

FPGA Implementation of Hardware Architecture for H264/AV Codec Standards

Prof. Naveen Jain

*Department of Mechanical Engineering SSIPMT Raipur
naveenjain@ssipmt.com*

<p>Article History</p> <p>Article Submission 18 October 2012</p> <p>Revised Submission 17 January 2012</p> <p>Article Accepted 15 February 2012</p> <p>Article Published 31st March 2013</p>	<p style="text-align: center;">Abstract</p> <p><i>The proposed work is a modern hardware based architecture for performing transformation, quantisation and prediction is designed which is used for H.264/AVC video standards. This designed hardware find its importance in advanced H264 encoders which are repeatedly find its application in HDTV applications. The H264/AV Codec does video compression and video decompression for prospect broadband and wireless networks. A low complexity discrete cosine transform is used by DSP embedded multiplier. An intra-prediction equation are employed to get low latency, high throughput, efficient utilization of resources. The proposed architecture also employs both pipeline & parallel process methods. The proposed architecture is implemented using VHDL and synthesised for Virtex 5, and the device is 5v1x50tff665.</i></p> <p>Keywords: H264/AV Codec, discrete cosine transforms, intra prediction.</p>
--	--

I. Introduction

We are living in an era of electronics and communication. Since the development of long distance communication techniques in the 19th century, evolution of communication has always been striving towards higher data rates and better quality of information exchange [1]. This has necessitated the invention of new and newer communication technologies in a very rapid phase. Among the various modes of communication, video transmission is a most important application. Higher data rates imply large amount of data to be transmitted. In order to transmit large amount of data, video transmitters use data compression techniques. Compression of video data is achieved through standardized video codecs [2]. These compression techniques involve encoding of information using fewer bits than the original data. This is achieved by removing redundant information in the video data. Currently, VC products like digital television, video mobile devices, entertainment based electronic devices use quite a few VC standards like Motion Picture Expert Group-2, MPEG4, H.264/AVC are used [3]. Among these standards H.264/AVC is the most important and widely used technique for compression and transmission of HD videos which was developed by ITU codec experts group and ISO codec expert groups. The standard has high bit-rate efficiency and has network friendly representations. Most video codecs are characterized by multiplications, followed by additions, Subtractions and accumulations. These codecs have clock between 1MHz to 1GHz. Performance is measured by the MAC operations ranging from 10 to 4000 instruction speeds. Given these requirements, the FPGAs can work up to thousand times faster than conventional processors, as conventional processors are prone to sequential computations.

In recent years, Field Programmable Gate Array has been the cornerstone for benchmarking digital signal processing algorithms. High gate densities, parallel computing hardware fabrics and dedicated cores for signal processing makes the FPGAs ideal for signal processing. FPGAs are preferred to conventional Digital Signal Processors (DSPs) due to the flexibility of reprogramming and capability to perform parallel computations. The flexibility to manage the logical problems at gate level enhances the construction of a custom processor to proficiently implement a desired signal processing algorithm [4][6].

Also, DSPs are one time programmable, whereas FPGAs can be reprogrammed infinite number of times. The reprogramming consumes only a very few seconds. Design changes for a given circuit can be incorporated and

implemented quickly. Reconfiguration helps to minimize hardware. Also FPGA synthesis tools allow “parameter sable” cores that accept word length changes of signals to meet the accuracy of signal processing algorithms. In this paper, a modern hardware based architecture for performing transformation, quantisation and prediction is designed which is used for H.264/AVC video standards [5].

The article is organised as given: The proposed hardware structure is shown as section II which consists of transform, quantization and prediction. Section-III gives the implementation. At last, section IV of the paper shows conclusion.

II. Proposed Hardware Structure For Encoder

The proposed hardware structure of the encoder is shown in fig 1.

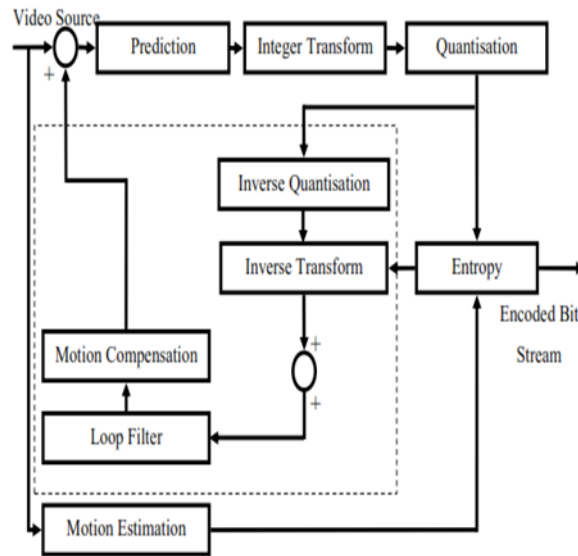


Fig1 H264/AV Codec encoder obstruct diagram

A. 4x4 forward transforms

In H264 video codec, 4x4 floating point transformations is employed, which tend be an approximation of real floating-point 4x4 transform. The 4x4 floating point transformation is expressed as

$$E_r = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 1 & -1 & -2 \\ 1 & -1 & -1 & 1 \\ 1 & -2 & 2 & -1 \end{bmatrix}$$

The information out of the intra-prediction block is then expressed in form of transform coefficients. The unique feature of the H264 video codec is that it uses a merely integer spatial transform which is usually 4x4 in shape as compared to the conventional 8x8 DCT. The integer transform eliminates any mismatch in encoder during the inverse transform process. Integer transforms also eliminate the losses introduced by rounding or truncation as compared to other transforms.

The 1-D integer transform can be computed by directly using a multiplier or by using a series of adders, subtractions and shift operations first by row and then proceeding by columns. In this paper, we will employ the direct multiplication method as embedded multipliers are computationally faster than cascaded adders and subtractions.

$$Y=(C_fXC_f^T)E_f \quad (1)$$

Where Y is the transformed matrix, C_f is the pixels of image in matrix format, E_f is the standard matrix.

The key features of DSP embedded multiplier is

1. Parallel and fixed constant coefficient multipliers
2. Fixed point multiplier with 2's complement
3. Input data width is to be 64 bits
4. Variable pipelining
5. Symmetric Rounding for DSP slice.

B. 4x4 Quantisation algorithm

Quantization algorithm scales down coefficients that are transformed to a pre-defined value. The quantization levels possible are 52 in number which is represented by the quantization parameter (QP). When quantization parameter is increased which leads to doubling of quantization step size by 2. A wider range of quantization levels enables the encoder to balance the trade-off that exists bit rate and its quality. between bit-rate and quality of the encoded bit stream.

$$Z_{ij} = \text{round}(Y_{ij} \frac{PF}{QStep}) \quad (2)$$

Where Z_{ij} is the coefficient after quantization.

$$|Z_{ij}| = (|Y_{ij}| .MF + f) \gg qbits \quad (3)$$

Where f – to avoid rounding errors. The quantization process is based on the step size. The step size increased for every profile. The quantization profile and step size is given in TABLE I.

Table I: Quantization Step Size

Quan_Prof	0	1	2	3	4	5
Quant_step	0.525	0.587	0.7125	0.775	1	1.025
Quan_Prof	6	7	8	9	10	11
Quant_step	1.025	1.3075	1.6025	1.705	2	2.205
Quan_Prof
Quant_step
Quan_Prof	48	49	50	51		
Quant_step	160.0	224.1		

The inverse quantization can also be implemented by

$$Y_{ij} = Z_{ij}.V_{ij}.2.\text{floor}(QP / 6) \quad (4)$$

The same multiplication factor which is used in quantization is also used in inverse quantization. These quantization is always has some errors but those errors are negligible in H.264.

Table ii: The MF Coefficient Table

Step size	Positions (0,0),(2,0),(0,2),(2,2)	Positions (1,1),(1,3),(3,1),(3,3)	Other positions
0	14107	6243	7166
1	11816	4560	6590
2	10182	4094	5254
3	9462	3547	6825
4	8092	3155	5143
5	7182	2793	4659

C. Prediction Modes Algorithm

Using the neighbouring blocks, intra-prediction algorithm calculate the necessary pixels in mb. A 16x16 luma block representation is formed for each luma component for prediction. A 4x4 luma block, 9 modes are available whereas in 16x16 luma block, 4 modes are available. These predictions are compared by mode decision algorithm and best luma is selected for the particular pixel.

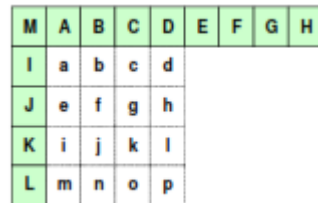


Fig 2 4x4 block & adjoining pixels

There are 9 4x4 luma modes designed in a direction based manner as shown in Figure 2. The individual pixel equations used in 4x4 diagonal down-left prediction modes are shown in figure 3.

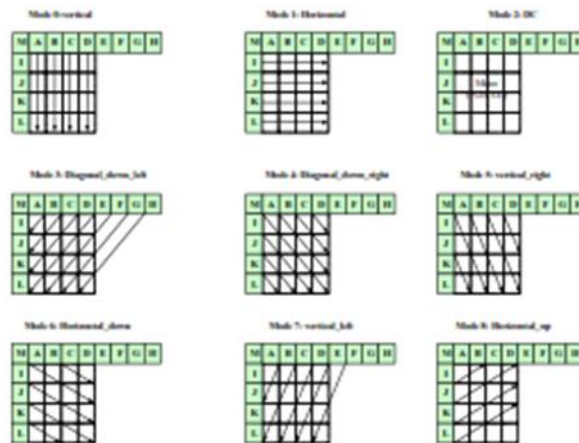
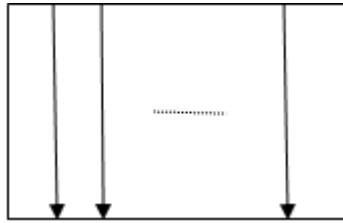
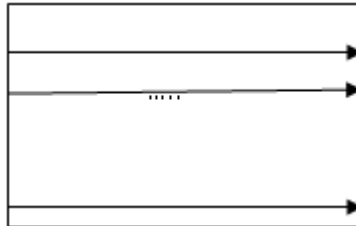


Fig 3: 4x4 prediction modes

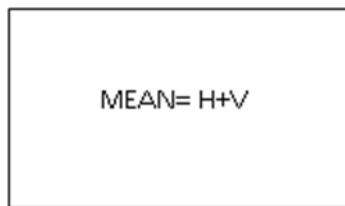
The prediction modes for 8x8 Chroma prediction is shown in figure 4 which is obtained from bilinear transformation of integer arithmetic. The various modes of 8x8 Chroma are given below:



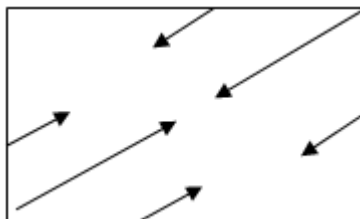
Mode-0 : vertical mode



Mode-1:horizontal mode



Mode-2-DC mode



Mode-3:plane mode

Fig 4: 8x8 chroma prediction

The Prediction Calculator gets the 13 neighbouring pixels from reconstructed blocks. Since all of the neighbouring pixels may be unavailable due to macroblock edges, there are valid inputs for each group of neighbouring pixels. The Prediction Calculator get all reconstructed pixels from reconstructed block and then calculates the equations needed to create all predicted values for all nine modes in parallel.[3]

III Simulation Results

The proposed H264/AV Codec hardware architecture is implemented using HDL. The proposed design for transform, quantization and intra-prediction was verified by simulation using Modelsim SE 10.1. Then the design was synthesized for a Virtex- 5v1x50tff665 Xilinx FPGA. The results are summarized in the table III.

Table 3: Resource Utilization for Transform, Quantization and Intra Prediction

FPGA Resource	Available	Used	Utilization in %
Registers	28800	2687	9
Slices	4005	1086	27
DSP 48E slices	48	12	25
Block RAM	2	60	3
Input/output	69	360	19

The presented architecture consumes less computational cycles. This is aided by mapping all complex multiplications to embedded multipliers (DSP48E slices). Simulations showed improvement in clock cycle count when mapping the multipliers to DSP48E slices instead of using distributed logic. The resource utilization of transform, quantization and intra prediction shown in table III shows better result than conventional approaches.

IV Conclusion

A modern hardware based architecture for performing transformation, quantisation and prediction is designed which is used for H.264/AVC video standards. This designed hardware find its importance in advanced H264 encoders which is repeatedly used in HDTV applications. Pipelining and use of DSP48E slices are the techniques used to reduce the computation cycles. The proposed H264/AV Codec hardware architecture was implemented using VHDL and verified by MODELSIM SE 10.1. The VHDL design was synthesized and works at a maximum clock frequency of 293.517 MHz for Xilinx Virtex-5 FPGA device.

References

- [1] E. G. Richardson, "H.264 and MPEG 4 Video Compression-Video Coding for Next Generation Multimedia", New York: Wiley, 2003.
- [2] 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.
- [3] Imen Werda, Haithem Chaouch, Amine Samet, Mohamed Ali Ben Ayed, Nouri Masmoudi, "Optimal DSP-Based Motion Estimation Tools Implementation for H.264/AVC Baseline Encoder," IJCSNS International Journal of Computer Science and Network Security, vol. 7, no. 5, 2007.
- [4] Imen Werda, Haithem Chaouch, Amine Samet, Mohamed Ali Ben Ayed, Nouri Masmoudi, "Optimal DSP-Based integer Motion Estimation Implementation for H.264/AVC Baseline Encoder," The International Arab Journal of information Technology, vol. 7, no. 1, January 2010.
- [5] Tham Y J, Ranganath S, Ranganath M et al, "A novel unrestricted center-biased diamond search algorithm for block motion estimation," IEEE Trans. on Circuits and Systems for Video Technology, vol.8, no. 4, pp 369-377, 1998.
- [6] C. Zhu, X. Lin, and L.P. Chau, "Hexagon-based search pattern for fast block motion estimation" ,IEEE Trans. on Circuits and Systems for Video Technology, vol. 12, no. 5, pp. 349–355, 2002.
- [7] D. Zhang, B. Li, J. Xu and H. Li, "Fast Transcoding from H.264 AVC to High Efficiency Video Coding," 2012 IEEE International Conference on Multimedia and Expo, Melbourne, VIC, 2012, pp. 651-656.

- [8] S. Qiao, Y. Zhang and H. Wang, "PI-Frames for Flickering Reduction in H.264/AVC Video Coding," 2012 International Conference on Computer Science and Service System, Nanjing, 2012, pp. 1551-1554.
- [9] A. Bjelopera and S. Grgić, "Scalable video coding extension of H.264/AVC," Proceedings ELMAR-2012, Zadar, 2012, pp. 7-12.
- [10] Y. Ismail, J. B. McNeely, M. Shaaban, H. Mahmoud and M. A. Bayoumi, "Fast Motion Estimation System Using Dynamic Models for H.264/AVC Video Coding," in IEEE Transactions on Circuits and Systems for Video Technology, vol. 22, no. 1, pp. 28-42, Jan. 2012.