ISSN No:-2456 –2165

# Implementation of Deblocking Filter for Reducing the Blocking Artifacts

P. Gowthami
PG Scholar (DSCE), Department of ECE, JNTUA CEA,
Ananthapuramu, Andhrapradesh, India

Dr. G. Mamatha
Assistant Professor, Department of ECE, JNTUA CEA,
Ananthapuramu, Andhrapradesh, India

**Abstract:-Blocking artifacts are one of the visible artifacts that will reduce the quality of an image. In order to improve the quality of a reconstructed image, a Deblocking filter is used. The main key role of Deblocking filter is to detect the artifacts & attenuate them by applying a selective filter around the boundaries of an image. In this paper, to remove the blocking effect we implemented the Deblocking filter architecture. We demonstrated the result by an image with & without blocking artifacts. The Experimental results show the area utilization & time delay is 4.663ns. The proposed deblocking filter operating clock frequency is 214MHZ.**

**Keywords**: Blocking Artifacts, Deblocking Filter, Filter Unit

## I. INTRODUCTION

Modern image coding standards try to remove as much as redundancy in the coded representation of video, to reduce the storage space, complexity, time delays. So one of the best coding technique is block based transform coding i.e.; Discrete cosine transform. This technique eliminates the redundancy bits which causes the discontinuities in an image .This effect is called "Blocking artifact". The blocking artifacts are visible discontinuities that occur at the boundaries of an image. It is one of the most annoying problems that degrade the quality of the reconstructed image. The Deblocking filter is used to remove the arifacts caused by block based transform coding & quantization. It enhances the image quality by applying a selective filter to smoothen the sharp edges around the boundaries.

The crucial point in designing a deblocking filter is to decide whether to filter or not, in addition to the strength of the filter to be applied. If the filtering is decided, it is applied to each edge of the blocks. The detailed algorithm & architecture of the deblocking filter is presented in this paper. The proposed deblocking filter architecture performs both horizontal & vertical filtering operations simultaneously. We focus on the quality of image, time delay, area occupation & operating clock frequency.

## II. RELATED WORK

Consider an image or frame, it consists of luminance & chrominance components. Luminance represents brightness (black& white) of an image. Chrominance represents colour .Let us consider a frame which is divided into equal blocks. Each block consists of one luminance & two chrominance components.

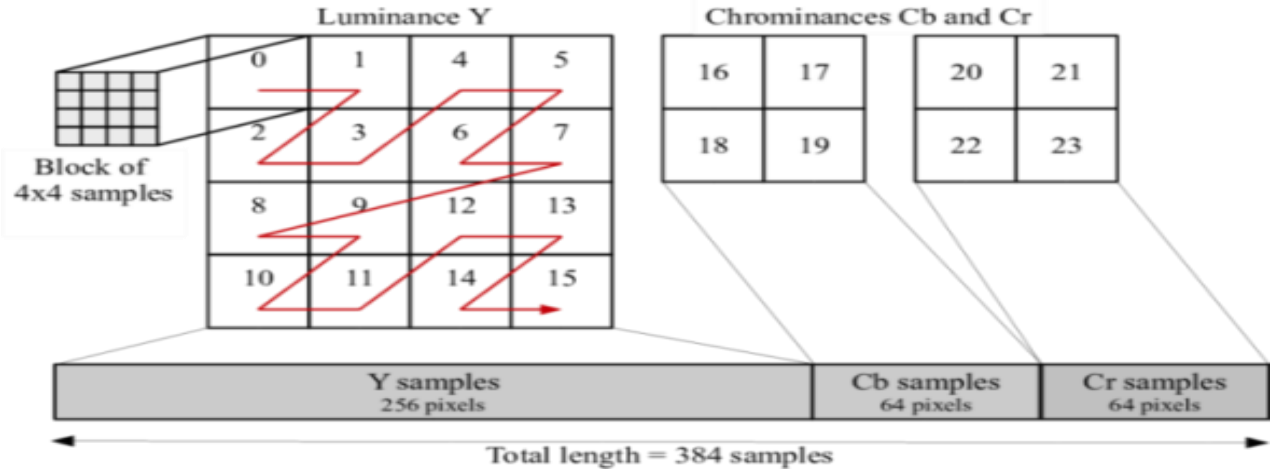


Fig. 1: Pixel Array of A Image Each Block is Divide Into 4x4 Samples with Luminance & Chrominance (ref4).

Suppose each block of a frame is divided into 4x4 blocks it will consists of one 4x4 luminance block & two 2x2 chrominance blocks .It is represented in the Fig1(ref4). In this paper we can consider each block as 8x8 in a frame, instead of 4x4 & applied as an input to the proposed architecture with blocking artifact data.

## III. PROPOSED ARCHITECTURE

In this paper we are implementing deblocking filter architecture to remove the artifacts & improve the visual quality of an image. Fig2 shows the proposed deblocking filter architecture.

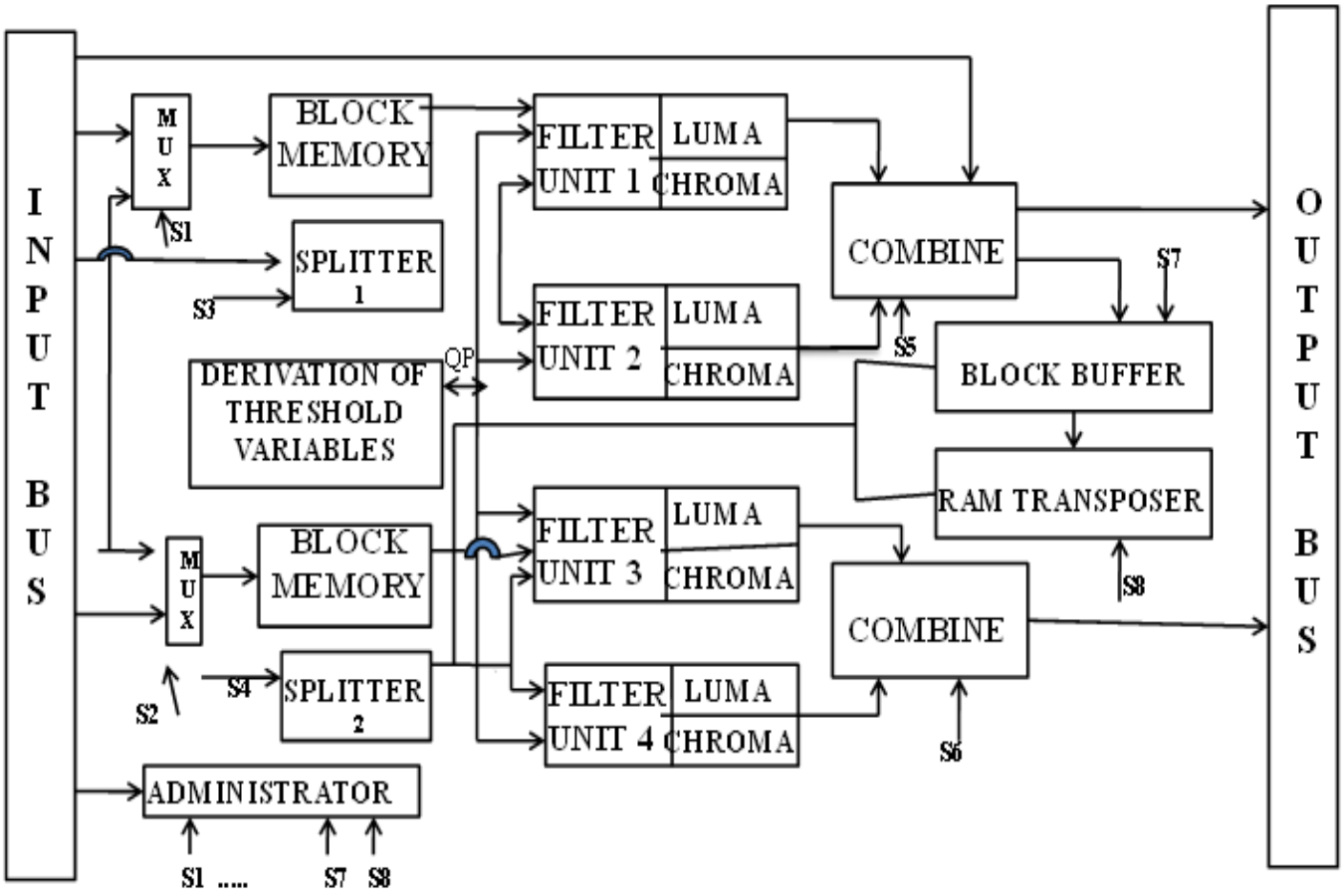


Fig.2: Proposed Deblocking Filter Architecture.

The deblocking filter architecture mainly consists of five components. The first component is the two block memories to store the data received from an external memory. Each memory is a 32x32 bit dual port SRAM. Block memory 1 is used to store left neighbouring blocks for filter unit 1. Block memory 2 is used to store right neighbouring blocks for filter unit 3.The second component is filter unit including two horizontal filters & two vertical filters. In addition, the luminance & chrominance are separated & executed at the same time. The third component is Derivation of threshold variables generating the threshold variables β & tc from the input quantization parameter (QP). In addition, a RAM transposer which is used to transpose the data flow of 8x8 or 4x4 blocks from row to column in order to reuse the data without storing back to the memory. The other two components in the proposed architecture are splitter & combine.

### *A. Filter Unit*

The crucial point in deblocking filter is to decide whether to filter or not, in addition to filter strength to be applied. It is decided in the filter unit module. The most important design in Filter unit is Filter Operator, which makes several decisions on filtering of each edge & mode of operation.

The proposed Filter unit has four modules are explained as follows. The architecture of filter unit is shown in Fig 3.

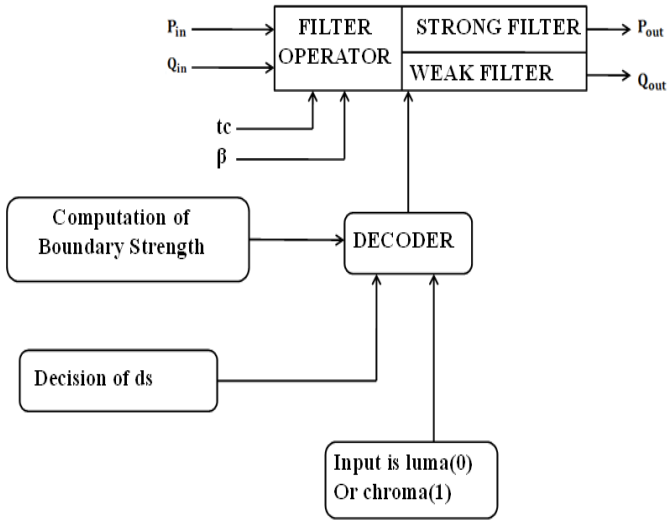


Fig.3: Filter Unit

*a). Computation of Boundary Strength*

To every edge between two 4x4 luminance sample blocks, boundary strength (Bs) parameter is assigned. There are three boundary strength values 0, 1 & 2 .The value depends on coding conditions of two adjacent blocks. For luma component, only block boundaries with Bs value equal to 1 or 2 are filtered. In case of chroma components, Bs equal to 2 are filtered.

*Determination of Bs value for the boundary between adjacent blocks (Fig4):*
If P & Q are intra blocks then Bs=2.
If P& Q satisfies the following conditions Non zero transform coefficients or Different number of reference frames Then Bs=1.Otherwise Bs=0.

*b). Decision of ds:*

If Bs is greater than zero, the decision of ds is applied on the deblocking of each edge.

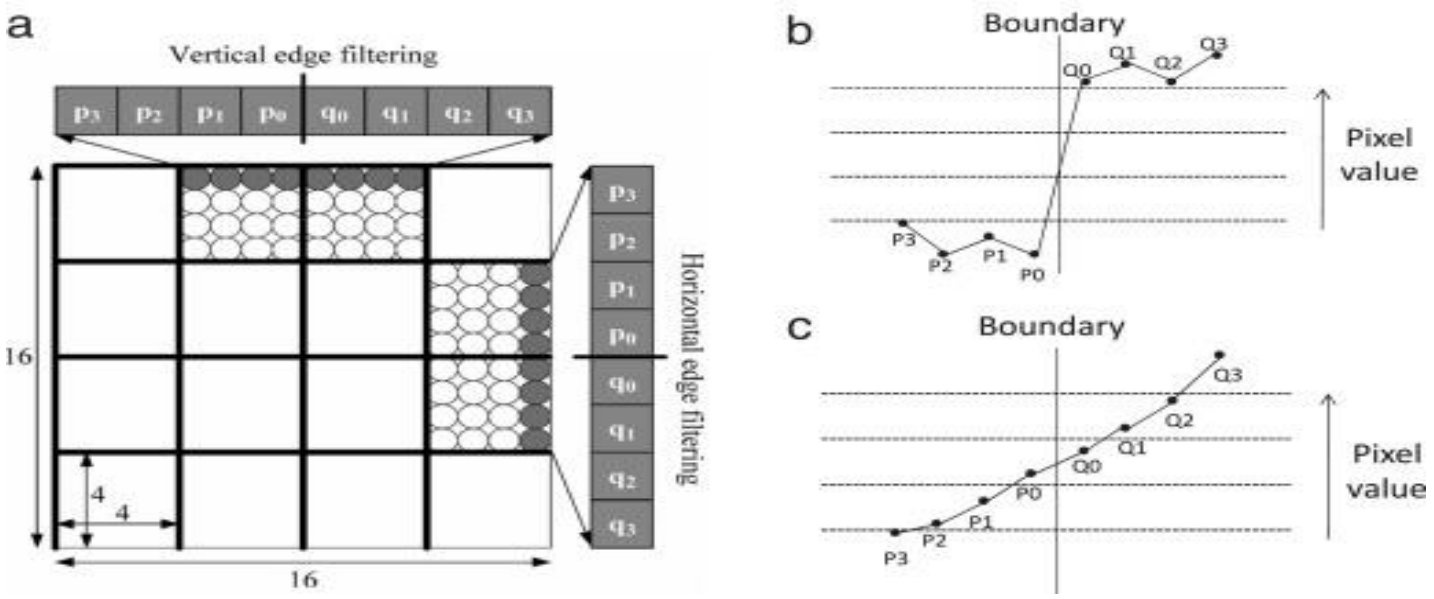


Fig.4: Boundary Filtering Applied Between two Adjacent Block Edges of 4x4 Samples (ref3).

To reduce the complexity, only the first & fourth lines in a block boundary of length4 are evaluated as shown in fig4(ref3). In the first step, the following condition is checked to decide whether the deblocking filter is applied or not:

$d$ = dp+ dq = $|p20 - 2p1\ 0 + p00| + |p23 - 2p1\ 3 + p03| + |q20 - 2q1\ 0 + q00| + |q23 - 2q1\ 3 + q03| < \beta$. (1)

Where, threshold β depends on QP. If d is smaller than β, filter is applied to the edge, otherwise not.

In the second step to determine the filter mode operation, the following condition is determined:

$dS = d < (\beta >> 2)\ \&\ (|p3i - p0i| + |q3i - q0i| < (\beta >> 3))\ \&\ (|p0i - q0i| < ((5tc >> 1))$ (2)

Where i=0,3. The threshold variables β and *tc* are derived from QP which is discussed in ref (1, 5) from standard lookup table.

*c). Decoder:*

Based on the above information, Decoder will decides which pixel should be filtered. When Bs is equal to 0, no filtering is applied. When Bs is not equal to 0 & d is smaller than, no filtering is applied either. When Bs & ds are not equal to 0, strong filtering is applied. Otherwise, if ds is equal to 0, the weak filter is applied. In addition, the control signals to distinguish between the luminance & chrominance a component is also sent to decoder. Decoder will generate the final decisions to Filter Operator.

*d). Filter Operator:*

The filter operator is the most important module in the filter unit. After receiving the values of Bs & checking several conditions, filter operator stars to filter pixels & executes two filter modes: strong & weak.

For Bs>0, if the condition (1) is only satisfied, Weak filter is applied. Then, p0, p1,q0 & q1 are the filtered(ref1).

If conditions (1) & (2) both are satisfied, then strong filter is applied. Then, p0 to p2 and q0 to q2 are the filtered pixels (ref1).

*Algorithm of Deblocking filter:*

- For each edge of 4x4 determine the filter boundary strength (Bs).
- Depending upon the coding conditions the Bs value is assigned to each edge.
- According to the filter strength & average quantization parameter determine threshold values tc & β.
- If Bs is greater than zero, apply the conditions (1) & (2).
- If condition (1) is only satisfied, weak filter is applied.
- If condition (1) & (2) both are satisfied, Strong filter is applied.
- According to the values of the edge pixels & tc, β modify (if needed) the pixels.
- Thus finally, reconstructed image is obtained after filtering.

## IV. EXPERIMENTAL RESULTS

Before simulation & synthesis, Consider an image with blocking artifacts & upload the image in MATLAB software & run the program. It displays the matrix values of blocking effect image.

The obtained matrix values are given as input for deblocking architecture in verilog code & simulated in Xilinx software. The processing time for each clock cycle is 3000ns. The overall simulation time is 16,709ns. Fig5 shows the simulation result of deblocking filter architecture.

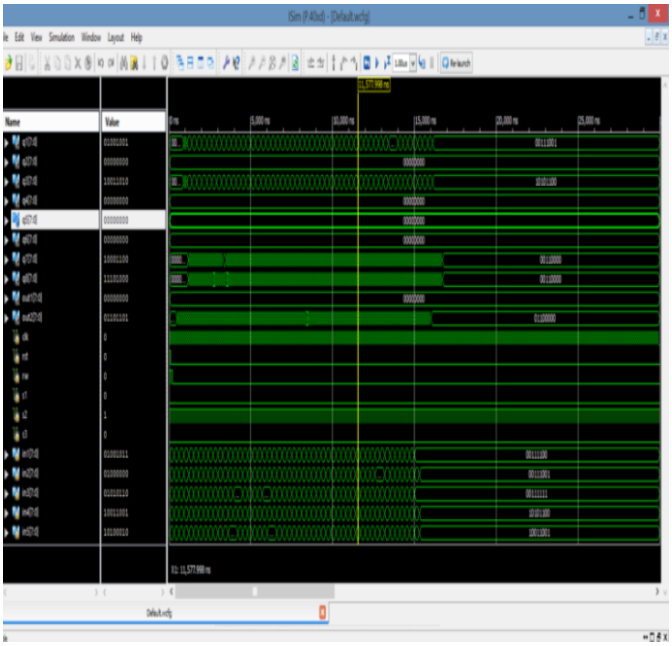

Fig.5: Simulation Result of Deblocking Filter Architecture.

The output values of architecture are uploaded into MATLAB. Finally the output reconstructed image without blocking artifact is displayed. Fig6 shows the input original image & output reconstructed image.

Fig.6 (a):Input Original Image with Blocking Artifacts.
(b): Output Reconstructed Image without Blocking Artifact.

## V. CONCLUSION

In this paper, we discussed about the reduction of blocking artifacts in a image. To study it, we proposed the deblocking filter architecture. The proposed architecture is implemented in Verilog HDL. The experimental results show the area utilization & memory usage. We also demonstrated the results in an image with & without blocking artifacts. The proposed architecture clock frequency is 214MHZ & time delay is 4.663ns.

## REFERENCES

[1]. Mihir Mody, "Understanding in-loop filtering in the HEVC standard," EDN Network June 21,2013.
[2]. C.karthikeyan&Dr.Rangachar,"Impleme-ntation of Deblocking filter algorithm using reconfigurable architecture," International journal of application, volume2, Issue 12, December 2013.
[3]. Tsung-Hsi Weng, Chung-ping chung , "Exploiting fine-grain parallelism in the H.264 deblocking filter by operating reordering," IEEE ,Vol.37,October 2013.
[4]. AlexasandroC.Bonatto,HenriqueA.Klein,Marcelo Negreiros,Andre B.Soares,Leticia V.Guimaraes & Altamiro A.susin, "Hardware Decoding Architecture for H.264/AVC digital video standard 2013.
[5]. G. J. Sullivan, J.R. Ohm, W.J. Han & T. Wiegand, "Overview of the high efficiency video coding standard," IEEE Transactions on Circuits & Systems for video Technology, Vol. 22,2012.
[6]. Andrey Norkin, Gisle Bjontegaard, Arild Fuldseth,Matthias Narroscjke,Masaru Ikeda, Kenneth Anderson, Minhua Zhou, and Geert Van der Auwera, "HEVC Deblocking Filter," IEEE Transaction on circuits & systems for video Technology, vol.22,No.12, December 2012.
[7]. Kharisha, Rakan, "Bit-rate Reconfigurable Architecture for H.264/avc Deblocking filters" (2010). Electronic Theses & Dissertations .Paper 1571.