Journal of Research in Science, Technology, Engineering and Management (JoRSTEM)
ISSN: 2456-0197

An Enhanced DCT Architecture for Power reduction in HEVC

Sowmya .S .Yeli[1], Ramya. G. R[2] and Sandeep. G. S[3]

[1, 2, 3]CS&E, G M Institute of Technology, Davanagere, Karnataka, India,
Email: sowmya_15cs049@gmit.ac.in[1], ramya_15cs036@gmit.ac.in[2], sanddepgs2gmit.ac.in[3]

Abstract—***For the current IoT technology, one of the most demanding application in terms of performance and energy is the video processing. At all levels of the video processing chain, energy reduction optimization has to be introduced in order to support real-time high definition video. Based on high efficiency video coding (HEVC), discrete cosine Transform (DCT) blocks employed in video compression are efficiently implemented in this paper. The approximate adders and subtractors that are obtained using genetic programming are used in this multiplierless 4-input DCT implementations. The worst and average errors were determined exactly by means of Binary decision diagrams (BDD) in order to manage the complexity of evolutionary approximations and provide formal guarantees in terms of errors of key circuit components.***

Keywords*—Binary decision diagrams, Discrete cosine transform, Genetic programming, High efficiency video coding, Multiple constant multiplier.

I. INTRODUCTION

The small embedded systems are expected to be a new infrastructure that is connected to IoTs in the information society. The range of these systems is from small sensors to the advanced embedded applications. The energy and performance are highly demanded by the video processing which is directly performed on IoTs. Energy consumption optimization has to be introduced at all levels of video processing for the purpose of supporting real time coding and encoding of high definition videos on low cost devices. The frequently performed operations which are energy demanding is the DCT block. In these blocks the multipliers are replaced by the adders, subtractors and shifts. Here the addition and subtraction are performed instead of multiplication. This is done in order to reduce the bit width of all operations. As well as the less important logic is pruned in order to reduce the power consumption. DCT can further be approximated as the video processing is an error resilient application.

In the state of the HEVC standard, the deep optimization and approximation of the DCT block is employed which is mainly dealt in this paper. In multiplierless DCT, the resulting error is kept under a predefined threshold when the approximate adders and subtractors are proposed in addition to common optimization.It is always guaranteed that the target error is never exceeded as the error of addition/subtraction is computed formally, without applying circuit simulation. The relaxed equivalence checking is used for error calculation procedure using BDD. Genetic programming-based approximation method is the one through which the approximate adders and subtractors are generated. The accurate logic circuits are gradually simplified when the functional gate-level approximation is performed and an acceptable error is tolerated. The DCT showed good trade-off between power consumption and quality of video processing in various approximate implementations.

II. RELATED WORK

There are many efficient implementations of particularly the DCT and other signal processing blocks in general for decades. The effort of this is leading to the establishment of new HEVC standard. There is a good potential for improving energy and implementation efficiency, especially of forward as well as inverse DCT blocks as per the results profiled on HEVC.The approximations introduced at various levels of the algorithm and its implementation can be considered as the improvements. They include bit width reduction, employing the integer data representation, multiplierless multiplication and coefficient approximation.

A. *DCT as a Test problem for Approximation Computing:* There are more radical approximations proposed with the development of the approximate computing paradigm. To evaluate the quality of general purpose approximation method, DCT blocks of JEPEG and MPEG serves as one of several test circuits.

B. *Approximate Adders:* The general purpose approximation methods or the problem specific methods are used to approximate the adders. Structures of conventional adders are exploited by problem solving methods. The

quality configurable adders which are another class of circuits allow for dynamic control of the error-power trade-off. The four types of approximate adders are: Speculative adders, segmented adders, Carry select adders, Approximate one bit full adders.

C. *Error calculation:* Circuit simulation is the one by which the error is computed in all circuit approximation methods. The worst errors checking can be done by satisfiability (SAT) solving. The average arithmetic error, worst error, error rate and average hamming distance are obtained using the Binary Decision Diagrams. Now days for determining some types of errors for certain types of circuits, the formal methods are not applicable.

III. DCT AND ITS APPLICATION

A. *Principle of DCT:* The given input vector is $X=[x_0, x_1,\ldots., x_{N-1}]^T$ and $Y=[y_0, y_1,\ldots., y_{N-1}]$ is the corresponding output. And this output is given by $Y=C_NX$, where C_N denotes the N-point DCT kernel. C_N most of all consist of real numbers. The elements $c_{ij} \epsilon C_N$ are defined as $c_{ij} = \frac{A}{\sqrt{N}} \cos [\frac{\pi}{N} (j + \frac{1}{2}) i]$, where i,j = 0,….,N-1. Here A is equal to 1 and $\sqrt{2}$ for i = 0 and i> 0 respectively.4 point, 8 point and 32 point DCT are utilized by HEVC standard both in forward and inverse form. The HEVC scales the coefficient by a power of two and rounds them to the integer value. This is for the complexity simplification of the hardware circuits computing the output of DCT.

The main properties and advantages were preserved by the rounding and scaling. When we restrict ourself to the 4 point 1D forward transform, definition of output is as follows:

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \\ y_3 \end{bmatrix} = \mathbf{C_4} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 64 & 64 & 64 & 64 \\ 83 & 36 & -36 & -83 \\ 64 & -64 & -64 & 64 \\ 36 & -83 & 83 & -36 \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (1)$$

Through the Straightforward matrix multiplication, the output of DCT could be determined. N2 multiplications and N(N-1) addition are required for this approach. Utilizing the symmetry properties of basics vector avoids the costly matrix multiplication. To reduce the complexity of DCT, Even- Odd decomposition technique is employed. The rewriting of eq.1 can be done as,

$$\begin{bmatrix} y_0 \\ y_2 \end{bmatrix} = \mathbf{C_2^o} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix} = \begin{bmatrix} 64 & 64 \\ 64 & -64 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \end{bmatrix}$$

$$\begin{bmatrix} y_1 \\ y_3 \end{bmatrix} = \mathbf{C_2^e} \begin{bmatrix} b_0 \\ b_1 \end{bmatrix} = \begin{bmatrix} 83 & 36 \\ 36 & -83 \end{bmatrix} \begin{bmatrix} b_0 \\ b_1 \end{bmatrix} \quad (2)$$

Where, a0 = x0 + x3, a1 = x1 + x2, b0 = x0 – x3, b1 = x1-x2. Then the hardware architecture is given as follows (Fig I). The first part calculates the partial sum ai and bi using w-bit adders and two w-bit subtractors, both producing a (w+1)-bit signed integer value. Multiplication is performed in the second part. Then at last two adders and two subtractors multiply the summed up values in third part.

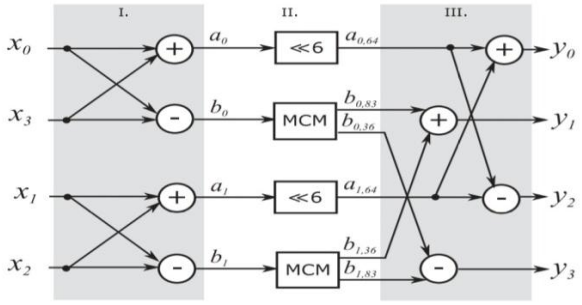

Fig. 1 Architecture of 4-point integer DCT.

To reduce the complexity of adders computing the output y0(y2), the multiplication by 64 which is performed through a logic shift may be proposed and executed after summing the partial sums a0 and a1. Here the bit wise adder is reduced by six. The coefficient used in eq 2 are constants. So the usage of costly multipliers is avoided and multiplierless multiple constant multiplier that determines the value of 36.bi and 83.bi efficiently is employed.

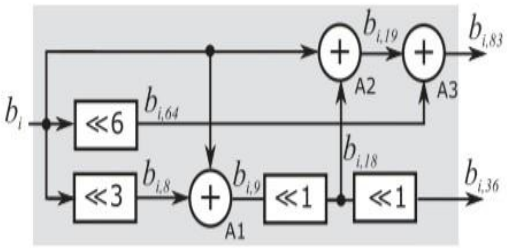

Fig. 2 Multiplier less multiple constant multipliers.

Figure 2 gives the most compact implementation of MCM for 83 and 36. It consists of a single input bi and two outputs bi, 83 and bi, 36. The three adders in it are arranged to get minimal possible bandwidth. One (w+4)-bit signed adder (A1) which adds (w+1)-bit and (w+4)-bit signed values is shared between the sub circuit outputting 83.bi and 36.bi. The remaining two adders are employed to determine 83.bi.

B. The Proposed Approximation of DCT: By reducing the number of utilized adders (subtractors) and/or employing approximate adders, DCT can be approximated. Here both are discussed.

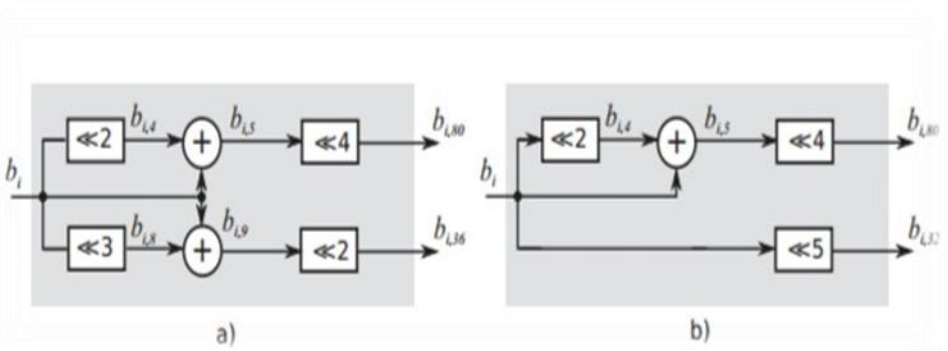

Fig. 3 Two variants of approximate MCM

The coefficient of the matrix Ce2 of eq 2 is approximated in order to reduce the number of components. That means the MCM blocks of architecture of fig 1 are replaced by different ones. Here all possible MCM implementations are enumerated with two coefficients within the range of ±16. We obtain 3 architectures representing the best trade-off among the considered parameters. Two are shown in Fig III. In the first case (Fig III:.a) the coefficient 83 is replaced with the coefficient 80. This small difference helps to reduce the number of adders by one. As the shift by 2 can be postponed, the bit-width of the adder and subtractors combining the outputs of MCM block in DCT can be reduced by 2. Another adder can be removed by using the coefficient 32 instead of 36 (Fig III.b). By this the bit-width is reduced by 4.

IV. Design of Approximate Adders and Subtractors

In order to approximate the arithmetic circuit, many approaches have been proposed. Cartesian Genetic Programming (CGP) is employed in this work as it can easily handle constraints given on candidate circuits. An array of programmable nodes constitutes the modelling of candidate circuit. 2 input Boolean functions are considered in this work. ni primary inputs and n0 primary outputs are utilized by the circuit. The primary inputs and the outputs of the nodes are labelled 0,1….N and taken as address which the node inputs can be connected to. The chromosome consists of candidate solution by N-ni triplets (x1, x2, Ψ) determining each node its function Ψ (Ψ ϵ Γ where Γ is the set of available functions) and input connections. The n0 which is the last part of chromosome specifies the nodes where the primary outputs are connected to. The CGP employs a simple search in which the initial population P contains at least one implementation of the accurate circuit and a few circuits generated using mutation of the accurate circuit. Then the candidate circuits are evaluated by the fitness function. Then the fitness score is received by the members of P and the new parent of the next population will be the highest-scored individual. Then using mutation, λ candidates are generated from this parent. The maximum number of iteration gives the termination criterion.

A. Error metrics: The average-case arithmetic error and worst-case arithmetic error must be kept under some level for designing the adders and subtractors by exhibiting suitable parameters. In this case, as a design constraint we employed the worst-case arithmetic error and average-case arithmetic error as a design objective. Using BDDs, the error metrics are determined. By subsequently representing BDDs, a virtual circuit is created for each candidate solution. The virtual circuit (Figure 4), takes a 2n-bit input vector x and produces absolute difference between the (n+1)-bit output of accurate adder (subtractor) f(x) and approximate adder (subtractor) f'(x). Subtraction in virtual circuit is calculated using m = n+2 full adders with first carry-in set to 1 and inverting each bit of the subtrahend. m-1half-adders and m-1 XOR gates are used to compute absolute value.

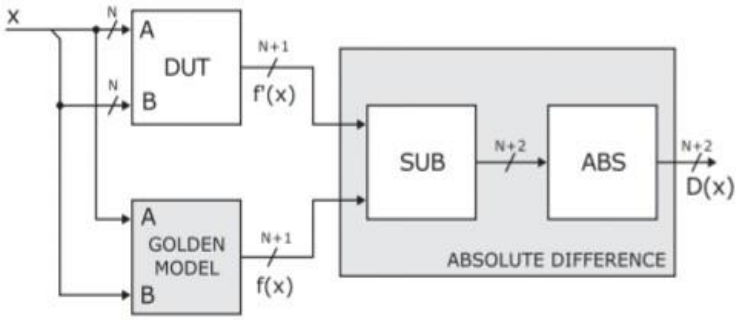

Fig. 4 Virtual circuit used for arithmetic error analysis

The difference D(x) id represented by m-bit binary vector which corresponds with a natural number. It holds that $D(x) = \sum 0 \leq i < m\ di(x) \cdot 2i$. Then, Algorithm 1 is used for the worst case analysis.

Algorithm 1: worst-case error analysis

Input: BDD representation of the virtual circuit (d)
Output: The maximum arithmetic error (ε_{max})

1 $\varepsilon_{max} \leftarrow 0, \mu \leftarrow true$;
2 for $i \in \{m-1, m-2, ..., 0\}$ do
3 if $satisfiable(\mu \wedge d_i)$ then
4 $\mu \leftarrow \mu \wedge d_i$; $\varepsilon_{max} \leftarrow \varepsilon_{max} + 2^{i}$;
5 return ε_{max} ;

The average case arithmetic error can be obtained by m calls of SAT count operation (one per each bit of D) which determines the number of input assignments that evaluates di to one.

Algorithm 2: average-case error analysis

Input: BDD representation of the virtual circuit (d)
Output: The average arithmetic error (ε_{avg})

1 $\varepsilon avg \leftarrow 0$;
2 for $i \in \{m-1, m-2, ..., 0\}$ do
3 $\varepsilon_{avg} \leftarrow \varepsilon_{avg} + 2^{i-2n} \cdot satcount\ (d_i)$;
4 return ε_{avg} ;

B .Fitness function: The following fitness function is used

$$\text{Fitness (A)} = -\begin{cases} Pwr(A) & if \in \max(A) \leq E \\ \infty & otherwise \end{cases}$$

Where, A represents a candidate circuit. E is the maximum acceptable error level and Pwr (A) is the power consumption of A.

Based on the switching activity estimation, the power consumption of a candidate circuit is estimated. At the final step, BDD for each output bit of f'(x) is available and transition probabilities are determined for each gate of DUT. To determine the switching activity and power consumption, these probabilities are employed.

V. EXPERIMENTAL SETUP

The impact on the 4-point DCT approximations on the quality on video processing and power consumption are evaluated. The accurate implementations of 4-point DCT require five different adders and two different subtractors. In addition to A1, three approximate architectures are considered: architecture A2 which utilizes MCM from fig 3a, A3 employing from MCM in fig 3b and finally A4 replaces MCM with logic shifts that can be moved after the adders and subtractors situated in the third part of DCT architecture. 6 additional adders and 3 subtractors are needed to design. To construct the four considered DCT architectures, we need to implement 16 different circuits (11 adders and 5 subtractors).

The goal of CGP is to design various approximate implementations of each circuit. We considered seven error levels defining the maximum acceptable E ϵ {0.1%, 0.2%, 0.5%, 1%, 2%, 5%, 10%}. Note that 100% with ϵ_{max} = 2w+1, where w is the circuit's bit-width. For each E, 20 independent CGP runs were executed with parameters: λ = 5, Γ = {BUF, AND, OR, XOR, NAND, NOR, XNOR}. CGP is terminated when 106 generations are exhausted or when its duration exceeded 120 minutes.

TABLE I: AVERAGE TIME NEEDED TO PERFORM ERROR ANALYSIS FOR W-BIT ADDERS/SUBTRACTORS.

	w =4	**w =8**	**w = 12**	**w = 16**
Simulation	4.5 µs	1.9 ms	682.4 ms	140.90 s
BDD ϵ_{max} Speedup	10.3 µs 0.43×	3.5 ms 0.54×	127.9 ms 5.33×	1.38 s 102.30×
BDD ϵ_{avg} Speedup	14.0 µs 0.32×	4.6 ms 0.42×	312.7 ms 2.18×	2.93 s 48.09×

Here we obtain 140 different solutions for each circuit and 2240 solutions in total. For each E and each circuit, we identified a single solution exhibiting average-case error. Then the circuits are chosen to create 7 approximate variants for each of 4 considered DCT architectures. We obtained 32 different implementations of DCT whose parameters were subsequently evaluated. At first, it is implemented in Verilogy, synthesized in ABD and measured their power consumption. Secondly, implemented in C language and employed in reference implementation of HEVC where the code performing the 4-point DCT is replaced by our implementations of DCT.

VI. RESULT

Table1 gives the result of performance analysis of the BDD based method calculating the error of approximate adders. Statically significant results are obtained by running CGP for 10 minutes. The comparison is done between proposed BDD-based technique and an optimization approach based on exhaustive simulation. As shown the BDD based method performs better for more complex circuits.

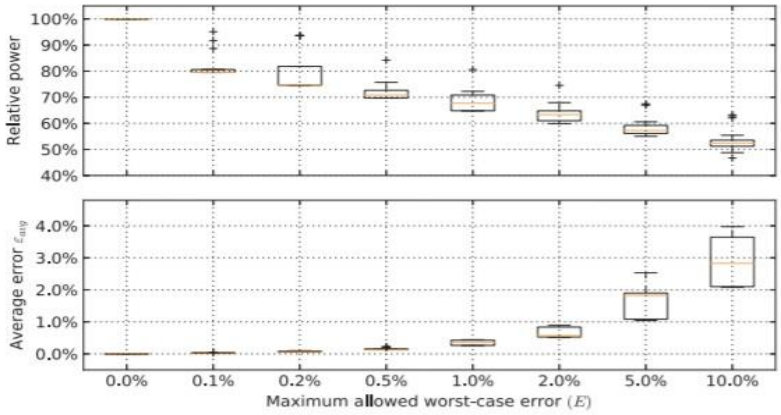

Fig. 5 Parameters of evolved approximate 16-bit adders for various error levels (for each E, 20 solutions are considered)

Figure 5 gives result of evolutionary approximation of one circuit instance-16 bit signed adder. Relatively stable results are provided by proposed evolutionary methods independently of the number of executed evolutionary runs when the boxplots size is observed. The power consumption decreases when error E is increased and it is evident. But the average case error follows opposite trend.

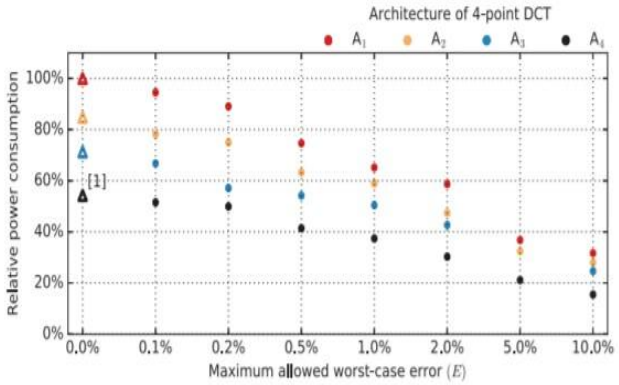

Fig. 6 Parameters of various DCT architectures Constructed using evolved adders and subtractors.

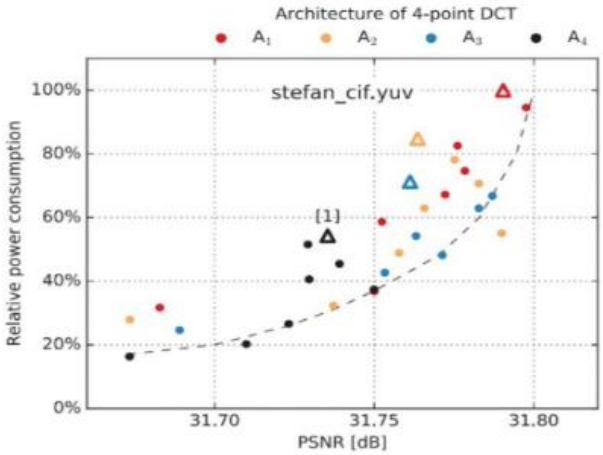

Fig. 7 The quality expressed in terms of average PSNR between original and encoded video sequence and relativepower consumption

The triangles represent the implementations employing the accurate components. The Fig VI gives the parameters of all obtained implementations of 4 point DCT summarized. The higher power consumption is achieved by the architecture that utilizes accurate adders and subtractors. This power consumption can be decreased by increasing the number of adders and decreasing bit-width of adders and subtractors.as this observation, the architecture A4 shows a 45% reduction in power consumption compared to A1.On the six common test video sequences, based on the quality of obtained video sequences the impact of the approximate DCT was evaluated. The Fig VII gives the result. Between the original frames and frames encoded, the peak signal to noise ratio (PSNR) was measured for each video sequence. Then they are decoded by the HEVC coding. A4 gives the worst results.A2 is better in providing result than A3 and A3 gives better result than A4.Quality difference between A3 and A2 is small. Here A3 achieves greater than 30% power reduction. The complexity of dependency between the quality and power consumption is increased as the approximate adders and subtractors are introduced to DCT. When an example is taken as A2 with approximation adder/subtractor giving 0.1% worst-case error give better PSNR than exact A2.In reference to HEVC implementation the phenomenon of hypothesis is related to the construction of DCT coefficient values.

VII. CONCLUSION

The new method for optimization and approximation of the DCT block were successfully proposed, which were employed in the HEVC standard. Using CGP, the approximate adders and subtractors were obtained which included

the multiplierless DCT implementations. The complexity of evolutionary approximation were managed by determining the worst and average errors by means of BDDs. Evolved implementations show better quality/power trade off than the relevant implementations available. Then the developed and proposed approximation methods will be applied to other blocks of HEVC in the future works in order to find much more power trade-off/better quality.

REFERENCES

[1] M. Jridi and P. Meher, *"A scalable approximate dct architectures for efficient HEVC compliant video coding,"* IEEE Trans. Circuits Syst. Video Technol, pp. 1–10, 2016.

[2] F. Bossen B. Bross et al, "*HEVC complexity and implementation analysis*" IEEE Trans. Circuits Syst. Video Technol, Vol. 22, No. 12, pp. 1685–1696, 2012.

[3] S. Venkataramani, A. Sabne, V. J. Kozhikkottu, K. Roy, and A. Raghunathan, "*SALSA: systematic logic synthesis of approximate circuits*" in The 49th Annual Design Automation Conference 2012, DAC'12. ACM, 2012, pp. 796–801

[4] V. Gupta, D. Mohapatra, A. Raghunathan, and K. Roy, *"Low-power digital signal processing using approximate adders*," IEEE Trans. on CAD of Integr. Circuits and Systems, Vol. 32, No. 1, pp. 124-137, 2013.

[5] H. Jiang, J. Han, and F. Lombardi, "A comparative review and evaluation of approximate adders," Proc. 25th Edition on Great Lakes Symposium on VLSI. ACM, 2015, pp. 343–348.