



# Implementation of 2D-DCT as an Efficient Accelerator for HEVC Video CODEC

Sumalatha. S, Rajeswari

**Abstract:** Programmable architectures like GPU based embedded system for video and imaging applications are widely used due to their high performance, as they allow flexibility for running customized functions. However these architectures do not allow reconfiguration of the architecture at run time and optimization of the hardware resources. This paper explores the FPGA based architecture suitable for all video CODEC standards used in multimedia applications which is both programmable and reconfigurable. The proposed architecture demonstrates an accelerator to perform two dimensional 8\*8 discrete Cosine Transform (DCT) and Inverse Discrete Cosine Transform (IDCT). The accelerator can be reconfigured to compute higher order two-dimensional DCT/IDCT according to different system requirements and is implemented on Xilinx Zynq evaluation board 7vx485tffg1157-1. The architecture is found to have a high scalability in terms of power and area. The synthesis results reads, 48% improvement in both dynamic and static power consumption, with optimal hardware utilization suitable for high performance video CODECs.

**Keywords:** Accelerator, DCT/IDCT, Micro-architecture, HEVC, Instruction Level Programming (ILP), VLIW, SIMD.

## I. INTRODUCTION

The two main video codec standard organizations ITU-T VCEG (Video Coding Expert Group) and ISO/IEC MPEG (Moving Picture Expert Group) in collaboration brought out the H.262/MPEG-2 Video and H.264/MPEG-4 Advanced Video Coding (AVC) standards. The two standards had a prominent impact and have been widely used in a wide variety of products. Series of efforts have been made to maximize compression capability and improve other characteristics like data loss, robustness, while considering the usage of computational resources to aid additional applications with extended-range, enhanced precision and color format support, scalable video coding, and 3-D/stereo/multi-view video coding. In this regard, a new standard HEVC is made as MPEG-H Part 2 (ISO/IEC 23008-2) and in ITU-T it is H.265.

The new Intra Prediction direction modes and variable size block prediction unit (PU) up to 64X64 were included as an additional features into HEVC to support various high resolution Video [1].

Each video frame is divided into smaller blocks of various sizes 16×16, 32×32 or 64×64 called Coding Tree Unit (CTUs). Normally the CTU includes one Luma and two Chroma Coding Tree Block (CTB) components [1]. The CTU is similar to standard Macroblock in AVC. Processing starts from dividing CTU into multiple Coding Unit (CU) and further CU can be split into four depth sizes of 8×8, 16×16, 32×32 and 64×64 pixels depending on coding Sequence resolution. Depending on the decision modes the coding unit can be divided into Prediction Unit (PU) for intra and inter block prediction with various sizes ranging from 4x4 to 64×64. Finally, these units were sent to Transform and quantization.

In the recent developments of VLSI technology, designing a Micro architecture and Application Specific Instruction Set Processors (ASIP) is emerging as a challenging and exploring domain to meet real time video processing applications. Micro architecture relates to suitable selection of different functional units like ALU (Arithmetic and Logical unit), MAC (Multiply and accumulate), FPU (Floating Point Units) and other signal processing accelerators which are designed using effective instruction sets to reach target cost and performance goals. The continuous evolution of Micro-architecture based processors is due to technology scaling and workload distribution. Micro-architecture designs are based on several factors like Pipelining, In-order/ Out-of-Order Processor execution, Scalar/Superscalar (ILP) – Processor executes single instruction per cycle in case of scalar and multiple instructions per cycle in case of superscalar SIMD (Single Instruction Multiple Data), also called vector processors permits data level parallelism.

Most of the Video codec standards demands efficient processor to perform transformation, Quantization and entropy coding efficiently to increase compression efficiency, speed and reduce bandwidth requirement. In this aspect most of the RISC processors are designed and tailored to do specific task in order to achieve both hardware efficiency and software flexibility. The computational intensive parts of the application are designed by hardware accelerators and control intensive tasks are performed by software programming.

An additional hardware called Accelerator to speed up the system performance can be used if the uni-processor is slow. This accelerator works along with the processor and executes key functions with higher speed than the processor. The possible means of accelerating a programmable core is by exploiting instruction and/or data parallelism with VLIW or SIMD extensions and the other option is to include special functional units like MAC circuits, barrel shifter in the data path of the programmable core to speed up the execution of DSP algorithms.

Revised Manuscript Received on December 30, 2019.

\* Correspondence Author

**Sumalatha. S\***, Department of Electronics and Communication Engineering, Acharya Institute of Technology, Bangalore, India. E-mail: sumalathas@acharya.ac.in

**Rajeswari**, Department of Electronics and Communication Engineering, Acharya Institute of Technology, Bangalore, India. E-mail: rajeswari@acharya.ac.in

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](http://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)



The DCT kernel matrix of size 4, 8, 16 and 32 can be obtained by equation (2) and (3). The 4-point and 8-point DCT kernel matrix given in [12] is represented as

$$T_4 = \begin{bmatrix} 64 & 64 & 64 & 64 \\ 83 & 36 & -36 & -83 \\ 64 & 64 & 64 & 64 \\ 36 & -83 & 83 & -36 \end{bmatrix} \quad (4)$$

and

$$T_8 = \begin{bmatrix} 64 & 64 & 64 & 64 & 64 & 64 & 64 & 64 \\ 89 & 75 & 50 & 18 & -18 & -50 & -75 & -89 \\ 83 & 36 & -36 & -83 & -83 & -36 & 36 & 83 \\ 75 & -18 & -89 & -50 & 50 & 89 & 18 & -75 \\ 64 & -64 & -64 & 64 & 64 & -64 & -64 & 64 \\ 50 & -89 & 18 & 75 & -75 & -18 & 89 & -50 \\ 36 & -83 & 83 & -36 & -36 & 83 & -83 & 36 \\ 18 & -50 & 75 & -89 & 89 & -75 & 50 & -18 \end{bmatrix} \quad (5)$$

These matrices are further modified to perform a direct 2D DCT on video blocks of lengths 4x4 to 32x32, which is discussed in the next section. As many of the hardware implementation involves two 1D transforms and in each 1D transforms having different stages like input adder unit (IAU), shift and add unit (SAU) and output adder unit (OAU)[9] the complexity of the circuit increases. To reduce the number of stages and to perform direct 2D DCT/IDCT a novel method is proposed as given in below paragraph.

### III. PROPOSED DIRECT 2D-DCT ARCHITECTURES

In this section, first the 4 point DCT architecture is proposed with necessary computation equations, later the design of higher order DCT architectures are built using the lower order blocks.

#### A. Four and Eight point 2D DCT Architecture

The 2D DCT matrix for proposed four point DCT is derived from (4) by performing matrix operation  $T^*T^*2$ . The new four point 2D DCT kernel is as given in equation (6).

$$C_4 = \begin{bmatrix} 32768 & 0 & 0 & 0 \\ 0 & 32740 & 0 & 0 \\ 0 & 0 & 32768 & 0 \\ 0 & 0 & 0 & 32740 \end{bmatrix} \quad (6)$$

The proposed 4-point DCT architecture consists of shift and add (SAU) operation, as shown in figure-1. Input pixel matrix X is applied to the four point DCT unit and output 2D DCT Y is obtained. The four point DCT requires four shift operation and one addition operation. Similarly the 8-point DCT kernel is obtained by  $T^*T^*$  of equation (5) and resultant 2D DCT kernel is depicted in equation (7).

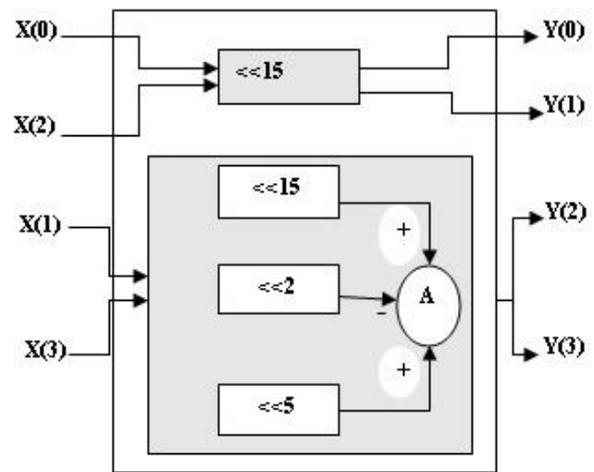


Fig. 1 Four point integer DCT architecture

$$C_8 = \begin{bmatrix} 32768 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 32740 & -50 & 0 & 0 & 50 & 0 & 0 \\ 0 & 0 & 32740 & 0 & 0 & 0 & 0 & 0 \\ 0 & -50 & 0 & 32740 & 0 & 0 & 0 & 50 \\ 0 & 0 & 0 & 0 & 32768 & 0 & 0 & 0 \\ 0 & 50 & 0 & 0 & 0 & 32740 & 0 & 50 \\ 0 & 0 & 0 & 0 & 0 & 0 & 32740 & 0 \\ 0 & 0 & 0 & 50 & 0 & 50 & 0 & 32740 \end{bmatrix} \quad (7)$$

The architecture of eight unit can be built by using the 4-point DCT unit as an (N/2) even transformation unit and the (N/2) odd DCT coefficients are obtained by a new shift and add(SAU) unit as shown in figure (2).

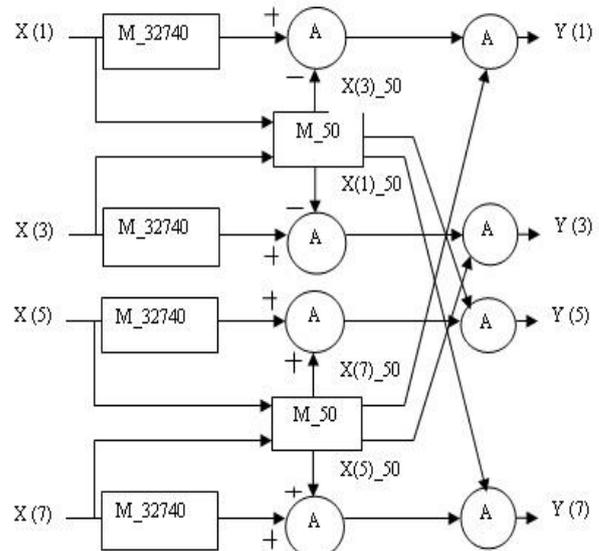


Fig. 2. An 8- point integer DCT architecture (ODD part of SAU).

In 8-point DCT architecture the 4-point architecture is used as a reusable structure for computing even part of the DCT. Hence the output  $y(0)$ ,  $y(2)$ ,  $y(4)$  and  $y(6)$  of 8-point DCT is obtained by architecture shown in figure(1). The odd part is computed by structure shown in figure(2). The M-32740 and M\_50 are shift and add blocks , its internal blocks are shown in figure(3) and (4). The complete structure of 8 point dct architecture using 4-point architecture is shown in figure 5.

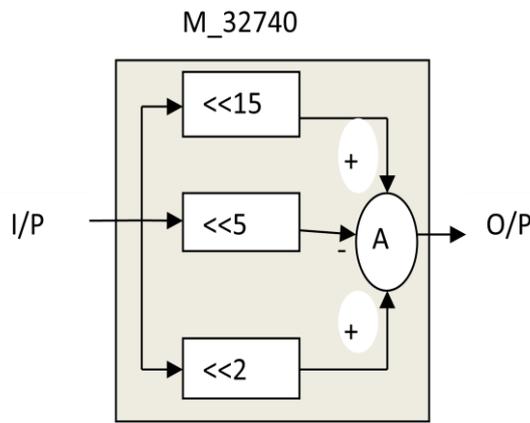


Fig. 3.M\_32740 SAU of 8-point DCT

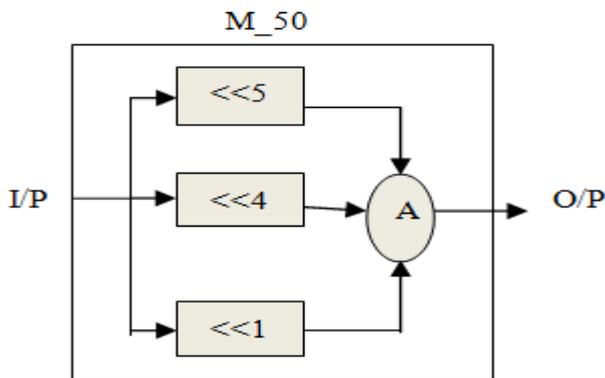


Fig. 4.M\_50 SAU OF 8-point DCT

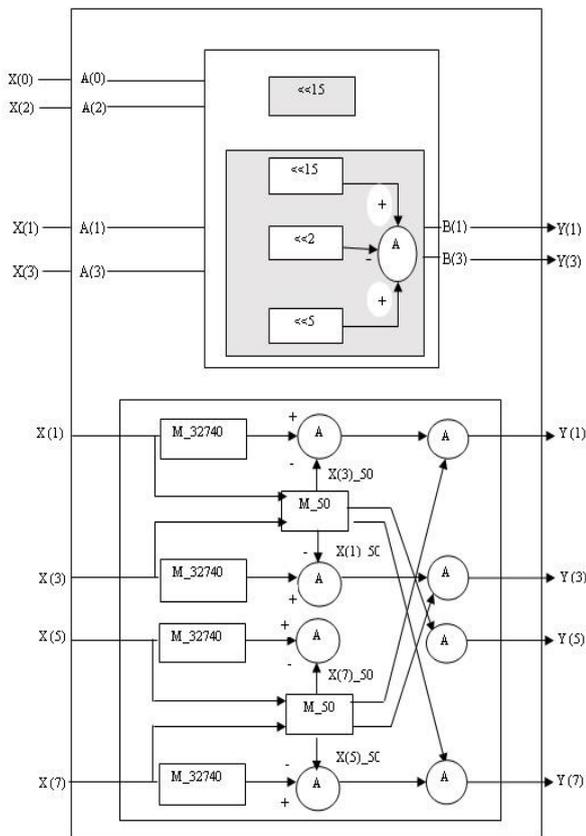


Fig. 5.8\*8 DCT architecture using reusable 4\*4 block and SAU

Generalized structure to compute DCT of varying length starting from 4\*4 to 32\*32 can be designed by splitting lower order architecture as even part block and higher order block as odd part. As it is clearly described in the design of 8-point DCT architecture the 4-point DCT architecture will become even computation block of 8-point DCT and odd part is defined with new SAU block. Similarly the 16 point DCT architecture can be built by using 8-point DCT architecture as even computation block of 16-point DCT and a new SAU block to be used for odd components. Likewise for 32-point DCT architecture can be extended to support all the block sizes of HEVC transform.

#### IV. IMPLEMENTATION RESULTS AND DISCUSSION

Input to DCT core is supplied as the hexadecimal pixel values of the frame which is a sampled video from MatLab program. First the stored video is sampled and converted to frames. Each frame is resized to 128\*128 and from this we considered 8\*8 block data which is in hexadecimal for discrete cosine transformation. The simulation results of 2D-DCT is verified by the Verilog HDL code written based on MCM based algorithm[9], as shown in figure 6. Y in figure 6 is the first 1D-transformation and Y1r is the final 2D transformation. To get 2D transformation we used the same module structure of 1D-DCT. The verilog code for our proposed design is also simulated and obtained same 2D DCT values. Hence the functionality of the proposed architecture is verified.

The integer DCT architecture proposed in [9] as well as the proposed architecture which is coded in Verilog were synthesized by Xilinx vivado design suit 2016.2 and Xilinx ISE design suit 2014.6. The 8\*8 blocks of pixel data is directly 2D transformed by the proposed architecture. But in case of reference algorithm first the transformation is done along row-wise and then that result is transposed to get 2D DCT by applying 1D DCT along column wise. The number of multiplications, addition and shift operations required for both the architecture and the standard method are listed in Table-I. The direct implementation of integer forward 2D DCT architecture can be used to perform 2D IDCT computation. The proposed architecture arithmetic complexity includes only shift and add units. For any higher order computation like 16\*16 can be performed by lower order 8\*8 block which produces even part of the transformed coefficients and odd part of the coefficients with a separate block of SAU. With the help of the reusable structures any higher block size transformation can more easily be performed. When 2D IDCT computation is to be performed, the DCT computed values are directly fed to these blocks based on selected block size and transformed coefficients will be obtained. The verilog coded proposed architecture and MCM based algorithm which are coded using Xilinx vivado design suit 2016.2 were implemented on zynq board 7vx485tffg1157-1.

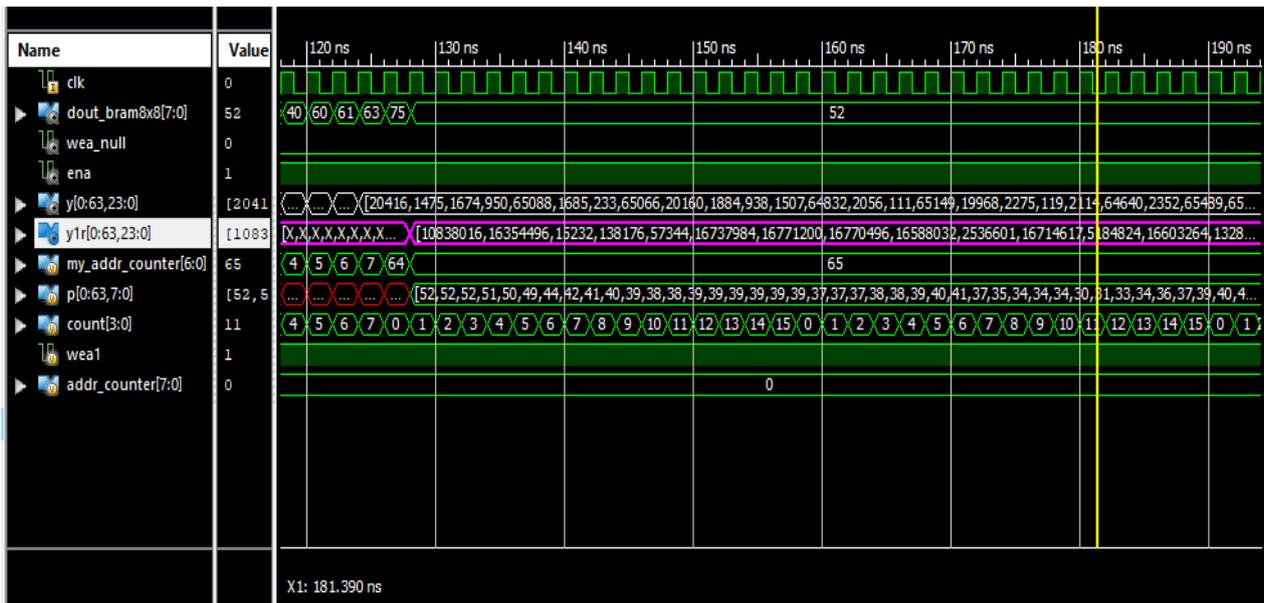


Fig.6. simulation results of 2D DCT architecture

Table- I: Arithmetic complexity of DCT architectures

Block Size	Arithmetic complexity								
	4*4			8*8			16*16		
operations	Standard	MCM based	Proposed	Standard	MCM based	Proposed	Standard	MCM based	Proposed
Multiplications	128	0	0	1024	0	0	8192	0	0
Additions	96	14	4	896	50	28	7680	186	116
Shifts	0	10	8	0	30	32	0	86	112

Table- II: Arithmetic complexity of DCT architectures

Resources	Proposed Architecture			MCM based Architecture		
	Used	Available	%	Used	Available	%
Number of Slice Registers:	52	11440	1	531	11440	4
Number of fully used LUT-FF pairs	52	266	19	521	7439	7
Number of Block RAM/FIFO:	1	32	3	1	32	3
Number of BUFG/BUFGCTRLs:	0	0	0	1	16	6

The implementation results of both the architecture in terms resource utilization and power consumption is shown in figure 7 and figure 8. As per the results obtained by implementation, the dynamic power utilization is only 0.49W for the proposed system as compared to the MCM based architecture which utilizes 12.614W. Even the static power utilization on programmable logic (PL) is less for proposed architecture it is only 0.127W as compared reference algorithm 0.411W. The number of LUT and FF used in the proposed architecture is very less. Similarly the architectures coded in verilog using Xilinx ISE design suit 2014.6. is targeted to the Spartan board xc6slx9-3tqg144. The resource utilization results are shown in table2. The time for execution of proposed system is 12.378ns for computing 8point 2D DCT.

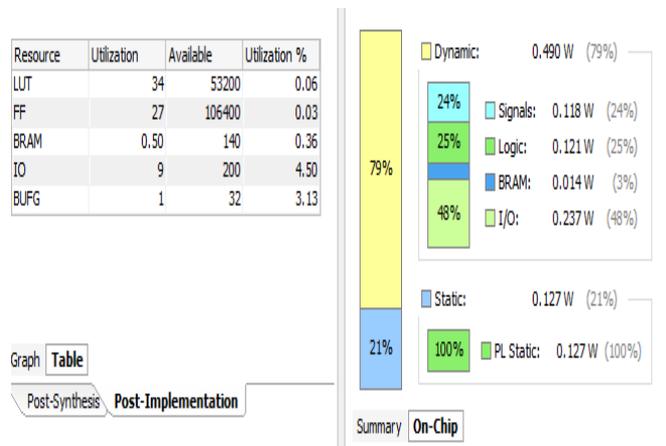


Fig.7.Proposed 2D DCT architecture results on Zynq board.

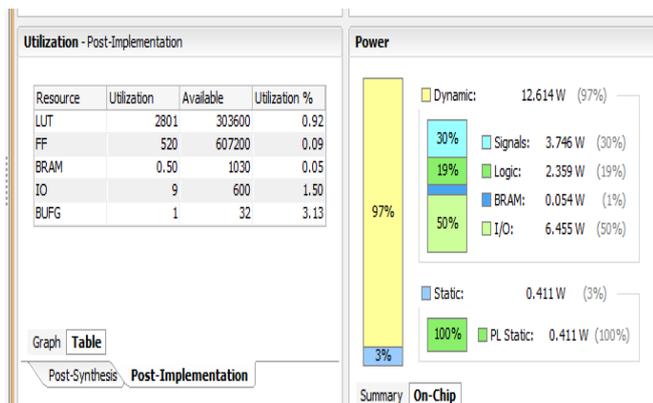


Fig. 8. MCM based 2D DCT architecture [9] results on Zynq board

### V. CONCLUSION

In this paper, reconfigurable and programmable architecture for 2D-DCT with optimal computation and hardware resource utilization is proposed and implemented on Xilinx zynq -zc702 board. The computation of the architecture is reduced by exploring orthogonal property of the DCT kernel matrices and thereby reducing the need of transpose buffer to save intermediate results. The synthesis results shows that the number of LUTs and FFs utilized by the architecture is less as compared to standard MCM based algorithms. In MCM based architecture, the decomposed kernel matrix is used and 2-D DCT/IDCT is performed by two 1D DCT structure with transpose buffer. The implementation and verification of the architecture on the board gives less dynamic and static power consumption. The architecture utilizes 12.37ns to perform DCT on 8\*8 block and to complete the entire DCT of UHD frame (3840x2160) a time of 1.6ms is observed. The proposed 2D DCT accelerator unit will be incorporated into the video CODEC micro-architecture and its performance will be evaluated in terms of area, speed and power in future work.

### REFERENCES

- G. J.Sullivan, J.R. Ohm, W.-J. Han and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp.1648-1667, Dec. 2012.
- A. Waterman, Y. Lee, D. Patterson, and K. Asanovic, "Volume I: User-Level ISA Version 2.0, The RISC-V Instruction Set Manual." [Online]. Available: <http://riscv.org/spec/riscv-spec-v2.0.pdf>.
- Ho-Cheung Ng, Cheng Liu, Hayden Kwok-Hay So "A Soft Processor Overlay with Tightly-coupled FPGA Accelerator" 2nd International Workshop on Overlay Architectures for FPGAs (OLAF2016), Monterey, CA, USA, Feb. 21, 2016.
- Bernardo Kastrup, Arjan Bink, Jan Hoogerbrugge "CONCISE:A Compiler driven CPLD based Instruction set accelerator" seventh Annual IEEE Symposium on Field Programmable Custom Computing Machines. August 2002.
- Antonino Tumeo, Matteo Monchiero, Gianluca Palermo, Fabrizio Ferrandi, Donatella Sciuto "A Pipelined Fast 2D-DCT Accelerator for FPGA-based SoCs" IEEE Computer Society Annual Symposium on VLSI(ISVLSI'07),2007.
- Wenqi Bao, Jiang Jiang, Qing Sun, Yuzhuo Fu "A Reconfigurable Macro-Pipelined DCT/IDCT Accelerator" 9th IEEE International Conference on ASIC ,2012.
- Budagavi, M., Fuldseth, A., Bjontegaard, G., Sze, V., & Sadafale, M. (2013). "Core Transform Design in the High Efficiency Video Coding (HEVC) Standard". *IEEE Journal of Selected Topics in Signal Processing*, 7(6), 1029-1041.
- Chatterjee, S., & Sarawadekar, K. (2018). *An Optimized Architecture of HEVC Core Transform using Real-valued DCT Coefficients. IEEE Transactions on Circuits and Systems II: Express Briefs*, 1-1.

- Pramod Kumar Meher, Pramod Kumar Meher, "Efficient Integer DCT Architectures for HEVC" *IEEE transactions on circuits and systems for video technology*, VOL. 24, NO. 1, JANUARY 2014.
- Abdessamad. EL ANSARI, Anass. MANSOURI, Ali. AHAITOUF, "An Efficient VLSI Architecture Design for Integer DCT in HEVC Standard" *IEEE/ACS 13th International Conference of computer system and applications*, 2016.
- Ken Cabeen , Peter Gent, "Image compression and the Discrete Cosine Transform" *Math 45*, college of the Redwood.
- A. Fuldseth, G. Bjontegaard, M. Budagavi, and V. Sze. (2011, Nov.). *JCTVC-G495, CE10: Core Transform Design for HEVC: Proposal for Current HEVC Transform* [Online]. Available: [http://phenix.int-evry.fr/jct/doc\\_end\\_user/documents/7\\_Geneva/wg11/JCTV%20C-G495-v2.zip](http://phenix.int-evry.fr/jct/doc_end_user/documents/7_Geneva/wg11/JCTV%20C-G495-v2.zip)
- Mert, A. C., Kalali, E., & Hamzaoglu, I. (2017). High performance 2D transform hardware for future video coding. *IEEE Transactions on Consumer Electronics*, 63(2), 117-125.

### AUTHORS PROFILE



**Sumalatha S.**, working as Assistant Professor in the Department of Electronics and Communication Engineering at Acharya Institute of Technology, Bangalore, Karnataka, India. She has 12 years of academic experience. She is pursuing her PhD from VTU, Belagavi and her research interests includes image processing, analog and digital circuits, VLSI and embedded systems. She is life time member of ISTE.



**Dr. Rajeswari**, is presently working as professor and HOD, Department of Electronics and Communication Engineering at Acharya Institute of Technology, Bangalore, Karnataka, India. She has completed her PhD in the field of signal processing. Her areas of interests include signal processing, Deep learning, Machine learning and control engineering.