules and only decoded a single type of video content in each module. To reach better area-efficiency and support multi-standard video requirements, a new architecture has been recently developed to integrate both MPEG-2 and H.264/AVC in a single chip [4].

In this paper, we present a low-power MPEG-2 SP@ML and H.264/AVC BL@L4 video decoder, which can be applied to mobile phone applications for both low-delay and low-memory (i.e., without B-frames) requirements. To realize an area/power-efficient architecture, IDCT and deblocking filter have been restructured in both algorithmic and architectural levels. Furthermore, three low-power techniques have been proposed. First, the conventional architectures [11]–[13] apply fixed pipelined registers without considering the cycle characteristics in each functional unit. In our design, whole modules have been partitioned into two pipelined domains: cycle-critical and non-cycle-critical domains. Based on different pipelined domains, the fine-level (i.e., $4 \times 4/8 \times 8$) and coarse-level (i.e., 16×16) pipelines are applied to non-cycle-critical and cycle-critical domains respectively. This method optimizes the number of pipelined registers according to the processing cycles. Second, we aim at a memory hierarchy where copies of data from larger memories that exhibit high data-correlation are stored to additional layers of smaller memories [14]. Specifically, we allocate a line buffer for the storage with rows of pixels [5], [10]. However, storing a full row of pixels is unnecessary since not all pixels will be referenced by the following decoding procedures. To deal with the above problems, we propose a line-pixel-lookahead (LPL) scheme to predict whether the pixel data should be kept or not, resulting in less power dissipation of a crucial part of H.264/AVC. Third, to achieve a high-throughput decoding procedure with low operating frequency, we reuse the neighboring data to reduce the processing cycles and lower the required working frequency in both motion compensation and de-blocking filter, which are the most critical part in our system profiling. Altogether, these techniques not only improve area efficiency but also save power.

The organization of this paper is as follows. Section II presents a cost-effective design integrating both MPEG-2 and H.264/AVC video standards. Section III describes three

Manuscript received May 1, 2006; revised July 22, 2006. This work was supported by the National Science Council of Taiwan, R.O.C., under Grant NSC94-2215-E-009-046, and by the NCTU-MTK Research Program.

The authors are with the Department of Electronics Engineering, National Chiao-Tung University, Hsinchu 300, Taiwan, R.O.C. (e-mail: mingle@si2lab.org; cylee@si2lab.org).

Digital Object Identifier 10.1109/JSSC.2006.886542

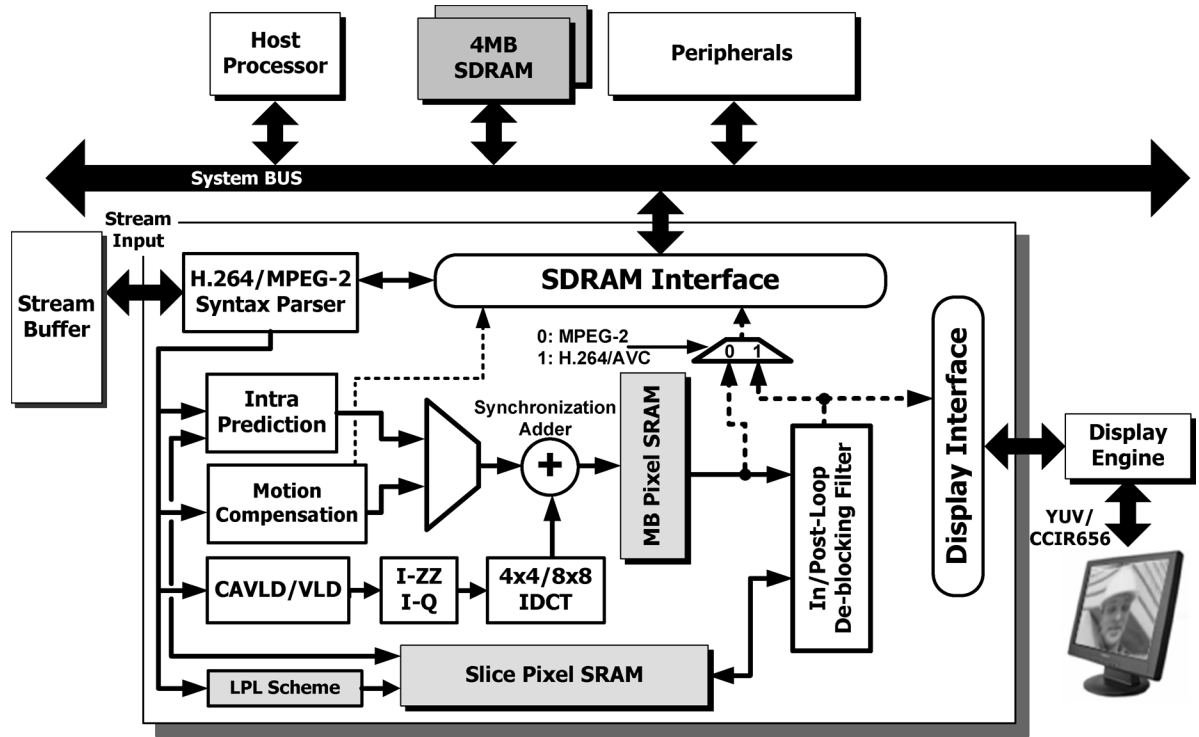


Fig. 1. System block diagram.

novel low-power techniques incorporated in this chip design. Implementation results are summarized in Section IV and conclusions are made in Section V.

II. MPEG-2 AND H.264/AVC DECODER ARCHITECTURE

A. Basic Configuration

Fig. 1 shows the basic configuration of the dual-mode video decoder chip. Host processor controls all modules and allocates the memory for different modules. The main block is a MPEG-2/H.264 video decoder that has a dedicated input for the MPEG-2 or H.264/AVC bistream and a dedicated output for displaying. In the dual-video decoder core, a 22.75 Kb embedded SRAM is employed to store local pixel data, which yields an acceptably small size and reduced power penalties. Two 4 MB external frame memories are connected to SDRAM interface (I/F) via a 64-bit system bus. Using two memories makes SDRAM I/F simple, and the wider bus can improve the external processing cycles. Accessing SDRAM is issued by both motion compensation and deblocking filter. To reduce the bandwidth between external frame memory and deblocking filter, a separate peripheral is explored for on screen display (OSD) through a direct display I/F. Other peripherals, interfaces for camera, mic/speaker, network, etc., are connected via a system bus. Note that most of functional blocks in H.264/AVC are similar to those in MPEG-2, including the syntax parser, entropy decoder, and motion compensation. First, we implement a custom-built syntax parser and exploit a register sharing technique to reduce register numbers. Second, one codeword cannot be the prefix of another codeword in a table but this rule does not hold among different standards. Hence, most of VLC code-words can be merged in both standards. Third,

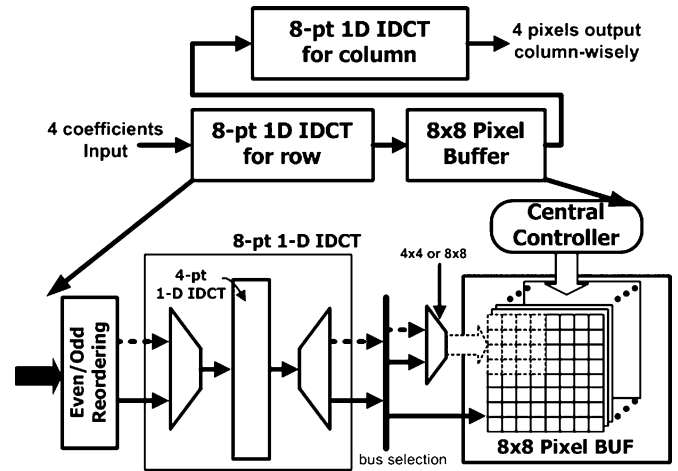


Fig. 2. The $4 \times 4/8 \times 8$ IDCT core block diagram.

both motion compensations intend to perform interpolation procedures. Several adders and multipliers can be combined by applying resource sharing techniques. Although the aforementioned modules improve the integration issue, the inverse transforms between MPEG-2 and H.264/AVC are so diverse that they are difficult to combine. Similarly, the integration of deblocking filters has the same problem as well. To enhance area-efficiency, we propose solutions for $4 \times 4/8 \times 8$ IDCT and in/post-loop deblocking filter in the following subsections.

B. $4 \times 4/8 \times 8$ IDCT Core

A major focus in integrating MPEG-2 and H.264/AVC is IDCT since it faces the most diverse algorithms over the whole design. As we know, the IDCT kernel of H.264/AVC is a 4×4

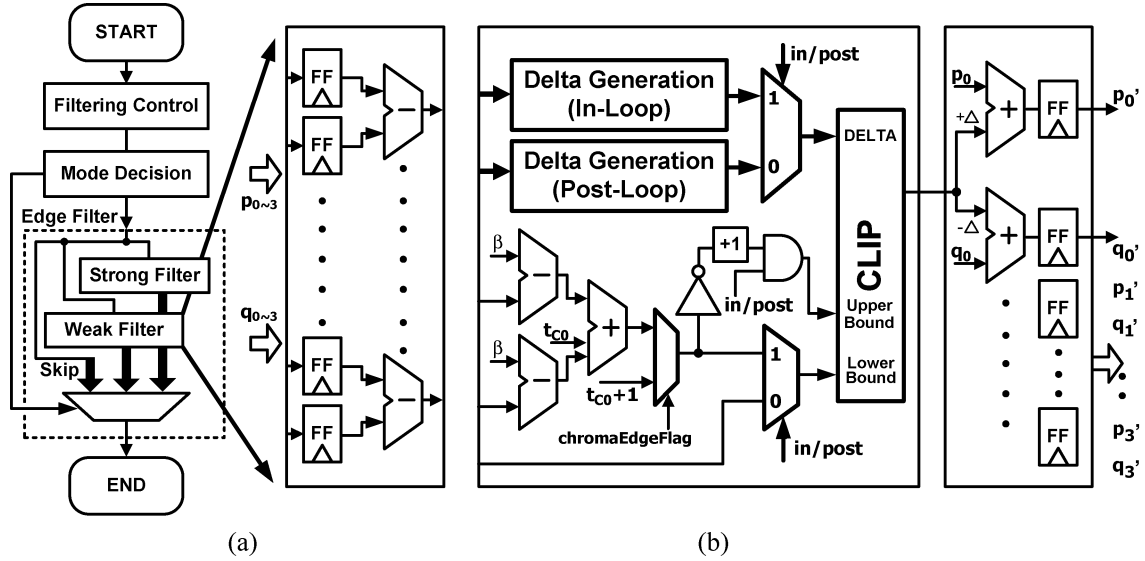


Fig. 3. (a) Combined in/post-loop filter block diagram. (b) Weak filter.

integer transform kernel but that of MPEG-2 is an 8×8 cosine transform kernel. Due to the difference, existing solutions contain two individual IDCT modules without sharing. By contrast, we propose a shared $4 \times 4/8 \times 8$ IDCT in Fig. 2. It is composed of two 1-D IDCTs for row and column transforms respectively, and an 8×8 pixel buffer for matrix transposition. The 8×8 IDCT can be computed by using 4×4 IDCT recursively [15]. In other words, $2^m \times 2^m$ IDCT can be decomposed into $2^{m-1} \times 2^{m-1}$ IDCT by reordering even and odd coefficients and selectively storing IDCT results into pixel buffers. The word-length of each operating unit is 16-bit in order to meet both standard requirements and each data path in Fig. 2 stands for four-pixel (i.e., 4×16 -bit) operation. The dotted lines perform the 4×4 IDCT in H.264/AVC. Moreover, the 8×8 IDCT in MPEG-2 can also be performed in a 4×4 fashion and one-fourth of pixel buffers are shared for different IDCT operations. Although 8×8 IDCT intends to use multipliers to realize inverse transforms, we replace these multiplications with a series of shifts and additions [16]. Therefore, these operating units can be shared with 4×4 IDCT to reduce silicon area. Finally, this proposal saves 15% gate-count than the one without exploiting any hardware sharing.

C. In/Post-Loop Deblocking Filter Core

The in-loop filter is standardized by H.264/AVC and the post-loop filter follows the prevalent MPEG-x standards. In general, the visual quality improvement is very small (0.04 dB) under mobile environments if we put the in-loop filter of H.264/AVC into MPEG-2 decoding flow. To alleviate this problem, we derive a new algorithm that can be reconfigured as an in-loop or post-loop filtering process. It is totally compatible to the in-loop filter but improves visual quality in the post-loop operations. Specifically, the derived algorithm can be divided into three main components. They are the filtering control, mode decision, and edge filter as shown in Fig. 3(a). The filtering control decides the filtered order and the size of filtered boundaries. The mode decision governs the filtering intensity in that boundary

and thereby dispatches the filtered mode to the select-pin of the multiplexer. The edge filter operates on a specified boundary and smoothes out the discontinuities with pre-defined coefficients. Fig. 3(b) depicts the detailed circuit of the weak filter. Generally, it needs a great number of operations and greatly influences the visual quality. It takes 4-pixel (i.e., $p_0 \sim p_3$, $q_0 \sim q_3$) on either side of the boundary to perform interpolation procedures. In particular, a pixel-wise difference is applied, and a delta metric is generated. A CLIP¹ operation limits the delta metric between y (i.e., Upper Bound) and x (i.e., Lower Bound). Finally, the CLIP's outputs add and subtract the raw pixels to obtain the filtered results. Because the weak filter is the most quality-intensive process in the deblocking filter, we share most of operations except "Delta Generation" and make a better trade-off between visual quality and area efficiency. As a result, the synthesized logic gate counts can be reduced by 30% compared to the preliminary design that implements in-loop or post-loop filter separately.

III. LOW-POWER TECHNIQUES

A. Reducing the Pipelined Registers

It is obvious that reducing the number of registers cuts the logic and clock-tree power dissipation. In general, pipelined registers are required for data transactions among modules. However, traditional pipelined methods used fixed number of pipelined registers over the whole design, leading to additional register power. To alleviate this problem, we propose a domain-pipelined scalability (DPS) in Fig. 4. This method partitions the circuits into two pipelined domains. One of them includes the cycle-critical path that consumes a great number of processing cycles from stream inputs to outputs (e.g., motion compensation, deblocking filter, display I/F), and

$$^1 \text{CLIP}(x, y, z) = \begin{cases} x; & z < x \\ y; & z > y \\ z; & \text{otherwise} \end{cases}$$

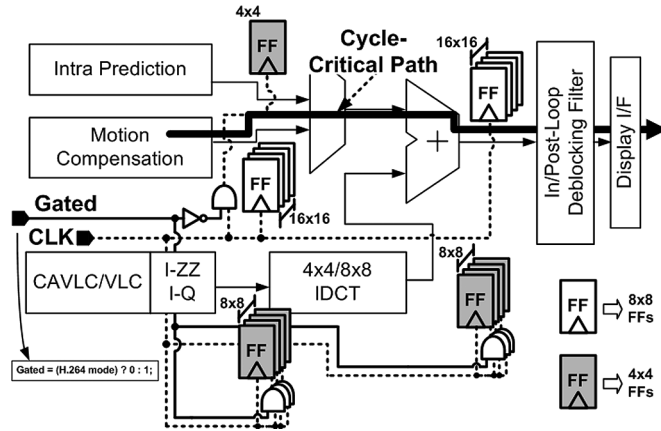


Fig. 4. Two pipelined domains by using the DPS method.

TABLE I
DIFFERENT PIPELINE LEVELS IN EACH MODULE

Cycle Characteristics	Key Module	Proposed	
		MPEG-2	H.264/AVC
Non-cycle-critical path	Intra Prediction	N/A	4x4
	VLD/CAVLD	8x8	
	4x4/8x8 IDCT		
Cycle-critical path	Motion Compensation	16x16	16x16
	In/Post-Loop Filter		
	Display I/F		

the other includes the non-cycle-critical path that indicates the path except for cycle-critical one (e.g., entropy decoder, IDCT, intra prediction). On the other hand, there are several pipelined alternatives at design time such as 4x4 (16 FFs), 8x8 (64 FFs) and 16x16 (256 FFs) levels. As we know, a fine-level (i.e., 4x4/8x8-level) pipeline introduces bubbles or waiting cycles frequently on each 4x4/8x8-level while a coarse-level (16x16-level) pipeline improves the processing cycles since the waiting cycles are reduced and only occur on each 16x16-level [5]. Therefore, we only apply the fine-level pipeline into the non-cycle-critical path for reducing the pipeline registers. By contrast, the coarse-level pipeline is utilized to eliminate the waiting cycles on the cycle-critical path. With regard to the integration issue, a 4x4 sub-block is the smallest pipelined element in H.264/AVC while an 8x8 block size is adopted by MPEG-2 video standard. Due to different pipelined sizes, we utilize AND gates to disable unused flip-flops according to the pipelined operation in each standard. Thus, the proposed DPS method considers the processing capabilities in each module when allocating the number of pipelined registers. It is suitable for determining the optimized pipelined levels during the system development. The detailed pipeline structures are listed in Table I. Compared to the unoptimized 16x16-level pipelines [11], the proposed DPS method reduces the number of pipelined registers by 37.5%, resulting in less power dissipation.

B. Improving the Memory Hierarchy

Improving the memory hierarchy or reducing the embedded SRAM size is very effective for achieving low power dissipation

because internal memories occupy about 70% of core power dissipation [5]. Fig. 5(a) depicts a three-level memory hierarchy where a slice pixel SRAM is allocated for the storage with rows of pixels since H.264/AVC features to access logically adjacent pixels in the vertical direction. However, storing all pixels in rows of vertical pixels is unnecessary when the following decoding process is unrelated to the upper neighboring pixels. Hence, we propose a line-pixel-lookahead (LPL) scheme to eliminate the unused pixels. This scheme achieves less power dissipation in a crucial part of video decoding system.

Fig. 5(b) depicts the slice pixel SRAM and LPL scheme to enhance access efficiency. In particular, a 19.2 kb slice pixel SRAM caches the pixels of upper neighbors, and a LPL scheme predicts whether the follow-up pixel data should be kept or not. In the LPL scheme, the TAG prediction issues a *Decoding TAG* (*D. TAG*) that contains a pair of signals for the purpose of deblocking filter and intra prediction units, and the *D. TAG* is equal to the *Neighboring TAG* (*N. TAG*) after buffering one row of TAGs. Two 2 W-bit TAG buffers record each *D. TAG*, where W means frame width. A TAG CMP (compare) unit perceives the contrast between *N. TAG* and *D. TAG*. A prediction miss will be noticed from the output of TAG CMP when current *D. TAG* differs from *N. TAG*.

To illustrate how the LPL scheme works, we partition it into two steps: 1) TAG prediction and 2) TAG buffer. Fig. 6 depicts the detailed circuit of the TAG prediction. The prediction step forecasts data accesses needed in advance, so a specific piece of data is pre-stored in the SRAM before it is actually desired by the follow-up decoding processes. A key observation is that not all upper neighboring pixels need to be pre-stored when they are determined as a “horizontal prediction mode” in intra prediction or a “SKIP mode” in deblocking filter. To realize the above behavior, we extract intra prediction mode, boundary strength (bS) and related header information from syntax parser as arrowed input signals. Each non-arrowed input is hard-coded and can be referenced by H.264/AVC specifications [1]. In the case of a TAG buffer, it is of size 2 W-bit and implemented by a single read and write port register file. The *N. TAG* will be read first from the TAG buffer for the TAG comparison. Afterward, the *D. TAG* signal writes into the TAG buffer for follow-up operations in next rows. Both reading and writing procedures are activated in different time slots to ensure that the TAG signal previously written to the buffer can be read back without error.

Fig. 7 describes the 4x4 intra prediction behavior of the LPL scheme through an example with a frame size of 48x32. Each square represents a 4x4 sub-block labeled by a 1-bit TAG signal. In the *N. TAG* field, we tag the 4x4 pixel data when a vertical prediction mode is applied. Furthermore, the untagged pixel will be discarded via *wen* [see the slice pixel SRAM in Fig. 5(b)], resulting in reducing memory size. The memory word-length is fixed at 8-bit and a correlation factor *f* is introduced to scale the address depth at design time. Thus, the memory size is scalable and proportional to W/f (instead of W in the beginning [5]) without degrading performance when the prediction hits (i.e., $D. TAG = N. TAG$). However, an error of prediction may occur (i.e., *miss*) so we need to fetch the missed data from the external memory. The miss rate stands for a probability of missing events and is equal to the

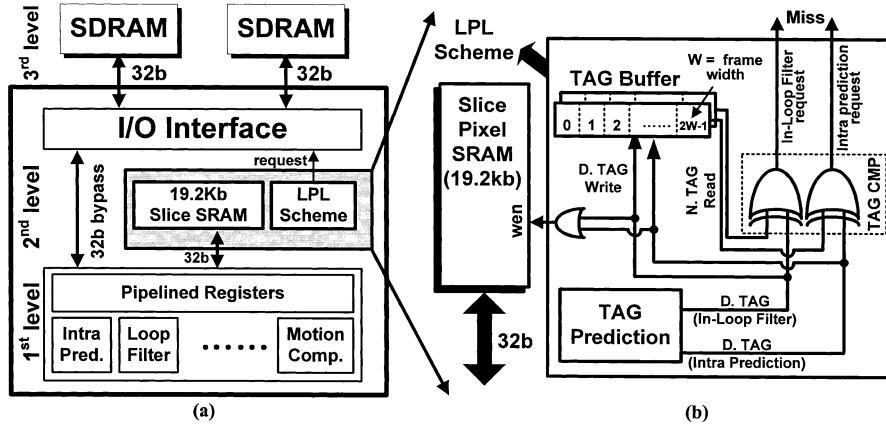


Fig. 5. (a) Three-level memory hierarchy. (b) LPL scheme.

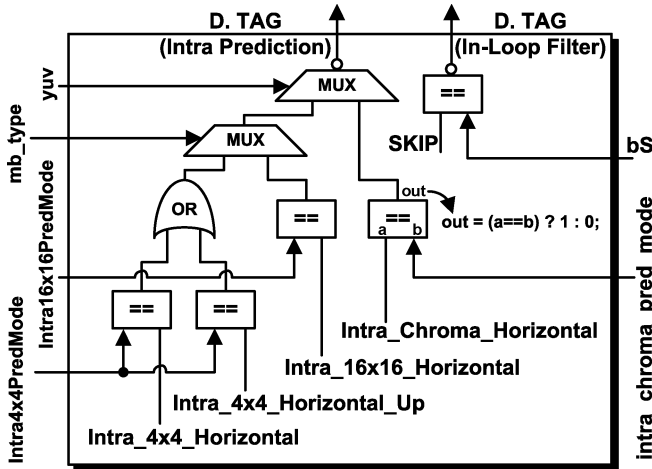


Fig. 6. Proposed prediction circuit for TAG prediction.

number of *miss* over one row of TAG. Although we reduce the memory size from $W \times 8$ to $(W/f) \times 8$ bits, 16.7% (i.e., 2/12) of miss rates are its penalties in this example. Therefore, it is indispensable to making a better compromise between the introduced miss rate and the correlation factor f .

Though the internal SRAM size can be reduced, there are penalties in terms of miss rate as well as external bandwidth, leading to the increment of DRAM power dissipation. An observation is that the curve between internal SRAM and external DRAM power consumption is shown in Fig. 8. Let us illustrate this property by choosing Micron’s SDRAM model (i.e., MT48LC2M32B2) with CAS latency = 2, BL = 1 and $t_{CK} = 7$ ns. In the X-axis, there are several design alternatives according to the factor f . In other words, we provide a scalable solution with a tradeoff at the architectural design time. As a result, a better compromise can be constrained by the minimal distance from origin (i.e., $f = 8$) since it achieves smaller SRAM size as well as SDRAM power dissipation. Note that this property can also be applied to DDR/DDR2 memory for high-resolution video decoding when adopting the proposed memory hierarchy with the LPL scheme.

Fig. 9 shows a power saving through the three-level memory hierarchy with the LPL scheme. The traditional three-level

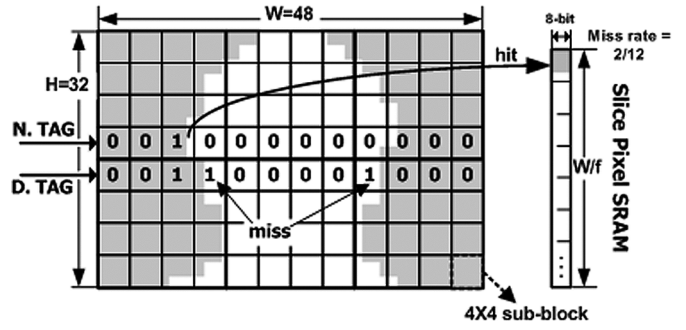


Fig. 7. Data organization between slice pixel SRAM and TAG prediction.

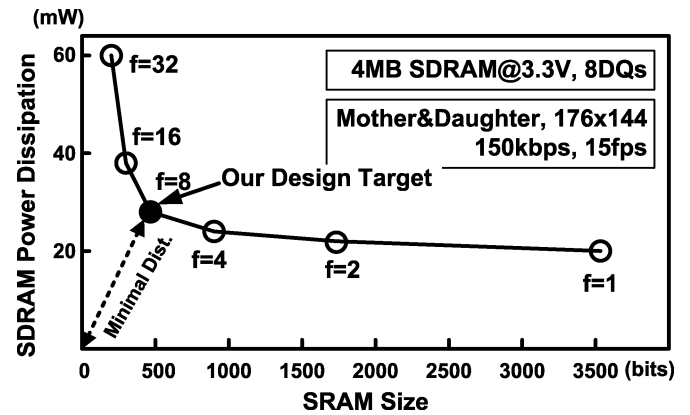


Fig. 8. Analysis of power dissipation on external SDRAM and internal SRAM.

memory hierarchy [5] in middle bar is just a special case when a correlation factor f equals one. While it reduces power dissipation by 44%, the SRAM power penalty is considerably high. To further reduce the SRAM power consumption, we propose the LPL scheme to make a better compromise from the observation in Fig. 8. Although the right-hand bar increases the SDRAM power by 4 mW, the SRAM power in the right-hand bar can be greatly reduced to 1/8 of that in the middle bar. Hence, the LPL scheme further gains 11% power dissipation. Altogether, the three-level memory hierarchy and LPL scheme achieve 51% power saving. The power improvement

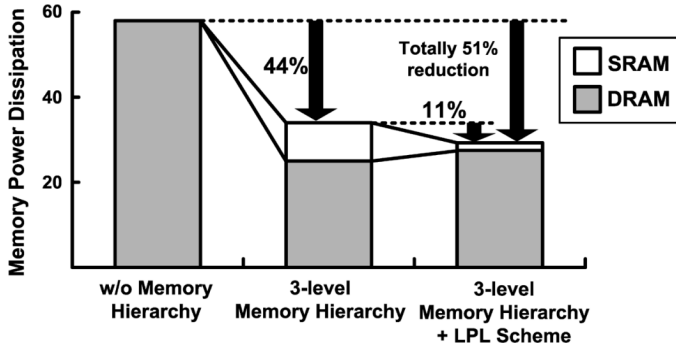


Fig. 9. Power saving on improved memory hierarchy.

for the LPL scheme becomes more significant when low-power SDRAMs are applied.

C. Low-Power Motion Compensation and Deblocking Filter

The proposed motion compensation (MC) and deblocking filter (DF) are designed to eliminate redundant memory accesses. They lower the required working frequency and thereby achieve low power consumption without degrading system performance. Under a 1080HD real-time decoding process, these low-power designs reduce the required frequency by approximately 60% with only a few additional buffers and logics.

The interpolation unit is always the most time-consuming module in the whole motion compensation core. A great deal of memory accesses degrade decoding throughput especially in the features of variable block size and quarter-pel resolution. To reduce memory access times, it is necessary to increase data reuse probability for overlapped regions of neighboring interpolation windows [17]. In Fig. 10(a) and (b), a dotted line shows the different scan orders in one luma macroblock, and the number in each square represents the decoding orders defined by H.264/AVC. In Fig. 10(a), a 2×2 raster scan is compliant to H.264/AVC, but extra cycles for data initialization are required when the dotted line turns. Compared to the 2×2 raster scan, a 4×4 raster scan features fewer turning events but violates the H.264/AVC standard. Since standard-limitation and cycle-efficiency are often at odds, an extended 2×2 raster scan has been proposed to improve decoding performance in Fig. 10(c). In particular, content registers, i.e., 6×9 pixel buffers, attached to shift registers for the interpolator are adopted. When sub-block #1 is decoded, overlapped windows in the right-hand side are stored in the background of the pixel buffers. These buffers switch into the foreground when decoding index moves to sub-block #3. Therefore, in the decoding of sub-block #4, the left overlapped window can be reused from the content registers instead of external memory, and the overall processing cycles can be reduced. Compared to the conventional design [12], the proposed motion compensation requires additional 6×9 pixel buffers (1% cost of MC) but saves 30% of access times.

The proposed deblocking filter reduces one-half of processing cycles with slightly incrementation of buffer cost [18]. Fig. 11(a) describes the filtering order where vertical edges are filtered first, followed by horizontal edges. Each square is sized to 4×4 pixel data. The numbers within the rectangles represent the processing order in one luma macroblock. The direct ap-

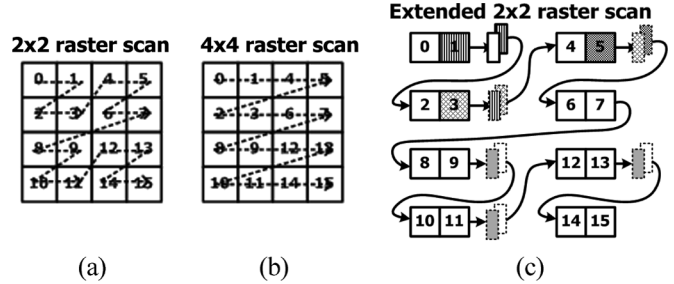


Fig. 10. Different scan orders in motion compensation: (a) 2×2 raster scan; (b) 4×4 raster scan; (c) extended 2×2 raster scan when decoding index is sub-block #1.

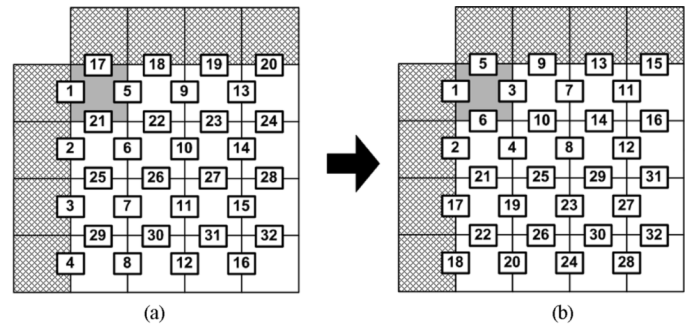


Fig. 11. H.264/AVC-defined filtering order. (a) Traditional scheduler. (b) Proposed hybrid scheduler.

proach induces a drawback that intermediate data have to be stored and loaded again when altering the filtered directions. For example, considering the gray region, the edge #1 will be filtered first followed by the edge #5. After that, the processing data in gray region cannot be reused since the distance between the vertical and horizontal edges (i.e., #5 versus #17) becomes longer. To alleviate this problem, we propose a hybrid scheduler to reorder the standard-defined edges in Fig. 11(b). The proposal reuses the intermediate data and eliminates the redundant accesses of transformation between the horizontal and vertical directions. Compared to traditional schedulers [19], [20], the proposed method reduces the processing cycles by 50%, and it combines both vertical and horizontal filters with standard compliance. Additionally, to support the proposed scheduler, extra control logics and four 4×4 pixel buffers are required and then contribute 17% of total gate counts in the deblocking filter core.

Fig. 12 exhibits a working frequency breakdown of different design phases for the H.264/AVC decoding of 1080HD@30 fps. At the beginning, the required working frequency (i.e., 242 MHz) is dominated by motion compensation (MC) due to its extensive processing cycles. Afterward, the operating frequency is lowered to 152 MHz through the low-power MC design. However, the de-blocking filter (DF) becomes the frequency bottleneck after applying the low-power MC. Therefore, the low-power DF is further applied to lower the required frequency to 100 MHz.

IV. IMPLEMENTATION RESULTS

The chip micrograph is shown in Fig. 13. The slice pixel SRAM is positioned on the left-top corner of this chip. The LPL

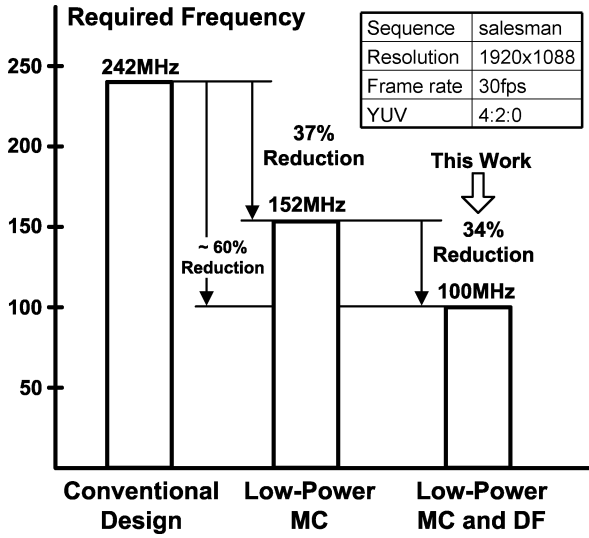


Fig. 12. Performance comparison at different architectural design phases.

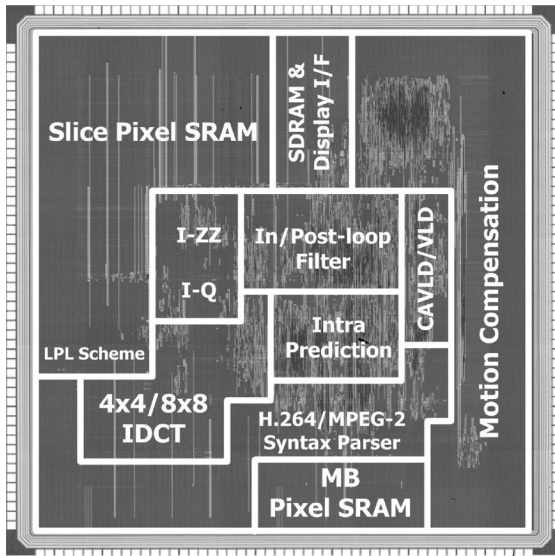


Fig. 13. Chip micrograph.

scheme is interfaced to the embedded SRAM for improving access efficiency. Several area-efficient and low-power architectures have been included in this chip. Chip features are summarized in Table II. It integrates MPEG-2 SP@ML and H.264/AVC BL@L4 and is fabricated using 0.18 μ m single-poly six-metal CMOS process. The die size is 3.9×3.9 mm². In addition, the logic gate counts are about 300 K excluding the memory. The memory consisting of 22.75 Kb SRAM and 8 MB SDRAM are exploited to store pixel data. The maximum working frequency is about 100 MHz and achieves 101.04 MPixel/s of maximum throughput rates.

In terms of core power measurements, a sub-mW of power dissipation can be achieved under decoding sequences of QCIF resolution and 15 fps for mobile applications. Because DRAM configurations are so diverse in existing designs and DRAM power can be optimized through other leading-edge techniques [21], [22], we only show core power dissipation to make a feasible comparison. Fig. 14 shows a measured power-throughput

TABLE II
CHIP FEATURES

Specification	Dual	MPEG-2 SP@ML H.264/AVC BL@L4
Technology	Standard 0.18 μ m 1P6M CMOS 1.8V core, 3.3V I/O	
Die Size	3.9mm \times 3.9mm	
Package	208-pin CQFP	
Logic Gates	303.78K	
Internal Memory	22.75Kb	SRAM
External	4MB \times 2	SDRAM
Max. System Clock	100MHz	
Max. Processing Throughput	101.04Mpixels/sec	
Core Power	MPEG-2	108 μ W (1.15MHz@1V, QCIF@15fps)
Consumption		10.4mW (16.6MHz@1.2V, D1@30fps)
	H.264/AVC	125 μ W (1.15MHz@1V, QCIF@15fps)
		12.4mW (16.6MHz@1.2V, D1@30fps)

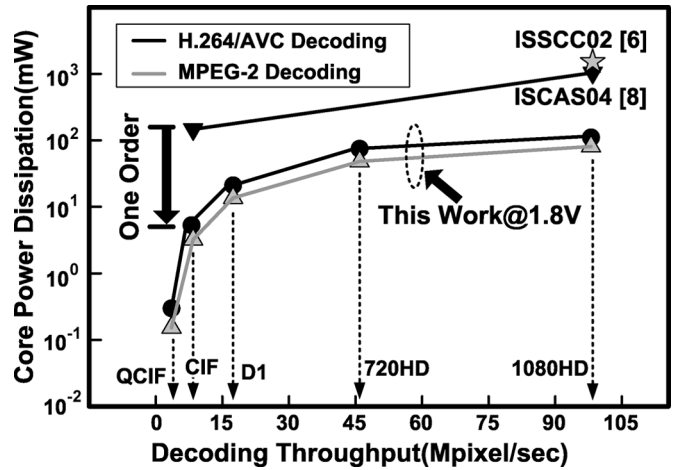


Fig. 14. Power dissipation.

curve. This plot represents characteristics of video decoding capability, where the bottom-right side of this figure indicates better system performance. The power dissipation of this chip is about 90 mW and 100 mW for the real-time decoding of high-definition video quality in MPEG-2 and H.264/AVC video standards, respectively. When we consider the mobile applications, the power consumption is only sub-mW for the real-time decoding of QCIF resolution and 15 fps. Therefore, this chip operates at a power level that is about one order of magnitude less than comparable decoders [6], [8].

The aforementioned power can be further improved through a voltage scaling. Under the H.264/AVC decoding mode, a shmoo plot obtained by a VLSI tester is shown in Fig. 15. It indicates that this chip operates at a working frequency of 1.15 MHz and 16.6 MHz with a supply voltage of 1 V and 1.2 V, respectively. As a result, 125 μ W and 12.4 mW of core power dissipation can be obtained for the H.264/AVC decoding of QCIF@15 fps and D1@30 fps. Similarly, a voltage scaling can be applied to

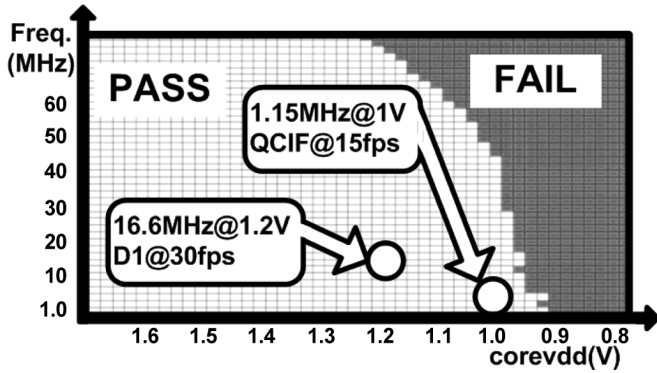


Fig. 15. Shmoo plot.

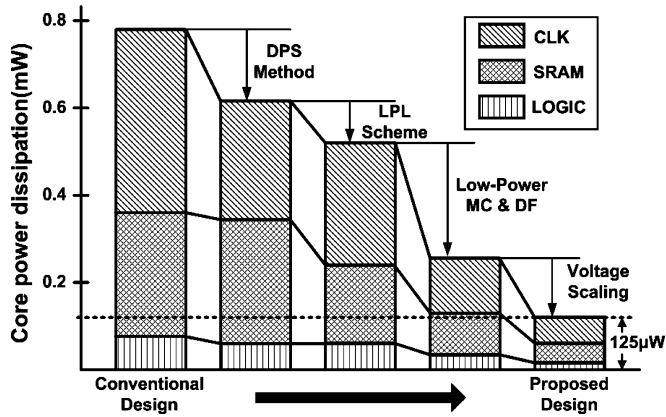


Fig. 16. Power consumption of MPEG-2/H.264 video decoding core.

MPEG-2 video decoding, and the associated power dissipation is summarized in Table II.

To summarize the low-power techniques, Fig. 16 shows the composition of the power consumption when the dual-video decoder chip runs at the H.264/AVC mode and meets the real-time decoding requirements of QCIF@15 fps. By applying DPS and LPL methods, the CLK and SRAM power are reduced due to the optimized register and memory allocations. Further reduction is obtained through the low-power architectures and voltage scaling. The power dissipation is dramatically reduced by lowering the required working frequency and supply voltage. As a result, the overall design consumes $125 \mu\text{W}$ when working at 1.15 MHz.

V. CONCLUSION

In this paper, a single-chip MPEG-2 SP@ML and H.264/AVC BL@L4 video decoder has been presented. The proposed $4 \times 4/8 \times 8$ IDCT and in/post-loop deblocking filter achieve 15% and 30% of gate-count reduction, respectively. A DPS method is developed to optimize the pipeline registers and processing cycles, resulting in saving 37.5% registers compared to an existing design [11]. Furthermore, the proposed design employs a LPL scheme, low-power motion compensation, and de-blocking filter to improve system performance, leading to saving 51% of memory power consumption and 60% of required working frequency. In total, the power

reduction of this chip is about one order of magnitude compared to state-of-the-art implementations [6], [8]. Measurement results show that H.264/AVC and MPEG-2 video decoding of quarter-common intermediate format (QCIF) sequences at 15 frames per second are achieved at 1.15 MHz clock frequency with power dissipation of $125 \mu\text{W}$ and $108 \mu\text{W}$, respectively, at 1 V supply voltage. This low-power and area-efficient feature makes our proposal very suitable for mobile applications where conservative power requirements are essential.

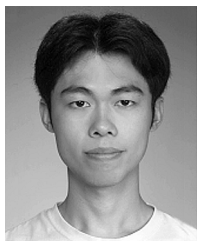
ACKNOWLEDGMENT

The authors would like to thank B.-J. Shieh, W.-C. Lee, J.-B. Chen, C.-C. Chung, W.-H. Peng, S.-M. Sun, Y.-F. Chuang, and colleagues of the SI2 Group of National Chiao Tung University for insightful discussions on this work. They also thank the Chip Implementation Center (CIC) for testing services.

REFERENCES

- [1] *Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification*, ITU-T Rec. H.264 | ISO/IEC 14496-10 AVC, May 2003.
- [2] L. Bolcioni, M. Borgatti, M. Felici, R. Rambaldi, and R. Guerrieri, "A 1 V 350 μW voice-controlled H.263 video decoder for portable applications," in *IEEE ISSCC Dig. Tech. Papers*, 1998, pp. 112–113.
- [3] M. Miyama *et al.*, "A sub-mW MPEG-4 motion estimation processor core for mobile video application," *IEEE J. Solid-State Circuits*, vol. 39, no. 9, pp. 1562–1570, Sep. 2004.
- [4] T.-M. Liu *et al.*, "A 125 μW , fully scalable MPEG-2 and H.264/AVC video decoder for mobile applications," in *IEEE ISSCC Dig. Tech. Papers*, 2006, pp. 402–403.
- [5] T.-M. Liu *et al.*, "An 865- μW H.264/AVC video decoder for mobile applications," in *Proc. IEEE Asian Solid-State Circuit Conf. (A-SSCC)*, 2005, pp. 301–304.
- [6] H. Yamauchi *et al.*, "A 0.8 W HDTV video processor with simultaneous decoding of two MPEG-2-MP@HL streams and capable of 30 frames/s reverse playback," in *IEEE ISSCC Dig. Tech. Papers*, 2002, pp. 372–474.
- [7] Y.-W. Huang *et al.*, "A 1.3TOPS H.264/AVC single-chip encoder for HDTV applications," in *IEEE ISSCC Dig. Tech. Papers*, 2005, pp. 128–129.
- [8] H.-Y. Kang *et al.*, "MPEG4 AVC/H.264 decoder with scalable bus architecture and dual memory controller," in *Proc. IEEE ISCAS*, 2004, pp. II-145–II-148.
- [9] T. Fujiyoshi *et al.*, "A 63-mW H.264/MPEG-4 audio/visual codec LSI with module-wise dynamic voltage/frequency scaling," *IEEE J. Solid-State Circuits*, vol. 41, no. 1, pp. 54–62, Jan. 2006.
- [10] Y. Hu, A. Simpson, K. McAdoo, and J. Cush, "A high definition H.264/AVC hardware video decoder core for multimedia SoCs," in *Proc. IEEE Int. Symp. Consumer Electronics*, 2004, pp. 289–385.
- [11] M. Toyokura *et al.*, "A video DSP with a macroblock-level-pipeline and a SIMD type vector-pipeline architecture for MPEG-2 CODEC," *IEEE J. Solid-State Circuits*, vol. 29, no. 12, pp. 1474–1481, Dec. 1994.
- [12] S.-H. Wang *et al.*, "A platform-based MPEG-4 advanced video coding (AVC) decoder with block level pipelining," in *Proc. 2003 Joint Conf. 4th Int. Conf. Information, Communications and Signal Processing and 4th Pacific Rim Conf. Multimedia*, Dec. 2003, pp. 15–18.
- [13] S. Park, H. Cho, H. Jung, and D. Lee, "An implemented of H.264 video decoder using hardware and software," in *Proc. IEEE Custom Integrated Circuits Conf. (CICC)*, 2005, pp. 271–275.
- [14] S. Wuytack, J.-P. Diguët, F. V. M. Cathoor, and H. J. De Man, "Formalized methodology for data reuse exploration for low-power hierarchical memory mappings," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 6, no. 4, pp. 529–537, Dec. 1998.
- [15] H. S. Hou, "A fast recursive algorithm for computing the discrete cosine transform," *IEEE Trans. Acoust., Speech Signal Process.*, vol. 35, no. 10, pp. 1455–1461, Oct. 1987.
- [16] M. Potkonjak, M. B. Srivastava, and A. P. Chandrakasan, "Multiple constant multiplications: efficient and versatile framework and algorithms for exploring common subexpression elimination," *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.*, vol. 15, no. 2, pp. 151–165, Feb. 1996.

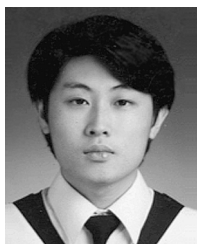
- [17] S.-Z. Wang, T.-A. Lin, T.-M. Liu, and C.-Y. Lee, "A new motion compensation design for H.264/AVC decoder," in *Proc. IEEE ISCAS*, 2005, pp. 4558–4561.
- [18] T.-M. Liu, W.-P. Lee, and C.-Y. Lee, "An area-efficient and high-throughput de-blocking filter for multi-standard video applications," in *Proc. IEEE Int. Conf. Image Processing*, 2005, pp. III-1044–III-1047.
- [19] Y.-W. Huang *et al.*, "Architecture design for deblocking filter in H.264/JVT/AVC," in *Proc. IEEE I. Conf. Multimedia and Expo*, 2003, vol. 1, pp. 1693–1696.
- [20] M. Sima, Y. Zhou, and W. Zhang, "An efficient architecture for adaptive deblocking filter of H.264/AVC video coding," *IEEE Trans. Consum. Electron.*, vol. 50, no. 1, pp. 292–296, Feb. 2004.
- [21] H. J. Oh *et al.*, "High-density low-power-operating DRAM device adopting 6F² cell scheme with novel S-RCAT structure on 80 nm feature size and beyond," in *Proc. 35th Eur. Solid-State Device Research Conf. (ESSDERC'05)*, Sep. 2005, pp. 177–180.
- [22] J.-W. Park *et al.*, "Performance characteristics of SOI DRAM for low-power application," *IEEE J. Solid-State Circuits*, vol. 34, no. 11, pp. 1446–1453, Nov. 1999.



Tsu-Ming Liu (S'04) was born in I-Lan, Taiwan, R.O.C., in 1980. He received the B.S. and M.S. degrees in electronics engineering from National Chiao-Tung University, Taiwan, in 2002 and 2004, respectively.

During 2004, he was an intern with Sunplus Technology Company, HsinChu, Taiwan. In 2004, he joined the Institute of Electronics Engineering of National Chiao-Tung University, where he is currently working toward the Ph.D. degree. His major research interests include binary shape coding, design, H.264/AVC video decoding and associated

joint source and channel VLSI architectures.



Ting-An Lin (S'05) received the B.S. and M.S. degrees in electronics engineering from National Chiao-Tung University, Taiwan, R.O.C., in 2003 and 2005, respectively.

He has been working for MediaTek Inc. since 2005. His major research interests include equalizer design of DVB-T inner receiver system, H.264 video decoder design, and associated VLSI architectures during graduate school. Currently, he is a Video Decoding System Designer with MediaTek Inc.



Sheng-Zen Wang received the B.S. and M.S. degrees in electronics engineering from National Chiao-Tung University, Taiwan, R.O.C., in 2003 and 2005, respectively.

In 2006, he joined MediaTek Inc., Hsinchu, Taiwan, where he develops digital TV backend related systems. His major research interests include memory controller design, H.264/AVC video decoding and associated VLSI architecture.



Wen-Ping Lee was born in Miaoli City, Taiwan, R.O.C., in 1983. He received the B.S. degree in electrical engineering from National Chiao Tung University, Hsinchu, Taiwan, R.O.C., in 2005. He has been working toward the M.S. degree in the Department of Electronics Engineering, National Chiao Tung University, on research multimedia communication.



Jiun-Yan Yang was born in Kaohsiung City, Taiwan, R.O.C., in 1982. He received the B.S. degree in electrical engineering from National Chiao Tung University, Hsinchu, Taiwan, R.O.C., in 2004.

Since 2004, he has been working toward the M.S. degree in the Department of Electronics Engineering, National Chiao Tung University, as part of the SI2 Research Group. His research interests include IC design flow, cell-based VLSI design, system-on-chip technology, and multimedia communication.



Kang-Cheng Hou was born in Taipei, Taiwan, R.O.C., in 1979. He received the B.S. degree in electronics engineering from National Kaohsiung University of Applied Sciences, Kaohsiung, Taiwan, in 2001, and the M.S. degree in electronics engineering from National Chiao Tung University, Hsinchu, Taiwan, in 2006.

His current research interests include VLSI design, multimedia application, and memory management for video application.



Chen-Yi Lee (M'01) received the B.S. degree from National Chiao Tung University, Hsinchu, Taiwan, R.O.C., in 1982, and the M.S. and Ph.D. degrees from Katholieke University Leuven (KUL), Belgium, in 1986 and 1990, respectively, all in electrical engineering.

From 1986 to 1990, he was with IMEC/VSDM, working in the area of architecture synthesis for DSP. In February 1991, he joined the faculty of the Electronics Engineering Department, National Chiao Tung University, Hsinchu, Taiwan, where he is currently a Professor. His research interests include VLSI algorithms and architectures for high-throughput DSP applications. He is also active in various aspects of system-on-chip design technology, very low power designs, multimedia signal processing, and wireless communications. He served as the Director of the Chip Implementation Center (CIC), an organization for IC design promotion in Taiwan, from 2000 to 2003.

Prof. Lee was the former IEEE CAS Taipei Chapter Chair (2000–2002), the SIP task leader of the National SoC Research Program (2003–2005), and the microelectronics program coordinator of the Engineering Division under the National Science Council of Taiwan (2002–2005). He also served as the Department Chair of Electronics Engineering, National Chiao Tung University (2003–2006).