A 140-MHz 94 K Gates HD1080p 30-Frames/s Intra-Only Profile H.264 Encoder

Yu-Kun Lin, Chun-Wei Ku, De-Wei Li, and Tian-Sheuan Chang, Senior Member, IEEE

Abstract—This paper presents a HD1080p 30-frames/s H.264 intra encoder operated at 140 MHz with just 94 K gate count and 0.72-mm$^2$ core area for digital video recorder or digital still camera applications. To achieve high throughput and low area cost for high-definition video, we apply the modified three-step fast intra prediction technique to reduce the cycle count while keeping the quality as close as full search. Then, in architecture scheduling, we further adopt the variable pixel parallelism instead of constant four-pixel parallelism to speed up performance on the critical intra prediction part while keeping other parts unchanged for low area cost. The achieved design only needs half of the working frequency and reduces the gate count cost by 23.5% compared with the previous design with the same HD720p 30-frames/s requirement. Besides, our design at 140 MHz can support HD1080p 30 frames/s for digital video encoder or 4096$\times$2304 images with 6.78 frames/s for digital still camera application.

Index Terms—H.264, intra prediction, VLSI architecture.

I. INTRODUCTION

The new video coding standard, known as H.264 or MPEG-4 Part 10 Advanced Video Coding (MPEG-4 AVC) [1], is developed by the Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG as the next-generation video compression standard for higher coding efficiency, in which the intra frame prediction technique can efficiently use the neighboring pixels to predict the current coding block from various directions. With this, the coding efficiency is even competitive with the latest still image coding standard, JPEG2000. Intra frame coding is now accepted as the intra-only profile [2] and is suitable for applications like digital video recorders (DVRs) and digital still cameras that cannot afford the inter prediction computing.

Several results on the H.264 intra frame encoder have been reported in [3]–[5]. However, these designs are still limited to HD720p resolution. In this paper, we propose a high-throughput-rate intra frame encoder targeted for HD1080p 30-frames/s encoding with low area cost. This is achieved by hardware-oriented fast algorithms as well as architecture optimization. In summary, this paper has the following contributions compared to previous works: modified three-step algorithm for fast intra prediction, variable pixel parallelism scheduling for parallel computation and low area cost, and interleaved best mode computation strategy to reduce the pipeline bubble cycles. With these techniques, we can reduce 44% of cycle count and 23.5% of area cost when compared with the HD720p intra encoder design reported in [3]. Our final design can support HD1080p 30-frames/s encoding at 140 MHz.

The remainder of this paper is organized as follows. In Section II, we briefly overview the H.264 intra coding. Then, we present our modified three-step intra prediction algorithm and its performance in Section III, and Section IV introduces the variable pixel parallelism architecture. The final implementation result and design comparison are shown in Section V. Finally, we conclude this paper in Section VI.

II. OVERVIEW OF H.264 INTRA FRAME ENCODING

Fig. 1 shows the intra encoding flow of H.264. Each input block will be predicted by the reconstructed boundary data of neighboring blocks. For intra predictor, there are nine modes for $4 \times 4$ luma prediction and four modes for $16 \times 16$ luma and $8 \times 8$ chroma prediction. The residuals generated from the difference between the current block and the best mode are further processed by the transform and quantization unit, and reconstructed by inverse operations to be the reference of next macroblock. The coefficients after quantization are encoded by entropy coding for final bitstream output.

In the intra coding of H.264, the intra prediction and SATD (sum of absolute transformed difference) cost function for mode decision, take almost 77% of computation in all functions [4] and are the bottleneck of intra encoder design.

III. PROPOSED HARDWARE-EFFICIENT FAST ALGORITHM

A. Review of Previous Approaches

Intra prediction algorithms can be classified as two categories. The first type of fast algorithms adopts the two-step approach [6]–[11]. The first step uses image features to select the possible modes, and the second step computes the selected modes. Thus, in the worst case, it will have to compute all modes [11]. Furthermore, this two-step dependency results in irregular data flow which is not suitable for hardware design. In some cases, the feature calculation in the first step will be too complex for hardware design. For example, the edge-based
method [9] uses an arc tangent function and two dividers to calculate the edge for possible mode selections. Thus, these feature calculation circuits introduce extra hardware overhead to the original intra prediction circuit. Even though this arc tangent function can be simplified as shown in [14], this simplification results in an overhead of 15 K gates, which is still large when compared to a full mode calculation circuit with 10 K gates [15]. The second type of algorithms uses a single-step approach [12], [13], [15]. This type of algorithm calculates the modes without extra overhead. However, full search implementations like those in [13] and [15] need large area cost and long computational cycles. Our previous algorithm [12], the three-step algorithm, divides the mode computation into three steps without complex feature calculation. It just needs to compute a constant six modes and is more suitable for hardware design. However, it still introduces pipeline bubble cycles.

B. Proposed Algorithms

The original three-step algorithm [12] in Fig. 2(a) needs 28 cycles to predict a block, in which ten bubble cycles are needed due to the dependent decision flow. Therefore, we adjust the order of prediction modes in the scheduling to reduce the bubble cycles. The key of order change is at the third step as shown in Fig. 2(a). The third step in Fig. 2(a) has to predict either mode 3 or mode 4 disregarding of the result of the second step. Thus, we can swap steps 2 and 3 to reduce the transition bubbles as shown in Fig. 2(b). Fig. 3 shows the pipeline scheduling of the modified three-step algorithm. In the diagram, the total cycles to predict a block are reduced to 20 and no bubble cycle exists although the number of prediction modes is increased from six to seven.

Table I shows the simulation results for two HD1080p sequences when compared with the reference software JM8.6 [16]. The test sequences are all encoded by intra prediction without rate-distortion optimization (RDO). In the simulation, we first show the results of the modified three-step algorithm. The proposed one has the negligible quality degradation with at most 1% of bit rate increase. Besides, the fast algorithms adopted in this paper also include our previously proposed plane mode removal and enhanced SATD [3]. The results of combining the modified three-step algorithm with the previous algorithms are also shown in Table I. In average, our algorithms can achieve 0.11-dB PSNR degradation and 1.03% of bit rate increase.

IV. PROPOSED HIGH-THROUGHPUT ARCHITECTURE

A. Variable Pixel Parallelism Architecture

To further speed up the intra prediction, we adopt some hardware parallelism strategies. However, only the most critical part, intra prediction, adopts the eight-pixel parallelism to double the throughput and thus reduces almost half of the computation cycles. Other parts like the quantization and reconstruction use four-pixel parallelism to save the area cost. This architecture with mixed eight-pixel and four-pixel parallelism is termed variable pixel parallelism architecture.

Fig. 4 shows the proposed intra frame encoder design with variable pixel parallelism architecture. The whole design works as the data flow in Fig. 1. This architecture is partitioned into four phases: prediction phase, reconstruction phase, quantization phase, and bitstream phase. The bitstream phase with CAVLC encoder is similar to the previous one [3].

This variable pixel parallelism has the benefit of low area overhead and high throughput. This parallelism will not introduce the performance bottleneck at the four-pixel part since only blocks with the best mode will be passed to the quantization phase and reconstruction phase. Data flow between different data parallelism is smoothed by several buffers, including the current block and best block registers in the quantization phase and the FIFO registers in the reconstruction phase. To
achieve the eight-pixel parallelism in prediction phase, we only add one more intra prediction engine, two 1-D four-point transform units, and a few small buffers. The gate count overhead of these new components is very little.

B. Schedule of Proposed Encoder

In the intra encoder design, the major scheduling problem is the data dependency of neighboring blocks since each intra prediction will use the reconstructed data from its left and upper blocks. During these reconstruction cycles, the intra prediction unit will be idle. Thus, the scheduling challenge is how to hide the reconstruction bubble cycles and keep the intra prediction unit busy. Besides, the newly adopted variable pixel parallelism architecture also introduces a problem that demands a new scheduling technique. In our variable pixel parallelism architecture, we adopt a block-size buffer at the boundary between four-pixel parallelism quantization phase and eight-pixel parallelism prediction phase. However, this is not enough since the recomputed coefficients for best luma and eight-pixel parallelism prediction phase. However, this is not enough since the recomputed coefficients for best luma and chroma modes cannot be passed directly because of different parallelism architectures. Thus, these coefficients will be blocked. This will result in a larger buffer to store temporarily blocked data or low hardware utilization with empty cycles in the prediction phase.

To solve these problems, we propose the interlaced best mode computation strategy as shown in Fig. 5. We interleave and insert the successively recomputed best modes for luma 16 × 16 and chroma components into the normal prediction modes that may not pass to quantization phase. If the best mode is a 16 × 16 mode, the recomputed luma 16 × 16 best mode is interleaved with modes of normal chroma components as shown in Fig. 5. This can improve the hardware utilization in the prediction phase without wasted cycles and keep the data flow continuous in the quantization phase. If the best mode is a 4 × 4 mode, only the best mode of chroma 8 × 8 is recomputed and interleaved with normal chroma modes because the best mode of intra 4 × 4 mode is already saved in the buffer and recomputation is not necessary. If the 4 × 4 mode is chosen as the best mode, the total cycle count for encoding a macroblock with the interlaced method can be reduced to 522, which is about only 52% of that in design [3].

C. Design of Eight-Pixel Parallelism Intra Prediction Generator and Transform

The eight-pixel parallelism intra prediction generator consists of two four-pixel parallelism units. One is for even rows and one is for odd rows in a 4 × 4 block. Fig. 6 shows the eight-pixel parallelism intra prediction generator. Its input ports can switch to select any neighboring data in registers for different modes except horizontal and vertical mode. The bypass input ports are used for horizontal and vertical modes because the inputs of these two modes are passed to output directly. After data input, the computation for each mode is done by selecting the appropriate data path through multiplexers.

In additional to eight-pixel parallelism intra predictor, the transform unit also adopts eight-pixel parallelism for the same data throughput. The eight-pixel parallelism transform design as shown in Fig. 7 includes two 1-D four-point row transform units and two 1-D four-point column transform units for computations. To fit such design, the corresponding transpose registers are decomposed as four 2 × 2 transpose register array. With such arrangement, we only need two additional 1-D transform units when compared to the design with four-pixel parallelism [3].

With above designs, we can double the architecture throughput with small area overhead.

V. IMPLEMENTATION AND DESIGN COMPARISON

A. Implementation Results

The proposed intra frame encoder was designed using Verilog HDL and implemented using TSMC 0.13-µm CMOS technology. Fig. 8 shows the effects of the proposed techniques. With these three techniques, we can process a macroblock (MB) with 560 cycles. Thus, the final design can achieve HD720p-30 frames/s encoding at 61 MHz and HD1080p 30 frames/s encoding at 140 MHz. For digital still camera applications, our design can process a 4096 × 2304 image with 6.78 frames/s. The total gate count is 94.7 K for HD1080p 30-frames/s encoding at 140 MHz. Table II lists the final results of gate count for each component. Most of the area is spent on boundary prediction buffer, quantization, DCT, and cost generator for mode decision as shown in Table II. Fig. 9 shows the circuit layout.

B. Comparison With Previous Works

Table III shows the comparison to other designs. For the same HD720p 30-frames/s encoding requirement, this design can reduce 48% of operating frequency compared with [3] (encoding
part, only) because of lower cycle count. With lower operating frequency, the critical path timing is thus relaxed and the area cost is 23.5% lower than [3]. Moreover, our design can support HD1080p 30-frame/s encoding at 140 MHz but with similar gate count as [3]. Compared with the standard definition (SD, 720 × 480)-sized encoder in [4], this design reduces the working frequency by 57.6% for SD-sized support. In addition, comparing with another HD-sized design [5], our gate count reduction reaches 50.8%.

For the intra predictor part, Table II shows that our design only needs 6 K gates, which save 40% of gate count when compared to the full search design [15]. For other fast algorithm designs, the gate count in [13] and [14] are 28.5 K gates and 15 K gates, respectively.

VI. CONCLUSION

A high-throughput and low-cost H.264/AVC intra frame encoder is presented in this paper with just 94 K gate and 0.72-mm² core area at 140 MHz. We have applied techniques such as fast prediction algorithms, variable-pixel parallelism, and other
scheduling techniques to optimize this design. Compared with previous designs for HD720p 30 frames/s, this design can reduce the gate count by 23.5% but only with 52% of working frequency. With these improvements, the new design can support DVR applications with HD1080p 30-frames/s resolution in real time. Besides, the encoder also can support digital still camera application with 4096 $\times$ 2304 resolution at 6.78 frames/s. Further extension to full intra-only profile is straightforward by including 8 $\times$ 8 transform and intra prediction and higher bit-width per pixel.

**ACKNOWLEDGMENT**

The authors would like to thank the National Chip Implementation Center (CIC) for providing cell-based design tools and cell library.

## REFERENCES


