International Journal of Advanced Information Science and Technology (IJAIST) ISSN: 2319:2682
Vol.4, No.2, February 2015 DOI:10.15693/ijaist/2015.v4i2.351-355

Integer DCT For High Efficiency Video Coding Using Verilog HDL

A.Shanmugakani	B.Santhanalakshmi	S.Sathyapriya	Mrs.V.Lalitha
Student	Student	Student	AP/ECE
P.S.R.R College Of EngineeringFor Women, Sivakasi.	P.S.R.R College Of EngineeringFor Women,Sivakasi.	P.S.R.R College Of Engineering ForWomen, Sivakasi.	P.S.R.R College Of Engineering For Women,Sivakasi.

***Abstract*-This paper presents integer DCT for high efficiency video coding. DCT has near-optimal decorrelation efficiency so that it plays a major role in video compression.The proposed architectures for integer DCT provides less area and delay when compared to the direct implementation of the architecture. The proposed architecture could be reusable for different length of DCT like 4,8,16.Reusable and generalized architecture are designed by using MCM scheme from which the performance of proposed architectures are analysed.The performance is measured interms of the hardware resources required, delay and execution time.**

Keywords-High Efficiency Video Coding (HEVC), Discrete Cosine Transform (DCT), Integer DCT.

1.INTRODUCTION

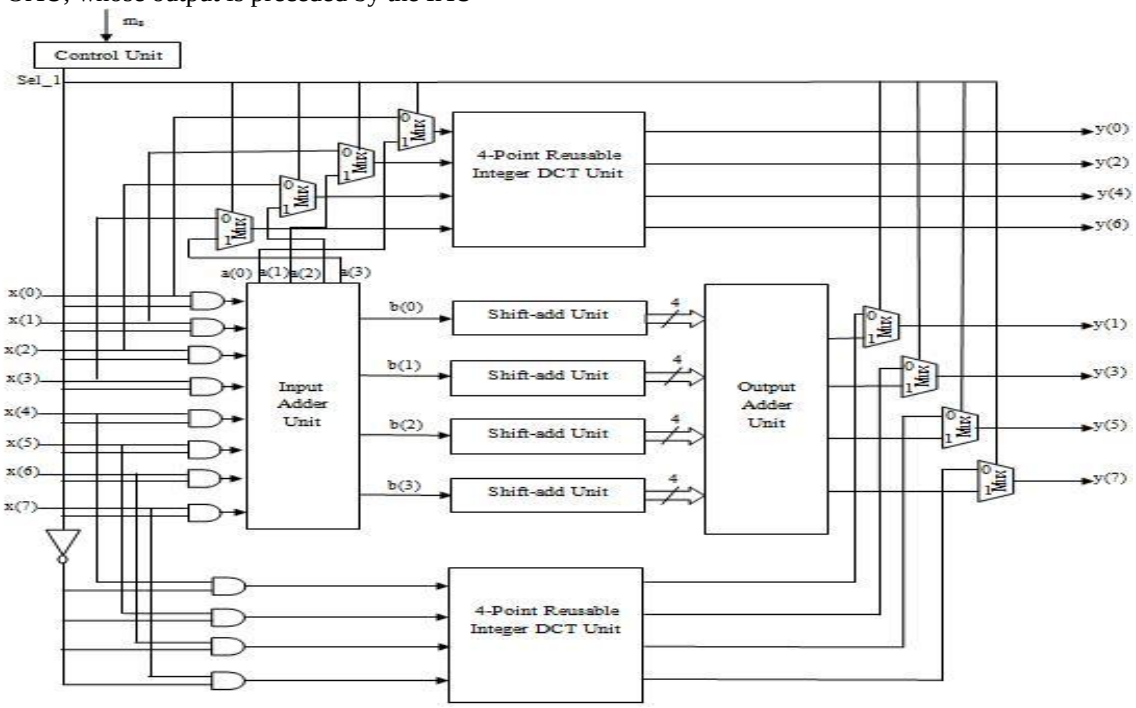
The discrete cosine transform has a major role in video compression.Several variation are introduced in integer DCT to reduce the computational complexity.Ahmed *et al.*[1]have used the lifting scheme to avoid the multiplication where DCT matrices are decomposed into sub-matrices.Shen *et al.*[2] used the multiplierless multiple constant multiplication and normal multipliers with sharing techniques for 4-point ,8-point and 16-point,32-point respectively.Park *et al.*[3] have used Chen's factorization of DCT where the processing elements are only shifters,adders,multiplexors.Macleod *et al.* [4] have used common subexpression elimination algorithm for low cost multiplierless implementation of matrix multipliers.J.Wu and Y.Li *et al.*[5]have used the rapid butterfly algorithm.W.Cham and Y.Chan *et al.*[6]have used the integer DCT to reduce the computational complexity.N.Boullis and A.Tisserand *et al*[7]have used the number recoding and dedicated common subexpression factorization technique.The previous designs for DCT based on CMM and MCM can provide optimal solutions but it cannot be reused for different length irrespective of the transform size.The proposed architecture for integer DCT can be reused for any length.

II. PROPOSED ARCHITECTURE FOR INTEGER DCT

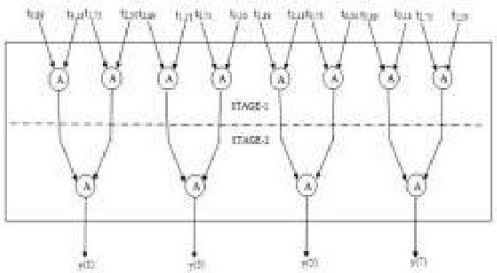
This section presents the proposed reusable and generalized architectures for the integer DCT of computation of length 8.It also has the 4-point integer DCT architecture.

A.Reusable Architeture For Integer DCT

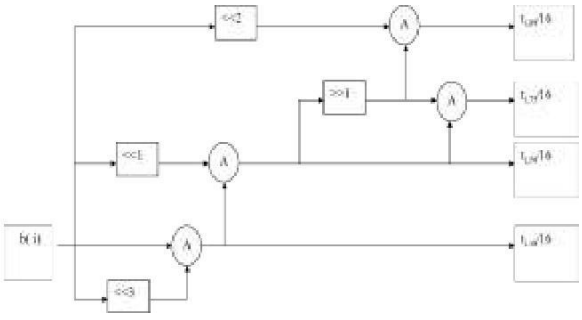
The proposed reusable architecture for DCT of length 8 is shown in fig.1(a).It consists of two 4-point DCT units. The input to one DCT is fed through four 2:1 MUXes that selects either [a(0),a(1),a(2),a(3)] or [x(0),x(1),x(2),x(3)],it depends on whether the input is used for 4-point DCT computation or a lower size. The another 4-point DCT takes the input [x(4),x(5)x(6),x(7)] When used for the computation of 4-point or a smaller size, else the input is reset by using a 4-AND gates to disable the 4-point DCT. The output of the 4-point DCT unit is multiplexed with the OAU, whose output is preceded by the IAU and SAUs in the structure. The 8 AND gates before IAU unit are used to disable the SAU, IAU, and OAU when the architecture is used for the DCT computation. In this architecture m_N is the input of the control unit used to select the size of the DCT computation.For N=32,m_{32} is the signal set to [00],[01],[10],[11] to compute the same for 4-,8-,16 respectively.The control unit produces sel_1,sel_2,where sel_1 is used as a control signals to 8 MUXes and input to 8 AND gates before input adder unit.sel_2 is used as a input to



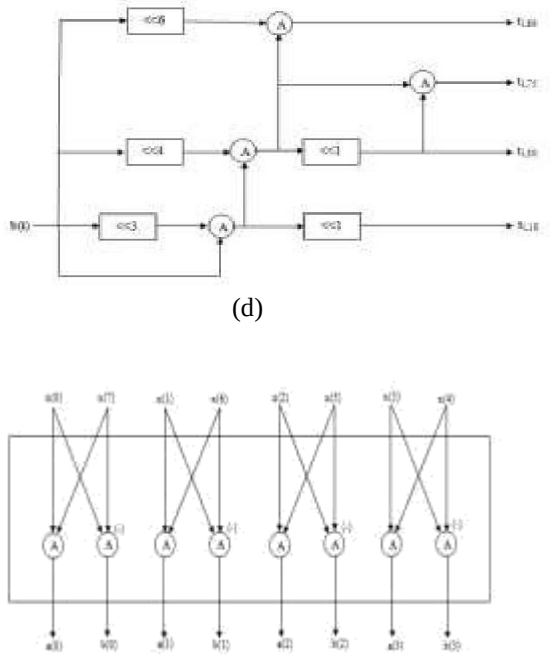
(a)



(b)



(c)



(d)

(e)

Figure 1. (A)Proposed Reusable Integer Dct (B)Structure Of Oau (C)Structure Of Msau (D)Structure Of Sau(E)Structure Of Iau

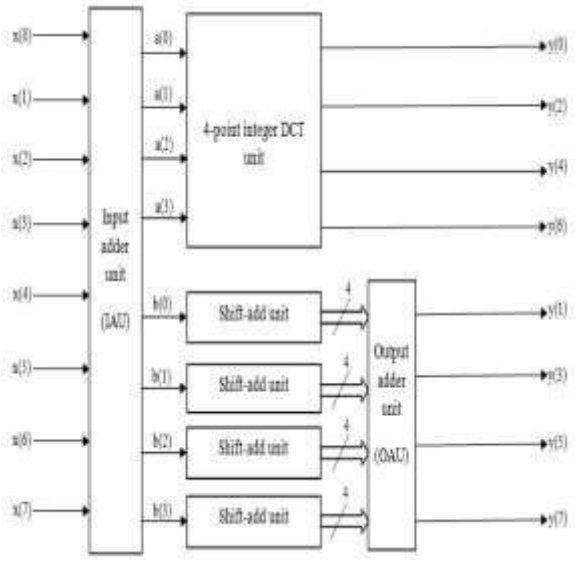
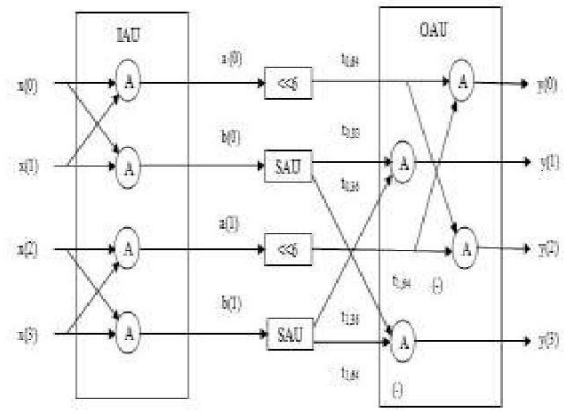
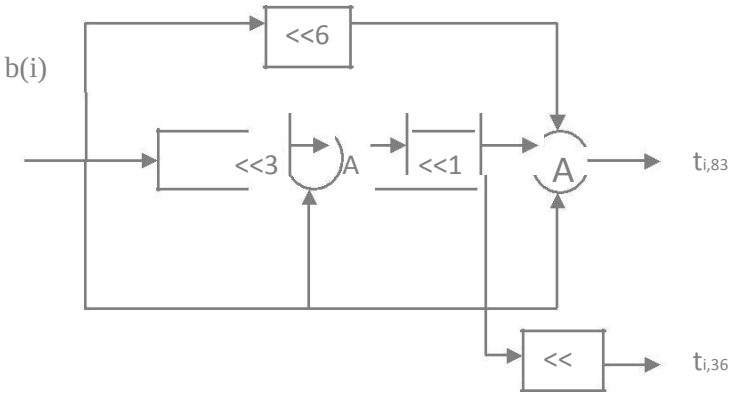


Figure .2 Proposed Generalized Architecture For Integer DCT (N=8)



(a)



(b)

Figure .3 (A) Four Point Integer DCT Unit (B) Structure Of SAU

the lower size DCT. For N=8 ,m_8 is the 1-bit signal so the sel_2 is not required.In Fig.1(d)shows the structure of SAU for N=8.It has the execution time higher than the proposed modified shift-addunit(MSAU).It has the input b(i) and outputs are $t_{i,89}$, $t_{i,75}$, $t_{i,50}$, $t_{i,18}$ Fig.1(c)shows the structure of MSAU has the less execution time with same delay and area.So the reusable architecture uses the modified shift-add unit.Fig.1(e) shows the structure of input adder unit which has the input x(0),x(1),.x(N-1) and computes the outputs are a(0),a(1),a(2),a(3),b(0),b(1),b(2),b(3).

B. Four Point Integer DCT

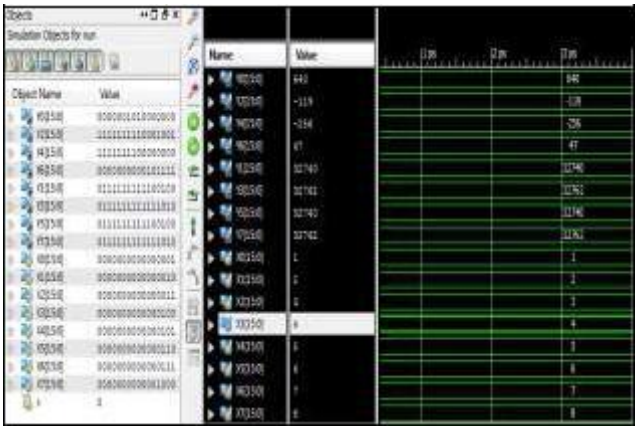
The proposed architecture for four point integer DCT is shown in fig.3(a).It has the following units such as input adder unit (IAU), output adder unit (OAU), shift-add unit (SAU).Where the input adder unit calculates the a(0),a(1),a(2),a(3) by using the stage-1 of the proposed algorithm.Fig.3(b)shows the structure of SAU which has the input b(i)and the outputs are $t_{i,36}$, $t_{i,83}$.The values for $t_{i,36}$, $t_{i,83}$ (i=0,1) are computed by using two SAUs that uses the stage-2 of the proposed algorithm.The $t_{0,64}$ and $t_{1,64}$ computation does not have any logic where the shift operations are rewired in hardware. The structure of 4 point SAU is shown in fig.2(a).T he output of OAU is adding the outputs of SAU using the stage-3 of the algorithm.

C. Generalized Architecture For integer DCT

The generalized architecture for 8-point integer DCT unit based on the proposed algorithm is shown in fig .2.The proposed architecture has the following units are 4-point integer DCT unit ,IAU ,OAU ,SAU. The 8-point IAU has the outputs are a(0), a(1), a(2), a(3), b(0), b(1), b(2), b(3). The 4-point integer DCT unit generates the output y(0), y(2), y(4), y(6) using the operation of DCT.Four SAU are required to calculate the $t_{i,89}$, $t_{i,75}$, $t_{i,50}$, $t_{i,18}$ values.For N=8 the range of i =1,2,3,4.Finally the output adder unit produces the output y(1), y(3), y(5), y(7) using the output of the SAUs.

III. RESULTS AND DISCUSSIONS

The code for the proposed architectures for different length are synthesized. The wordlength of input samples are chosen.The area, computation time , delay are obtained from the synthesis reports. The pruning scheme is more efficient for DCT length is high since the area occupied by the SAUs increases when DCT length increases.



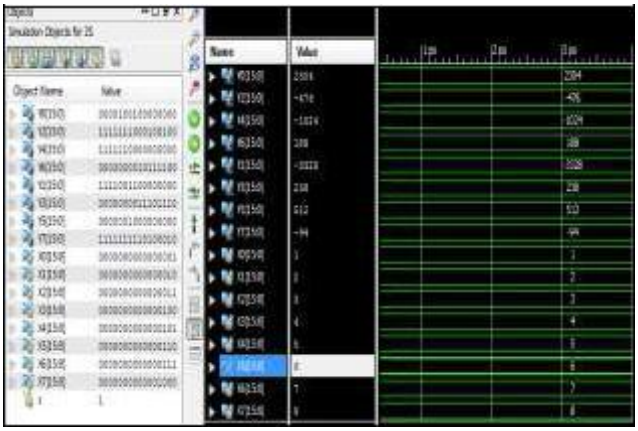


Figure 4 Synthesis Results For Reusable Integer Dct

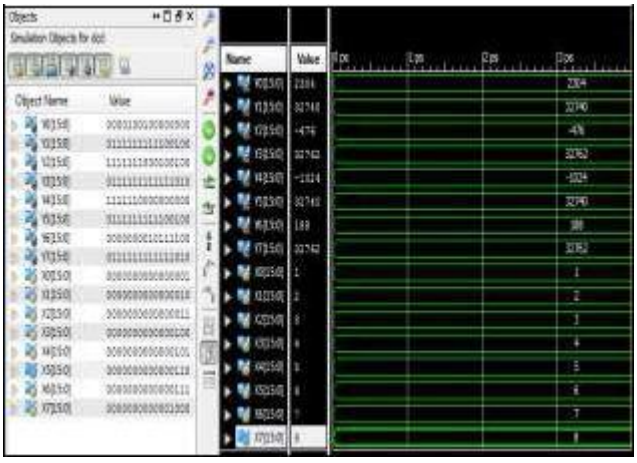


Figure.5 Synthesis Results For Generalized Integer DCT

IV.CONCLUSION

In this paper,the proposed area and delay efficient architectures for the integer DCT of

different length used in HEVC.By the use of the MSAU,the execution time is less when compared to SAU.Also shown that the architecture based on MCM is highly regular and involves less area-delay complexity and less energy consumption when compared to the direct implementation of the architecture. The proposed architecture can be used to derive a reusable architecture for DCT of different length.

Table 1
Comparision between generalized and reusable integer DCT

DESIGN	REUSABLE DESIGN	GENERALIZED DESIGN
Architecture Size	N=8	N=8
Number of 4 input LUTs used(28800)	644(2.23%)	454(1.57%)
Number of occupied slices used(7200)	185(2.56%)	125(1.73%)
Number of bonded IOBs(480)	257(53.5%)	256(53.3%)
Combinational Path Delay	13.030ns	9.626ns
Execution Time(MSAU)	12.26secs	7.86secs
Execution Time(SAU)	16.47secs	9.00secs

REFERENCES

1. A.Ahmed, M. U. Shahid , and A. Rehman, "N Point DCT VLSI Architecture for Emerging HEVC Standard,"in *Proc. VLSI Design,* vol.2012,Article 752024, pp. 1-13,2012.
2. S. Shen, W. Shen, Y. Fan, and X. Zeng,"**A** unified 4/8/16/32-point integer IDCT architecture for multiple video coding standards,"*inProc.IEEE Int. Conf. Multimedia Expo*, Jul. 2012, pp.788–793
3. J.-S. Park, W.-J. Nam, S.-M. Han, and S.Lee,"2-D large inverse transform (16x16,32x32) for HEVC,"*J Semicond. Technol. Sci*., vol. 12,no. 2,pp.203-211,Jun. 2012.
4. M. D. Macleod and A. G. Dempster, "Common subexpression elimination algorithm for low-cost multiplierless implementation of matrix multipliers*," Electron. Lett.*, vol. 40, no. 11, pp. 651–652, May 2004.
5. J. Wu and Y. Li, "A new type of integerDCT transform radix and its rapid algorithm," in Proc. International Conference on Electric Information and Control Engineering, Apr. 2011, pp. 1063–1066.
6. W. Cham and Y. Chan, "An order-16 integer cosine transform," IEEE Trans. Signal Process., vol. 39, no. 5, pp. 1205– 1208, May 1991.
7. N.Boullis and A.Tisserand, "Some optimizations of hardware multiplication by constant matrices",IEEE Trans. Comput., vol.54,no. 10,pp.1271-1282,oct,2005.