1D Discrete Cosine Transform using Distributed Algorithm

Trinadh Balaga, B.Bhaskar Rao

Abstract: ***Discrete Cosine Transform (DCT), which is an important component of image and video compression, is adopted in various standardized coding schemes, such as JPEG, As the ongoing demand increases, for better compression performance of the latest video coding standard, the H.264/AVC (Advanced Video Coding) is formulated .The H.264/AVC is also known as MPEG-4 .An advantage of the H.264/AVC is the simplicity of its transform. Distributed Arithmetic (DA) is an effective method for computing inner products when one of the input vectors is fixed. It uses pre computed look-up tables and accumulators instead of multipliers for calculating inner products and has been widely used in many DSP applications such as DFT, DCT, convolution, and digital filters. In particular, there has been great interest in implementing DCT with parallel distributed arithmetic and in reducing the ROM size required in the implementations since the DA-based DCT architectures are known to have very regular structures suitable for VLSI implementations. Low hardware circuit cost as well as low power consumption. Low hardware cost is achieved by exploiting redundant computational units and a technique to reduce error introduced by sign extension is also presented. The results indicate the considerable power as well as hardware savings in presented architecture.***

Index Terms**: *Distributed Arithmetic (DA), JPEG, Discrete Cosine Transform (DCT), MPEG.***

I. INTRODUCTION

The arrival of portable devices processing audio and video data endorses the need for solutions providing high-speed and low-power consumption for implementing the compute-intensive multimedia calculations. Hardwired implementations of such algorithms are not suitable, as a margin of flexibility is required due to the constantly changing algorithms and DSPs provide low-throughput and high power-consumption. In the past years reconfigurable hardware has emerged as a low-cost and flexible solution for high-throughput custom hardware at the cost of increased power consumption and area. As reported earlier in [1], a reconfigurable array specific to one type of calculation provides a good compromise between flexibility, power-consumption, area and performance when compared to DSPs, FPGAs and hardwired solutions. This paper presents a reconfigurable architecture specific to computations that can be implemented in Distributed Algorithms [2]; this includes computations such as DCT and FIR filtering used in video and audio systems. Previous domain-specific and coarse-grain reconfigurable architectures are more processor based; e.g. [3] provides simple programmable processors interconnected together for the execution of complex algorithms. In [4] the data path of a processor can be reconfigured during run-time to adapt to calculations. The architecture proposed in this paper is based on a heterogeneous array with a mesh of interconnects that is able to provide more parallel computations and a higher throughput at a lower frequency. Previous programmable and configurable architectures for DCT such as the one presented in [5] provide limited flexibility in the wide range of possible implementations that could be suitable. By using FPGA-style interconnects and elements we can provide greater flexibility at a lower-level.

The paper is organized as follows: In section 2 the algorithms to be supported are overviewed. Section 3 describes the area and power efficient architecture for 1D DCT computation, and in section 4 the implementation and comparison results. Conclusions are drawn in section V.

Revised Manuscript Received on February 05, 2012.

Trinadh Balaga, ECE Department ,St.Teressa Institute of Engineering and technology.Vizianagaram,INDIA, email: Vicky.balaga@gmail.com).

B.Bhaskar Rao, HOD , ECE Department ,St.Teressa Institute of Engineering and technology.Vizianagaram,INDIA email: bhaskar6278@gmail.com).

II. ERROR REDUCTION TECHNIQUE FOR 8X1 1D DCT

A. Computation of 2D -DCT

For a 2-D data X(p,q), $0 \le p \le 7$ and $0 \le q \le 7$, 8x8 2-D DCT is given by

$$F(a,b) = \frac{2}{8} C(a) C(b) \sum_{p=0}^{7} \sum_{q=0}^{7} X(p,q) \text{ x } \cos\left(\frac{(2i+1)a\pi}{16}\right) \cos\left(\frac{(2j+1)b\pi}{16}\right) \quad (1)$$

where $0 \le a \le 7$ and $0 \le b \le 7$ and c(a),c(b) = $\frac{1}{\sqrt{2}}$ for a,b=0, c(a),c(b) =1 otherwise. Implementation computation is reduced by decomposing (1) in two 8x1 1-D DCT given by,

$$F(a,b) = \frac{1}{2} C(a) \sum_{i=0}^{7} X(p) COS\left(\frac{(2i+1)a\pi}{16}\right) \quad (2)$$

For 2-D DCT computation of a 8x8 2-D data, first row-wise 8x1 1-D DCT is taken for all rows followed by column-wise 8x1 1-D DCT to all columns. Intermediate results of 1-D DCT are stored in transposition memory [10].

In [9], (2) can be simplified as,

$$F(0) = [X(0) + X(1) + X(2) + X(3) + X(4) + X(5) + X6 + X7 \quad (3a)$$

$$F(1) = [X(0 - X(7)]A + [X(1) - X(6)]B + X(2) - X5C + X3 - X(4)]D \quad (3b)$$

$$F(2) = [X(0) - X(3) - X(4) + X(7)]M + X(1) - X(2) - X(5) + X(6)]N \quad (3c)$$

$$F(3) = [X(0) - X(7)]B + [X(1) - X(6)](-D) + [X(2) - X(5)](-A) + [X(3) - X(4)](-C) \quad (3d)$$

$$F(4) = [X(0) - X(1) - X(2) + X(3) + X(4) - X(5) - X(6) + X(7)]I \quad (3e)$$

$$F(5) = [X(0) - X(7)]C + [X(1) - X(6)](-A) + [X(2) - X(5)]D + [X(3) - X(4)]B \quad (3f)$$

$$F(6) = [X(0) - X(3) -$$

$X(4)+X(7)]N+[X(1)-X(2)-X(5)+X(6)](-M)$ (3g)

$F(7)=[X(0)-X(7)]D+[X(1)-X(6)](-C)+[X(2)-X(5)]B+[X(3)-X(4)](-A)$ (3h)

Where

M=$\frac{1}{2}\cos\frac{\pi}{8}$, N=$\frac{1}{2}\cos\frac{3\pi}{8}$, I=$\frac{1}{2}\cos\frac{\pi}{4}$

A=$\frac{1}{2}\cos\frac{\pi}{16}$, B=$\frac{1}{2}\cos\frac{3\pi}{16}$, C=$\frac{1}{2}\cos\frac{5\pi}{16}$, D=$\frac{1}{2}\cos\frac{7\pi}{16}$,

Distributed arithmetic is used to compute the 8 equations above where cosine terms are expressed in DA form. Implementation is realized by using shift and adds operations. Shifting operation is performed by wiring. Each value of F is computed in parallel and hence faster speed is achieved [11]. Shifted data are represented by less number of bits and hence adder bit-width is reduced resulting in less hardware cost. For DCT computation image data is represented in signed 2's complement form range -128 to 127. Bit width of shifted data is determined by number of times shift operation is done. So different bit-width intermediate data are present which are to be added. For 2-input adder both input data width has to be equal and hence sign extension is done in smaller bit-width data. Shifting and addition with sign extension creates error. For example if initial value is -2, in 8-bit 2's complement representation this can be written 11111110. If we take 4 shifted sample of this value a=111111 (shifting 2 times), b=1111 (shifting 4 times), c=111 (shifting 5 times), d=11 (shifting 6 times) all are -1. But all these data should be zero. If we add these values in cascade, result we will have -4 (which should be zero). To overcome this problem we have realized adder as shown in fig.1. If one of the input is -1

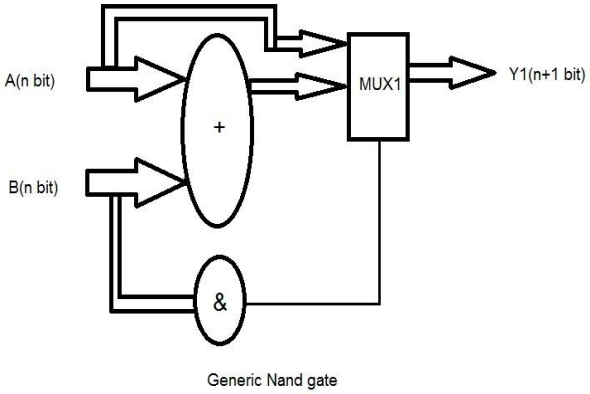
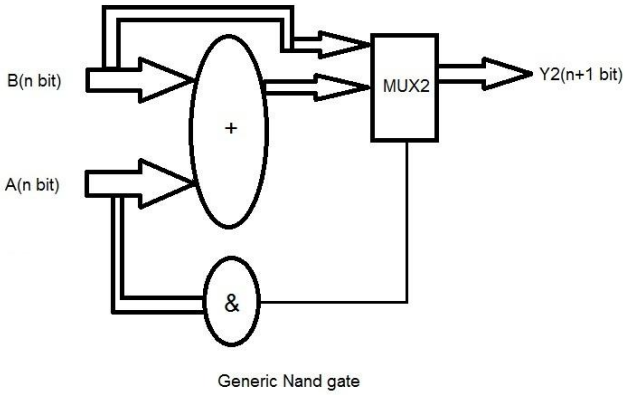
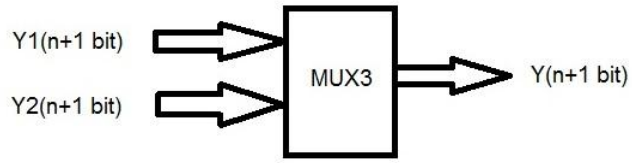


Fig.1: Circuits to reduce sign extension error propagation when number is negative

(a) MUX1 selects A if B is -1 else sum of A and B



(b) MUX2 select B if A is -1 else sum of A and B



(c) Final sum is from MUX1 or MUX2 output.

then output is other one. To verify this VHDL implementation of 1-D DCT and simulation is done using Xilinx ISE simulator for 8x1 data matrix of X given by,

X = [60, 40, 25, 55, 40, 42, 82, 84]

Matlab simulation result for 1-D DCT gives,

Y = [151.3209 -32.4895 33.1588 -1.7108 17.6777 18.5074 -16.0309 -5.0975]

Fig.2 shows the result of Xilinx ISE simulator using simple adder and proposed adder scheme. It is evident with matlab comparison that error due to sign extension is less in proposed adder scheme.

a0	60
a1	40
a2	25
a3	55
a4	40
a5	42
a6	82
a7	84
b0[10:0]	149
b1[10:0]	-36
b2[10:0]	31
b3[10:0]	-5
b4[10:0]	16
b5[10:0]	14
b6[10:0]	-18
b7[10:0]	-11

(a)

a0	60
a1	40
a2	25
a3	55
a4	40
a5	42
a6	82
a7	84
b0[10:0]	149
b1[10:0]	-32
b2[10:0]	31
b3[10:0]	-2
b4[10:0]	16
b5[10:0]	18
b6[10:0]	-18
b7[10:0]	-9

(b)

Fig 2: Simulation result of 1D DCT architecture using Xilinx ISE simulator (a) Addition Operation (b) Proposed adder.

III. PROPOSED ARCHITECTURE OF 8X1 1D DCT

Instead of computing $F(0)to\,F(7)$ in parallel process, they can be computed in pipeline process and there are only seven oscine terms which can be seen from equation 3a to 3h. Hence we can implement a Distributed Arithmetic based module which gives the multiplication and accumulation results by taking the inputs.

Let us consider,

a1=X(0)+X(1)+X(2)+X(3)+X(4)+X(5)+X(6)+X(7),

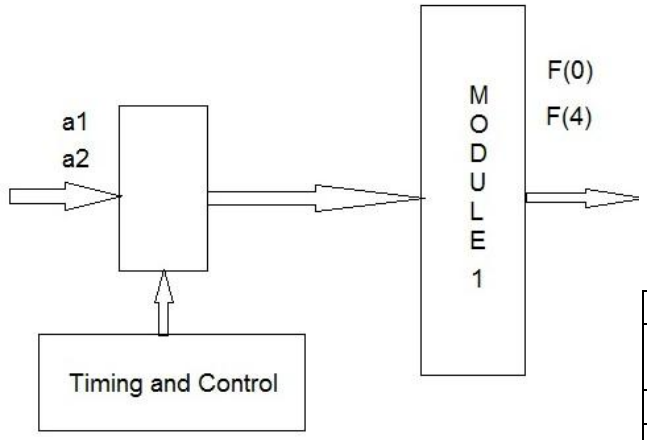
a2=X(0)-X(1)-X(2)+X(3)+X(4)-X(5)-X(6)+X(7),

b1=X(0)-X(7),b2=X(1)-X(6),b3=X(2)-X(5),

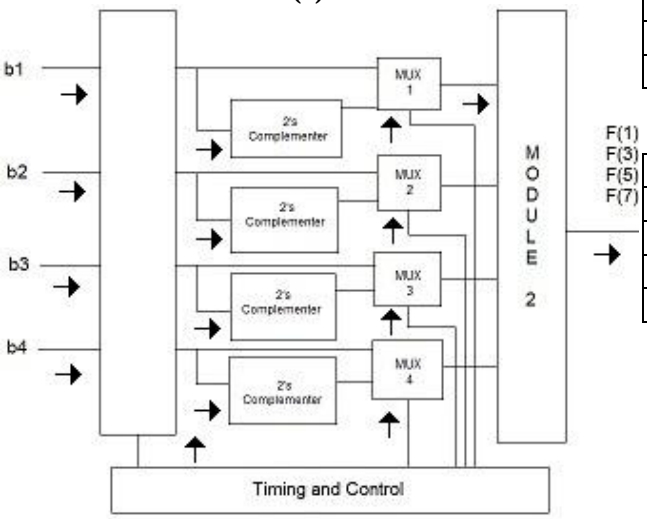
b4=X(3)-X(4),

c1=X(0)-X(3)-X(4)+X(7) and c2=X(1)-X(2)-X(5)+X(6).

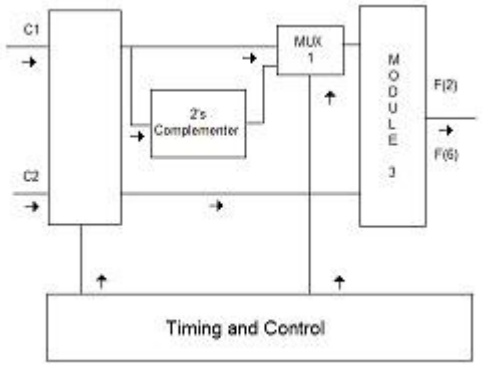
Then from (3a) to (3h), F(0)=a1xP, F(4)=a2xP,
F(1)=b1xA+b2xB+b3xC+b4xD,
F(3)= b1xB-b2xD-b3xA-b4xC,
F(5)= b1xC-b2xA+b3xD+b4xB,
F(7)= b1xD-b2xC+b3xB-b4xA,
F(2)=c1xM+c2xN, and F(6)=c1xN-c2Xm



(a)



(b)



(c)

Fig.3. VLSI architecture for computation of 8 point DCT in pipeline manner for (a) computation of F(0) and F(4) (b) computation of F(1), F(3), F(5) and F(7) and (c) computation of F(2) and F(6).

For the 8 coefficients computations three module namely MODULE1, MODULE2 and MODULE3 is constructed as shown in fig.3. For MODULE1 (1/2)cos(π /4) is expressed in DA form with one input. For MODULE2, 1/2cos(π /16), (1/2)cos(3π /16), (1/2)cos(5π /16), and (1/2)cos(7π /16) are expressed in DA form with four inputs and for MODULE3 (1/2)cos(π/8) and (1/2)cos(3π /8) is expressed in DA form with two inputs. Table I, II and III show the inputs given to these module on clock cycle basis provided by timing and control unit. Outputs obtained are stored in 8 registers.

Table I: Pipeline Computation Of Dct Coefficients F(0) And F(4)

MODULE 1		
	CLOCK CYCLE 1	CLOCK CYCLE 2
Input	a1	a2
Output	F(0)	F(4)

TABLE IIPIPELINE COMPUTATION OF DCT COEFFICIENTSF(1) ,F(3),F(5),F(7)

MODULE 2				
	CLOCK CYCLE 1	CLOCK CYCLE 2	CLOCK CYCLE 3	CLOCK CYCLE 4
Input 1	b1	-b3	-b2	-b4
Input 2	b2	b1	b4	b3
Input 3	b3	-b4	b1	-b2
Input 3	b4	-b2	b3	b1
Output	F(1)	F(3)	F(5)	F(7)

TABLE IIIPIPELINE COMPUTATION OF DCT COEFFICIENTSF (2) AND F(6)

MODULE 3		
	CLOCK CYCLE 1	CLOCK CYCLE 2
Input 1	c1	-c2
Input 1	c2	c1
Output	F(2)	F(6)

IV. IMPLEMENTATION RESULTS AND COMPARISIONS

VHDL code is written for the implementation of 1-D DCT architecture in (9) and proposed architecture. Adders used in both the implementations are proposed scheme (as explained in section II). Registers are added at the output of 1-D DCT architecture in (9). Code is synthesized in Xilinx xc2vp30 FPGA device. Table IV shows the device utilization summary for the FPGA implementation. From the FPGA implementation comparison results, it is evident that the proposed architecture is efficient in terms of area (FPGA resources in case of FPGA implementation) and power.

TABLE IV DEVICE UTILIZATION FOR THE FPGA IMPLEMENTATION

FPGA CHIP XILINX XC2VP30		
	1-DDCT architecture in (9)	Proposed1-D DCT architecture
# of 4 input LUTs	1268	696
# of slices	694	370
# of slice Flip Flops	0	97
# of IOB Flip Flops	88	0
Min. Period (ns)	32.6	16.29
Power (W)	13.1	2.06

8x8 2-D DCT is implemented using row column decomposition technique. Table V shows FPGA implementation device utilization summary.

TABLE V DEVICE UTILIZATION SUMMARY FOR 2-D DCT IMPLEMENTATION USING ROW-COLUMN DECOMPOSITION TECHNIQUE OF PROPOSED 1-D DCT ARCHITECTURE

FPGA-chip Xilinx XC2VP30	
# of 4 input LUTs	2522
# of slices	1701
# of slice Flip Flops	1025
Max. Freq.(MHz)	45.173
Power (W)	0.751

V. CONCLUSION

We have proposed an area and power efficient architecture for the computation of 1-D DCT using ROM free DA is implemented in Xilinx FPGA. Comparison with recently reported DA based 1-D DCT, a significant area reduction as well as low power consumption is achieved in proposed architecture. With proposed 1-D DCT architecture, 2-D DCT is implemented using row column decomposition technique and area and power results are tabulated. An adder for the error reduction introduced due to shift and adds with less number of bit representation is also proposed and comparative HDL simulation is done for the accuracy. Proposed adder shows less error as compared to conventional adder.

REFERENCES

1. T. Acharya and P. Tsai, "JPEG2000 Standard for Image Compression: Algorithms and VLSI Architectures" J. Wiley & sons. NJ, 2005.
2. Gregory K. Wallace, "The JPEG Still Picture Compression Standard,"
3. IEEE Transactions on Consumer Electronics , vol.38(I), Feb. 1992. [3] R. C. Gonzalez, R. E. Woods, "Digital Image Processing," 2nd.Ed.,Prentice Hall, 2002.
4. F.H.P. Fitzek, M. Reisslein, "MPEG-4 and H.263 Video Traces for
5. Network Performance Evaluation ," IEEE Network , vol.15, no.6, pp.40-54, Nov/Dec 2001.
6. Luciano Volcan Agostini, Ivan Saraiva Silva and Sergio Bampi,
7. "Multiplierless and fully pipelined JPEG compression soft IP targeting
8. FPGAs," Microprocessors and Microsystems, vol. 31(8), 3 pp.487-497, Dec. 2007.
9. S. A. White, "Applications of distributed arithmetic to digital signal processing: a tutorial review," IEEE ASSP Magazine , vol.6, no.3, pp.4-19,Jul.1989.
10. M.-T. Sun, T.-C. Chen, A.M. Gottlieb, ''VLSI Implementation of a 16x16
11. Discrete Cosine Transform," IEEE Transactions on Circuits and Systems, vol.36, no. 4, pp. 610 – 617, Apr. 1989.
12. A. Shams, A. Chidanandan, W. Pan, and M. Bayoumi, "NEDA: A low power high throughput DCT architecture," IEEE Transactions on Signal Processing, vol.54(3), Mar. 2006.
13. Peng Chungan, Cao Xixin, Yu Dunshan, Zhang Xing, "A 250MHz optimized distributed architecture of 2D 8x8 DCT," 7th International Conference on ASIC, pp. 189 – 192, Oct. 2007.
14. M. Kovac, N. Ranganathan, "JAGUAR: A Fully Pipelined VLSI
15. Architecture for JPEG Image Compression Standard," Proceedings of the IEEE, vol.83, no.2, pp. 247-258,Feb.1995.
16. Yuan-Ho Chen, Tsin-Yuan Chang, Chung-Yi Li, "High Throughput DA-Based DCT With High Accuracy Error-Compensated Adder Tree," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. PP, issue 99, pp. 1-5, Jan 2010.
17. A. Kassem, M. Hamad, E. Haidamous, "Image Compression on FPGA using DCT," International Conference on Advances in Computational Tools for Engineering Applications, 2009 , ACTEA '09, pp.320-323, 15-17 July 2009.
18. Leila Makkaoui, Vincent Lecuire and Jean-Marie Moureaux, "Fast Zonal DCT-based image compression for Wireless Camera Sensor Networks," 2nd International Conference on Image Processing Theory Tools and Applications (IPTA), pp. 126-129, 2010.
19. Byoung-Il Kim and Sotirios G. Ziavras, "Low-Power Multiplierless DCT for Image/Video Coders," IEEE 13th International Symposium on Consumer Electronics, 2009. ISCE '09, pp. 133-136.
20. C. H. Chen, B. D. Liu and J. F. Yang, "Direct Recursive Structures for Computing Radix-r Two-Dimensional DCT/IDCT/DST/IDST", IEEE Transactions On Circuits And Systems ,I Regular Papers , vol. 51, no. 10, October 2004.
21. S. An C. Wang, "Recursive algorithm, architectures and FPGA implementation of the two-dimensional discrete cosine transform," IET Image Process ., vol. 2(6), pp. 286–294, 2008. Xilinx Inc., Web: www.xilinx.com.

AUTHOR'S PROFILE:

Trinadh Balaga is pursuing his M.Tech in St. Theressa Institute of Engineering and Technology under the guidance of Mr. B.Bhaskar Rao, Assoc. Professor, HOD, and Department of ECE. My research interests include VLSI design and digital signal processing.

Sri. B.Bhaskar Rao working as Assoc. Professor and Head, Department of ECE at St. Theressa Institute of Engineering and Technology, Vizianagaram . His research interests include VLSI design and digital signal processing.