

Design of High-Performance Intra Prediction Circuit for H.264 Video Decoder

Jihye Yoo, Seonyoung Lee, and Kyeongsoon Cho

Abstract—This paper proposes a high-performance architecture of the H.264 intra prediction circuit. The proposed architecture uses the 4-input and 2-input common computation units and common registers for fast and efficient prediction operations. It avoids excessive power consumption by the efficient control of the external and internal memories. The implemented circuit based on the proposed architecture can process more than 60 HD (1,920x1,088) image frames per second at the maximum operating frequency of 101 MHz by using 130 nm standard cell library.

Index Terms—Intra prediction, H.264, video decoder, circuit architecture

I. INTRODUCTION

The Joint Video Team of ISO/IEC MPEG and ITU-T VCEG proposed a video compression standard known as H.264 [1] with the emphasis on the efficiency and robustness. The intra prediction in the H.264 video compression makes use of similarities among the neighbors in the current frame while the inter prediction uses the previous or future frames as a reference frame. The intra prediction has nine modes of operation for a luma 4x4 block, four modes of operation for a luma 16x16 block and four modes of operation for a chroma 8x8 block. Each prediction mode includes various computations such as addition and multiplication, and many of the modes require a large amount of computational efforts. Furthermore, a larger image resolution is required in order to provide a better image quality and it

results in the significant increase of complexity. Therefore the circuit architecture for the intra prediction should be very efficient to manage such a large amount of computations.

This paper proposes an efficient architecture of the intra prediction circuit for the H.264 video decoder. The intra prediction circuit based on the proposed architecture uses the 4-input and 2-input common computation units for fast prediction operations. Common registers are used to store the data computed by the common computation units. Many of the data are reused by the proper control of the common registers. An efficient management of the data required in the prediction operations using the external and internal memories reduces the power consumption caused by the complex memory accesses. Our circuit can process more than 60 frames of high definition (HD) image with 1,920x1,088 pixels per second by using 130 nm standard cell library.

This paper consists of four sections. In Section II, the proposed architecture is described. Section III presents the experimental results and finally Section IV concludes the paper.

II. PROPOSED ARCHITECTURE

1. Overall Intra Prediction Circuit

The base architecture of our intra prediction circuit is the one described in [2]. As illustrated in Fig. 1, the overall architecture of the proposed intra prediction circuit consists of four modules: 1) neighboring samples buffer ('NSB') module to store the neighbor sample pixels for the prediction operations of the next sub-macroblock; 2) syntactic elements decoder ('SED') module to decode the intra prediction modes transferred from the variable length decoding (VLD) module; 3) predict

Manuscript received Aug. 23, 2009; revised Nov. 1, 2009.
Department of Electronics and Information Engineering, Hankuk
University of Foreign Studies Yongin, Korea
E-mail : kscho@hufs.ac.kr

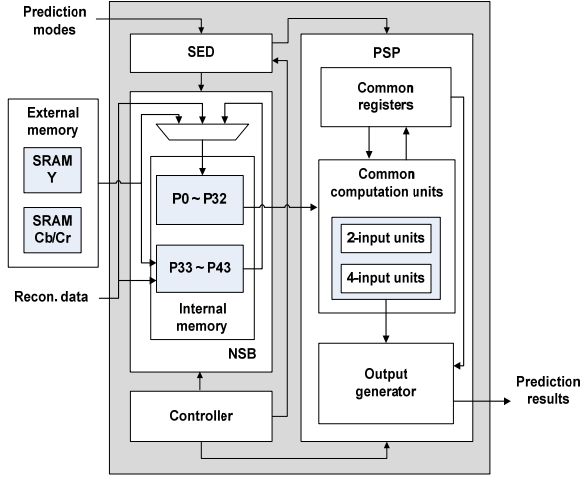


Fig. 1. Overall architecture of intra prediction circuit.

samples processor (‘PSP’) module to compute the intra prediction results and transfer them to the outside of the intra prediction circuit; 4) ‘Controller’ module to control the above three modules.

Since we maintain the modularity of each module, the operations to store the pixels in the external and internal memories can be performed in parallel with the intra prediction operations. These parallel operations improve the overall performance of the intra prediction circuit.

2. Common Computation Units and Common Registers

There are a total of 17 modes of intra prediction operations: 1) nine modes for a luma 4x4 block; 2) four modes for a luma 16x16 block; 3) four modes for a chroma 8x8 block. While the vertical and horizontal prediction modes are straightforward and do not require any computation, the other prediction modes require various kinds of computations. In [3], the computations involved in all of the 17 prediction modes are expressed by the following equation:

$$F(W, X, Y, Z, \alpha) = (W + X + Y + Z + 2) \gg \alpha \quad (1)$$

The common computation unit [3] has been proposed to implement the function described by Equation (1). The unit accepts four inputs and consists of four adders and one shifter, as shown in Fig. 2 (a). We further investigated each prediction mode and found that some of the computations can be expressed by the following simpler equation:

$$F(a, b, \beta) = (a + b + 1) \gg \beta \quad (2)$$

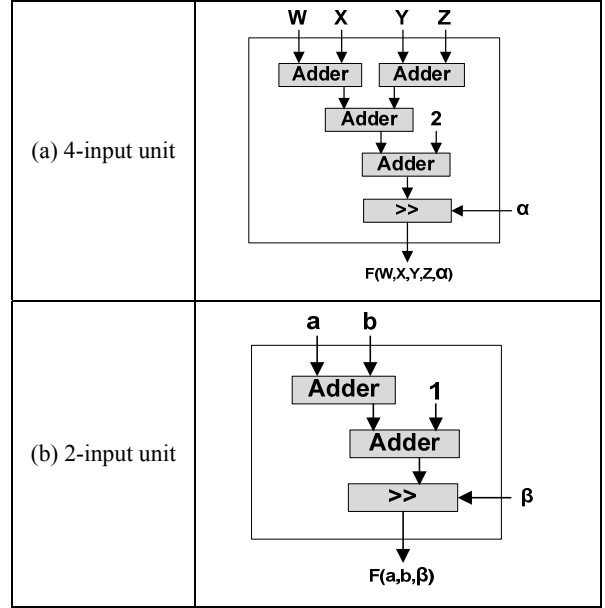


Fig. 2. Common computation units.

We propose to use another common computation unit to implement the function described by Equation (2). As shown in Fig. 2 (b), it accepts two inputs and consists of two adders and one shifter. Notice that the 2-input unit is smaller and faster than the 4-input unit. Since they are not only compact but also reusable, the various computations for all of the prediction modes can be performed by using them. Eight common computation units (five 4-input units and three 2-input units) are required to process all the prediction modes. One multiplier and several shifters are additionally required for the plane mode.

The outputs of the all common computation units are transferred to the outside of the intra prediction circuit as the prediction results. As shown in Fig. 3, we use seven 14-bit common registers. The prediction results of a sub-macroblock for all the prediction modes except the DC, plane, horizontal and vertical modes are generated at a rate of 16 pixels per clock cycle using eight common computation units. Some of the eight prediction results computed at the first clock cycle are stored in the common registers to be reused. They are not computed at the second clock cycle to avoid unnecessary power consumption. The intermediate prediction results for the DC and plane modes are also stored in the common registers and reused.

Fig. 4 shows an example of one of the nine prediction modes for a luma 4x4 block: mode 6, i.e., the horizontal

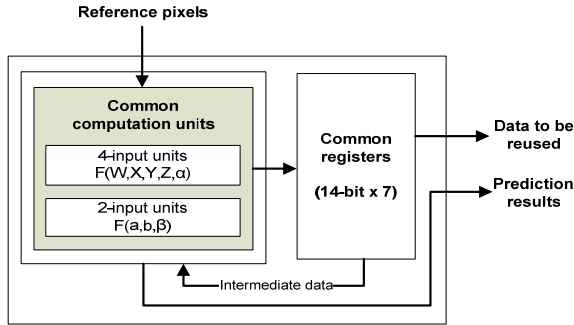


Fig. 3. Data reuse with common registers.

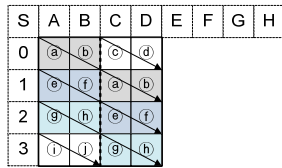


Fig. 4. Horizontal down mode for a luma 4x4 block.

down mode. In this figure, the pixels denoted by 0~3, A~H and S represent the neighboring sample pixels and the pixels denoted by (a)~(j) are the ten different prediction results. In the horizontal down mode, the predictions are performed according to the direction denoted by the arrows. Eight prediction results out of 16 in the left half of the 4x4 sub-macroblock are generated at the first clock cycle. Six prediction results (a),(b),(c),(f),(g),(h) are stored in the common registers and reused in the next clock cycle. As another example of data reuse, Equation (3) shows one of the four prediction modes for a luma 16x16 block: mode 3, i.e., the plane mode. In this equation, pred16x16_L is the final prediction results of the plane mode. The intermediate results t1, t2, t3, t3x3, t3x5, t3x6 and t3x7 are stored in the common registers and reused when necessary. Without the common registers, the same predictions would be made in duplicate causing unnecessary power consumption. Since the common registers are used in most of the prediction modes, the reusability is very high.

$$pred_{16 \times 16_L} = Clip_1((t1 + t2 \times (x - 7) + t3 \times (y - 7) + 16) \gg 5, \quad \text{with } x, y = 0..15$$

$$\text{where,}$$

$$t1 = 16 \times (p[-1,15] + p[15,-1])$$

$$t2 = (5 \times H + 32) \gg 6$$

$$t3 = (5 \times V + 32) \gg 6$$

$$H = \sum_{x'=0}^7 (x'+1) \times (p[8+x',-1] - p[6-x',-1])$$

$$V = \sum_{y'=0}^7 (y'+1) \times (p[-1,8+y'] - p[-1,6-y'])$$

3. External and Internal Memories

The prediction modes for a luma 4x4 block require more memory accesses than other prediction modes. It results in longer processing time and larger power consumption. In order to reduce the external memory accesses, the internal memory is used to store the reference pixels to be used right away or in the near future as shown in Fig. 5 (a). The internal memory consists of 42 8-bit words (0~15, A~P, S, x0~x5 and C0~C2). The neighboring sample pixels, i.e., reference pixels for a macroblock are stored in 0~15 and A~P. The left reference pixels of sub-macroblocks 0, 1, 4, 5 (2, 3, 6, 7) are stored in 0~3 (4~7) and the upper reference pixels of sub-macroblocks 0, 2, 8, 10 (1, 3, 9, 11) are stored in A~D (E~H). In case of prediction modes 4, 5 and 6 for a luma 4x4 block, we need the pixels in the left upper corners. x0~x5, C0~C2 and S are used to store them. The pixels stored in C0~C2 are used for the predictions of the next macroblock. After the predictions are completed, the internal memory is overwritten by the reconstructed data as shown in Fig. 5 (b). For example, sub-macroblock 3

| | | | | | | | | | | | | | | | | |
|----|----|---|----|---|---|----|----|---|---|----|----|---|---|----|---|----|
| S | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P |
| 0 | | | | | | | | | | | | | | | | |
| 1 | 0 | | | | | 1 | | | | 4 | | | | 5 | | |
| 2 | | | | | | | | | | | | | | | | |
| 3 | | | x0 | | | | x1 | | | | x4 | | | | | C0 |
| 4 | | | | | | | | | | | | | | | | |
| 5 | 2 | | | | | 3 | | | | 6 | | | | 7 | | |
| 6 | | | | | | | | | | | | | | | | |
| 7 | | | x2 | | | | x3 | | | | x5 | | | | | C1 |
| 8 | | | | | | | | | | | | | | | | |
| 9 | 8 | | | | | 9 | | | | 12 | | | | 13 | | |
| 10 | | | | | | | | | | | | | | | | |
| 11 | | | x0 | | | | x1 | | | | x4 | | | | | C2 |
| 12 | | | | | | | | | | | | | | | | |
| 13 | 10 | | | | | 11 | | | | 14 | | | | 15 | | |
| 14 | | | | | | | | | | | | | | | | |
| 15 | | | x2 | | | | x3 | | | | x5 | | | | | |

(a)

| | | | | | | | | | | | | | | | | |
|----|----|--|----|---|---|----|---|---|----|----|---|---|---|----|---|---|
| | | | | | | | | | I | J | K | L | M | N | O | P |
| | | | | | | | | | 0 | 1 | | | | | | |
| | | | | | | | | | 0 | 1 | | | | | | |
| | | | | | | | | | 0 | 1 | | | | | | |
| C0 | | | x0 | | | E | F | G | H3 | | | | | | | |
| | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | |
| C1 | | | A | B | C | | | | | | | | | | | |
| 8 | | | | | | | | | | | | | | | | |
| 9 | 8 | | | | | 9 | | | | 12 | | | | 13 | | |
| 10 | | | | | | | | | | | | | | | | |
| C2 | | | | | | | | | | | | | | | | |
| 11 | | | | | | | | | | | | | | | | |
| 12 | | | | | | | | | | | | | | | | |
| 13 | 10 | | | | | 11 | | | | 14 | | | | 15 | | |
| 14 | | | | | | | | | | | | | | | | |
| 15 | | | | | | | | | | | | | | | | |

(b)

Fig. 5. Internal memory management for reference pixels.

Table 1. Comparison of implementation results

| | Proposed | [4] | [5] | [6] |
|-------------------------|----------|-----------|----------|---------|
| Area (gates) | 26,607 | 49,126 | 28,707 | 20,400 |
| SRAM (Kbytes) | 3.75 | N.A. | 4.93 | N.A. |
| Technology (nm) | 130 | 180 | 180 | 250 |
| Image size | HD 1080 | CIF, QCIF | HD 1024p | HD 1080 |
| Maximum frequency (MHz) | 101 | 108 | 120 | 104 |
| Frames/sec | 60 | 30 | 30 | N.A. |
| Clock cycles/ MB | 112 | N.A. | 490 | 450 |

uses four sample pixels stored in 4~7, four sample pixels stored in E~H and one sample pixel stored in x0. The pixels stored in C0~C2 are used in the prediction for sub-macroblocks 2, 8 and 10, respectively. Two more internal memories are used for the chroma blocks: one for Cb and the other for Cr.

III. EXPERIMENTAL RESULTS

We designed the proposed intra prediction circuit at register transfer level (RTL) using Verilog hardware description language (HDL). The RTL circuit was verified using the simulator NC-Verilog from Cadence and synthesized into the gate-level circuit using the logic synthesizer Design Compiler from Synopsys and 130 nm standard cell library. The maximum operating frequency of the synthesized gate-level circuit is 101 MHz. Since our circuit requires 112 clock cycles to process one macroblock including luma and chroma data, it can process more than 60 frames of HD image with 1,920x1,088 pixels per second. The number of gates in the synthesized circuit is 26,607. The size of the dual-port static random access memory (SRAM) used in our circuit is 3.75 Kbytes. Table 1 shows the comparison of the implementation results. The size of the proposed circuit is smaller than [4] and [5]. It is larger than [6], but the number of clock cycles required to process one macroblock is much smaller than [6]. We process eight pixels per clock cycle for the most of the prediction modes by using the common computation units. Only

two clock cycles per sub-macroblock are required to make predictions for a luma 4x4 block. By utilizing external and internal memories efficiently, the memory access time is greatly reduced. All these techniques resulted in the performance improvement compared to others.

IV. CONCLUSIONS

In this paper, we proposed the architecture of the intra prediction circuit for the H.264 video decoder. In order to process HD video in real time, we used the 4-input and 2-input common computation units and common registers with high reusability. For an efficient memory management we used the internal memory to store the data to be used right away or in the near future and thereby reduced the external memory accesses. The proposed circuit can process more than 60 frames of HD image at the maximum operating frequency of 101 MHz by using 130 nm standard cell library.

ACKNOWLEDGMENTS

This work was supported by Hankuk University of Foreign Studies Research Fund of 2009.

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), Mar, 2003.
- [2] W. T. Staehler, E. A. Berriel, A. A. Susin, and S. Bampi, "Architecture of an HDTV Intraframe Predictor for a H.264 Decoder," *2006 IFIP International Conference*, Oct. 2006, Page(s):229–233.
- [3] J. Shim, S. Lee, and K. Cho, "Design of Intra Prediction Circuit for H.264 Decoder Sharing Common Operations Unit," *Journal of the Institute of Electronics Engineers of Korea*, Vol.45-SD, Issue 9, Sep. 2008, Page(s):103–109.
- [4] J. Park and S. Lee, "Design of Memory-Access-Efficient H.264 Intra Predictor Integrated with Motion Compensator," *Journal of the Institute of Electronics Engineers of Korea*, Vol.45-SD, Issue 6, Jun. 2008, Page(s):611–616.

- [5] T-C. Chen, C-J. Lian, and L-G. Chen, "Hardware Architecture Design of an H.264/AVC Video Codec," *Asia and South Pacific Design Automation Conference*, Jan. 2006, Page(s):750–757.
- [6] C. Lee, "Design of Scalable Intra-Prediction Architecture for H.264 Decoders," *Journal of the Institute of Electronics Engineers of Korea*, Vol 45-SD, Issue 11, Nov. 2008, Page(s):1108–1113.



Jihye Yoo received the B.S. degree in the Department of Electronics and Information Engineering from Hankuk University of Foreign Studies, Korea, in 2008. She is currently pursuing the M.S. degree in the Department of

Electronics and Information Engineering at Hankuk University of Foreign Studies, Korea. Her research interests include SoC architecture and design for H.264 video codec.



Seonyoung Lee received the B.S. and M.S. degrees in the Department of Electronics and Information Engineering from Hankuk University of Foreign Studies, Korea, in 1998 and 2000, respectively. From 2001 to 2006, he

was a researcher of Enhanced Chip Technology. He is currently pursuing the Ph.D. degree in the Department of Electronics and Information Engineering at Hankuk University of Foreign Studies, Korea. His research interests include SoC architecture and design for multimedia.



Kyeongsoon Cho received the B.S. and M.S. degrees in Electronics Engineering from Seoul National University, Korea, in 1982 and 1984, respectively. He received the Ph.D. degree from the Department of Electrical and Computer Engineering at Carnegie Mellon

University, U.S.A. in 1988. From 1988 to 1994, he was a senior researcher in Semiconductor ASIC Division of Samsung Electro-nics Company. He was responsible for research and development of ASIC cell library and design automation. Since 1994, he has been a professor in the Department of Electronics and Information Engineering at Hankuk University of Foreign Studies. In parallel with the academic research and education, he has been also very active in the industrial sector. From 1999 to 2003, he was a senior director of Enhanced Chip Technology. From 2003 to 2004, he was a head of CoAsia Korea Research and Development Center. Since 2005, he has been a technical advisor of Dongbu HiTek and a vice director of a Collaborative Project for Excellence in System IC Technology sponsored by the Ministry of Knowledge Economy, Korea. His current research activities include SoC architecture and design for multimedia and communications, SoC design and verification methodology, and very deep submicron cell library development.