Hardware Implementation of Discrete /Inverse Discrete Cosine Transform Using Redundant Number System CORDIC Processors

Jayashri M. Rudagi, Dr. (Smt.) Shaila Subbaraman.
Associate Professor, KLE College of Engineering & Technology, Chikodi, Karnataka, India.
Walchand College of Engineering (WCE), Sangli, Maharashtra, India.

Abstract

Rapid enhancement in multimedia service running on portable application has forced the development of high quality and low power implementation of complex signal processing algorithms. Image and video processing are typical application of multimedia system. Image compression and decompression using DCT/IDCT are the two algorithms commonly used in MPEG standard of image processing. For an image of N x N size, DCT/IDCT would require N^4 multiplications and increases complexity. To reduce the complexity and increase the speed many algorithms have been proposed. This paper presents the implementation of DCT/IDCT using Loeffler Algorithm, radix-2, radix-4 and radix-8 CORDIC processors. The implementation of high radices results into reduction in the computations thus making it faster and power efficient. By using higher radix CORDIC processor 50% advantage for speed enhancement and 80% power reduction have been observed respectively. The results also indicate that radix-8 CORDIC Processor implementing DCT/IDCT (CRDCT/IDCT) in comparison with CRDCT/IDCT using radix-2 and radix-4 yields reduction in critical path delay by 47% and 28% with a sacrifice on area in terms of more hardware by 12% and 30% respectively.

Keywords: Radix-2, Radix-4, Radix-8, DCT, IDCT, CORDIC

INTRODUCTION

Digital Signal Processing has received a wide spread importance and acceptance in last two to three decades for processing the information. The basic transform used in DSP is a Fourier transform with extension to Fast Fourier transform which treats the signal as complex quantity. In many practical applications especially with reference to audio and video processing, one can deal with only the real part of the signal and hence use Discrete Cosine Transform (DCT) to convert time domain signal to frequency domain signal [1]. The established image compression/decompression standards viz. MPEG, JPEG, H.261 makes use of DCT/IDCT in their implementation. The transforms of the data from spatial domain to spatial frequency domain takes place in DCT. It decorrelates image data, which are usually highly correlated for small areas of an image. Generally a lot of redundant information is available in these heavily correlated data samples which can be eliminated while a small amount of un-correlated information can be retained to represent the data more efficiently. For a (N x N) 2-D data, straight forward implementation of DCT/IDCT requires N^4 multiplications and is not the preferred way of implementation. Most of the DCT/IDCT implementations use fast algorithms which reduce the number of multiplication to O(N^2). These fast algorithms not only improve the speed of DCT/IDCT but also reduce the number of computations required.

The direct implementation of DCT/IDCT equations is computationally expensive. Most implementations employ fast algorithms that reduce the computational cost. Fast algorithms can be classified into two broad categories: a) row/column approach and b) direct fast 2D/Chen approaches. In both approaches 1D-DCT is the basic building block. In the former approach, 1D-DCT/IDCT of each row of 8x8 input data is taken which are then transposed to form the column of 2D-DCT/IDCT [2]. The row/column approach results in very simple and regular implementations, but it is not computationally efficient. It also suffers from the drawback of increased latency due to the transposition operation. Row-column approach of computing 2-D DCT/IDCT has been reported to be faster using Chen algorithm [3, 4] and other similar algorithms [5]. The details of these faster algorithms are available in the references mentioned. This fast 2-D DCT/IDCT Chen approach requires computation of only half the multiplications compared to that by row/column approach. The number of multiplications required using Chen algorithm for computing an eight point 1D-DCT/IDCT is only 16 as against 256 multiplications for fast 2D-DCT/IDCT approach while 1024 multiplications using row-column approach. The major concern in the DCT/IDCT architecture is method of implementing the multipliers, because of the large number of multiplication operations required in DCT/IDCT implementation. Direct approach makes use of more number of multipliers and adders. Even the matrix operation makes use of multipliers, where it occupies more area. CORDIC, which is multiplier less operates at faster rate and power consumption is also less [6, 7, 8]. The conventional binary number system for the representation of operands of the computation to be implemented using CORDIC is not effective from delay point of view. Generally in any digital system performing mathematical functions, the number of adders and the number of bits of operands i.e. word size determines the complexity of that system. The literature review indicates that the redundant number representation which is one of the signed number representation of operands results in reducing the delay of the computation besides offering an advantage of making the delay independent of the width of the operands in terms of number of bits. Algorithm and architecture of COR2DCT, COR4DCT and COR8DCT
has been covered in section II. Section III discusses the simulation and synthesis results. Section IV concludes the paper.

ALGORITHMS AND ARCHITECTURE

The DCT typically operates on 8x8 pixel block of image data. For an N x N blocks of samples x (m, n) the two dimensional 2D- DCT is formulated as below [1].

\[ z(k, l) = \frac{1}{N} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} x(m, n) \cos \left[ (2m + 1)\pi k / 2N \right] \cos \left[ (2n + 1)\pi l / 2N \right] \]

(1)

Since for the image block size is considered is 8x8 i.e. N=8. Therefore DCT/IDCT equation becomes

\[ z(k, l) = \frac{1}{4} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} x(m, n) \cos \left[ (2m + 1)\pi k / 16 \right] \cos \left[ (2n + 1)\pi l / 16 \right] \]

(2)

The Loeffler algorithm which makes use of above equation to calculate DCT/IDCT needs 11 multiplication and 27 additions. The total area occupied and power consumption is also very large.

When using the CORDIC to replace the multiplications of 8 point DCT, the angles \( \theta \) are fixed. Therefore the unnecessary iterations can be skipped without losing accuracy. The following Table 1 gives the detailed number of iterations and compensation for the CORDIC based algorithm.

<table>
<thead>
<tr>
<th>Processors</th>
<th>Angle</th>
<th>CORDIC Iterations(( \sigma, i ))</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>CORDIC 1</td>
<td>( \pi / 4 )</td>
<td>-1,0</td>
</tr>
<tr>
<td>CORDIC 2</td>
<td>( 3\pi / 8 )</td>
<td>1,0</td>
</tr>
<tr>
<td>CORDIC 3,6</td>
<td>( 7\pi / 16 )</td>
<td>1,0</td>
</tr>
<tr>
<td>CORDIC 4,5</td>
<td>( 3\pi / 16 )</td>
<td>1,1</td>
</tr>
</tbody>
</table>

Recalling from the equation (2), 8 point DCT is represented by the following equation

\[ z_n = \frac{1}{2} \alpha \left( n \right) \sum_{i=0}^{7} x_i \cos \left( \frac{(2i+1)n\pi}{16} \right) \]

(4)

Here \( n = 0, 1, 2, ..., 7 \). For image processing applications, a tensor product of two 1D- DCT are used to obtain 2D-DCT.

Figure 1 shows the top level architecture of COR2DCT. The design mainly consists of CORDIC processor units, transpose memory to transpose and to store intermediate values of DCT/IDCT operation and it also includes a control logic block to control and supervise the overall operation of processor, radix-2 adder module and butterfly units. The DCT/IDCT using CORDIC can be split into two parts, one is butterfly addition and subtraction and another is CORDIC computation and accumulation. The butterfly operator performs the conventional butterfly shuffling, addition and subtraction. The result of two additions continues performing the butterfly computations, the result of two subtractions are sent to CORDIC unit, which computes the DCT coefficients. The CORDIC array performs the fixed angle rotation in the DCT algorithm.

The IDCT operation is just the reverse process of DCT. The following architecture depicts the COR2IDCT architecture.

Table 1: CORDIC iterations

Figure 1: Top level Architecture of COR2DCT Processor
COR2IDCT is higher than that for other algorithms.

<table>
<thead>
<tr>
<th>Processors/Parameters</th>
<th>PSNR(dB) (Leena)</th>
<th>PSNR(dB) (Grass)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCT/IDCT (Loeffler Algorithm)</td>
<td>40.2</td>
<td>39.8</td>
</tr>
<tr>
<td>COR2DCT/IDCT</td>
<td>42.1</td>
<td>41.9</td>
</tr>
<tr>
<td>COR4DCT/IDCT</td>
<td>42.8</td>
<td>42.2</td>
</tr>
<tr>
<td>COR8DCT/IDCT</td>
<td>43.6</td>
<td>42.9</td>
</tr>
</tbody>
</table>

**Results and Discussion:**

The designs were coded in Verilog and synthesized using Cadence RTL Encounter. Standard benchmark images Leena (Low Frequency Image) of size 512 x 512 and Grass (High Frequency) of size 256 x 256 respectively taken for study purpose. These images are obtained from Standard Public Image database (SIP). The quantization factor of 50 is considered. Both images are compressed and decompressed using Loeffler algorithm and also using developed CORDIC algorithms namely COR2DCT/IDCT, COR4DCT/IDCT and COR8DCT/IDCT. PSNR and CR values of these reconstructed images are presented in the Table 2. The image reconstructed using COR8DCT/IDCT has better visual quality compared to that of other. Also the PSNR value for radix-8
Table 5: Summary of Synthesis Results of IDCT

<table>
<thead>
<tr>
<th>Processors/ Parameters</th>
<th>Tool Generated Parameters</th>
<th>Derived Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Delay (ns)</td>
<td>Power (mW)</td>
</tr>
<tr>
<td>IDCT (Loeffler Algorithm)</td>
<td>8.2</td>
<td>21.3832</td>
</tr>
<tr>
<td>COR2IDCT</td>
<td>7.7</td>
<td>12.01</td>
</tr>
<tr>
<td>COR4IDCT</td>
<td>5.9</td>
<td>13.76</td>
</tr>
<tr>
<td>COR8IDCT</td>
<td>4.1</td>
<td>14.01</td>
</tr>
</tbody>
</table>

From the above table, it can be observed that COR8IDCT processor uses 12% and 30% more hardware compared to that of COR4IDCT and COR2IDCT respectively. But when it is compared with DCT which makes use of Loeffler algorithm there is 50% reduction in hardware. When delay is considered, there is a 50%, 47%, 28% reduction in critical path delay compared to that of DCT/IDCT using Loeffler algorithm, COR2DCT and COR4DCT respectively. Total power dissipated by CORDIC DCT/IDCT cores is about 37% of the Loeffler DCT core with marginal increase from radix-2 to radix-8 CORDIC core. This is a very striking result that CORDIC algorithm coupled with radix representation leads to high speed and low power VLSI designs.

From the design, it can be concluded that reduction in the delay comes at the cost of area i.e. as the radix of the CORDIC processor increases the hardware complexity also increases. There is 62%, 25% and 21% reduction in ADP. So it can be concluded that AD product can be used to find the point at which least amount of area is sacrificed for the improvement in speed.

Figure 3: Comparison of (a) PDP and (b) EDP

Figure 4: Comparison of (a) Area and (b) ADP
CONCLUSIONS

This paper presents the algorithm and architectures of Loeffler, COR2DCT, COR4DCT and COR8DCT Processors, which are intended for low power and high speed DCT/IDCT applications. The most efficient architecture for DCT/IDCT is using higher radix CORDIC processor i.e. radix-8, which has higher throughput, high PSNR, lower PDP and EDP compared to all other architectures, but which has extra hardware overhead. The average PDP value of COR8DCT/IDCT is less by 68%, 40% and 26% and also energy saving of 85%, 68% and 49% compared to Loeffler, COR2DCT/IDCT and COR4DCT/IDCT architectures respectively. Even with increase in area of higher radix processors the ADP value decreases.

REFERENCES