HARDWARE ARCHITECTURE DESIGN FOR H.264/AVC INTRA FRAME CODER

Yu-Wen Huang, Bing-Yu Hsieh, Tung-Chien Chen, and Liang-Gee Chen

DSP/IC Design Lab., Graduate Institute of Electronics Engineering and
Department of Electrical Engineering, National Taiwan University
{yuwen, bingyu, djchen, lgchen}@video.ee.ntu.edu.tw

ABSTRACT

In this paper, we contributed a VLSI architecture design for H.264/AVC intra frame coder. First, analysis of coding algorithm is provided by using a RISC model to obtain the proper degrees of parallelism under SDTV specification. Second, a two-stage macroblock pipelining is proposed to double the processing capability and hardware utilization. Third, Hadamard-based mode decision is modified as DCT-based version to reduce the 40% of memory access. To sum up, our system architecture achieves 215 times of speed compared with RISC-based software implementation in terms of processing cycles. In addition, we also made a lot of efforts on developing area-speed efficient modules. Reconfigurable intra predictor generator can support all kinds of prediction modes. Parallel multi-transform has four times throughput of the serial one with little area overhead. CAVLC engine can efficiently provide coding information for the bitstream packer. A prototype chip was fabricated with TSMC 0.25 μm CMOS technology and is capable of encoding 720x480 4:2:0 30Hz video in real time at the working frequency of 54 MHz. The transistor count is 429K, and the core size is only 1.855x1.885 mm^2.

1. INTRODUCTION

H.264/AVC intra frame coder [1] is competitive with the latest image coding standard, JPEG2000 [2], in coding performance. According to the experimental results of JPEG2000 VM7.2 and H.264/AVC JM7.3, the rate-distortion curve of H.264/AVC Main Profile intra frame coder (CABAC, high complexity mode decision) is almost the same as that of JPEG2000 DWT97. H.264/AVC Baseline Profile intra frame coder (CAVLC, low complexity mode decision) is 0.2-1.0 dB better than JPEG2000 DWT53. For encoding and decoding the "Bike" image (2048x2560x8b), JPEG2000 DWT53 requires 3430 and 3180 Mega-instructions, respectively, while H.264/AVC Baseline Profile requires 3648 and 584 Mega-instructions. For applications whose key functionality is compression instead of scalability, such as digital storage camera, digital scanner, digital video editing, and digital video surveillance, H.264/AVC intra frame coder may be a more attractive solution due to the hardware-friendly block-based algorithm. In JPEG2000, DWT is a frame-based transform that requires a huge amount of memory, and EBCOT is a sequential bitplane processing that requires a high operating frequency.

Intra prediction with rate-distortion constrained mode decision is the most important technology in H.264/AVC intra frame coder. The predictor generation engine for intra prediction and the transform engine for mode decision are critical because these operations occupy 80% of the computation time of the whole compression process, and it is difficult for general purpose processors (GPP) to meet the real-time constraints. In this paper, we will analyze the coding algorithm to develop the VLSI architecture of H.264/AVC Baseline Profile intra frame coder targeted for SDTV specification (720x480 4:2:0 30Hz video). The rest of this paper is organized as follows. In Section 2, the fundamentals of H.264/AVC intra frame coding is first reviewed. In Section 3, system design is proposed according to deep analysis. Module design and implementation results are described in Section 4 and Section 5, respectively. Finally, Section 6 gives a conclusion.

2. FUNDAMENTALS

The encoding flow of each macroblock (MB) can be separated into mode decision phase and residue encoding phase. In the mode decision phase, 17 kinds of prediction modes are generated for one MB (9 I4MB modes for luma, 4 I16MB modes for luma, 4 modes similar to I16MB for chroma), and distortion cost is evaluated by sum of absolute values of 2-D 4x4 Hadamard transformed differences (SATD), and rate cost is estimated by quantization parameter and number of bits required to code the mode information. Then, the best MB mode is chosen by minimizing the Lagrangian cost value (distortion cost plus rate cost) [3]. In the residue encoding phase, prediction residues are transformed and quantized [4]. The mode information and residues are then compressed by Exp-Golomb code and context-based adaptive variable length code (CAVLC) [5], respectively.

The instruction profile of H.264/AVC Baseline Profile intra frame coder with low complexity mode decision for SDTV specification is shown in Table 1. Real-time processing requires 10,829 million instructions per second (MIPS), which is far beyond the capability of today's GPP. The instructions are classified as three categories: computing, controlling, and memory access. It is shown that memory access operations are the most highly demanded. This reveals that local SRAM and registers are critical to reduce the bus bandwidth. Figure 1 shows the runtime percentages of several major functional modules. As can be seen, transform for cost generation (SATD computation) and mode decision take the largest portion of computation, and intra predictor generation is the second. These two functions take 77% of computation and obviously are the processing bottleneck.

3. SYSTEM DESIGN

In this section, a parallel H.264/AVC intra frame coding architecture will be proposed for SDTV specification, which requires to

Table 1: Instruction profile for SDTV specification.

Instruction Type	MIPS	%	Category
Arithmetic	1,785	16.5	Computing
Logic	83	0.77	Computing
Rotate and Shift	279	2.58	Computing
Jump and Compare	1,558	14.4	Controlling
Stack Instruction	3,154	29.15	Memory Access
Data Instruction	3,961	36.6	Memory Access
Total	10,820	100	

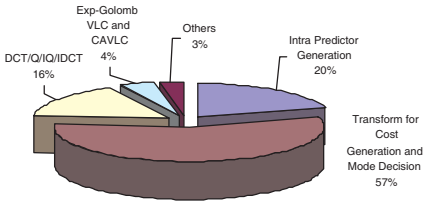


Figure 1: Run-time percentages of various functional modules.

encode about 16 Mega-pixels within one cycle. The detailed analysis and system/module designs will be described as follows.

3.1. Exploration of Parallelism

First, two assumptions are made: a RISC is able to execute one instruction in one cycle with an exception of multiplication requiring two cycles; a processing element (PE) is capable of generating the predictor of one pixel in one cycle. Next, we compute the average instruction counts required for intra predictor generation. For example, the operation "c = a + b" requires two load instructions and one add instruction. Table 2 shows that it takes 3.2629 and 3.9610 cycles for a RISC to generate one luma predictor and one chroma predictor, respectively.

We first discuss three possible solutions for intra predictor generation, as shown in Fig. 2. The first one is a RISC solution, which requires to run at 521.9 MHz to generate the predictors in

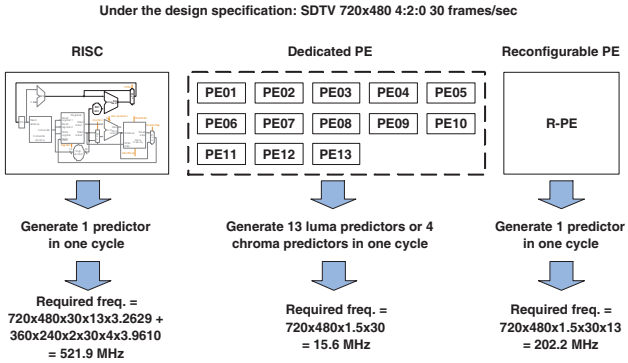


Figure 2: Three possible solutions and the required working frequency to meet the real-time requirement of SDTV specification.

Table 2: Analysis of instructions for intra predictor generation.

	Intra Prediction Modes	Average Cycles to Generate the Predictor of a Pixel
LUMA	Intra4x4 Vertical	(0+0+0+4 x4)/16 = 1
	Intra4x4 Horizontal	(0+0+0+4 x4)/16 = 1
	Intra4x4 DC	(8+1+0+4 x4)/16 = 1.5625
	Intra4x4 Diagonal Down/Left	(6x6+4+7+12+4x4)/16 = 4.6875
	Intra4x4 Diagonal Down/Right	(3x7+7+7+4x4)/16 = 3.1875
	Intra4x4 Vertical Left	(2x4+3x6+10+6+4x4)/16 = 3.625
	Intra4x4 Horizontal Down	(2x4+3x6+10+6+4x4)/16 = 3.625
	Intra4x4 Vertical Right	(2x4+3x4+6x2+10+9+4x4)/16 = 4.1875
	Intra4x4 Horizontal Up	(2x4+3x4+6x2+10+9+4x4)/16 = 4.1875
	Intra16x16 DC	(15 x2+2+1+0+16x16)/256 = 1.1289
	Intra16x16 Vertical	(0+0+0+16 x16)/256 = 1
	Intra16x16 Horizontal	(0+0+0+16 x16)/256 = 1
	Intra16x16 Plane	(8x3x2+2x2+2+2+2+(3+2x2+2+2)x256+16x16)/256 = 12.2266
	Average	**3.2629 (cycles/pixel required for RISC)**
CHROMA	DC	(3x4+4+4+0+8x8)/64 = 1.3125
	Vertical	(0+0+0+8 x8)/64 = 1
	Horizontal	(0+0+0+8 x8)/64 = 1
	Plane	(4x3x2+2x2+2+2+2+(3+2x2+2+2)x64+8x8)/64 = 12.5313
	Average	**3.9610 (cycles/pixel required for RISC)**

Table 3: Hardware complexity and operating frequency under different degrees of parallelism.

Parallelism / Solution	No Parallelism		Two-Parallel		Four-Parallel		Eight-Parallel	
	Hardware Complexity	Operating Frequency	Hardware Complexity	Operating Frequency	Hardware Complexity	Operating Frequency	Hardware Complexity	Operating Frequency
RISC	~A	>>521.9 MHz	~2A	>>261.0 MHz	~4A	>>130.5 MHz	~8A	>>65.2 MHz
Dedicated PE`s	<13A	15.6 MHz	<26A	7.8 MHz	<52A	3.9 MHz	<104A	1.9 MHz
Reconfigurable PE	A	202.2 MHz	2A	101.1 MHz	**4A**	**50.6 MHz**	8A	25.3 MHz

time, not to mention transform, entropy coding, and other system jobs. Consequently, RISC seems to be impractical. The second solution is a set of 13 different PE's. The hardware can generate 13 kinds of predictors in one cycle. The architecture only needs to operate at 15.6 MHz, but the cost is very high. The third choice is to design a reconfigurable PE to generate all the intra predictors with different configurations. This solution targets at higher area-speed efficiency. Nevertheless, it still requires to operate at 202.2 MHz. Thus, parallel reconfigurable PE's become the most promising solution. Table 3 lists the hardware complexity and required frequency of the three solutions under different degrees of parallelism. We conclude this subsection by adopting four-parallel reconfigurable PE for intra predictor generation in our design.

3.2. System Architecture

We divide our system into two main parts, the encoding loop and the bitstream generation unit, as illustrated in Fig. 3. Assume one row of reconstructed pixels/coding information is buffered in the external DRAM. At the beginning, current MB pixels and upper reconstructed pixels/coding information are loaded from external DRAM to on-chip SRAM. The reconstructed pixels/coding information of the previous (left) MB can be directly kept in registers to save bus bandwidth. With on-chip SRAM and coding information registers, the bus bandwidth is reduced from hundreds of Mbytes to about 20 Mbytes/sec. Then, we start the intra prediction block by block. According to the previous analysis, four pixels should be processed (predictor generation and SATD computation) in one cycle. Therefore, the number of cycles required

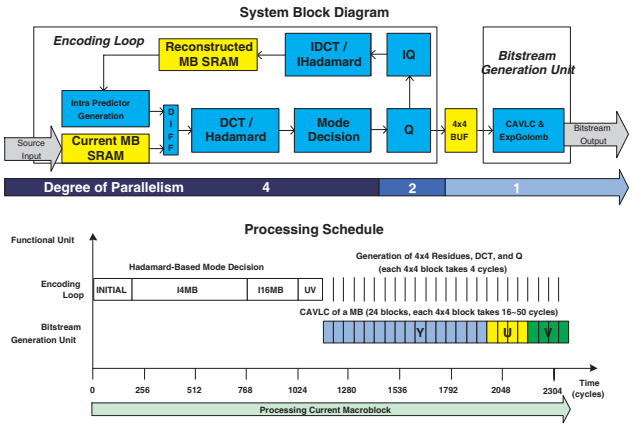


Figure 3: Illustration of initial system architecture.

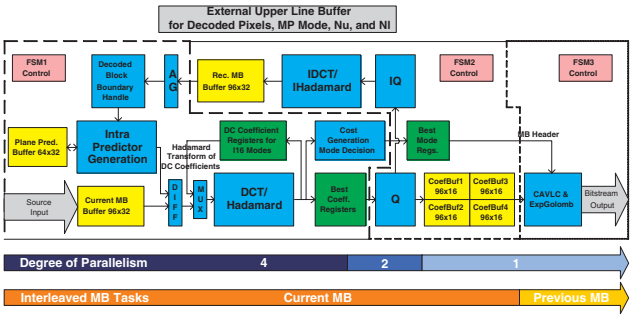


Figure 4: Illustration of final system architecture.

Table 4: Comparison of system architectures.

Architecture	Initial	MB Pipelining	MB Pipelining and DCT-Based Mode Decision
Parallelism Optimized	YES	YES	YES
Task Schedule	Sequential	Interleaved	Interleaved
Mode Decision Method	Hadamard-Based	Hadamard-Based	DCT-Based
Processing Cycles / MB	< 2400 (cycles)	< 1300 (cycles)	< 1300 (cycles)
Required Frequency	97.2 MHz	52.7 MHz	52.7 MHz
Bus Bandwidth	~20 (Mbytes/s)	~20 (Mbytes/s)	~20 (Mbytes/s)
On-Chip SRAM Access	113.17 (Mbytes/s)	113.17 (Mbytes/s)	72.25 (Mbytes/s)
Coefficient Buffer	16x16 (bits)	96x64 (bits)	96x64 (bits)

for luma intra prediction and mode decision is 832 (4x13x16), and that for chroma is 128 (4x4x8). Before DCT/Q/IQ/IDCT finishes the previous 4x4-block, current 4x4-block cannot proceed to intra prediction, so four-parallel DCT/IDCT and two-parallel Q/IQ are adopted to reduce the latency. CAVLC is a sequential algorithm and its computational loading is not very large. Hence, parallel processing is not a must. In sum, the initial system architecture needs about 2400 cycles to encode a MB. Also, with the straightforward data flow, the residues are generated twice. One time is used for intra predictor generation and mode decision while the other is for entropy coding of the best mode.

In the previous paragraph, it is observed that the number of processing cycles for encoding loop is about the same as that for bitstream generation unit in the worst case. These two procedures are separable because there is no feedback loop between them. Therefore, a MB pipelining is incorporated into the system to accelerate the processing speed at the cost of coefficient buffer for a MB. When current MB is processed by encoding loop, bitstream generation unit processes previous MB simultaneously. The number of processing cycles is reduced to less than 1300 cycles.

In the reference software, Hadamard transform is involved in SATD. The transform coefficients in the mode decision phase cannot be reused for CAVLC. Thus, we modify the SATD computation by using DCT. Luma mode decision is performed block by block and 13 kinds of predictors are generated for each 4x4-block. The former nine prediction modes decide the best I4MB mode and its quantized transform coefficients will be stored in the coefficient buffer. In this way, if I4MB mode is chosen, regeneration of luma transform quantized residues can be avoided. The improvement will be significant in high quality applications where almost all MB's select I4MB. In our experience, when QP is smaller than 25, the percentage of I16MB mode is less than 10%. The amount of on-chip memory access can thus be reduced from 113.17 Mbytes/sec to 72.25 Mbytes/sec. Also, the proposed mode decision does not suffer any quality loss compared with the mode decision in the reference software. Fig. 4 shows the final system block diagram of our proposed H.264/AVC intra frame encoder.

Table 4 shows the comparison of the three developed architectures. The last version has the fewest processing cycles and the least memory access. Compared with software implementation on RISC, which requires 0.28M cycles to encode one MB, the performance of proposed architecture is 215.4 times faster than the software implementation. The chip only needs to operate at about 50 MHz to meet the SDTV specification.

4. MODULE DESIGN

We developed a four-parallel reconfigurable intra predictor generator to achieve resource sharing between all kinds of prediction modes. Due to the limited space, only I16MB plane prediction will be explained. We also developed an area-speed efficient four-parallel multi-transform engine. Details can be found in [6]. CAVLC engine will also be described later.

4.1. I16MB Prediction Modes

The detailed definition of I16MB plane prediction mode, which is an approximation of bilinear transform, is described as follows.

$$Pred[y,x] = Clip1((a + b \cdot (x-7) + c \cdot (y-7)) >> 5)$$

We proposed a decomposition technique to avoid multiplications. First, there is a short setup period to precompute a, b, c, H and V and buffer them in registers. Next, four seed values, $Pred[0,0]$, $Pred[0,4]$, $Pred[0,8]$, and $Pred[0,12]$ are computed. With the precomputed b, c, and these seed values, all the other I16MB plane predictors can be computed by add and shift operations, as expressed in Fig. 5. The proposed PE is shown in Fig. 6.

4.2. Bitstream Generation

Figure 7 shows the bitstream generator. The macroblock header is first produced. Then, the CAVLC forms bitstream block by block.

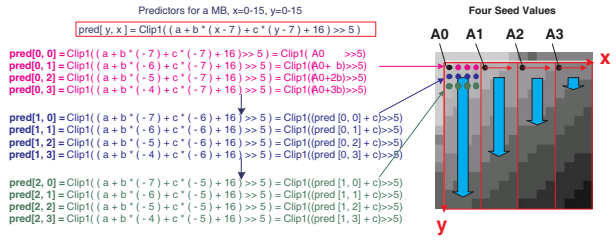


Figure 5: Decomposition of I16MB plane prediction mode.

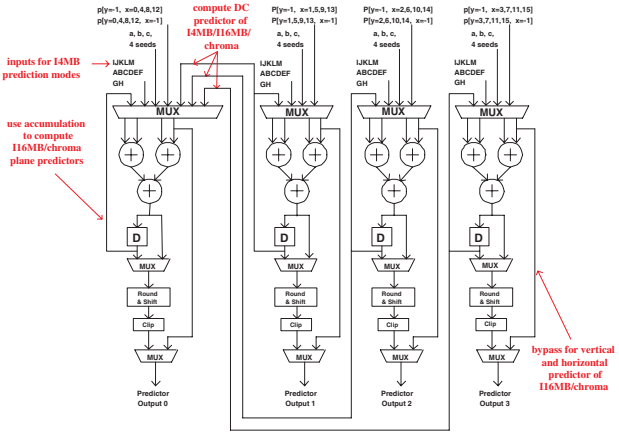


Figure 6: Proposed four-parallel reconfigurable intra predictor generator.

It takes sixteen cycles to load coefficients of a 4x4-block from the memory in a reverse zigzag scan order. During the loading, the level detection checks if the coefficient is zero. If the level is nonzero, it will be stored into the level-FIFO, and the corresponding run information will also be stored to the run-FIFO. At the same time, the trailing one counter, total coefficient counter, and run counter will update the corresponding counts into registers. After scan, the total coefficient/trailing one module will output the code word to packer by looking up the VLC table according to the results of total coefficient register and trailing one register. Next, level code information is sent to the packer by looking up the VLC table for levels followed by total zero and runs.

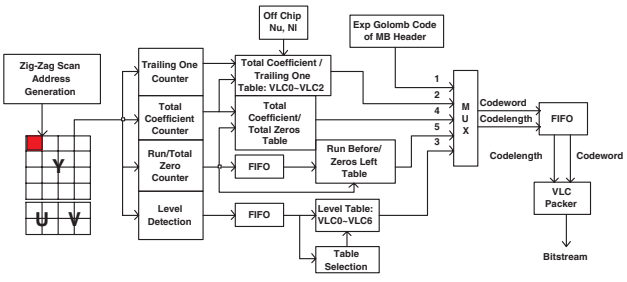
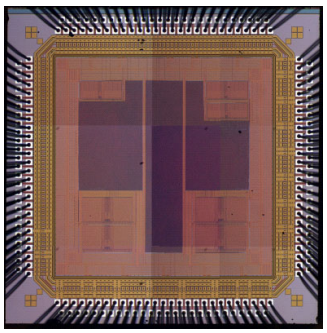


Figure 7: Hardware architecture of bitstream generation engine.



Technology	TSMC 0.25um CMOS 1P5M
Package	CQFP 208
Core Size	1.855 x 1.885 mm²
Logic Gate Count	84,985
On-Chip SRAM	Single Port 64x32 (x1)
	Single Port 96x32 (x2)
	Dual Port 96x16 (x4)
Transistor Count	429,139
Max. Clock Rate	55 MHz
Processing Capability	434 fps for 4:2:0 QCIF (176X144)
	107 fps for 4:2:0 CIF (352X288)
	31 fps for 4:2:0 SDTV (720x480)
	16.26 Mega-pixels within 1 sec

Figure 8: Chip photo and specifications.

5. IMPLEMENTATION RESULTS

The chip photo and specifications are shown in Fig. 8. I16MB plane predictors are buffered in one 64x32 RAM to save the regeneration of predictors when selected as best MB mode. Two 96x32 RAM's are used to save current MB and reconstructed MB. The other four RAM's are used as residue buffer for MB pipelining.

6. CONCLUSION

This paper presents a VLSI architecture design for H.264/AVC intra frame coder. We provide analysis to obtain the suitable degrees of parallelism under SDTV specification. MB pipelining and DCT-based mode decision are then proposed to double the speed and to reduce the 40% of memory access, respectively. Area-speed efficient modules are also designed. Our implementation is capable of encoding 16 Mega-pixels within one second at 54 MHz with 1.855x1.885 mm^2 core area.

7. REFERENCES

[1] Joint Video Team, *Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification*, ITU-T Rec. H.264 and ISO/IEC 14496-10 AVC, May 2003.

[2] *JPEG 2000 Part I*, ISO/IEC JTC1/SC29/WG1 Final Committee Draft, Rev. 1.0, Mar. 2000.

[3] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, and G. J. Sullivan, "Rate-constrained coder control and comparison of video coding standards," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 688–703, July 2003.

[4] H. S. Malvar, A. Hallapuro, M. Karczewicz, and Louis Kerosfsky, "Low-complexity transform and quantization in H.264/AVC," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 598–603, July 2003.

[5] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, July 2003.

[6] T. C. Wang, Y. W. Huang H. C. Fang, and L. G. Chen, "Parallel 4x4 2D transform and inverse transform architecture for MPEG-4 AVC/H.264," in *Proc. of IEEE International Symposium on Circuits and Systems*, 2003.