ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 1, Issue 4, December 2014

A Novel VLSI Architecture MST Core Supported Video Codec Using CSDA

Mr.B.Mohan, M.E.,
Assistant Professor of ECE[1],
B.Sutha Sivakumari, II M.E VLSI Design[2]
SCAD College of Engineering and Technology
Tirunelveli, India
mohan.me@gmail.com[1]
suthasivakumarib@gmail.com[2]

Abstract— This paper presents a low cost effective multistandard transform supported video codec using combination of factor sharing and distributed arithmetic called CSDA (Common Sharing Distributed Arithmetic) method. Several researches have worked on transform core in order to reduce the hardware cost. This new architecture only uses adders and shifters instead of multipliers. This architecture shares available hardware resources. So that it will reduce the hardware cost very efficiently. To support different standards such as MPEG1/2/4,VC-1,H.264, it uses selection signals of multiplexers. The language which is used to build this architecture is Verilog HDL. The simulation of this MST core is done by using Model Sim.

Index Terms—Common Sharing Distributed Arithmetic (CSDA), Discrete Cosine Transform (DCT), integer transform, Multistandard Transform (MST).

I. INTRODUCTION

Transforms are widely used in image and video applications. Several groups, such as The International Telecommunication Union Telecommunication Standardization Sector (ITU-T), Microsoft Corporation and International Organization for Standardization (ISO), has developed various transforms. Several researchers have worked on transform core designs, including Integer Transform and Discrete Cosine Transform (DCT), Distributed Arithmetic (DA), Factor Sharing (FS) technique and matrix decomposition methods to reducing hardware cost and achieving high throughput rate. In image processing, the samples can be the values of pixels along a row or column of a raster image.

A Discrete Cosine Transform (DCT) expresses a finite sequence of data points in terms of a sum of cosine functions oscillating at different frequencies. It is a real transform with better computational efficiency than DFT which by definition is a complex transform. DCT does not introduce discontinuity while imposing periodicity in the time signal.

A Distributed arithmetic is a bit level rearrangement of a multiply accumulate to hide the multiplications. It is a powerful technique for reducing the size of a parallel hardware multiply-accumulate that is well suited to FPGA designs. It can also be extended to other sum functions such as complex multiplies, Fourier transforms and so on. Distributed Arithmetic enjoyed widespread application in Very Large Scale Integration (VLSI) implementations of Digital Signal Processing (DSP) algorithms.

The circuit area can be efficiently reduced by adopting appropriate circuit share strategies. Although some elements in the integer Inverse Discrete Cosine Transforms (IDCT) matrix are different, some Sums of their Binary Factors (SBFs) are possibly the same. The same SBFs can be shared in the multiplier-less implementation of the integer IDCT. This circuit optimization strategy is called as Factor Sharing (FS). New Distributed Arithmetic (NEDA) is one of the techniques to implement many digital signal processing system that require multiply and accumulate units. DA has become an efficient tool to implement multiply and accumulate unit in many DSP systems. It eliminates the need of a multiplier that is used as a part of multiply and Accumulate unit. NEDA is a method which does not require any Read Only Memory (ROM), thus making the design to have reduced hardware.

A. Video Codec Design

A video codec is a device or software that enables compression or decompression of digital video. Historically,

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 1, Issue 4, December 2014

video was stored as an analog signal on magnetic tape. Around the time when the compact disc entered the market as a digital-format replacement for analog audio, it became feasible to also begin storing and using video in digital form, and a variety of such technologies began to emerge. There is a complex relationship between the video quality, the quantity of the data needed to represent it, the complexity of the encoding and decoding algorithms, robustness to data losses and errors, ease of editing, random access, and end-to-end delay.

MPEG (Motion Picture Experts Group) is one of the biggest families in video codec, and it is the most common video format. MPEG-1, its compression algorithm is widely used in the production of VCD and the download of some video clip. Almost all VCD is compressed using the Mpeg-1 format.

MPEG-2, its compression algorithm used in the production of the Digital Video Disc (DVD), and also in some of the High Definition Television(HDTV) and high demand video editing, processing of the application. MPEG-4 is a new compression algorithm; the use of this compression algorithm can be a 120 minute film is compressed to about 300MB. H.264 is a video compression format that is currently one of the most commonly used formats for the recording, compression, and distribution of video content.

VC-1 is a video coding standard developed by Microsoft. It began as Windows Media Video 9. It is prevalent in Advanced System Format (ASF) files downloaded from the internet. It is also supposed to be used on High Definition/Density Digital Versatile Disc (HD-DVDs).

II. EXISTING SYSTEM

The existing methods that are used to design the transform core are Discrete Cosine Transform (DCT), and integer transform, using Distributed Arithmetic (DA), Factor Sharing (FS), and Matrix decomposition methods to reduce the hardware cost. The inner products can be implemented using Read Only Memory (ROM) and accumulators instead of multipliers to reduce the hardware cost. Distributed Arithmetic enjoyed widespread application in Very Large Scale Integration (VLSI) implementations of digital signal processing algorithms. But it also increases the hardware cost. So that several architectures without ROM with DA are designed to reduce the hardware cost. Bit-level sharing scheme to construct the adder-based butterfly matrix called New Distributed Arithmetic (NEDA) are designed to reduce the hardware cost which is better than the previous methods. To improve the throughput rate of the NEDA method adder trees are introduced.

FS method is used to share hardware resources in delta matrices. Matrices are derived for multistandards as linear combinations from the same matrix and delta matrix, and show that the coefficients in the same matrix. To further reduce the area, the combination of Factor Sharing (FS) and Adder Sharing (AS) are used for multistandard applications. It reduces area from the previous methods. The shifting and addition computation uses a shift-and-add operator in Very Large Scale Integration (VLSI) implementation in order to reduce hardware cost. However, when the number of the shifting and addition words increases, the computation time will also increase.

III. PROPOSED SYSTEM

A. *Introduction*

This session discusses the method that is chosen to design the 2-D Multi Standard Transform (MST) core that supports MPEG-1/2/4(8x8), H.264(8x8, 4x4), and VC-1(8x8, 8x4, 4x8, 4x4) transforms and also about how the hardware resources used for this design be reduced.

B. *Proposed CSDA Algorithm*

The proposed CSDA combines Distributed Arithmetic (DA) and Factor sharing (FS) methods. These two methods are applied to the coefficients in the matrixes. The same factor in each coefficient is first shared by the FS method, and then the same combination of input among each coefficient position is shared by the DA method. The proposed CSDA algorithm in matrix inner product can explain as follows:

$$\begin{bmatrix} \\ = \end{bmatrix} \begin{bmatrix} D11 & D12 \\ D21 & D22 \end{bmatrix} \begin{bmatrix} \end{bmatrix} \quad (1)$$

where the coefficients D11 ~ D22 are all five bits CSD numbers.

$$\begin{aligned} D11 &= \begin{bmatrix} 1 \end{bmatrix} \\ D12 &= \begin{bmatrix} 1 \end{bmatrix} \\ D21 &= \begin{bmatrix} 1 \end{bmatrix} \\ D21 &= \begin{bmatrix} 0 \end{bmatrix} \end{aligned} \quad (2)$$

Fig 3.1 shows the proposed CSDA sharing flow. The shared factor FS in four coefficients is [1 -1] and D11 ~ D22 can use instead of [1 -1], with the corresponding position under the FS method. The DA is applied to share the same position for the input, and the DA shared coefficient

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 1, Issue 4, December 2014

DA1=(Y1+Y2) FS. Finally, the matrix inner product in (1) above equation can be implemented by shifting and adding every nonzero weight position.

Z1= FS method DA

method

Z2= FS method DA

method

Fig. 1 Example of CSDA Algorithm

The proposed CSDA combines the FS and DA methods to obtain better resource sharing for inner product operation. The FS method is adopted first to identify the factors that can achieve higher capability in hardware resource sharing, where the hardware resource is defined as the number of adder usage. Based on the results of the FS method, the DA method is used to find the shared coefficient. The proposed CSDA circuit will follow the adder tree circuit. Thus, the CSDA method aims to reduce the non-zero elements to as few as possible. The CSDA shared coefficient is used for estimating and comparing the number of adders in the CSDA loop. The large numbers of loops are used to determine the smallest hardware resource by these steps, and the CSDA shared coefficients can be established. So we used the iteration searching loop to determine the better CSDA coefficients.

C. MODULES

The modules which are used to build 2-D CSDA MST core are explained below.

a. *1-D Common Sharing Distributed arithmetic-MST*

The architecture of the 1-D eight-point MST core is shown in Fig 3.2, which consists of a Selected Butterfly (SBF) module, an Even part CSDA (CSDA_E), an Odd part CSDA (CSDA_O), Eight Error Compensated Trees (ECATs) and a permutation module. Based on the proposed CSDA algorithm, the coefficients for MPEG-1/2/4, H.264, and VC-1 transforms are chosen to achieve high sharing capability for arithmetic resources. Note that the choice of shared coefficient is obtained by some constraints. Thus, the chosen CSDA is just a local or suboptimal solution and not a global optimal solution. By using the proposed CSDA algorithm, the chosen coefficients of CSD expression can achieve high sharing capability for arithmetic resources. The SBF module executes for the eight-point transform and by passes the input data for two four-point transforms. After the SBF module, the CSDA_E and CSDA_O execute and by feeding input data a and b, respectively. The butterfly operation circuit comprises a coefficient register configured and structured to store coefficients employed by the butterfly operations. This 1-D Common Sharing Distributed Arithmetic (CSDA) MST Core is designed with seven multiplexers in order to choose the appropriate standards.

The selection signals of multiplexers for appropriate standards are as follows: For supporting MPEG standard (8-point transform) we have to select mux,mux-1,mux-2,mux-3,mux-4,mux-6. For supporting H.264 standard (8 point transform), we have to select mux and mux-3. We have to select mux-5 and mux-6 to support H.264 standard transform (4 point). By giving the selection signals for mux,mux-2,mux-5,mux-6,it will support VC-1 transform standard(8 point).For to support VC-1 standard transform(4-point) we give the selection signals for mux-2,mux-4,mux-5,mux-6.

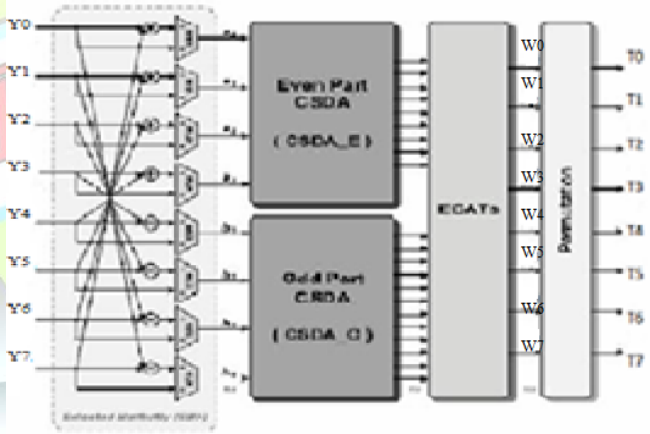


Fig. 2 Architecture of the proposed 1-D CSDA-MST

b. *Even part Common Sharing Distributed Arithmetic Circuit*

The even part of the eight-point transform which is calculated by the CSDA_E, similar to the four-point transform for H.264 and VC-1 standards. Two pipeline stages existwithin the architecture of CSDA_E.A four-input butterfly matrix circuit is executed by the first stage, and the

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 1, Issue 4, December 2014

second stage of CSDA_E then executes by using the proposed CSDAalgorithm to share hardware resources in variable standards.

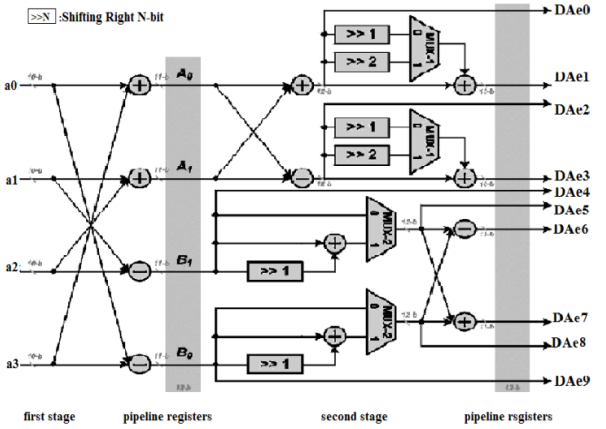


Fig. 3 Architecture of even part CSDA

c. Odd part Common Sharing Distributed Arithmetic circuit

The CSDA_O also has two pipeline stages which is similar to the CSDA_E,. Based on the proposed CSDA algorithm, then CSDA_O efficiently shares the hardware resources among the odd part of the eight-point transform and four-point transform for variable standards. It contains selection signals of multiplexers (MUXs) for different standards.

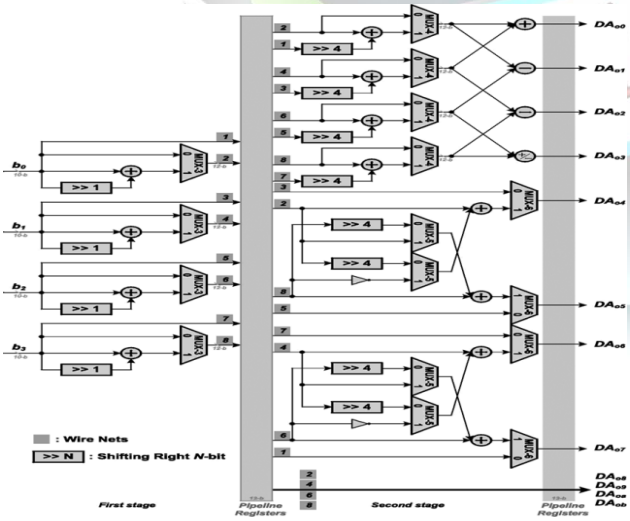


Fig. 4 Architecture of odd part CSDA

d. Error Compensation Adder Trees (ECAT)

The ECATs circuits can assuage truncation error efficiently in small area design when summing the nonzero data all together. Eight adder trees with error compensation (ECATs) are followed by the CSDA_E and CSDA_O, which add the nonzero CSDA coefficients with corresponding weight as the tree-like architectures. In eight output from ECAT directly given to permutation. Permutation relates to the act of rearranging, or permuting, all the members of a set into some sequence or order unlike combination, which are selections of some members of the set where order is discarded. It is used for encode output matrix. The Error-Compensated Circuit diminishes the truncation error for high accuracy design. Therefore, the hardware size and cost is reduced, and the speed is improved by using the ECAT architecture. ECAT architecture shown in fig 3.5

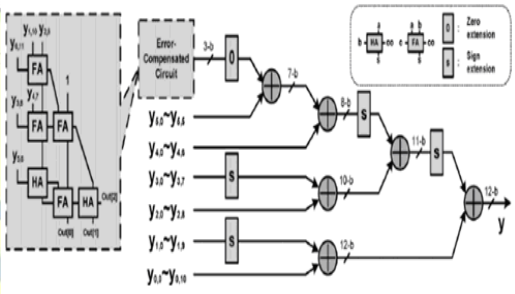


Fig. 5 ECAT Architecture

e. Transpose Memory (TMEM)

The TMEM is implemented using 64-word 12-bit dual-port registers and has a latency of 52 cycles. Based on the time scheduling strategy and result of the time scheduling strategy, the 1st-D and 2nd-D transforms are able to be computed simultaneously. The transposition memory is an 8×8 register array with the data width of 16 bits and is shown in Fig 3.6.

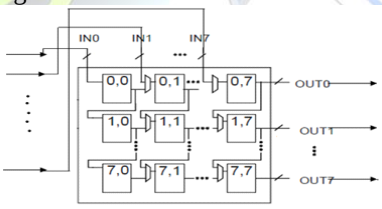


Fig. 6 Transpose memory

A. Proposed 2d CSDA-MST Core

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 1, Issue 4, December 2014

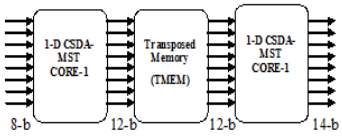


Fig. 3.7 Proposed 2-D CSDA-MST core

The proposed 2-D CSDA-MST core consists of two 1-D CSDA-MST core (Core-1 and Core-2) with a transposed memory (TMEM).TMEM is used to interchange the row and column. Core-1 and Core-2 are different in word length for each arithmetic, MUX, and register, and the TMEM is designed using sixty-four 12-bit registers, where the output data from Core-1 can be transposed and fed into Core-2. Each core has four pipeline stages: two in the even and two in the odd part CSDA circuit. The advantages of proposed system are low power consumption and less area. It supports MPEG-1/2/4,H.264,VC-1 transforms with a reduced hardware cost.

IV. A Mathematical Derivation Of Eight-Point And Four-Point Transforms

We introduces the proposed 2-D CSDA-MST core implementation. Neglecting the scaling factor, the one dimensional (1-D) eight-point transform can be defined as follows:

=D. (3)

where D is given in the equation (4)

D= (4)

The eight-point coefficient structures in H.264, MPEG- 1/2/4, and VC-1 standards are the same;the same mathematic derivation can be used for the eight-point transform for these standards. The 1-D eight point transform in (3) can be divided into even and odd two four-point transforms, We and Wo, as listed in (5) and (6),respectively based on the symmetry property.

We= (5)

=De.a

We= (6)

=Do.b

where

a= , b= (7)

The even part of the operation in (5) is the same as that of the four-point transformation of H.264 and VC-1. Moreover, the even part **W***e* can be further decomposed into even and odd parts as **W***ee* and **W***eo* .

Wee= = (8)

= Dee.A

We0= = (9)

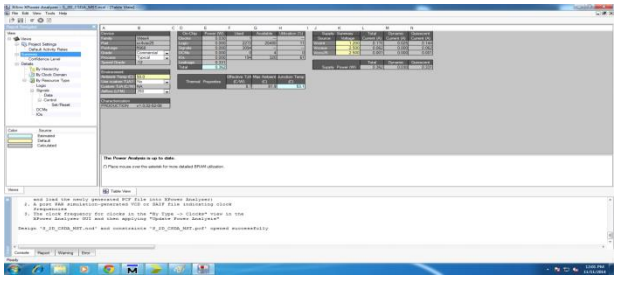
=Deo.B

where

A= , B= (10)

V. Simulation Results

The language which is used to build this architecture is Verilog HDL. The simulation of this MST core is done by using Model Sim.



ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 1, Issue 4, December 2014

Fig. 8 Output for 2-D Common Sharing Distributed arithmetic-MST architecture Power

The output for 2-Dimensional Common Sharing Distributed Arithmetic MST architecture power is 0.362W.

Device Utilization Summary			
Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slice Flip Flops	1,259	9,312	13%
Number of 4 input LUTs	2,097	9,312	22%
Logic Distribution			
Number of occupied Slices	1,312	4,656	28%
Number of Slices containing only related logic	1,312	1,312	100%
Number of Slices containing unrelated logic	0	1,312	0%
Total Number of 4 input LUTs	2,277	9,312	24%
Number used as logic	2,097		
Number used as a route-thru	180		
Number of bonded IOBs	194	232	83%
Number of GCLKs	1	24	4%
Total equivalent gate count for design	28,624		
Additional JTAG gate count for IOBs	9,312		

Fig. 9 Design summary for 2-D MST architecture using Common Sharing Distributed Arithmetic algorithm

The Output for 2-D Common Sharing Distributed Arithmetic- MST architecture gate count is 28.6 k.

A. Comparison Table

Table 5.1 Comparison table between existing transform Architecture with proposed transform architecture

Method used	**EXISTING SYSTEM**		**PROPOSED SYSTEM**
	New Distributed Arithmetic algorithm [8]	Adaptive block-size transform [9]	Common sharing Distributed Arithmetic Algorithm
Supporting standards	MPEG1/2/4 H.264	H.264	MPEG1/2/4 H.264 VC-1
Gate counts	39.8K	63.6K	28.6K
Power consumption	38.7mw	86.9mw	36.2 mW

The comparison table between the existing transform architecture with the proposed transform architecture is shown in the table 5.1. The comparison results shows that the proposed system of CSDA algorithm supports three standards as well as the power consumption and gate counts are reduced.

VI. CONCLUSION

The CSDA-MST core can achieve high performance, with low-cost VLSI design supporting MPEG-1/2/4, H.264, and VC-1 standards. This MST core can support MPEG-1/2/4, H.264 and VC-1 standards by using the selection signals of multiplexers. By using the proposed CSDA method, the gate counts and power consumption can be reduced efficiently. Common sharing distributed arithmetic (CSDA) combines factor sharing and distributed arithmetic sharing techniques, efficiently reducing the number of gate counts by high hardware-sharing capability.

ACKNOWLEDGEMENT

I would like to articulate my profound gratitude and indebtedness to Assistant Prof. Mr.B.Mohan M.E., for guiding and encouraging me in all aspects.

REFERENCES

[1]. August N.J., and Ha D.S.,(2004) 'Low power design of DCT and IDCT for low bit rate video codecs', *IEEE Trans. Multimedia*, vol. 6, no. 3, pp. 414–422.

[2]. Butts M.,(2007) 'Synchronization through communication in a massivelyparallel processor array', *IEEE Micro*, vol. 27, no. 5, pp. 32–41.

[3]. Chen J.W., Chen K.H., Wang J.S., and Guo J.I.,(2006) 'A performance aware IP core design for multi-mode transform coding using scalable-DA algorithm' in *Proc. IEEE Int. Symp. Circuits Syst.*,pp. 1904–1907.

[4]. Chen K.H., Guo J.I, and Wang J.S.,(2006) 'A high-performance direct 2D transform coding IP design for MPEG-4AVC/H.264', *IEEE Trans.Circuits Syst. Video Technol.*, vol. 16, no. 4, pp. 472–483.

[5]. Chen Y.H., Chang T.Y. and Li C.Y.,(2011) 'High throughput DA-based DCT with high accuracy error-compensated adder tree', *IEEE Trans. VeryLarge Scale Integr. (VLSI) Syst.*, vol. 19, no. 4, pp. 709–714.

[6]. Fan C.P. and Su G.A.,(2009) 'Fast algorithm and low-cost hardware- sharing design of multiple integer transforms for VC-1', *IEEE Trans. CircuitsSyst.*, vol. 56, no. 10, pp. 788–792.

[7]. Fan C.P. and Su G.A.,(2009) 'Efficient fast 1-D 8×8 inverse integer transform for VC-1 application', *IEEE Trans. Circuits Syst. Video Technol.*,vol. 19, no. 4, pp. 584–590.

[8]. Huang C.Y., L. F. Chen L.F., and Y. K. Lai Y.K.,(2008)'A high-speed 2D transform architecture with unique kernel for multi-standard video applications',in *Proc. IEEE Int. Symp. Circuits Syst.*, pp. 21–24.

[9]. Hwangbo W. and Kyung C.M.,(2010) 'A multitransform architecture for H.264/AVC high-profile coders', *IEEE Trans. Multimedia*, vol. 12, no.3,pp. 157–167.

[10]. Li.W.,(2001),'Overview of fine granularity scalability in MPEG-4 video Standard', *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 3,pp. 301–317.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 1, Issue 4, December 2014

[11]. Schwarz H., Marpe D., and Wiegand T.,(2007) 'Overview of the scalable video coding extension of the H.264/AVC standard', *IEEE Trans. Circuits Syst.Video Technol.*, vol. 17, no. 9, pp. 1103–1120.