International Journal of Power Control and Computation(IJPCSC)
Vol 6. No.1 – Jan-March 2014 Pp. 29-35
©gopalax Journals, Singapore
available at : www.ijcns.com
ISSN: 0976-268X

COMMON SHARING DISTRIBUTED ARITHMETIC METHOD WITH EIGHT PARALLEL COMPUTATION PATHS USED EFFECTIVE MULTI STANDARD TRANSFORM CORE SUPPORTING THE STANDARDS MPEG, H.264, VC-1

S.Sudha Mr.S.Nivash,
Department Of ECE, SRM University.
Email: sudhaselvaraj1991@gmail.com, nivashs09@gmail.com

ABSTRACT

This paper proposes a low-cost high-throughput multi standard transform (MST) core, which can support MPEG- 1/2/4 (8 × 8), H.264 (8 × 8, 4 × 4)[2][3], and VC-1 (8 × 8, 8 × 4, 4×8, 4×4) transforms[4]. Common sharing distributed arithmetic (CSDA) combines factor sharing and distributed arithmetic sharing techniques, efficiently reducing the number of adders for high hardware-sharing capability[5][7]. This achieves a reduction in adders in the proposed MST, compared with the direct implementation method. With eight parallel computation paths, the proposed MST core has an eightfold operation frequency throughput rate.The CSDA-MST[8][9] core thus achieves a high-throughput rate supporting multi standard transformations at low cost.

KEYWORDS: Common sharing distributed arithmetic (CSDA), discrete cosine transform (DCT), integer transform, multistandard transform (MST).

1. INTRODUCTION

H.264 and VC-1 are popular video compression standards.The VC-1 codec is designed to achieve state-of-the-art compressed video quality at bit rates that may range from very low to very high[6]. VC-1 Advanced Profile is also transport and container independent. This provides even greater flexibility for device manufacturers and content services.

Innovations:

VC-1 includes a number of innovations that enable it to produce high quality content. This section provides brief descriptions of some of these features.

2. ADAPTIVE BLOCK SIZE TRANSFORM:

Traditionally, 8 × 8 transforms have been used for image and video coding. However, there is evidence to suggest that 4 × 4 transforms can reduce ringing artifacts at edges and discontinuities. [6]VC-1 is capable of coding an 8 × 8 block using either an 8 × 8 transform, two 8 × 4 transforms, two 4 × 8transforms, or four 4 × 4 transforms. This feature enables coding that takes advantage of the different transform sizes as needed for optimal image quality.

16-Bit Transforms:

In order to minimize the computational complexity of the decoder, VC-1 uses 16-bit transforms. This also has the advantage of easy implementation on the large amount of digital signal processing (DSP) [2][11]hardware built with 16-bit processors. Among the constraints put on VC-1 transforms is the requirement that the 16-bit values used produce results that can fit in 16 bits. The constraints on transforms ensure that decoding is as efficient as possible on a wide range of devices.

Motion compensation:

Motion compensation is the process of generating a prediction of a video frame by displacing the reference frame. Typically, the prediction is formed for a block (an 8 × 8 pixel tile) or a macro block (a 16 × 16 pixel tile) of data[9][10]. The displacement of

International Journal of Power Control and Computation(IJPCSC)
Vol 6. No.1 – Jan-March 2014 Pp. 29-35
©gopalax Journals, Singapore
available at : www.ijcns.com
ISSN: 0976-268X

data due to motion is defined by a motion vector, which captures the shift along both the x- and y-axes. The efficiency of the codec is affected by the size of the predicted block, the granularity of sub-pixel data that can be captured, and the type of filter used for generating sub-pixel predictors. [11][12]VC-1 uses 16 × 16 blocks for prediction, with the ability to generate mixed frames of 16 × 16 and 8 × 8 blocks. The finest granularity of sub-pixel information supported by VC-1 is 1/4 pixel. Two sets of filters are used by VC-1 for motion compensation. The first is an approximate bicubic filter with four taps. The second is a bilinear filter with two taps.

VC-1 combines the motion vector settings defined by the block size, sub-pixel granularity, and filter type into modes. The result is four motion compensation modes that suit a range of different situations[12][13]. This classification of settings into modes also helps compact decoder implementations.

Loop Filtering:

VC-1 uses an in-loop de-blocking filter that attempts to remove block-boundary discontinuities introduced by quantization errors in interpolated frames. These discontinuities can cause visible artifacts in the decompressed video frames and can impact the quality of the frame as a predictor for future interpolated frames. The loop filter takes into account the adaptive block size transforms. The filter is also optimized to reduce the number of operations required.

Interlace Coding:

Interlaced video content is widely used in television broadcasting. When encoding interlaced content, the VC-1 codec can take advantage of the characteristics of interlaced [4][5]frames to improve compression. This is achieved by using data from both fields to predict motion compensation in interpolated frames.

Advanced B Frame Coding:

A bi-directional or B frame is a frame that is interpolated from data both in previous and subsequent frames. B frames are distinct from I frames (also called key frames), which are encoded without reference to other frames. B frames are also distinct from P frames, which are interpolated from previous frames only. VC-1 includes several optimizations that make B frames more efficient.

Fading compensation:

Due to the nature of compression that uses motion compensation, encoding of video frames that contain fades to or from black is very inefficient. With a uniform fade, every macro block needs adjustments to luminance. VC-1 includes fading compensation, which detects fades and uses alternate methods to adjust luminance. This feature improves compression efficiency for sequences with fading and other global illumination changes.

Differential quantization:

Differential quantization, or dquant, is an encoding method in which multiple quantization steps are used within a single frame. Rather than quantize the entire frame with a single quantization level, macro blocks are identified within the frame that might benefit from lower quantization levels and greater number of preserved AC coefficients. Such macro blocks are then encoded at lower quantization levels than the one used for the remaining macro blocks in the frame[5]. The simplest and typically most efficient form of differential quantization involves only two quantizer levels (bi-level dquant), but VC-1 supports multiple levels, too.

2. EXISTING SYSTEM

A significant amount of research has been conducted to efficiently combine and implement the transform units for multiple codec's[5][7]. On the other hand little research is focused on the implementation of multi-quantized unit. Among the multiple-transform units, a unified Inverse Discrete Cosine Transform (IDCT) architecture to support five standards (such as, AVS, H.264, VC-1, MPEG-2/4 and JPEG) is presented. The authors in offer an area efficient architecture to perform a DCT-based transform for JPEG, MPEG-4, VC-1 and H.264 using delta mapping. The design in is an IDCT and IQ circuit for H.264, MPEG-4 and VC-1. The MJPEG standard defines quantization as the division operation of the DCT coefficient coming from the transform unit by the corresponding Q value (specified by the quantization matrix). MJPEG allows specification of Q-matrices that facilitates the allocation of more bits for the representation of coefficients which are visually more significant.

The intercommunications between the video devices using different standards are so much

International Journal of Power Control and Computation(IJPCSC)
Vol 6. No.1 – Jan-March 2014 Pp. 29-35
©gopalax Journals, Singapore
available at : www.ijcns.com
ISSN: 0976-268X

inconvenient,[8][10] thus video codec supporting multiple standards are more useful and more attractive. In this brief, low cost very large scale integration (VLSI) architecture is designed for multi standard inverse Discrete Cosine transform. It is used in multi standard decoder of MPEG-2, MPEG-4 ASP, and VC-1[4][6] .Two circuit share strategies, factor share (FS) an adder share (AS) are applied to the inverse transform architecture for saving its circuit resource. Pipelined stages are used in this Multistandard inverse transform to increase the operational speed.

The possibility to employ hierarchical prediction structures for providing temporal scalability with several layers while improving the coding efficiency and increasing the effectiveness of quality and spatial scalable coding. New methods for inter-layer prediction of motion and residual improving the coding efficiency of spatial scalable and quality scalable coding. The concept of key pictures for efficiently controlling the drift for packet-based quality scalable coding with hierarchical prediction structures. Single motion compensation loop decoding for spatial and quality scalable coding providing a decoder complexity close to that of single-layer coding.

3. PROPOSED SYSTEM

The proposed CSDA algorithm combines the FS and DA methods. By expanding the coefficients matrix at the bit level, the FS method first shares the same factor in each coefficient; the DA method is then applied to share the same combination of the input among each coefficient position. An example of the proposed CSDA algorithm in a matrix inner product is as follows the proposed CSDA combines the FS and DA methods[1][2]. The FS method is adopted first to identify the factors that can achieve higher capability in hardware resource sharing, where the hardware resource in defined as the number of adder usage. Next, the DA method is used to find the shared coefficient based on the results of the FS method. The adder-tree circuits will be followed by the proposed CSDA circuit.[4][8] Thus, the CSDA method aims to reduce the nonzero elements to as few as possible. The CSDA shared coefficient is used for estimating and comparing thereafter the number of adders in the CSDA loop.

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). 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. The registers in TMEM are designed the same way as the architecture. Each core has four pipeline stages: two in the even and odd part CSDA circuit, and two in ECATs. Therefore, the proposed 2-D CSDA-MST core has a latency of 16 clock cycles (= 4 + 8 + 4), and the TMEM executes transposed operation after 12 clock cycles (= 4 + 8) when 8 pixels are input.

The CSDA-MST core can achieve high performance, with a high throughput rate and low-cost VLSI design, supporting MPEG-1/2/4, H.264, and VC-1 MSTs. By using the proposed CSDA method, the number of adders and MUXs in the MST core can be saved efficiently. Measured results show the CSDA-MST core with a throughput rate of 1.28 G-pels/s, which can support (4928 × 2048@24 Hz) digital cinema format with only 30 k logic gates. Because visual media technology has advanced rapidly, this approach will help meet the rising high-resolution specifications and future needs as well.

4. MODULES

- 1-D Common sharing distributed arithmetic-MST
- Even part Common sharing distributed arithmetic circuit
- Odd part Common sharing distributed arithmetic circuit
- 2-D Common sharing distributed arithmetic -MST core

5. MODULE DESCRIPTION

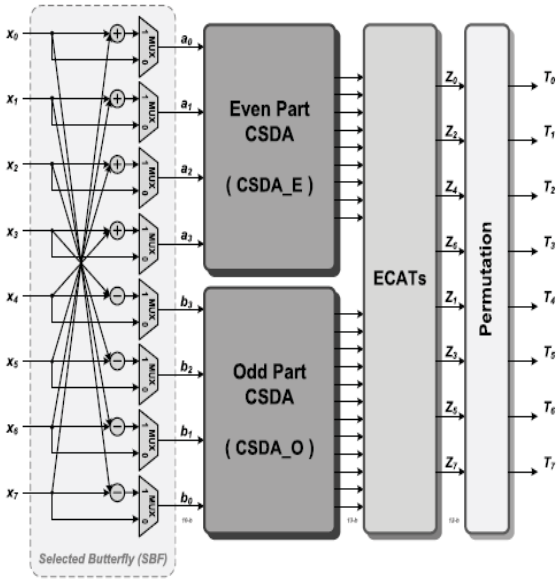
1-D Common Sharing Distributed Arithmetic-Mst:

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. To adopt the searching flow, software code will help to do the iterative searching loops by setting a constraint with minimum nonzero elements. In this paper, the constraint of minimum nonzero elements is set to be five. After software searching, the coefficients of the CSD expression, where 1 indicates −1. Note that the choice of shared coefficient is obtained by some constraints. Thus, the chosen CSDA coefficient is not a global optimal solution. It is just a local or suboptimal solution. Besides, the CSD codes are not the optimal expression, which have minimal nonzero bits.

International Journal of Power Control and Computation(IJPCSC)
Vol 6. No.1 – Jan-March 2014 Pp. 29-35

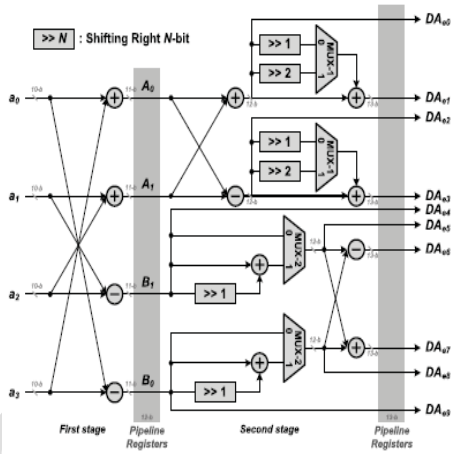
available at : www.ijcns.com
ISSN: 0976-268X

However, the chosen coefficients of CSD expression can achieve high sharing capability for arithmetic resources by using the proposed CSDA algorithm. More information about CSDA coefficients for MPEG-1/2/4, H.264, and VC-1 transforms.



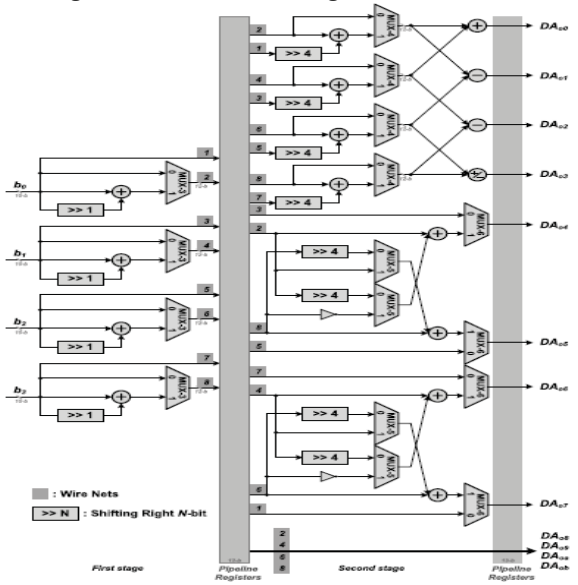
Even part common sharing distributed arithmetic circuit:

The SBF module executes for the eight-point transform and bypasses 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[10][11]. The CSDA_E calculates the even part of the eight-point transform, similar to the four-point Trans form for H.264 and VC-1 standards. Within the architecture of CSDA_E, two pipeline stages exist (12-bit and 13-bit). The first stage executes as a four-input butterfly matrix circuit, and the second stage of CSDA_E then executes by using the proposed CSDA algorithm to share hardware resources in variable standards.



6. ODD PART COMMON SHARING DISTRIBUTED ARITHMETIC CIRCUIT

Similar to the CSDA_E, the CSDA_O also has two pipeline stages. Based on the proposed CSDA algorithm, the 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. 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[11][12]. The ECATs circuits can alleviate truncation error efficiently in small area design when summing the nonzero data all together.

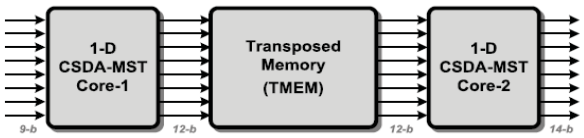


International Journal of Power Control and Computation(IJPCSC)
Vol 6. No.1 – Jan-March 2014 Pp. 29-35

available at : www.ijcns.com
ISSN: 0976-268X

7. 2-D COMMON SHARING DISTRIBUTED ARITHMETIC -MST CORE

This section provides a discussion of the hardware resources and system accuracy for the proposed 2-D CSDA-MST core and also presents a comparison with previous works. Finally, the characteristics of the implementation into a chip are described.



8. COMMON SHARING DISTRIBUTED ARITHMETIC

Distributed Arithmetic (DA) has been wildly adopted for its computational efficiency in many digital signal processing applications such as DCT (Discrete Cosine Transform), DFT (Discrete Fourier Transform), FIR (Finite Impulse Response), and DHT (Discrete Hartley Transform)[4][7]. These applications involve the computation of inner products between two vectors, one of which is a constant. The general method to generate the products is to use a MAC (multiply and accumulate) unit, which is fast but has a high cost in the case of long-length inner-products.

The idea behind the conventional DA, called ROM-based, is to replace multiplication operations by pre-computing all possible values and storing these in a ROM. According to the ROM based DA can reduce the circuit size by 50-80% on average. Custom reconfigurable technology emerged to satisfy the simultaneous demand for flexibility and efficiency. Custom/domain-specific reconfigurable arrays can be programmed to adapt for different applications, so the efficiency of the hardware and flexibility of the whole system is improved. Earlier works, such as, show good performance in area, power consumption and speed. Since a domain-specific reconfigurable architecture targets few application fields, it achieves better performance than a general purpose FPGA device.

9. PROPOSED 2-D CSDA-MST CORE DESIGN

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

$$\begin{pmatrix} Z0 \\ Z1 \\ Z2 \\ Z3 \\ Z4 \\ Z5 \\ Z6 \\ Z7 \end{pmatrix} = C \cdot \begin{pmatrix} x0 \\ x1 \\ x2 \\ x3 \\ x4 \\ x5 \\ x6 \\ x7 \end{pmatrix}$$

Where

$$C = \begin{pmatrix} C4 & C4 & C4 & C4 & C4 & C4 & C4 & C4 \\ C1 & C3 & C5 & C7 & -C7 & -C5 & -C3 & -C1 \\ C2 & C6 & -C6 & -C2 & -C2 & -C6 & C6 & C2 \\ C3 & -C7 & -C1 & -C5 & C5 & C1 & C7 & -C3 \\ C4 & -C4 & -C4 & C4 & C4 & -C4 & -C4 & C4 \\ C5 & -C1 & C7 & C3 & -C3 & -C7 & C1 & -C5 \\ C6 & -C2 & C2 & -C6 & -C6 & C2 & -C2 & C6 \\ C7 & -C3 & C3 & -C1 & C1 & -C3 & C5 & -C7 \end{pmatrix}$$

Because the eight-point coefficient structures in MPEG- 1/2/4, H.264, and VC-1 standards are the same, the eight-point transform for these standards can use the same mathematic derivation. According to the symmetry property, the 1-D eight point transform in (8) can be divided into even and odd two four-point transforms, **Z***e* and **Z***o*, as listed in (9) and (10), respectively

$$Ze = \begin{pmatrix} Z0 \\ Z2 \\ Z4 \\ Z6 \end{pmatrix} \begin{pmatrix} C4 & C4 & C4 & C4 \\ C2 & C6 & -C6 & -C2 \\ C4 & -C4 & -C4 & C4 \\ C6 & -C2 & C2 & -C6 \end{pmatrix} \begin{pmatrix} a0 \\ a1 \\ a2 \\ a3 \end{pmatrix}$$

$$= Ce.a$$

$$Zo = \begin{pmatrix} Z1 \\ Z3 \\ Z5 \\ Z7 \end{pmatrix} \begin{pmatrix} C1 & C3 & C5 & C7 \\ C3 & -C7 & -C1 & -C5 \\ C5 & -C1 & C7 & C3 \\ C7 & -C3 & C3 & -C1 \end{pmatrix} \begin{pmatrix} b0 \\ b1 \\ b2 \\ b3 \end{pmatrix}$$

$$= Co.b$$

International Journal of Power Control and Computation(IJPCSC)
Vol 6. No.1 – Jan-March 2014 Pp. 29-35
©gopalax Journals, Singapore
available at : www.ijcns.com
ISSN: 0976-268X

Where

$$a = \begin{pmatrix} X0+X7 \\ X1+X6 \\ X2+X5 \\ X3+X4 \end{pmatrix}, \quad b = \begin{pmatrix} x0-x7 \\ x1-x6 \\ x2-x5 \\ x3-x4 \end{pmatrix}$$

The even part of the operation in (10) is the same as that of the four-point H.264 and VC-1 transformations. Moreover, the even part **Z***e* can be further decomposed into even and odd parts: **Z***ee* and **Z***eo*

$$Zee = \begin{pmatrix} Z0 \\ Z4 \end{pmatrix} = \begin{pmatrix} C4 & C4 \\ C4 & -C4 \end{pmatrix} \begin{pmatrix} A0 \\ A1 \end{pmatrix}$$

$$= Cee.A$$

$$Zeo = \begin{pmatrix} Z2 \\ Z6 \end{pmatrix} = \begin{pmatrix} C2 & C6 \\ C6 & -C2 \end{pmatrix} \begin{pmatrix} B0 \\ B1 \end{pmatrix}$$

$$= Ceo.B$$

Where

$$A = \begin{pmatrix} a0+a3 \\ a1+a2 \end{pmatrix} \qquad B = \begin{pmatrix} a0 - a3 \\ a1 - a2 \end{pmatrix}$$

Applications:

- Video and image applications

10. CONCLUSION

The CSDA-MST core can achieve high performance, with a high throughput rate and low-cost VLSI design, supporting MPEG-1/2/4, H.264, and VC-1 MSTs. By using the proposed CSDA method, the number of adders and MUXs in the MST core can be saved efficiently. Measured results show the CSDA-MST core with a throughput rate of 1.28 G-pels/s, which can support (4928 × 2048@24 Hz) digital cinema format with only 30 k logic gates. Because visual media technology has advanced rapidly, this approach will help meet the rising high-resolution specifications and future needs as well.

FUTURE ENHANCEMENT

I will modify the proposed system by reducing the Area of converting one dimensional to two dimensional core designs.And also will reduce the total number of adders by using the CSDA Common Sharing Distributed Arithmetic method in the proposed system.

REFERENCES

[1] S. Uramoto, Y. Inoue, A. Takabatake, J. Takeda, Y. Yamashita, H. Terane, and M. Yoshimoto, "A 100-MHz 2-D discrete cosine transform core processor," IEEE J. Solid-State Circuits, vol. 27, no. 4, pp. 492–499, Apr. 1992.

[2] S. Yu and E. E. Swartzlander, "DCT implementation with distrib- uted arithmetic," IEEE Trans. Comput., vol. 50, no. 9, pp. 985–991, Sep. 2001.

[3] A. M. Shams, A. Chidanandan, W. Pan, and M. A. Bayoumi, "NEDA: A low-power high-performance DCT architecture," IEEE Trans. Signal Process., vol. 54, no. 3, pp. 955–964, Mar. 2006.

[4] C. Peng, X. Cao, D. Yu, and X. Zhang, "A 250 MHz optimized distributed architecture of 2D 8×8 DCT," inProc. 7th Int. Conf. ASIC, Oct. 2007, pp. 189–192.

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

[6] Y. H. Chen, T. Y. Chang, and C. Y. Li, "High throughput DA-based DCT with high accuracy error-compensated adder tree," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 19, no. 4, pp. 709–714, Apr. 2011.

[7] Y. H. Chen, T. Y. Chang, and C. Y. Li, "A high performance video transform engine by using space-time scheduling strategy," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 20, no. 4, pp. 655–664, Apr. 2012.

[8] Y. K. Lai and Y. F. Lai, "A reconfigurable IDCT architecture for universal video decoders," IEEETrans.

International Journal of Power Control and Computation(IJPCSC)
Vol 6. No.1 – Jan-March 2014 Pp. 29-35
©gopalax Journals, Singapore
available at : www.ijcns.com
ISSN: 0976-268X

Consum. Electron., vol. 56, no. 3, pp. 1872–1879, Aug. 2010.

[9] H. Chang, S. Kim, S. Lee, and K. Cho, "Design of area-efficient unified transform circuit for multi-standard video decoder," in Proc. IEEE Int. SoC Design Conf., Nov. 2009, pp. 369–372.

[10] S. Lee and K. Cho, "Circuit implementation for transform and quanti- zation operations of H.264/MPEG-4/VC-1 video decoder," in Proc. Int. Conf. Design Technol. Integr. Syst. Nanosc., Sep. 2007, pp. 102–107.

[11] S. Lee and K. Cho, "Architecture of transform circuit for video decoder supporting multiple standards," Electron. Lett., vol. 44, no. 4, pp. 274–275, Feb. 2008.

[12] H. Qi, Q. Huang, and W. Gao, "A low-cost very large scale integration architecture for multistandard inverse transform," IEEE Trans. Circuits Syst., vol. 57, no. 7, pp. 551–555, Jul. 2010.