

# Verilog realization of Diagonal-Down-Left intra prediction for H.264 Video Encoder

Bharathi.S.H\*, Dr.K. Nagabhushana Raju\*\*

\* Associate Professor in Department of Electronics and Communication Engineering at Reva Institute of Technology and Management, Bangalore.

\*\*Associate Professor in Department Instrumentation Sri Krishnadevaraya University, Anantapur Andrapradesh.

## Abstract

*This paper proposes a Verilog realization of Diagonal-Down-Left intra prediction for H.264 Video Encoder. The algorithm is capable of processing video frames with the preferred compression controlled by the user input. The algorithm and architecture for Diagonal-Down-left mode of Intra prediction were designed and developed in Verilog. The complete H.264 Advanced Video Codec for the above said mode of intra prediction was coded in MATLAB first in order to verify the functionality of each block. The same is realized in Verilog. PSNR is computed for the reconstructed picture. It is found that the quality of reconstructed picture in verilog realization is more than 32 dB.*

## 1. Introduction

Video compression systems are used in many commercial products, from consumer electronic devices such as digital camcorders, cellular phones to video teleconferencing systems. These applications make the video compression hardware devices an inevitable part of many commercial products. To improve the performance of the existing applications and to enable the applicability of video compression to new real-time applications, a new international standard for video compression is developed. This new standard, offering significantly better video compression efficiency than previous video compression standards, developed with the collaboration of ITU and ISO standardization organizations. Hence it is called with two different names, H.264 and MPEG4 Part 10 [1, 2, 3, and 4].

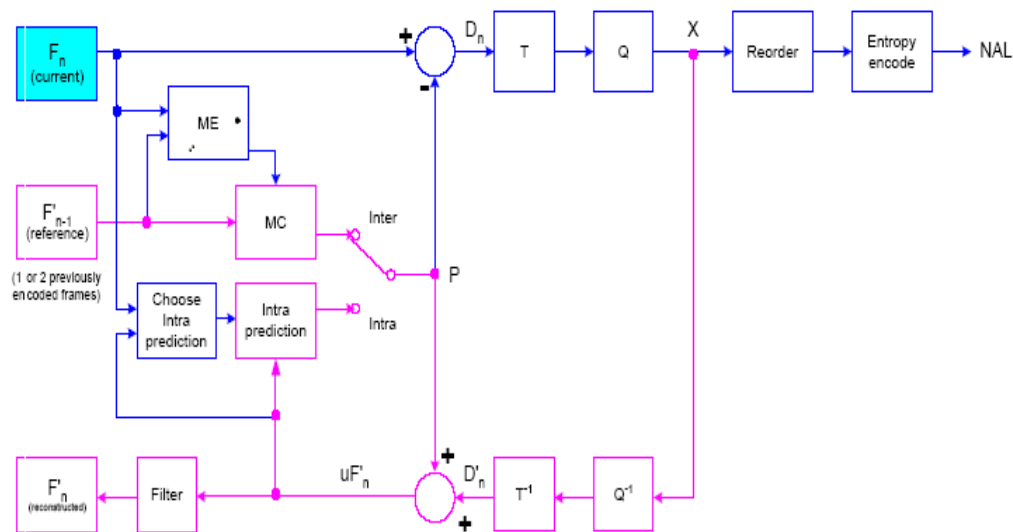
H.264 offers a significant performance improvement over previous video coding standards in terms of better peak signal to noise ratio and visual quality of variable block sizes for motion compensation, multiple reference frames, Integer Transform, Deblocking Filter, and Context Adaptive Variable Length Coding (CAVLC). The Intraprediction technique is one of the most important features that contribute to the success of H.264/AVC [1, 2]. The present work has realized one of the Intra prediction techniques called Diagonal-Down-Left for 4x4 luma components in achieving compression. The implementation conforms to the baseline, main as well as extended profiles since only Intra (I) frames are used. The H.264 Video coding standard supports intraprediction for various block sizes. For coding the luma signal, one 16x16 macro block may be predicted as a whole or as individual 4x4 sub blocks. There are nine modes of Intra-prediction for 4x4 sub-blocks.

In this work, we have realized Diagonal-Down-Left intra prediction technique as stated in the standard for H.264 Video Encoder in Verilog hardware language. The algorithm and architecture

for Diagonal-Down-Left mode of intra prediction has been developed. The algorithm is capable of processing video frames with the preferred compression controlled by the user input.

## 2. H.264/AVC Encoder

The block diagram of H.264/AVC Encoder is presented in Fig. 1 which includes two dataflow paths, a 'forward' path and a 'reconstruction' path. An input frame  $F_n$  is presented for encoding. Every frame is processed in units of a Macroblock (MB) of size 16x16 pixels. Each macro block is encoded in intra or inter mode. In both cases a prediction macro block  $P$  is formed based on a reconstructed frame. In intra mode,  $P$  is formed from samples in the current frame 'n' that has been previously encoded and reconstructed. The prediction  $P$  is subtracted from the current macro block to produce a residual or difference macro block  $D_n$ .



“Figure1. Block Diagram of H.264/AVC Encoder”

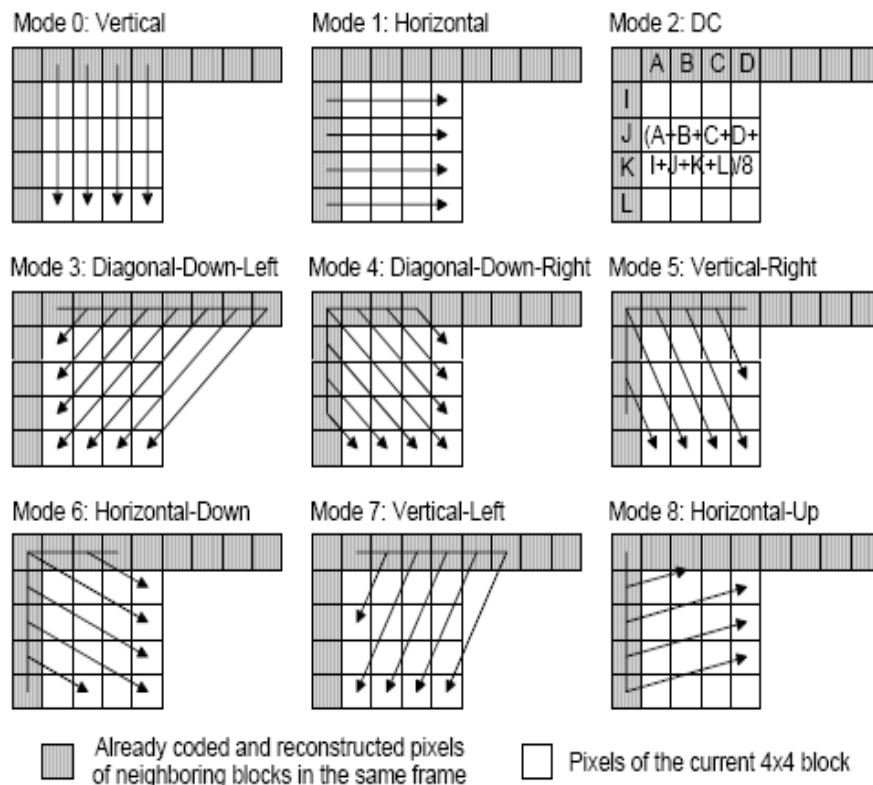
This prediction block  $P$  is transformed and quantized to give a set of quantized transform coefficients  $X$ . These coefficients are reordered and entropy encoded. The compressed bit stream is transmitted over a band-limited serial transmission channel as Network Abstraction Layer (NAL) [5]. The quantized macro block coefficients  $X$  are decoded to reconstruct a frame for encoding of further macro blocks. The coefficients  $X$  are inverse quantized and inverse transformed to produce a difference macro block ' $D_n$ '. The prediction macro block  $P$  is added to ' $D_n$ ' to create a reconstructed macro block after a filter, which improves the quality of the reconstructed picture [6, 7, 8, 9]. The video sequences may also be encoded optionally using motion estimation and compensation to get a further compression of 10 % or more exploiting temporal redundancy.

### 2.1. H.264 Intra Prediction

Intra prediction algorithm predicts the pixels in a MB using the pixels in the available neighbouring blocks. For the luma component of a MB, a 16x16 predicted luma block is formed by performing intra predictions for each 4x4 luma block in the MB. There are nine prediction modes for each 4x4 luma block and four prediction modes for a 16x16 luma block.

M	A	B	C	D	E	F	G	H
I	a	b	c	d				
J	e	f	g	h				
K	i	j	k	l				
L	m	n	o	p				

“Figure2. 4x4 luma block and neighbouring pixels in a sub block”



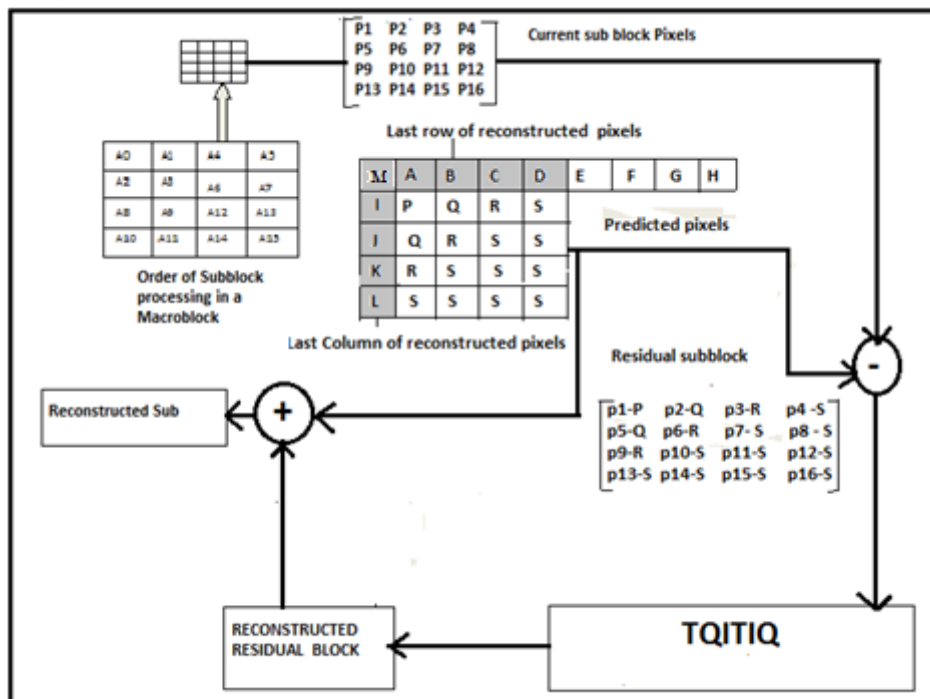
“Figure3. 4x4 luma prediction modes”

The nine 4x4 luma intra prediction modes namely Vertical mode, Horizontal, DC, Diagonal-Down-left, Diagonal-Down-Right, Vertical-Right, Horizontal-Down, Vertical-Left and Horizontal-Up are designed in a directional manner [2,3]. A 4x4 luma block consisting of the pixels ‘a’ to ‘p’ are shown in Fig.2. The pixels A to M belong to the neighbouring blocks and are assumed to be already encoded and reconstructed and are therefore available in the encoder and decoder to generate a prediction for the current MB. Each 4x4 luma prediction mode generates 16 predicted pixel values using some or all of the neighbouring pixels A to M as shown in Fig.3. The arrows indicate the direction of prediction in each mode. The predicted pixels are calculated by a weighted average of the neighbouring pixels A-M for each mode except Vertical and Horizontal modes [1, 2].

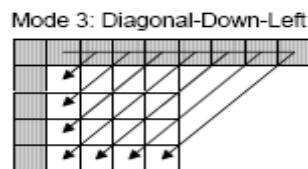
DC mode is always used regardless of the availability of the neighbouring pixels. However, it is adopted based on which neighbouring pixels A-M are available. If pixels E, F, G and H have not yet been encoded and reconstructed, the value of pixel D is copied to these positions and they are marked as available for DC mode. The other prediction modes can only be used if all of the required neighbouring pixels are available [4, 5]. Available 4x4 luma prediction modes for a 4x4 luma block depends on the availability of neighbouring 4x4 luma blocks.

### 3. Architectural Details of Diagonal-Down-Left (Mode 3) Intra prediction module

Intra prediction is done to achieve compression within a frame. The neighboring pixels within the picture frame tend to have similar values in order to exploit the spatial redundancy the prediction is done based on the values of reconstructed pixels of previous sub block. The Diagonal-Down-Left Intra prediction (Mode 3) technique predicts the pixel values to be processed in Diagonal-Down-Left direction. The current pixel prediction can be made on the availability of last row of reconstructed pixels. A, B, C, and D are the neighboring pixel values of last row of reconstructed pixels as shown in fig.4. The direction of prediction and the prediction equations are shown in fig.5.



“Figure 4. Architecture of Diagonal-Down-Left intra prediction”



a) The direction of prediction for Diagonal-Down-Left mode

pred[0,0]	$A + 2B + C + 2 \gg 2$
pred[0,1]	$B + 2C + D + 2 \gg 2$
pred[0,2]	$C + 2D + E + 2 \gg 2$
pred[0,3]	$D + 2E + F + 2 \gg 2$
pred[1,0]	$B + 2C + D + 2 \gg 2$
pred[1,1]	$C + 2D + E + 2 \gg 2$
pred[1,2]	$D + 2E + F + 2 \gg 2$
pred[1,3]	$E + 2F + G + 2 \gg 2$
pred[2,0]	$C + 2D + E + 2 \gg 2$
pred[2,1]	$D + 2E + F + 2 \gg 2$
pred[2,2]	$E + 2F + G + 2 \gg 2$
pred[2,3]	$F + 2G + H + 2 \gg 2$
pred[3,0]	$D + 2E + F + 2 \gg 2$
pred[3,1]	$E + 2F + G + 2 \gg 2$
pred[3,2]	$F + 2G + H + 2 \gg 2$
pred[3,3]	$G + 3H + 2 \gg 2$

**“Figure 5. (b) Prediction equation for 4×4 Diagonal-Down-Left mode intra prediction”.**

After studying the above equations it is observed that there are common parts in the equations and some equations are similar also. Hence for implementation of these equations for prediction the equations are reorganized as shown in Table 1.

**“Table 1.Reorganized prediction equations for 4×4 Diagonal-Down-Left Mode”**

Pixel Position			Prediction Equation
pred[0,0]			$[(A+B)+(B+C)+2] \gg 2$
pred[0,1]	Pred[1,0]		$[(C+D)+(B+C)+2] \gg 2$
pred[0,2]	Pred[1,1]	Pred[2,0]	$[(C+D)+(D+E)+2] \gg 2$
pred[0,3]	Pred[1,2]	Pred[2,1]	$[(E+F)+(D+E)+2] \gg 2$
pred[3,0]			$[(E+F)+(D+E)+2] \gg 2$
pred[1,3]	Pred[2,2]	Pred[3,1]	$[(E+F)+(F+G)+2] \gg 2$
pred[2,3]	Pred[3,2]		$[(G+H)+(F+G)+2] \gg 2$
pred[3,3]			$[(G+H)+(H+H)+2] \gg 2$

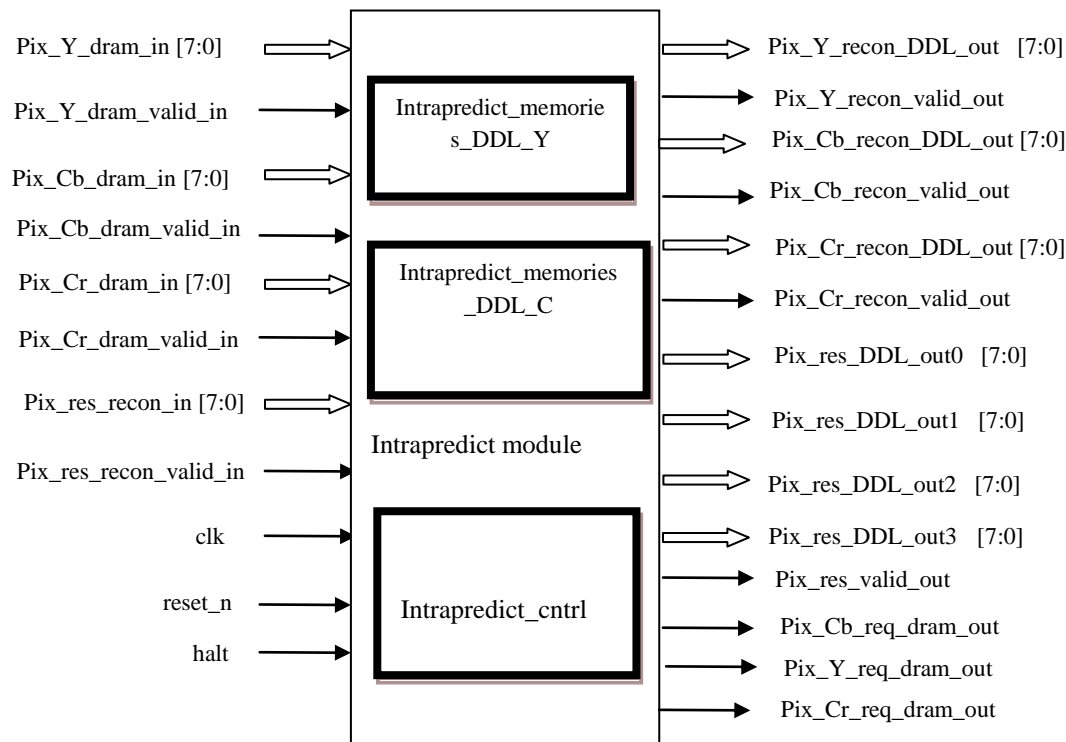
G and H have not yet been encoded and reconstructed, the value of pixel D is copied to these positions and they are marked as available for DC mode. The other prediction modes can only be used if all of the required neighbouring pixels are available [4, 5].

Fig.4 shows the architecture of Diagonal-Down-Left intra prediction. The prediction technique exploits only the last row of previous reconstructed pixel values of sub block namely A, B, C, D as indicated in fig.4. The part of equation like (A+B), (B+C), (C+D) and (D+D) are computed and stored as intermediate values P1, P2, P3 and P4. P, Q, R, S are the predicted pixels obtained as per the equations explained. The picture is divided in to Macro blocks of size 16x16 pixels. A picture is processed macro block by macro block in the order from top to bottom and from left to

right. The order of the macro block process is also shown in fig.4. Further each macro block is divided into 4x4 pixels sub blocks A0 to A15. The order of sub block process is A0, A1, A2, and A3; A4, A5... A15. The pixel values of current sub block are p1, p2, p3.....p16. In the fig.4 A4 is the current sub block required to be processed, A3 is the sub block which has already processed and reconstructed. For processing the first sub block A0 the prediction pixels are taken as 0,0,0,0. Fig.4 also presents the reconstruction of A4 current sub block as an example. The reconstruction is by processing TQIQIT. For processing the integer transform, the residual values of the sub block are taken as the inputs. The residual values are got by taking the pixel-wise difference between the current sub block (A4) and the prediction sub block (A3). The reconstructed residual pixels of the sub block are obtained after processing of TQIQIT. Subsequently these reconstructed residual pixels are added with the corresponding prediction sub block pixels to get the reconstructed sub block A4. For the first sub block A0 of a macro block no pixels are available to generate the predicted blocks. Therefore, the predicted block of such a block has its entire pixel values as “0”, i.e., the block is processed without prediction.

### 3.1. Detailed architecture of Diagonal-Down-left Intra prediction module

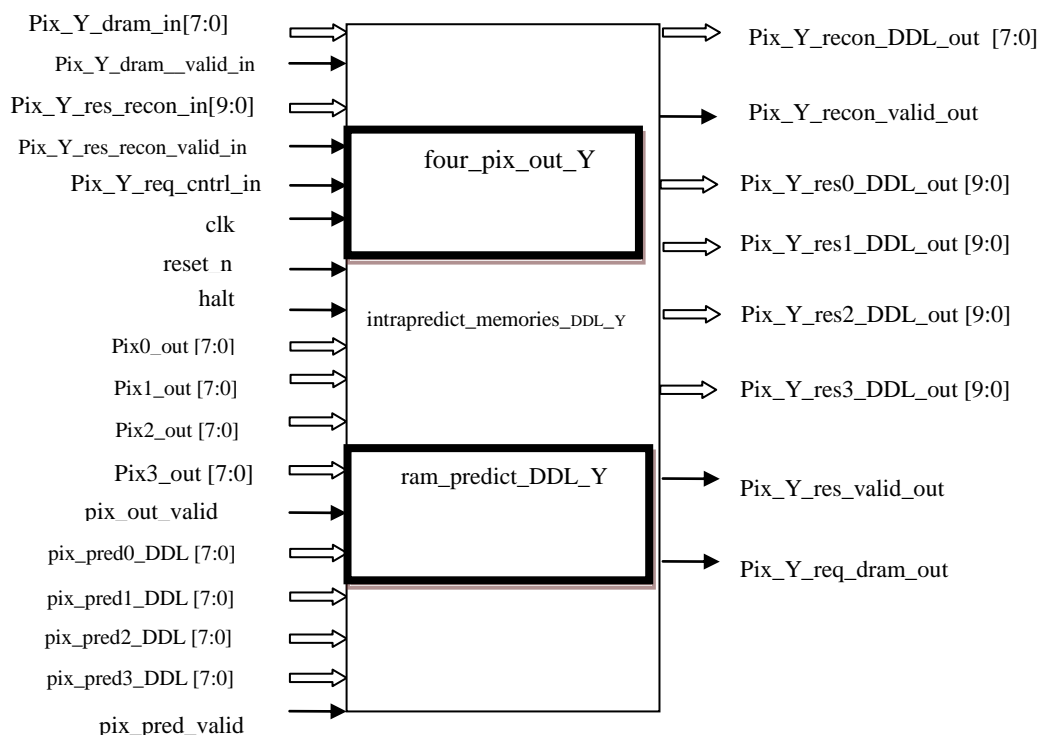
Diagonal-Down-left Intra predict module applies Diagonal-down-left intra prediction (Mode 3) mode of intra prediction to predict the values of pixels to be processed. This method utilizes top row of pixel values of previous reconstructed sub block. The intrapredict module of this mode as coded in this work has three sub modules like Intrapredict\_memories\_DDL\_Y, Intrapredict\_memories\_DDL\_C and Intrapredict\_Control. The detailed architecture of intrapredict module of Diagonal-Down-left intra prediction is presented in fig.6. The inner blocks of the fig.6 shows the three sub modules of Diagonal-Down-left Intra predict module. Intrapredict\_memories\_DDL\_Y module comprises four\_pix\_out\_Y and ram\_predict\_DDL\_Y sub modules for luma components. Further Intrapredict\_memories\_DDL\_C sub module in turn has four\_pix\_out\_C and ram\_predict\_DDL\_C sub modules for chroma components respectively. The detailed architecture of each sub modules is presented in next sections.



“Figure 6. Architecture of Diagonal\_Down\_left intra prediction Module”

“Pix\_Y\_dram\_in” is the luma (Y) pixel values from dual ram which is of 8 bit length. The luma(Y) pixel values are validated by “pix\_Y\_dram\_valid\_in” signal. When this signal is high luma pixel values are considered as valid values. Similarly “pix\_Cb\_dram\_in” and “pix\_Cr\_dram\_in” are the chroma pixel values from dual ram (chroma) whose values are ascertain by “pix\_Cb\_dram\_valid\_in” and “pix\_Cr\_dram\_valid\_in” respectively.

Intrapredict module generates residual pixels. Residual pixel values are the difference between actual pixel values and the predicted pixel values. These residual pixels are applied to Transform, Quantization, Inverse Transform and Inverse Quantization (TQIQIT) module to get residual reconstructed pixel values. “pix\_res\_recon\_in” is the reconstructed pixel values which forms input to intrapredict module. The validity of “pix\_res\_recon\_in” is ascertained by “pix\_res\_recon\_valid\_in” signal with high status. Intrapredict module outputs the luma reconstructed pixel called “pix\_Y\_recon\_DDL\_out”. “pix\_Cb\_recon\_DDL\_out” and “pix\_Cr\_recon\_DDL\_out” are the chroma reconstructed pixel values. The values of these reconstructed pixel values are ascertained by “pix\_Y\_recon\_valid\_out”, “pix\_Cb\_recon\_valid\_out” and “pix\_Cr\_recon\_valid\_out” respectively. “pix\_res\_DDL\_out0”, “pix\_res\_DDL\_out1”, “pix\_res\_DDL\_out2” and “pix\_res\_DDL\_out3” are the residual pixel values obtained by applying Diagonal-Down-left mode of intra prediction. These are the pixel values of one column of the 4×4 residual pixel block. Residual pixel values are the difference between actual pixel values and predicted pixel values. These residual pixel values are applied to TQIQIT module to get “pix\_res\_recon” pixel values. “pix\_Y\_req\_dram\_out” is a handshaking signal. When this signal is high, request is sent to dual ram to output pixels.

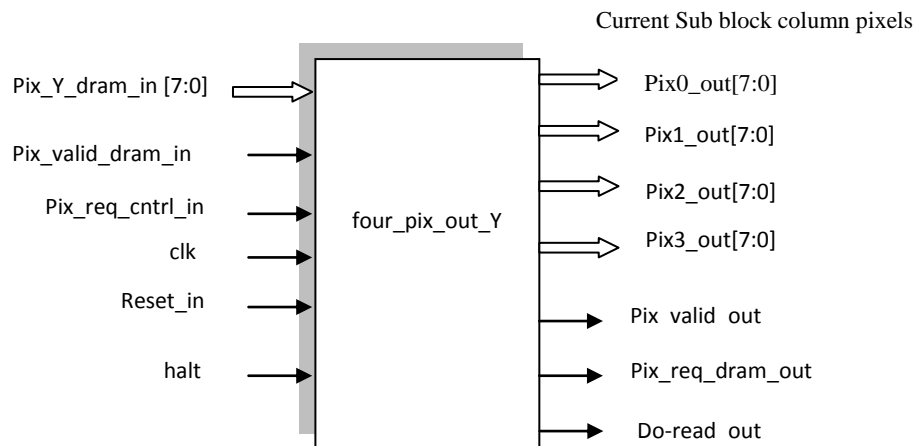


**“Figure 7. Architecture of Intrapredict\_memories\_DDL\_Y Module”**

Intrapredict module of Diagonal-Down-Left Intraprediction module is described by three sub modules intrapredict\_memories\_DDL\_Y, intrapredict\_memories\_DDL\_C and intrapredict\_cntrl. The following section explains in detail the sub module of intrapredict\_memories\_DDL\_Y sub module.

Intrapredict\_memories\_DDL\_Y module includes two rams. One ram for storing 4×4 block of the original image and the other ram for storing the reconstructed pixel values. The reconstructed pixel values are needed to generate the predicted sub block. These predicted values are saved to reconstruct the pixel values. Fig.7 presents the architecture of intrapredict\_memories\_DDL\_Y in depth. “pix\_Y\_dram\_in” is luma pixel values from dual ram. Whose values are ascertained by the signal “pix\_Y\_dram\_valid\_in”, “pix\_Y\_res\_recon\_in” is the input signal to intrapredict\_memories\_DDL\_Y sub block which is from TQIQIT module. TQIQIT module applies Transformation Quantization Inverse Quantization and Inverse Transformation to residual pixels to reconstruct the residual sub block. The validity of “pix\_Y\_res\_recon\_in” pixel values are ascertained by making “pix\_Y\_res\_recon\_valid\_in” high. “pix\_Y\_req\_cntrl\_in” is the request signal sent to dual ram for pixel request.

Intrapredict\_memories\_Y is the sub module of Intrapredict module which generates “pix\_Y\_recon\_DDL\_out”. “pix\_Y\_recon\_DDL\_out” is the luma reconstructed pixel values which can be obtained by adding predicted pixel values to “pix\_Y\_res\_recon\_in” signal. The validity of “pix\_Y\_recon\_DDL\_out” pixel values is validated by “pix\_Y\_recon\_valid\_out”. “pix\_Y\_res0\_DDL\_out”, “pix\_Y\_res1\_DDL\_out”, “pix\_Y\_res2\_DDL\_out” and “pix\_Y\_res3\_DDL\_out” are the luma residual pixel values generated by the difference between actual pixel values to be processed and predicted pixel values generated by ram\_predict\_DDL\_Y module using Diagonal-Down-left mode of intra prediction. These residual pixel values are applied to TQIQIT module. The pixel values are ascertained by making “pix\_Y\_res\_valid\_out” signal. “pix\_Y\_req\_dram\_out” is the hand shaking signal, when this signal is high the request is sent to dual ram to output the pixels.



**“Figure 8. Architecture of four\_pix\_out\_Y module”**

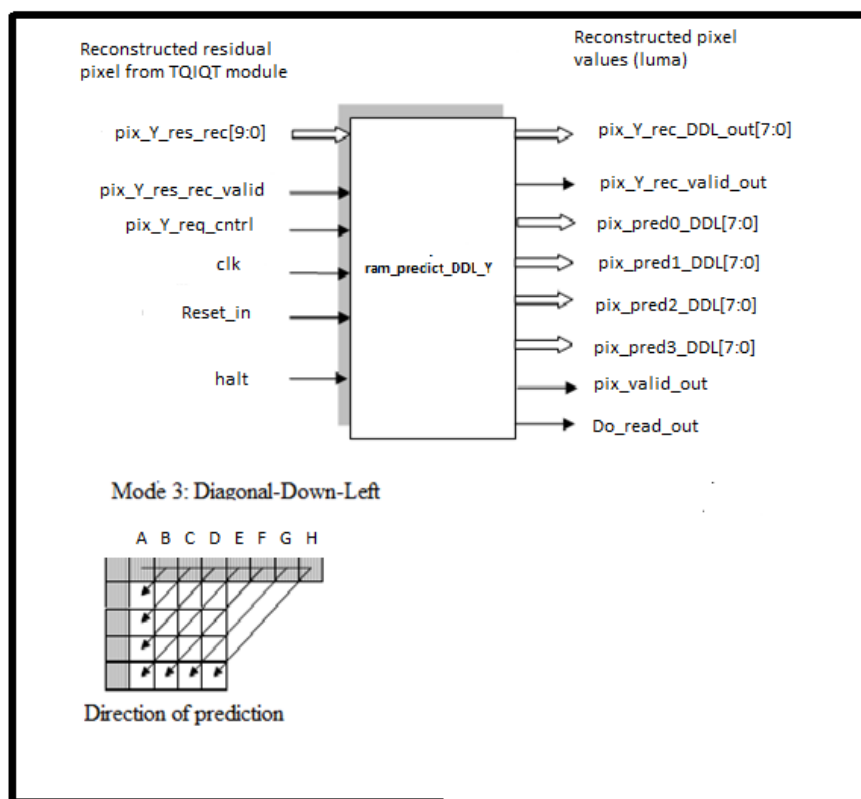
Intrapredict\_memories\_DDL\_Y has again two sub modules “four\_pix\_out\_Y” and “ram\_predict\_DDL\_Y”. As intrapredict\_memories\_DDL\_Y module generates the residual pixel values and luma reconstructed pixels, it is required to build two rams in this module. The detailed architecture of “four\_pix\_out\_Y” and “ram\_predict\_DDL\_Y” are presented below. Fig.8 presents the architecture of four\_pix\_out\_Y module.



The current sub block pixel values from the “dual RAM” module is input to the module “four\_pix\_out\_y” using the data bus “pix\_y\_dram\_in [7:0]”. Its validity is signaled by simultaneously asserting the signal “pix\_valid\_dram\_in”. The “four\_pix\_out\_y” module outputs current pixel values, “pix0\_out”, to “pix3\_out”. The values of these pixels are ascertained by “pix\_valid\_out” signal. “pix\_Y\_req\_dram\_out” is the hand shaking signal, when this signal is high the request is sent to dual ram to output the pixels.

### 3.2 Architecture of ram\_predict\_DDL\_Y Module

The module as shown in fig. 9 called “ram\_predict\_DDL\_Y”, which outputs predicted pixel values as per the direction of prediction as shown in fig.9. It is explained in the earlier section, that the Diagonal-Down-Left mode of intra prediction is achievable if the top neighbouring pixels A, B, C, and D of reconstructed sub block are available. The predicted pixels “pix\_pred0\_DDL” to “pix\_pred3\_DDL” are computed using prediction equations as explained in the algorithm.



**“Figure 9 Architecture of ram\_predict\_DDL\_Y Module”**

Table 2 describes the predicted pixel assignment in Diagonal-Down-Left intra prediction. Where [0,0], [0,1],[0,2].....[3,3] are the positions of pixels in 4x4 luma matrix. After careful analysis of the equations used in 4x4 luma prediction mode of Diagonal-Down-Left mode of intra prediction, it is observed that some part of the equation like (A+B), (B+C), (C+D), (D+E), (E+F), (F+G), (G+H) and (H+H) are common in the equation. Where A, B, C, D, E, F, G, H are the top neighbouring pixels of reconstructed pixels. G and H have not yet been encoded and reconstructed, the value of pixel D is copied to these positions and they are marked as available for DC mode [6]. Therefore the present architecture first calculates

the results of the common parts in the equations and stores them in temporary registers P1, P2, P3 and P4. The ram\_predict\_DDL\_Y module computes the predicted values and sends output at every clock cycle.

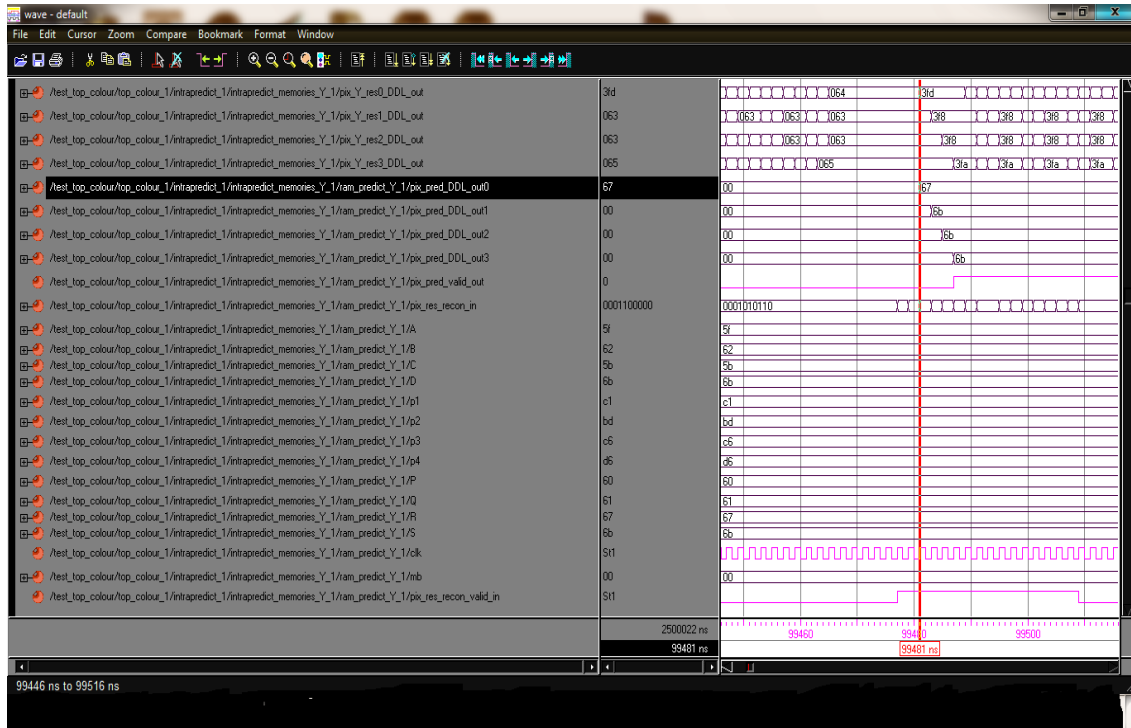
**Table 2. Predicted pixel assignment in Diagonal-Down-Left intra prediction**

Pixel position			
pred[0,0]			$[(A+B)+(B+C)+2] \gg 2 = \mathbf{P}$
pred[0,1]	Pred[1,0]		$[(C+D)+(B+C)+2] \gg 2 = \mathbf{Q}$
pred[0,2]	Pred[1,1]	Pred[2,0]	$[(C+D)+(D+E)+2] \gg 2 = \mathbf{R}$
pred[0,3]	Pred[1,2]	Pred[2,1]	$[(E+F)+(D+E)+2] \gg 2 = \mathbf{S}$
pred[3,0]			$[(E+F)+(D+E)+2] \gg 2 = \mathbf{S}$
pred[1,3]	Pred[2,2]	Pred[3,1]	$[(E+F)+(F+G)+2] \gg 2 = \mathbf{S}$
pred[2,3]	Pred[3,2]		$[(G+H)+(F+G)+2] \gg 2 = \mathbf{S}$
pred[3,3]			$[(G+H)+(H+H)+2] \gg 2 = \mathbf{S}$

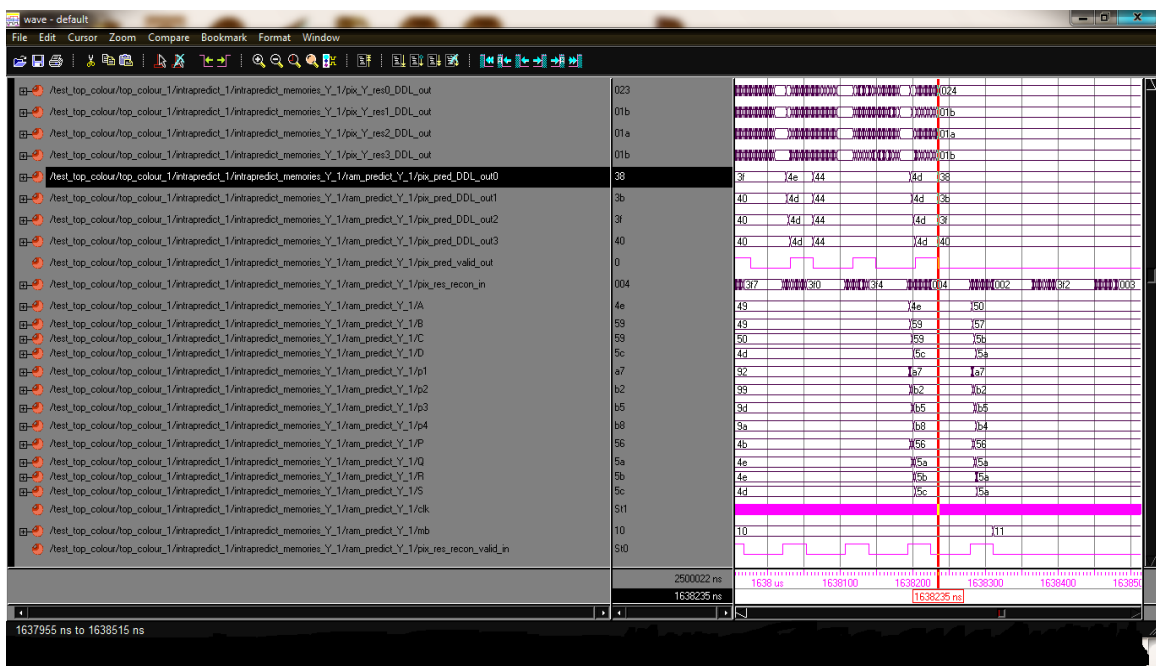
The TQIQIT module computes the residual values to get reconstructed signal for luma component called “pix\_Y\_res\_rec” these values are added with the predicted pixel values “pix\_pred0\_DDL” to “pix\_pred3\_DDL” in the “ram\_predict\_DDL\_Y” module. “pix\_Y\_rec\_DDL\_out [7:0]” is the reconstructed value of pixel. The last row pixel values of this reconstructed sub block are also output as “pix\_pred0\_DDL” to “pix\_pred3\_DDL” with “pix\_pred\_valid” as the valid signal.

#### 4. Simulation Results of Diagonal-Down-Left Intra prediction

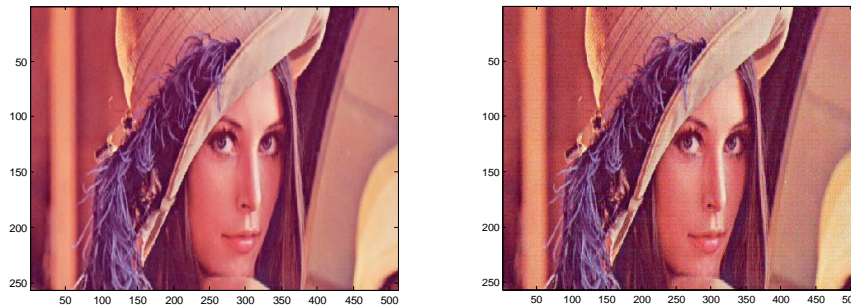
The H.264 video codec with Diagonal-Down-Left intra prediction was first implemented in Matlab in order to verify the functionality of various modules of codec and also to estimate the quality of the reconstructed image. The quality of the reconstructed image is tested various quantization steps. After confirming the results of Matlab, the Diagonal-Down-Left intra prediction module is implemented in verilog and simulated using Modelsim for Quantization step of 8. The verilog module generates reconstructed Y component Cb and Cr components as text files. C++ program is written to convert the text files to “tif” format. Matlab program is written to reconstruct the image, which also computes the quality of reconstructed image. Fig.10 shows the waveforms for Diagonal-Down-Left intra prediction. “pix\_pred\_DDL\_out0” to “pix\_pred\_DDL\_out3” are the predicted pixels obtained from ram\_predict\_Y module. “pix\_Y\_res0\_DDL\_out”, “pix\_Y\_res1\_DDL\_out”, “pix\_Y\_res2\_DDL\_out” and “pix\_Y\_res3\_DDL\_out” are the luma residual pixel values generated by the difference of actual pixels “pix0\_out”, to “pix3\_out” and predicted pixels. The first prediction starts at 99481ns. The intra prediction process continues up to 1638235ns as shown in fig.11. The original and the reconstructed pictures of Lena are shown in “Fig.12” as an example. The Matlab program computes the quality of reconstructed picture. The PSNR obtained for Lena image for Verilog encoder employing Diagonal-Down-Left intra prediction is 37.3 dB.



“Figure 10. Waveforms for Diagonal-Down-Left Intra prediction: First pixel residual Value after prediction”



“Figure 11. Waveforms for Diagonal-Down-Left Intra prediction: End of intra prediction”



(a)

(b)

**“Figure 12 Simulation Results Diagonal-Down-Left Intra prediction for H.264 Video Encoder”.**

(a)Original Lena image (512x256 pixels)

(b) Reconstructed Lena Image using Verilog, PSNR: 37.3 dB

## 4. Conclusion

The architectural design of H.264 Video encoder using Diagonal-Down-Left intra prediction was presented. The intra prediction Module is coded in verilog and integrated with other functional module such as Transform and Quantization. The complete video codec was also realized using Matlab for validating the verilog design. The results show that the reconstructed picture obtained after applying Diagonal-Down-Left Intra prediction is indistinguishable from the original.

## References

- [1] ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG-4) AVC, “Advanced Video Coding for Generic Audiovisual Services”, Version 3: 2005.
- [2] Joint Video Team, Draft ITU-T Recommendation and Final Draft International Standard of joint video specifications, ITU-T Recommendation H.264 and ISO/IEC 14496-10 AVC, March 2005.
- [3] Iain E. G. Richardson, “H.264 and Video Compression”, John Wiley and Sons, 2003.
- [4] Feng Pan, Xiao Lin, Susanto Rahardja, Keng Pang Lim, Z. G. Li, Dajun Wu, Si Wu, “Fast mode decision algorithm for intraprediction in H.264/AVC Video coding”, Circuits and Systems for Video Technology, IEEE Transactions on Volume 15, July 2005.
- [5] S. Kwon, A. Tamhankar, K. R. Rao, “Overview of H. 264/MPEG-4 Part 10”, IEEE Transactions on Circuits and Systems for Video Technology, 2003.
- [6] Thomas Weigand, Gary J. Sullivan Gisle Bjonte gaard, and Ajay Luthra, “Overview of the H.264/AVC Video Coding Standard”. IEEE Transactions on Circuits and Systems for Video Technology, July 2003.
- [7] Gulistan Raja, Sadiquallah Khan, Muhammad Javed Mirza, “VLSI architecture and implementation of H.264 integer transform,” The IEEE 2005 Workshop on Signal Processing Systems (SIPS’05), 2005.

- [8] N. Keshaveni, S. Ramachandran, K.S. Gurumurthy, "Design and Implementation of Integer Transform and Quantization Processor for H.264 Encoder on FPGA", International Conference on Advances in Computing, Control and Telecommunication Technologies, December 2009.
- [8] N. Keshaveni, S. Ramachandran, K.S. Gurumurthy, "Implementation of Context Adaptive Variable Length Coder for H.264 Video Encoder", International Journal of Recent Trends in Engineering [ISSN: 1797-9617] by the Academy Publishers, Finland, Jan. 2010.