OPEN ACCESS
International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-4, April 2019

Synthesis And Implementation Of Transform Coding Through C Based Vivado Hls

P.Srikanth Reddy, M.Sridhar, Y.Viswanadh

Abstract: ***This paper presents an approach for hardware implementation of 8X8 DCT(discrete cosine transform) and IDCT(inverse discrete cosine transform) through HLS.DCT is a frequently used algorithm for image compression in HEVC,XVID CODECS.DCT/IDCT are generally implemented through HDL approach, but the implementation through HDL is a time taking process and has higher computational complexity. In this paper an 8X8 DCT&IDCT are implemented through HLS along with HDL. HLS is a new emerging methodology which is utilized when there is a requirement for faster design and implementation of hardware modules .In this paper an implementation of 8X8 DCT & IDCT through CHEN-WANG algorithm in HLS is also demonstrated which is used in XVID CODECS. The results obtained are compared with HDL, which shows that HLS provides same result as HDL approach .However, HLS compared with HDL has limitation in terms of resource allocation but provides easier implementation in terms of design, verification time and computational complexity***

Index Terms: ***HLS(high level synthesis), DCT(Discrete Cosine Transform), HEVC(High Efficient Video Coding).***

I. INTRODUCTION

The amount of data coming out the back of a HD camera is in very massive range, reduction in file size allows more images to be stored in a given amount of disk or memory space and also the time taking process to be sent over the internet or downloaded from web pages will be reduced. Image compression is the technique that reduces the size of image file by effecting up to a considerable range which is not easily noticed. HEVC was developed with the goal of providing twice the compression efficiency of the previous standard H.264/AVC.HEVC[7] is developed by the joint collaborative team on video coding (JCT-VC), and the first version of the standard was completed, approved and published in 2013. The second version was completed and approved in 2014 and was published in early 2015. HEVC (H.265) enables video to be compressed to a file that is about half the size (or half the bit rate) of H.264.when compressed to the same file size or bit rate as H.264 ,H.265 delivers significantly better visual quantity. DCT is preferred compression technique for HEVC[1].IDCT is used for data retrieval at the receivers end, for the image reconstruction. Discrete Cosine Transform (DCT) is a lossy image compression algorithm. DCT can be applied in the form of matrix form on the image, which is also taken in the form of matrix. HEVC deals with 4x4, 8x8, 16x16, 32x32 DCT which provides an efficient compression factor also involves great computational complexity. DCT/IDCT Hardware modules are generally designed through HDL like Verilog, But it is a difficult and time taking procedure as it requires conversion of algorithm to a standard values. For example the dct formula is converted from algorithm level to floating point values and then to fixed point values to implement in Verilog. HLS provides an optimal solution to this problem as it allows to implement by the formula itself, no intermediate conversions are required. HLS is an approach of utilizing well-known software programming languages such as C and C++ to describe the designs at behavioral level and automatically converting the HDL from it. This way, the code is more readable, design and verification times are shorter, and the design reusability is better compared with handwritten HDL approach[2].

In this paper 8x8 DCT &IDCT are implemented through traditional HDL and new HLS[8] methodologies . A comparison between HDL and HLS approach in terms of outputs and resources is shown. CHEN WANG 8x8 DCT & IDCT is implemented in HLS and its results are compared with traditional dct and idct. Chen Wang algorithm has significant importance in XVID codec in terms of compression factor as per MPEG standards.

XVID: It is a image or video decoder and encoder library providing the best compression efficiency at a better quantization factor. Besides the core library, the codec project also covers integrations into various media frameworks like VfW, DirectShow or GStreamer.Xvid (formerly "XviD") is a video codec libraryfollowing the MPEG-4 video coding standard, specifically MPEG-4 part 2 Advanced simple Profile(ASP). It uses ASP features such as b-frames, global and quarter pixel motion compensation, lumi masking, trellis quantization, and H.263, MPEG and custom quantization matrices. The designed modules in HLS are converted to HDL, and the modules in both approaches are simulated and synthesized. The results obtained are same in HLS and HDL approaches ,however HLS deals with higher resource utilization. The rest of the paper is organized as follows: Section II describes about the implementation of conventional DCT & IDCT in Verilog. Section III deals with implementation of same algorithm in HLS along with the implementation of chenwang DCT &IDCT. Section IV discusses about results and resources followed by future scope in section V.

Manuscript published on 30 April 2019.
***** Correspondence Author (s)
P.Srikanth Reddy*, E.C.E, KLEF(Deemed to be Univrsity), Guntur, India
M.Sridhar,V.L.S.I,/KLEF(Deemed to be Univrsity), Guntur, India
Y.Viswanadh V.L.S.I,KLEF(Deemed to be Univrsity), Guntur, India

II. METHODOLOGY

Existing or general approach for hardware implementation of DCT/IDCT is through Verilog HDL. DCT is similar to FFT, the following process is an overview

1. The image is divided into 8 by 8 blocks of pixels.
2. Working from top to bottom, left to right the dct should be applied.
3. Quantization is done on each block. (here actual compression occurs)
4. The desired image is reconstructed through IDCT (inverse dct) ,i.e.., the decompression process.
5.Thus the redundant data is compressed and after reconstruction we will image of reduced size with less distortions.

DCT&IDCT algorithm[2] provides a formula which is the basis for hardware implementation. The values required for hardware implementation is obtained from the formula shown below

1–Dimension DCT equation :-

Forward DCT of a sequence u(n), is defined as

$$\vartheta(k) = \alpha(k) \sum_{n=0}^{N-1} u(n) \cos\frac{\pi(2n+1)k}{2N} \quad 0 \le k \le N-1$$

Where $\alpha(0) = \frac{1}{\sqrt{N}} \& \alpha(k) = \frac{2}{\sqrt{N}}$

Inverse DCT (IDCT) is defined as

$$u(k) = \sum_{n=0}^{N-1} \alpha(k)\vartheta(n) \cos\frac{\pi(2n+1)k}{2N} \quad 0 \le n \le N-1$$

2- Dimension DCT Equation :-

For a N X N Image u(m,n), Forward DCT is defined as

$$\vartheta(k,l) = \alpha(k)\alpha(l) \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} u(n) \cos\frac{\pi(2n+1)k}{2N} \cos\frac{\pi(2n+1)l}{2N} \quad 0 \le k \le N-1, \quad 1 \le k \le N-1$$

Where $\alpha(0) = \frac{1}{\sqrt{N}} \& \alpha(k) = \frac{2}{\sqrt{N}}$

Inverse DCT (IDCT) is

$$u(m,n) = u(k) = \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} \alpha(k)\alpha(l)\vartheta(k,l) \cos\frac{\pi(2n+1)k}{2N} \cos\frac{\pi(2n+1)}{2N} \quad 0 \le m,n \le N-1$$

Coefficient matrix

N*N cosine transform matrix is given by C = {c(k,n)}

$$C(k,n) = \begin{cases} \dfrac{1}{\sqrt{N}}, & k = 0, 0 \le n \le N-1 \\ \sqrt{\dfrac{2}{N}} \cos\dfrac{\pi(2n+1)k}{2N}, & 1 \le k \le N-1 \\ & 0 \le n \le N-1 \end{cases}$$

For N=4

$$C = \begin{bmatrix} \frac{1}{\sqrt{4}} & \frac{1}{\sqrt{4}} & \frac{1}{\sqrt{4}} & \frac{1}{\sqrt{4}} \\ \sqrt{\frac{2}{4}}\cos\frac{\pi}{8} & \sqrt{\frac{2}{4}}\cos\frac{3\pi}{8} & \sqrt{\frac{2}{4}}\cos\frac{5\pi}{8} & \sqrt{\frac{2}{4}}\cos\frac{7\pi}{8} \\ \sqrt{\frac{2}{4}}\cos\frac{\pi}{4} & \sqrt{\frac{2}{4}}\cos\frac{3\pi}{4} & \sqrt{\frac{2}{4}}\cos\frac{5\pi}{4} & \sqrt{\frac{2}{4}}\cos\frac{7\pi}{4} \\ \sqrt{\frac{2}{4}}\cos\frac{3\pi}{8} & \sqrt{\frac{2}{4}}\cos\frac{9\pi}{8} & \sqrt{\frac{2}{4}}\cos\frac{15\pi}{8} & \sqrt{\frac{2}{4}}\cos\frac{21\pi}{8} \end{bmatrix}$$

$$C = \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 \\ 0.65 & 0.27 & -0.27 & -0.65 \\ 0.5 & -0.5 & -0.5 & 0.5 \\ 0.27 & -0.65 & 0.65 & -0.27 \end{bmatrix}$$

$$C^t = \begin{bmatrix} 0.5 & 0.65 & 0.5 & 0.27 \\ 0.5 & 0.27 & -0.5 & -0.65 \\ 0.5 & -0.27 & -0.5 & 0.65 \\ 0.5 & -0.65 & 0.5 & -0.27 \end{bmatrix}$$

Likewise for 8 x 8 matrix the co efficient matrix are

For n=8

$$C = \begin{bmatrix} 0.353553 & 0.353553 & 0.353553 & 0.353553 & 0.353553 & 0.353553 & 0.353553 & 0.353553 \\ 0.490393 & 0.415735 & 0.277785 & 0.097545 & -0.097545 & -0.277785 & -0.415735 & -0.490393 \\ 0.461940 & 0.191342 & -0.191342 & -0.461940 & -0.461940 & -0.191342 & 0.191342 & 0.461940 \\ 0.415735 & -0.097545 & -0.490393 & -0.277785 & 0.277785 & 0.490393 & 0.097545 & -0.415735 \\ 0.353553 & -0.353553 & -0.353553 & 0.353553 & 0.353553 & -0.353553 & -0.353553 & 0.353553 \\ 0.277785 & -0.490393 & 0.097545 & 0.415735 & -0.415735 & -0.097545 & 0.490393 & -0.277785 \\ 0.191342 & -0.461940 & 0.461940 & -0.191342 & -0.191342 & 0.461940 & -0.461940 & 0.191342 \\ 0.097545 & -0.277785 & 0.415735 & -0.490393 & 0.490393 & -0.415735 & 0.277785 & -0.097545 \end{bmatrix}$$

In the matrix form FDCT= $C * U * C^T$

IDCT =$C^T * U * C$therefore $C^T = C$ Transpose

HDL way of approach involves converting DCT formula to floating point coefficient matrix and to fixed point form. For a 8x8 data DCT is performed in Verilog which compresses the data, the compressed data is reconstructed using 8x8 idct through which original values are obtained with little difference[3].

8x8 coefficients with 2^{15} fixed point converted values are taken for DCT and IDCT are

$$C = \begin{bmatrix} 64 & 64 & 64 & 64 & 64 & 64 & 64 & 64 \\ 89 & 75 & 50 & 18 & -18 & -50 & -75 & -89 \\ 83 & 36 & -36 & -83 & -83 & -36 & 36 & 83 \\ 75 & -18 & -89 & -50 & 50 & 89 & 18 & -75 \\ 64 & -64 & -64 & 64 & 64 & -64 & -64 & 64 \\ 50 & -89 & 18 & 75 & -75 & -18 & 89 & -50 \\ 36 & -83 & 83 & -36 & -36 & 83 & -83 & 36 \\ 18 & -50 & 75 & -89 & 89 & -75 & 50 & -18 \end{bmatrix}$$

ARCHITECTURE OF DCT &IDCT for HDL approach is shown in fig

DCT

For DCT module stage I multiplier contains image values and DCT coefficients as inputs. Stage II multiplier has intermediate outputs and transposed DCT coefficients as inputs, which produces DCT values for the input image values

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-4, April 2019

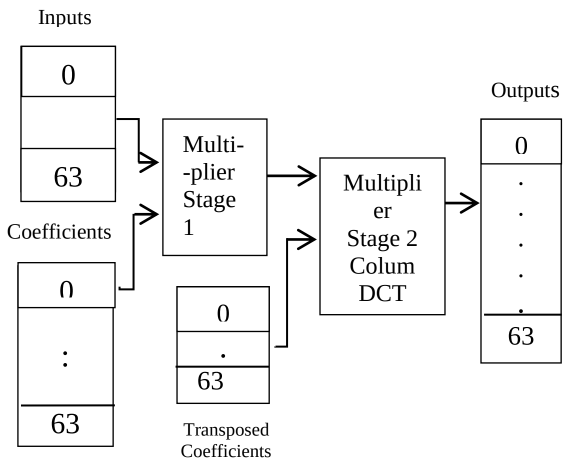


Figure 1:- Block diagram of DCT

IDCT

It is similar to DCT except the coefficient matrix ,its transpose position changes with primer inputs as DCT outputs

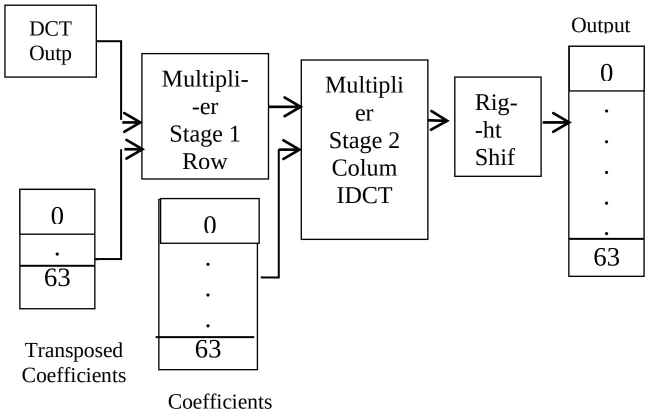


Figure 2:- Block diagram of IDCT

III. PROPOSED METHODOLOGY

Instead of dealing with traditional HDL approach which deals with complexity in design and verification, new trend of HLS approach is utilized[4]. 8x8 DCT & IDCT are implemented in HLS through traditional DCT and IDCT algorithm along with CHENWANG algorithm.

In HLS through C language forward DCT formula is directly applied on input data for compression and the original data is reconstructed through IDCT formula[5].

HLS approach of implementation is shown in the form of flow graph shown in figure 3.

HIGH LEVEL SNTHESIS DESIGN FLOW:-

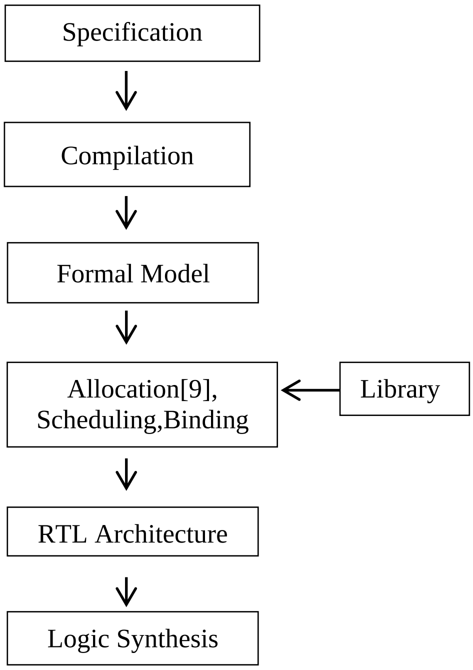


Figure 3 :Flow Chart of HLS

The output data obtained through DCT &IDCT is taken as reference for verifying the outputs of CHEN WANG[6] algorithm. Although resources utilization is more through HLS approach but as mentioned earlier design time is simple and implementation and verification approach is easy. CHEN WANG 2D 8x8 DCT & IDCT are Implemented in HLS through C LANGUAGE ,through which DCT & IDCT implementation is simple for better compression in XVID CODEC preferred by MPEG.

CHEN WANG 2D DCT/IDCT:-The algorithm is implemented in two stage as horizontal and vertical DCT in HLS through C LANGUAGE.

Chen propose the fast DCT and IDCT algorithm. They proposed a algorithm to factor any N-point DCT with N = 2m, m ≥ 2 into butterfly operations The factorization has a very regular structure and is 6 times faster than the fast Fourier transform based algorithms.. For an 8 point DCT this number is 16 multiplications and 26 additions. The butterfly implementation of 8 point DCT is shown in below figure.

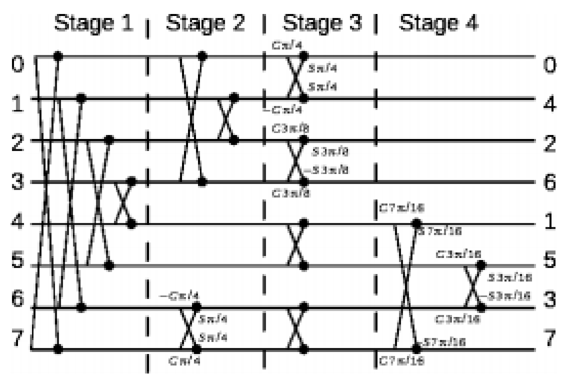


Fig 4.Butterfly Implementation of chenwangdct.

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)

Snipped code of Implementation of WANG DCT/IDCT in HLS in row and column wise is shown below

```
  // horizontal DCT
L_hor: for (i=0; i<DCT_SIZE; i++)
    {
    //wang_dct_1d( loc_xx+i*DCT_SIZE, loc_yy, i, 1);
    #pragma HLS PIPELINE
    wang_dct_1d( xx+i*DCT_SIZE, loc_xx, i, 1);
    }
  //vertical DCT
 L_vert: for (i=0; i<DCT_SIZE; i++)
     {
    //wang_dct_1d( loc_yy+i*DCT_SIZE, loc_xx, i, 0);
```

2D IDCTCHEN WANG :Similarly the IDCT implemented as row wise and column wise as

Row(horizontal) IDCT :-

$$dct[k] = \sum_{l=0}^{7} c[l] * src[l] * \cos\left[\frac{\pi}{8} * \left(k + \frac{1}{2}\right) * l\right]$$

Where:- $c[0] = 128$

$c[1 \ldots .7] = 128\sqrt{2}$

Column (vertical) IDCT:-

$$dct[k] = \sum_{l=0}^{7} c[l] * src[8 * l] * \cos\left[\frac{\pi}{8} * \left(k + \frac{1}{2}\right) * l\right]$$

Where:- $c[0] = 1/1024$

$c[1 \ldots .7] = (1/1024) * \sqrt{2}$

IV. IMPLEMNTATION & RESULTS

Results of 2D 8x8 DCT and IDCT implemented in HLS are as follows

An 8x8 block is taken as input matrix for entire implementation shown below

40	33	33	22	26	40	36	26
43	26	33	22	29	43	22	36
43	40	19	36	36	33	15	26
36	43	26	33	29	29	19	4
33	33	29	26	15	33	26	4
36	33	33	26	22	29	26	12
33	40	29	26	33	12	12	4
26	33	29	22	12	4	8	0

The results obtained by implementing DCT and IDCT in HLS is snniped from the tool and shown as follows

DCT

214	49	-3	20	-10	-1	1	-6
34	-25	11	13	5	-3	15	-6
-6	-4	8	-9	3	-3	5	10
8	-10	4	4	-15	10	6	6
-13	5	-1	-2	-15	9	-5	-1
5	9	-8	3	4	-7	-14	2
2	-2	3	-1	1	3	-3	-4
-1	1	0	2	3	-2	-4	-2

IDCT

40	33	33	22	26	40	36	26
43	26	33	22	29	43	22	36
43	40	19	36	36	33	15	26
36	43	26	33	29	29	19	4
33	33	29	26	15	33	26	4
36	33	33	26	22	29	26	12
33	40	29	26	33	12	12	4
26	33	29	22	12	4	8	0

The results obtained for CHEN WANG DCT are

215	49	-3	20	-10	-1	1	-6
34	-25	11	13	5	-3	15	-6
-6	-4	8	-9	3	-3	5	10
8	-10	4	4	-15	10	6	6
-12	5	-1	-1	-15	9	-5	-1
5	9	-8	3	4	-7	-14	2
2	-2	3	-1	1	3	-3	-4
-1	1	0	2	3	-2	-4	-3

CHEN WANG is taken as DESIGN UNDER TEST(DUT) and it is compared with reference DCT which is implemented through DCT algorithm

Difference between REF and DUT FDCT data

1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1

IDCT: The results obtained through WANG IDCT are:

40	33	33	22	26	41	36	26
43	26	33	22	29	42	22	36
43	40	19	36	36	33	15	26
36	43	26	33	29	29	19	4
34	33	29	26	15	33	26	4
36	33	33	27	22	29	26	12
33	40	29	26	33	13	12	4
26	33	29	22	12	4	9	1

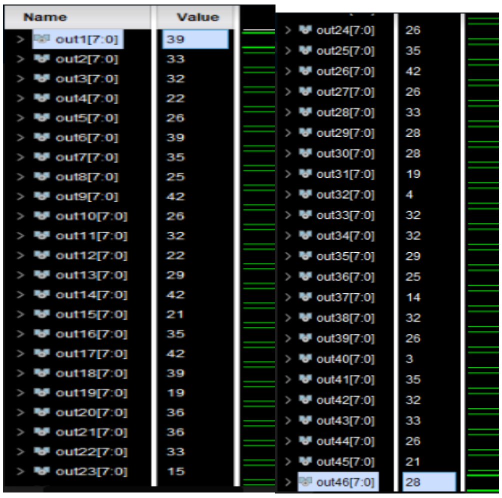
The difference between original input values and WANG IDCT values are:

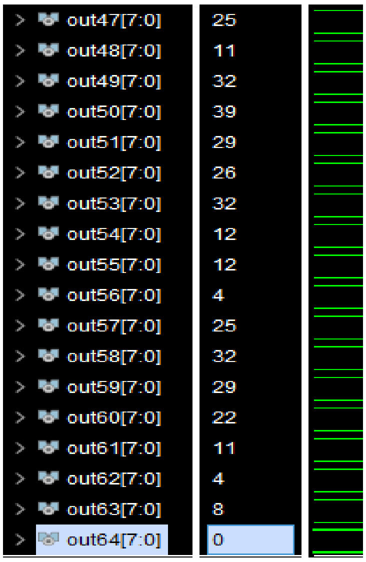
0	0	0	0	0	1	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0
0	0	0	0	0	1	0	0
0	0	0	0	0	0	1	1

HDL outputs:

The original matrix obtained through implementation in VERILOG for same inputs is:

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication (BEIESP)





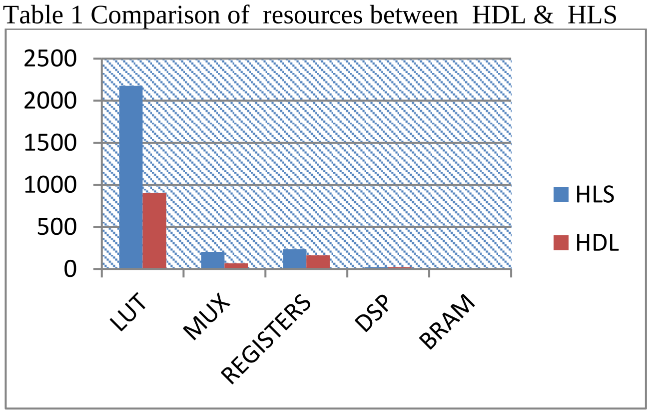
Output graph in vivado HDL

COMPARISION OF HADWWARE :-Both HDL and HLS implemented designs are synthesized and resources obtained are noticed

The resource utilization comparison table for HLS and HDL is shown below

S.NO	NAME	HLS	HDL
1	LUT	2176	900
2	MUX	203	67
3	REGISTERS	233	162
4	DSP	22	22
5	BRAM	1	1

Table 1 Comparison of resources between HDL & HLS



graph 1: comparing the HLS & HDL

The results obtained shows that DCT&IDCT implementation through HDL, HLS and CHEN WANG DCT&IDCT in HLS are same but the resource utilization in HLS is more compared to HDL

V. CONCLUSION AND FUTUREWORK:

Implementation of 8X8 DCT and IDCT in HLS is easier in terms of code design, reusability and verification compared to HDL ,at expense of much hardware utilization. Implementing 8X8 DCT AND IDCT through CHEN WANG algorithm is useful in terms of compression for XVID CODEC compared to traditional DCT &IDCT algorithm as per MPEG standards. HLS implementation provides an optimal solution for software designers to generate hardware modules in short time perriods .HLS has advantage in terms of loop unrolling and easier verification. Further future work can be done for resource optimization in HLS.

REFERENCES

1. High Efficiency Video Coding, document ITU-T Rec. H.265 and ISO/IEC 23008-2 (HEVC), ITU-T and ISO/IEC, Apr.2013.
2. A DCT based algorithm for blocking artefact reduction from dct coded images,2006 IEEE international conference on industrial technology.
3. FPGA Implementation of Secure Image Compression with 2D-DCT using Verilog HDL, 978-1-4799-1356-5/14/2017,IEEE.
4. High-Level Synthesis Hardware Implementation and Verification of HEVC DCT on SoC-FPGA, 978-1-5386- 4266-5 ,2017 IEEE.
5. High Level Synthesis Implementation Of HEVC 2-D DCT/DST On FPGA978-1-5090-4117-6/17 /2017 IEEE
6. Early Determination of Zero-Quantized 8 × 8 DCT Coefficients1051-8215/c 2009 IEEE publishers by Xiangyang Ji, Sam Kwong, Debin Zhao, Hanli Wang, Member, IEEE, C.-C. Jay Kuo
7. A Low Energy HEVC Inverse DCT Hardware.978-1-4799-1412-8/13/$31.00 ©2013 IEEE public by ErcanKalali, ErdemOzcan, OzgunMertYalcinkaya, IlkerHamzaoglu
8. High-Level Synthesis: Past, Present, and Future.0740-7475/09/$26.00 c 2009 IEEE. Published by Grant Martin,Gary Smith.
9. Scheduling and allocation in high level synthesis using stochastic technique.https://doi.org/10.1016/0026-2692(96)00001-8

AUTHORS PROFILE

Mr. P Srikanth Reddy has more than 5 years of industrial experience working on FPGA &Hardware Implementation of video processing codecs in TYNETEE and PATHPARTNER TECHNOLOGIES. He obtained his masters degree in microelectronics and system automation at NORTHUUMBRIA university, newcastle upon tyne U.K .He has completed his bachelors degree inhe is currently working as assistant proffessor in klef deemed to be university.His research activities are related to vlsi design, Verification ,implementation and signal processing applications, machine learning algorithms..

Mr.M Sridhar is currently pursuing his masters degree in vlsi design at klef deemed to be university. He obtained his bachelors degree in electronics & communication engineering at vishnu institute of technology. His working area of interest includes digital vlsi design & hardware implementation, software to hdl code conversions, hls methodologies.

.

MrY Viswanadh is currently pursuing his masters degree in vlsi design at klef deemed to be university.He obtained his bachelors degree in electronics and communication engineering at v.k.r,v.n.b&a.g.k college of engineering. His research activities includes hardware design & implementation of dct,idct modules for fvc&hevccodecs,digitalvlsi modules design and verification through hdlsHis other works includes hardware implementation of various protocols like spi,uart.