A 100 MHZ 1920x1080 HD-PHOTO 20 FRAMES/SEC JPEG XR ENCODER DESIGN

Ching-Yen Chien[1], *Sheng-Chieh Huang*[1], *Shih-Hsiang Lin*[1], *Yu-Chieh Huang*[1], *Yi-Cheng Chen*[1], *Lei-Chun Chou*[1], *Tzu-Der Chuang*[2], *Yu-Wei Chang*[2], *Chia-Ho Pan*[2] *and Liang-Gee Chen*[2]

[1]Sense/TCM SOC Lab, Department of Electrical and Control Engineering,
National Chiao Tung University, HsinChu, Taiwan
[2]Graduate Institute of Electronics Engineering and Department of Electrical Engineering,
National Taiwan University, Taipei, Taiwan
E-mail: gxinside@gmail.com

ABSTRACT

A JPEG XR chip for HD-Photo is implemented with 25 mm^2 area in TSMC 0.18um CMOS 1P6M technology at 100MHz. According to the simulation results, the 4:4:4 1920x1080 HD-Photo 20 frames/sec can be encoded smoothly.

***Index Terms*—** JPEG XR, HD-Photo, Encoder, Chip

1. INTRODUCTION

JPEG XR is a new still image coding standard, derived from the window media photo standard (WMP) [1-3]. Due to improvements in display devices, the JPEG XR is designed for high dynamic range (HDR) and HD Photo size. The coding flow is shown in the Figure 1.

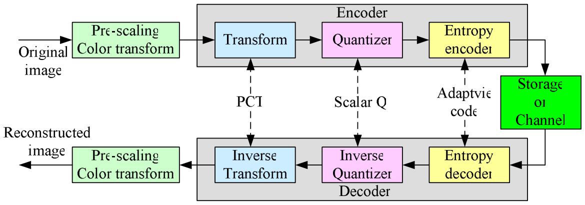


Fig. 1. The JPEG XR coding flow.

Compared with that of the JPEG2000, the coding flow of the JPEG XR has lower complexity at similar PSNR quality at the same bit rate. Hence, JPEG XR is very suitable for implementation with the dedicated hardware used to manage HD-Photo size images for HDR display requirements. In the following section, the novel JPEG XR encoder architecture design is presented.

2. ALGORITHM

In the following section, the different conditions are discussed.

2.1. Tiles

In the first step of the JPEG XR compression, the tile size has to be decided for optimization of the compression result. Because different tile sizes result in different compression outcomes, the Table I shows the compression results under the same quantization with different tile size for different benchmark images.

When the image has been divided into the different tiles, the Huffman table has to be changed accordingly and retransmitted, and the adaptive scan order has to be rebuilt as well. This method is beneficial for the progressive mode while the image size will be bigger.

Table I. The different tiles for the benchmark image

Q = 5	**No tiles**	**8x8**	**4x4**	**1x32**	**1x1**
Lena	344 K	346 K	350 K	345 K	433 K
Baboon	484 K	490 K	504 K	497 K	704 K
Airplane	274 K	276 K	279 K	279 K	332 K
Pepper	387 K	390 K	397 K	392 K	501 K
Q = 70	**No tiles**	**8x8**	**4x4**	**1x32**	**1x1**
Lena	8.34 K	9.55 K	12.6 K	10.9 K	38.9 K
Baboon	31.8 K	32.8 K	35.7 K	34.3 K	60.7 K
Airplane	9.27 K	10.5 K	13.8 K	12.3 K	39.6 K
Pepper	10.4 K	11.6 K	14.6 K	12.8 K	39.7 K

According to the above table, when choosing a small tile size, the encoded image size is bigger, and the row-by-row style is the best for the progressive mode and the compression ratio.

2.2. Overlapping, PCT, Quantization and the Prediction

There is three comparisons of overlap functions, each with different trade-offs: Non-overlap, One-level overlap and the Two-level overlap. The non-overlap condition enables faster encoding and decoding but is useful in low compression ratios mode only. The one-level overlap function, compared to the non-overlap, has a higher compression ratios but it needs more time for those results. The two-level overlap has

978-1-4244-1764-3/08/$25.00 ©2008 IEEE

the highest compression ratios but poor image quality. The one-level overlap has the best overall performance quality among all three.

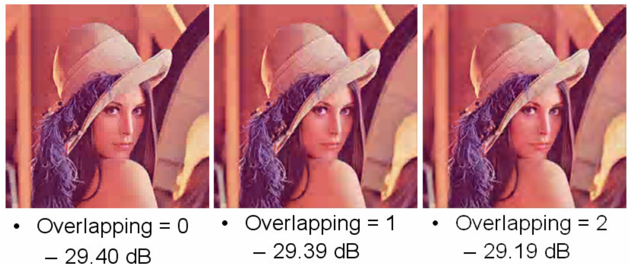


Fig. 2. The effect of the overlapping function.

There are two Photo Core Transform (PCT) stages. In PCT, the 16x16 block is partitioned into 16 4x4 blocks. The first PCT stage is on the 4x4 blocks. The second stage is on the 4x4 DC block from the first stage. Fig. 3 shows the PCT transformed method. In each stage, each 4x4 block is transformed by 4x4 PCT. So, there are three bands after PCT transformed. First band is DC band, next band is low pass band (AD), and last band is high pass band (AC).

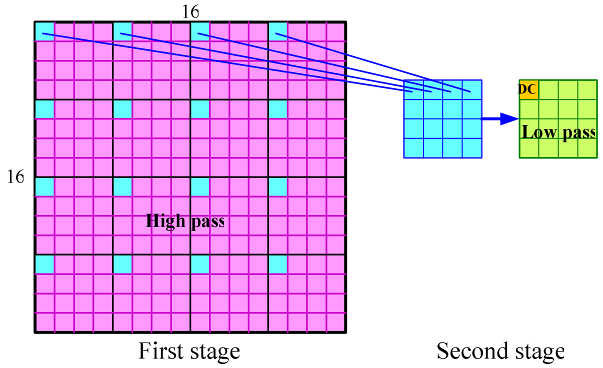
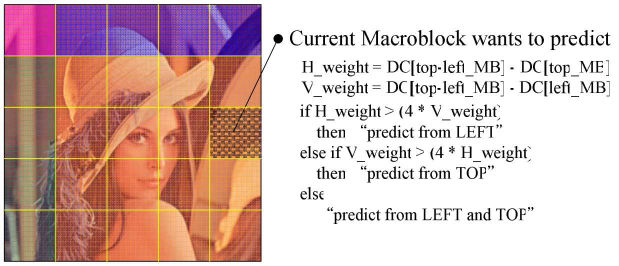


Fig. 3. The 1st stage and 2nd stage for PCT.

The DC prediction is very different from that of traditional encoder design. As shown in Figure 4, the JPEG XR uses the left and top block to a predict block's position instead of using the block's own position value.



```
H_weight = DC[top-left_MB] - DC[top_MB]
V_weight = DC[top-left_MB] - DC[left_MB]
if H_weight > (4 * V_weight)
   then   "predict from LEFT"
else if V_weight > (4 * H_weight)
   then   "predict from TOP"
else
      "predict from LEFT and TOP"
```

Fig. 4. DC prediction model.

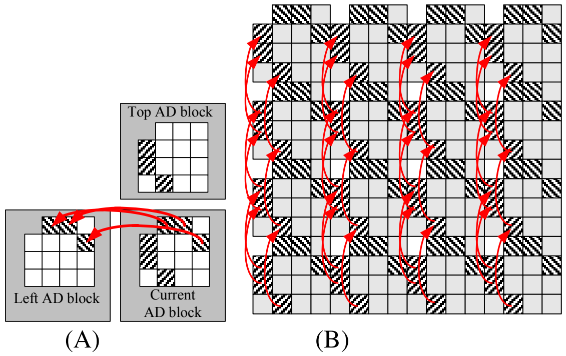
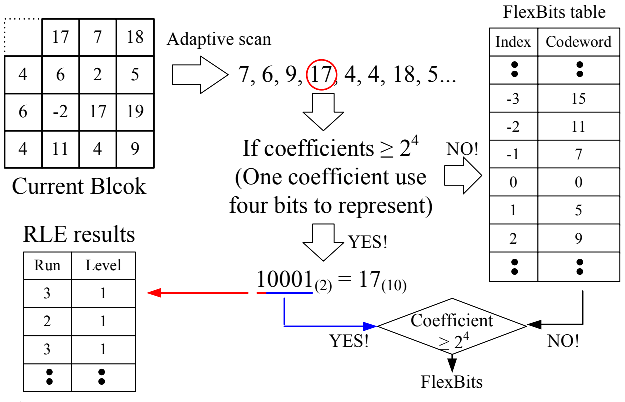


Fig. 5. (A)The example of AD block predict from LEFT
(B)The example of AC block predict from TOP

As shown in the Fig. 5(A), the AD block can be predicted from blocks on its TOP or LEFT. The AD prediction direction follows DC prediction model that can be decided in the Fig 4. After comparisons between the horizontal weighting and the vertical weighting of AD block, the prediction can be decided whether to be used or not. The AC computation is similar to AD, so the same prediction method can be used to design the hardware architecture. Fig. 5 shows the prediction methods.

2.3. Adaptive Encode (AE)

The adaptive scan based on the latest probability is used for the most efficient entropy coding. The most probable non-zero coefficients are scanned first and the probability is counted by numbers of the non-zero coefficients. The Run-Level Encode (RLE) function is added into the bitstream length for the overhead coefficient. The scheme is shown as the following (Figure 6). As illustrated, the 4 bit can not represent 17, and the encoding block will increase extra bit for the new RLE coding.



	17	7	18
4	6	2	5
6	-2	17	19
4	11	4	9

Current Blcok

RLE results

Run	Level
3	1
2	1
3	1
⋮	⋮

FlexBits table

Index	Codeword
⋮	⋮
-3	15
-2	11
-1	7
0	0
1	5
2	9
⋮	⋮

Fig. 6. Adaptive scan and Run-Level Encode - AD & AC.

2.3. The Simulation Results

The following shows the bit per pixel results for JPEG XR. As compared with the JPEG2000, the JPEG XR has the low computation cost and the similar PSNR quality at the same bitrate. Hence, the JPEG XR is very suitable to be implemented with the dedicated hardware to deal with HD photo size image for the HDR display requirement.

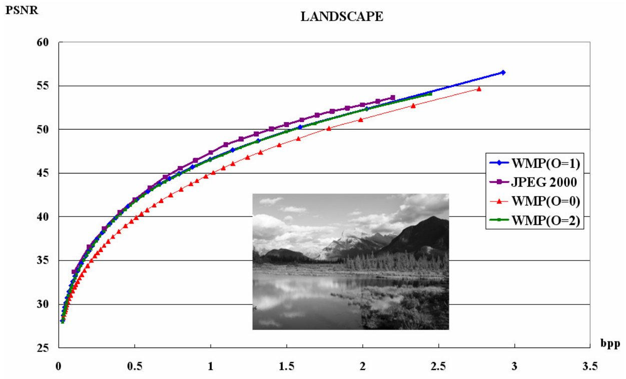


Fig. 7. Comparison of 3072x2048 benchmark image.

3. ARCHITECTURE

The proposed chip speed is 100 MHz for 4:4:4 1920x1080 20 frame/sec. The pipeline stage consideration, the pre-filter, transform and the quantization architecture are all discussed in the following sections in regards to the chip implementation and system requirement. Because the color conversion, pre-filter and PCT are computing with the 4x4 block matrix style, they are arranged into the same stage at the beginning.

For memory allocation, the color conversion needs 6 blocks per column, the pre-filter needs 5 blocks, and the PCT and quantization needs 4 blocks in order to execute the encoding function. Whenever a new pipeline starts, there is one pipeline bubble that can be used to process the DC block and improve the efficiency of encoding, as shown in the following figure:

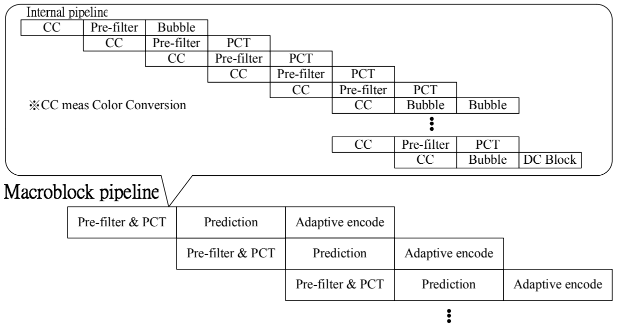


Fig. 8. JPEG XR chip pipeline stage.

3.1. Data reuse skill, Pre-filter, PCT and Quantization

When the red range is to be processed after the brown, the pre-filter function only needs to deal with a small amount of new data coming into the register. This is due to the last macroblocks still being stored in the registers and can be utilized by the red range. Therefore, the memory size for pre filter will be reduced. Otherwise, the additional SRAM will be needed to store the data of the whole red range to the off-chip memory and execution time will be increased.

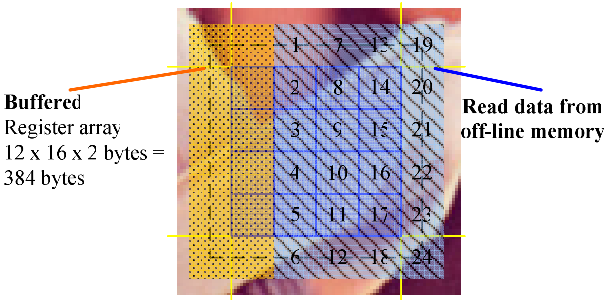


Fig. 9. The Data Reuse.

As shown in the Fig. 10, The PCT consists of two parts: Function 1 (Hardmard transform, T_h), and Function 2 (Hardmard transform, T_h), with sub-function blocks as the T_odd and the T_odd_odd. The DC function will be processed after the PCT.

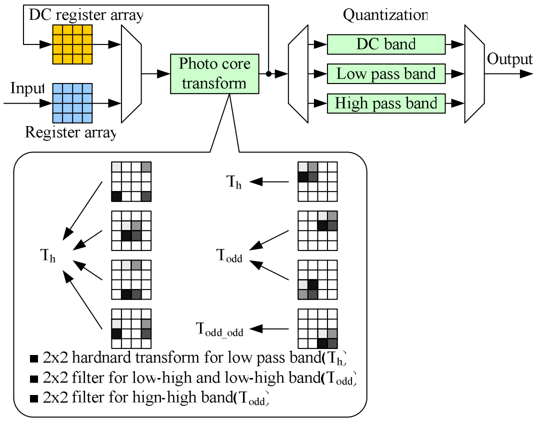

DC register array
Input
Register array
Photo core transform
Quantization
DC band
Low pass band
High pass band
Output
T_h
T_h
T_{odd}
T_{odd_odd}
■ 2x2 hardnard transform for low pass band(T_h)
■ 2x2 filter for low-high and low-high band(T_{odd})
■ 2x2 filter for hign-high band(T_{odd})

Fig. 10. PCT and Quantization.

3.2. Prediction

The data are arranged into the SRAM blocks and processed with the subtract operation as the prediction algorithm. The data of AD block are stored at the AD SRAM for buffering because the current AD block may be predicted from top as shown in Fig. 11.

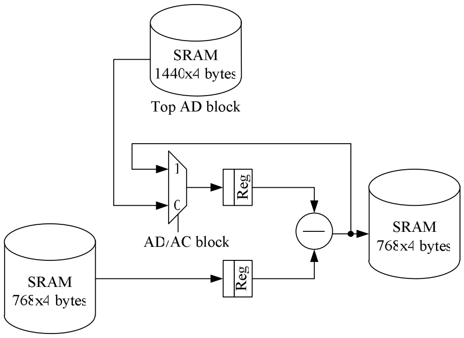


Fig. 11. AD/AC prediction architecture.

3.3. Adaptive Encode

There are three complex feedback loops in the adaptive encode function block. The first one is the update-scan-order block used to refresh the scan order. The second is the update ModelBits MB block, which decides how many bits are necessary to represent one coefficient. And the adaptive Huffman encode block to choose the most efficient code and the adaptive Huffman encode block to choose the most efficient Huffman table. In this stage, entropy coder only encodes the DC coefficients. The AD and AC coefficients are skipped.

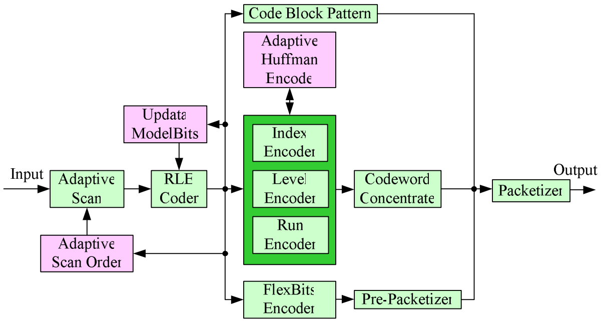


Fig. 12. The Adaptive Encode Architecture.

4. IMPLEMENTATION

The synthesis results are shown on the following table, where Table II shows the results with the Synopsys DC using tsmc 0.18 um 1P6M technology.

Table II. Power Comparison

Function Blocks	Power (mW)
Color conversion	9.4
Pre-filter	82
PCT/Quantization	79.6
Prediction	10
Adaptive Encode	35

The chip layout diagram is shown in the following:

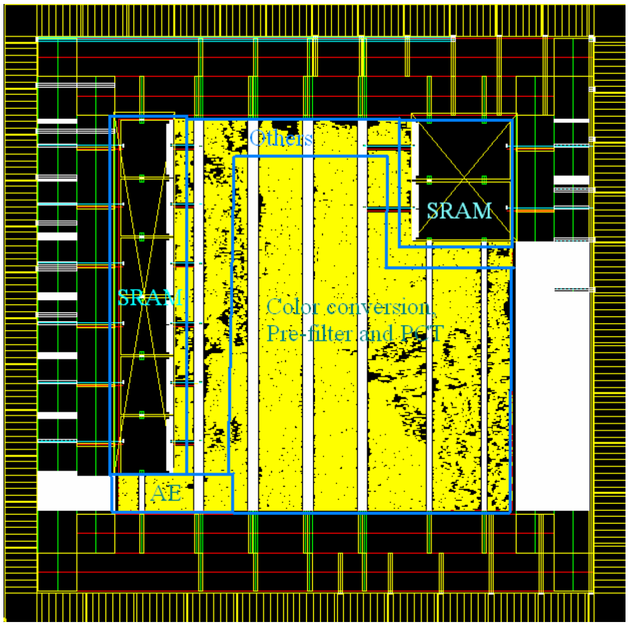


Fig. 13. Chip layout.

5. CONCLUSION

In this paper, a novel 100 MHz 1920x1080 HD-Photo 20 frames/sec JPEG XR encoder design is proposed. For the next generation HDR display, the digital surveillance, the mobile phone, cameras of all uses, etc, such a powerful and advanced still image compression chip will be required and will be used extensively. In comparing with the existing JPEG2000 encoder chip, the JPEG XR has the higher cost-performance benefit.

ACKNOWLEDGMENTS

The authors thank Megan Montgomery and Hsia-Ching Lee for editing support.

6. REFERENCES

[1] Windows Media Photo. Microsoft, 2007 [online]. Available: http: /www.microsoft.com/whdc/xps/wmphoto.mspx

[2] B. Crow, "Windows Media Photo: A new format for end-to-end digitalimaging," *Windows Hardware Engineering Conference*, Seattle, 2006.

[3] D.D.Giusto, T.Onali, "Data Compression for Digital Photography: Performance comparison between proprietary solutions and standards," *ICCE* 2007.

[4] L.V. Agostini, I.S. Silva, S. Bampi, "Pipelined Entropy Coders for JPEG compression," *Integrated Circuits and System Design*, 2002.