

Research Article

Minimized Area and High Speed EBCOT Architecture for JPEG 2000

¹S. Vijayaraghavan and ²C. Parthasarathy

¹Department of ECE,

²Department of IT, SCSVMV University, Kanchipuram, India

Abstract: In this study we have proposed minimized area and high speed EBCOT architecture for JPEG 2000. Embedded block coding with optimized truncation is an algorithm in JPEG 2000 image compression system. In several existing high speed EBCOT architecture is there in our proposed it overcomes and produce code generation. In our study open that image rate of more context dual generation is about 74.8%. To encoding the all image samples in a column, a new formulated named as pact context coding is invented as a important, high devised is used for less hardware. The proposed architecture is described in VHDL language, verified by simulation and successfully implemented in a Cyclone II and Stratix III FPGA. It provides a major reduction in memory access requirements, as well as a net increase of the processing speed as shown by the simulations. The C*D Quantizer coder is improved by the operating system and stage. The full design of processor encoder is tested on FPGA based. The results show that invented of the proposed architecture 172.28 M samples/sec is equaling to encode 1920*1080 (4:3:3) HD camera picture sequence at 39 f/sec. the bit plane architecture operates 315.06 MHZ which that implies that it is 4.03 times faster than the but plan coder so far. It is used many applications like satellite image, medical image and image compression system.

Keywords: Architecture, bit plane coder, EBCOT, FPGA, JPEG 2000

INTRODUCTION

The newest international standard of JPEG 2000 (ISO/IEC 15444-1, 2000) was proposed in December 2000. It has better quality at low bit rate and higher compression ratio than the widely used still image compression standard JPEG. The decompressed image is more refined and smoother (Acharya and Tsai, 2004; Rabbani and Joshi, 2002). Furthermore, JPEG 2000 has more novel functions such as progressive image transmission by quality or resolution, lossy and lossless compressions, region-of-interest encoding and good error resilience. Based on these advantages, JPEG 2000 can be used in many applications such as digital photography, printing, mobile applications, medical imagery and Internet transmissions (JPEG Official Website, 2000; Lee, 2005; Schelkens *et al.*, 2009).

The architecture of JPEG 2000 consists of Discrete Wavelet Transform (DWT), scalar quantization, context-modeling arithmetic coding and post compression rate allocation. It handles both lossless and lossy compressions using the same transform-based framework and adopts the idea of the Embedded Block Coding with Optimized Truncation (EBCOT) (Kishor and Swapna, 2012; Pearlman *et al.*, 2004). Although the EBCOT algorithm offers many benefits for JPEG 2000,

the EBCOT entropy coder consumes most of the time (typically more than 50%) in software-based implementations. In EBCOT, each sub band is divided into rectangular blocks (called code blocks) and the coding of the code blocks proceeds by bit-planes. To achieve efficient embedding, the EBCOT block coding algorithm further adopts the fractional bit-plane coding ideas and each bit-plane is coded by three coding passes (Lee, 2005).

In this study we present an efficient VLSI architecture for EBCOT. It's based on an optimized data organization and a new memory arrangement as well as a simple state machine and combinatorial logics of encoding part. Our proposed architecture makes the four bits to be processed and their neighbors available at one clock cycle and consequently a complete column is processed in only four clock cycles during each pass. This proposed architecture is implemented on FPGA without using any external memory.

EBCOT algorithm: EBCOT algorithm consists of two processing one is bit plane coding and another is Matrix Quantizer Coder (MQC). Bit plane coding operates on data stored in a CB and produces Context Decision (C*D) pairs the MQ codes saves these C*D pairs produces an embedded bit stream (Wang *et al.*, 2008; Xiong *et al.*, 2005).

METHODOLOGY

Bit plane coding: To using Discrete Wavelet Transforms (DWT) coefficients into sign magnitude format, after that data stored in CB memory.

During the encoding process image scanning from MSB to LSB. After the bit plane is further partitioned into stripes to assume all states variable are assumed zero. Finally, to take samples are encoded and C*D pairs are generated.

Matrix quantizer coder: The MQ coder is based on binary arithmetic coder the coder is contains index look up table and probability estimation table for JPEG 2000 standard shown in Fig. 1 (Rhu and Park, 2009). The procedures to run the input symbol are selected index lookup table. It is used to code operation a register denotes B is used to proper interval and register D is used to data stored the partial codeword.

In this compressed bit stream the output of MQ coder. If MQ coder is cutoff at the end of all bit planes single bit stream is generated. The results are high efficiency coding.

Existing system: The existing architecture for bit plane coding is presented architecture design of block coding engine for EBCOT. To improve encoder throughput, sample skipping and group of columns skipping techniques are used. About 13 kb on chip memory is required in this architecture and 4.6 MS/sec (M Samples/sec) throughput is reported at 50 MHz. The MQ coder is terminated and flushed after coding the entire CB. However, the effect of significance propagation is not considered.

Dynamic significance states that restoring technique. Here, instead of storing state variables in separate memory, they are reconstructed by analyzing several bit planes concurrently. The designed MQ coder is capable of consuming two CxD pairs simultaneously. At 66 MHz, throughput reported is 22 MS/sec. However, to reconstruct state variables, four magnitude bit planes are essential. Due to this, large on chip memory is required. Besides this, if magnitude of any sample is zero to restore its pass membership, it is necessary to perform significance propagation test in each pass.

Parallelism at three levels bit planes, coding passes and coding samples. All bit planes are coded concurrently. However, single first in, first out is used to store CxD pairs generated by two different bit planes. Similarly, one MQ coder is shared between the two-bit planes. To encode a CB of N×N size about

0.35-0.46×N×N clock cycles are required. However, two context windows are used to code all samples in a column.

Pass-parallel method is adopted to code all samples in a column within one clock cycle. Here, using data of various bit planes, state variables are computed on the fly. A state variable schedule unit is designed for this purpose. However, this unit constitutes many logic levels which might have restricted throughput to 50 MS/sec at 100 MHz. BPC architecture is capable of generating two CxD pairs simultaneously. This design has two MQ coders. One is shared between passes 1 and 2, whereas a separate coder is used for pass 3. The average throughput of BPC is 40.47 MS/sec at 100 MHz. However, to generate two CxD pair's implementation of four ZC, four SC and four MRC primitives are inessential.

Architecture for BPC and MQ coder based on concurrent symbol processing. The BPC design presented in this is capable of producing ten CxD pairs concurrently. Additionally, it can skip multiple columns, if required. In the MQ coder design, leading zero forwarding technique is implemented and it is capable of consuming two symbols at a time. But, PET memory requirement of this design is approximately three times that of the reference MQ coder.

Proposed system: The proposed EBCOT architecture is illustrated shown in Fig. 2. The memory controller selects a magnitude and sign bit plane to be encoded. Additionally, it reads σ and σ_{-} state variable data for the selected CB. Since magnitude and sign data are never updated, these planes are implemented using Single Port RAM (SPRAM). The state memories are implemented using Dual Port RAM (DPRAM). Once a bit plane is selected, the stripe controller selects a stripe to be processed. While processing the first column of every stripe, its left side neighboring column is assumed to be zero. Similarly, while processing the last column of every stripe, its right side neighboring column is assumed to be zero. These boundary conditions are handled by the boundary handler.

In order to process all magnitude bits concurrently, entire magnitude column and the corresponding sign bits column are read in one clock (Wintner, 2006). In addition, four sign neighbors, eight σ neighbors and four σ_{-} values of the column to be processed are read which form reference windows. Based on the values in these windows, pass detector determines the coding pass to be run and enables necessary primitive coding modules. With the help of one RLC, four ZC, four SC

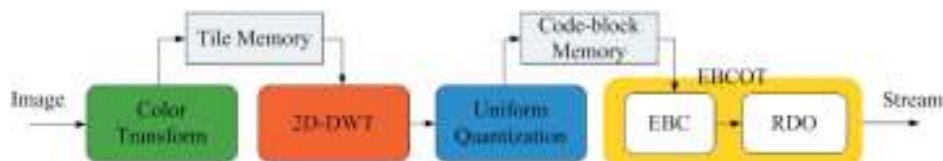


Fig. 1: JPEG 2000 system block diagram

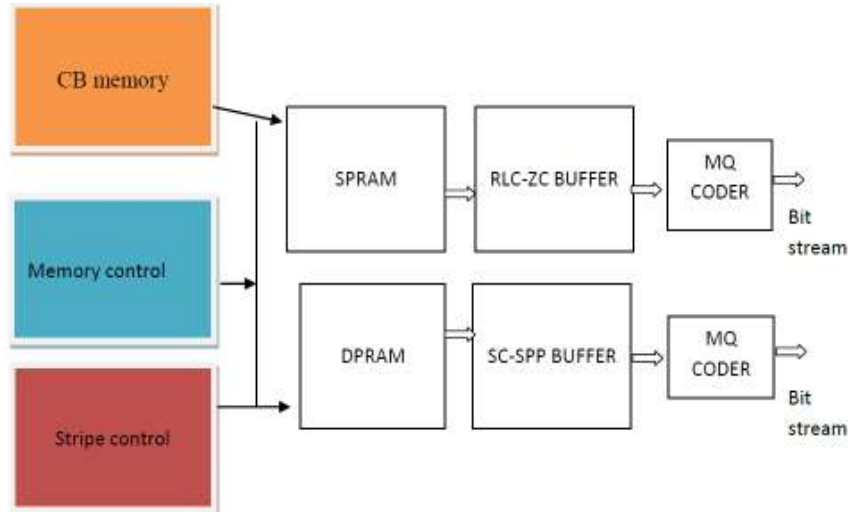


Fig. 2: Proposed EBCOT processor

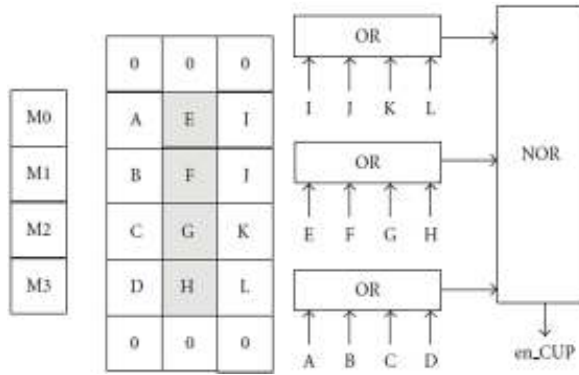


Fig. 3: Pass detection logic

and four MRC primitives, all samples in a column are coded concurrently. As a result, 1 to 10 $C \times D$ pairs get generated simultaneously. The context sequencer segregates these $C \times D$ pairs and stores them in the respective pass buffers. The MQ coders consume buffered $C \times D$ pairs and produce separate compressed bit stream corresponding to each coding pass.

Zero coding: To encode DWT coefficients context tables are provided in the JPEG 2000 standard. The ZC primitive encodes DWT coefficients in LL and LH sub band using single table, which is provided in the standard. First, summation of horizontal, vertical and diagonal σ values is computed and next they are compared against the standard table to determine context of a sample. Therefore, first σ values are summed and next contexts are produced. The VLSI realization of ZC primitive for LL and LH sub band. In ZC, current sample's magnitude bit (i.e., M_x) is treated as a decision bit. To encode all samples, in a column, concurrently four such modules are implemented in this architecture shown in Fig. 3.

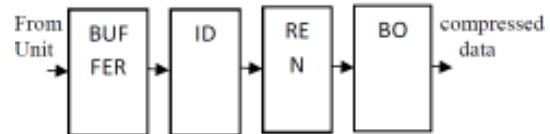


Fig. 4: Proposed MQ coder

MQ coder: Block level representation of the proposed MQ coder is depicted in Fig. 4. This coder consumes one $C \times D$ pair in one clock cycle. It is partitioned into three stages: Interval Detector (ID), Renormalized (REN) and Byte Out (BO). The $C \times D$ pairs generated by BPC are stored in the $C \times D$ buffer and are supplied to the ID stage. This stage reads an index value from the ILT and puts it on the address bus of the PET ROM. Depending on the index value received; PET ROM emits Q_e , NMPS, NLPS and switch information.

RESULTS AND DISCUSSION

To investigate relationship between contents of a CB and number of $C \times D$ pairs generated, five grayscale ISO images Lena, Boboon, Boat, Peppers and Barbara of size 512×512 are used. In this experiment, Daubechies (9, 7) filter bank is used. After three levels of image decomposition ten bits of each DWT coefficient are stored in 256 code blocks. Using C simulation all CBs in an image is encoded and context pairs are generated for analysis purpose. This study demonstrates that in an image the rate of $C \times D$ pair generation is totally dependent on the contents of a CB. Hence, it is difficult to predict the number of contexts produced in a bit plane.

The proposed pass parallel, concurrent sample code is described using Verilog and prototyped on the Xilinx XC4VLX80-12 FPGA. The design implementation summary which demonstrates that the area requirement of the proposed design is less, whereas operating speed

is very high. The BPC module requires only 288 clock cycles to encode one bit plane. To cope up with the BPC speed three MQ coders are used in this design. The speed of renormalizes is three times that of the MQ coder. This implies that if more than three rotations are required, one more clock is required to encode the symbol. But it is a rare situation. Therefore, it can be assumed that one C×D pair is consumed in a clock cycle. To achieve the best case performance, multiple clock domains are used in this design. The MQ coder renormalizes and BPC modules are operated at 108, 324 and 432 MHz, respectively. Considering 12 bps in the DC application, 8 μS are sufficient to encode one CB. Thus, the proposed design is capable of processing 2048×1080 size 57 DC frames in a second.

The proposed architecture is implemented on XC2V1000 FPGA for the imprison purpose. BPC architecture is sequential in nature. Moreover, it uses two processing elements to speed up the EBCOT performance. As a result, it has demanded large number of hardware resources. During MQ coding operation, a maximum 15 shifts may occur. Data width of the registers to be shifted (i.e., A and C) are much larger. Therefore, if barrel shifters are used in the re normalizes stage, hardware cost increases and at the same time operating frequency decreases.

CONCLUSION

In this study we compared contexts of a CB and number of C×D pairs generated are surveyed. In our proposed MQ coder produces continuous in nature and it is used to speed up in this module. To encoding the two symbols per clock, as output the high speed MQ coder can be realized at low hardware cost. The C×D Quantizer coo improve the overall efficiency of the proposed EBCOT processor multiple clock domains are used. The full design of processor encoder is tested on FPGA based. The results show that invented of the proposed architecture 172.28 M samples/sec is equaling to encode 1920*1080 (4:3:3) HD camera picture sequence at 39 f/sec. the bit plane architecture operates 315.06 MHz which that implies that it is 4.03 times faster than the but plan coder so far.

REFERENCES

- Acharya, T. and P.S. Tsai, 2004. JPEG 2000 Standard for Image Compression: Concepts, Algorithms and VLSI Architectures. John Wiley and Sons, Hoboken, New Jersey.
- ISO/IEC 15444-1, 2000. Information Technology-JPEG 2000 Image Coding System-Part 1: Core Coding System, 2000.
- JPEG Official Website, 2000. Retrieved from: www.jpeg.org/jpeg2000.html.
- Kishor, S. and B. Swapna, 2012. Area efficient, high speed EBCOT architecture for digital cinema. *ISRN Signal Process.*, 2012(2012): 9, Article ID 714176.
- Lee, D.T., 2005. JPEG 2000: Retrospective and new developments. *P. IEEE*, 93(1): 32-41.
- Pearlman, W.A., A. Islam, N. Nagaraj and A. Said, 2004. Efficient, low-complexity image coding with a set-partitioning embedded block coder. *IEEE T. Circ. Syst. Vid.*, 14(11): 1219-1235.
- Rabbani, M. and R. Joshi, 2002. An overview of the JPEG 2000 still image compression standard. *Signal Process-Image*, 17(1): 3-48.
- Rhu, M. and I.C. Park, 2009. A novel trace-pipelined binary arithmetic coder architecture for JPEG2000. *Proceeding of IEEE Workshop on Signal Processing Systems (SiPS, 2009)*, pp: 243-248.
- Schelkens, P., A. Skodras and T. Ebrahimi, 2009. The JPEG 2000 Suite. J. Wiley, Hoboken, N.J.
- Wang, R., B. Li, H. Jiang and H. Cao, 2008. High-throughput and hardware-efficient architecture of MQ arithmetic coder. *Proceeding of the 11th IEEE Singapore International Conference on Communication Systems (ICCS, 2008)*, pp: 783-787.
- Wintner, R., 2006. Bits on the big screen. *IEEE Spectrum*, 43(12): 42-48.
- Xiong, C.Y., J.W. Tian and J. Liu, 2005. Efficient word-level sequential coding scheme of bit plane coding for EBCOT in JPEG2000. *Proceeding of the 9th International Symposium on Consumer Electronics*, pp: 468-472.