Implementation of Two Dimensional Discrete Cosine Transform Using Field Programmable Gate Array

PUNYA THITIMAJSIMA
Department of Telecommunications Engineering, Faculty of Engineering
King Mongkut’s Institute of Technology Ladkrabang, Bangkok 10520, Thailand.

THAILAND

Abstract: This paper presents an architectural design of two-dimensional discrete cosine transform for implementation using FPGA. The technique consists of separating two-dimension transform into a cascade of one-dimension transform. Then, the number of multiplications is reduced by taking into account the redundancy of calculations. A hardware architecture of such an algorithm is proposed. It requires only 12 multipliers for one dimensional transform. A software simulation of this architecture comparing to the direct transform illustrates the usefulness of the proposed design.

Key-Words: Architectural Design, Two-Dimensional Discrete Cosine Transform (DCT), Image Processing, Field Programmable Gate Array (FPGA).

1 Introduction

The discrete cosine transform (DCT) is widely used for image processing and video signal coding. It also is the most important part of image compression standards such as JPEG, MPEG, and H.261. In videotelephony applications, for instance, it is the most computationally intensive part of both the encoder and decoder. A direct implementation of a two-dimensional (2D) DCT of 8x8-pixel block requires an immense amount of computation, typically 4096 multiplications per block. This 2D transformation, however, can be decomposed into 8x1 1D DCTs across the rows, followed by 8 1D DCTs down the columns. Fig. 1. shows the architecture of such a system.

2 Two Dimensional Discrete Cosine Transform Implementation

The 2D DCT of 8x8-pixel image block can be defined as:

\[
Y(k_1,k_2) = \frac{1}{2} \left[ C(k_1) C(k_2) \sum_{n_1=-48}^{47} \sum_{n_2=-48}^{47} x(n_1,n_2) \cos \left( \frac{(2n_1+1)k_1\pi}{16} \right) \cos \left( \frac{(2n_2+1)k_2\pi}{16} \right) \right]
\]

where \( x(n_1,n_2) \) is 8x8 pixel input image data. A direct calculation of this equation requires 4096 multiplications per block. This 2D transformation, however, can be decomposed into 8x1 1D DCTs across the rows, followed by 8 1D DCTs down the columns. Fig. 1. shows the architecture of such a system.

3 Architectural Design of 8x1 1D DCT

The following will describe the design of a single 8x1 1D DCT operation, which is defined by:

\[
y(k) = \sum_{n=-48}^{47} x(n) \cdot \cos \left( \frac{2(n+1) \cdot k \cdot \pi}{16} \right)
\]

This can be more easily explained by writing in a matrix form as:

\[
Y = C \cdot X
\]
changing the redundancy of multiplication matrix product technique \[7\] is furthermore applied to multiplications required from 48 to 20 at the

\[
\begin{bmatrix}
\begin{array}{ccccccc}
   C_4 & C_3 & C_2 & C_1 & C_0 & C_0 & C_0 \\
   C_3 & C_2 & C_1 & C_0 & C_0 & C_0 & C_0 \\
   C_2 & C_1 & C_0 & C_0 & C_0 & C_0 & C_0 \\
   C_1 & C_0 & C_0 & C_0 & C_0 & C_0 & C_0 \\
   C_0 & C_0 & C_0 & C_0 & C_0 & C_0 & C_0 \\
   C_0 & C_0 & C_0 & C_0 & C_0 & C_0 & C_0 \\
   C_0 & C_0 & C_0 & C_0 & C_0 & C_0 & C_0 \\
   C_0 & C_0 & C_0 & C_0 & C_0 & C_0 & C_0
\end{array}
\end{bmatrix}
\]

\(x(0)\)

\(x(1)\)

\(x(2)\)

\(x(3)\)

\(x(4)\)

\(x(5)\)

\(x(6)\)

\(x(7)\)

when the coefficients of the transformation matrix

\[c_k = \cos(2\pi k / 32)\]

By manipulating the terms in the input matrix, This equation can be rearranged in a sparse matrix form as:

\[
\begin{bmatrix}
\begin{array}{ccccccc}
   0 & 0 & 0 & 0 & 0 & 0 & 0 \\
   0 & -C_2 & C_2 & 0 & 0 & 0 & 0 \\
   0 & 0 & C_1 & C_1 & 0 & 0 & 0 \\
   0 & 0 & 0 & C_1 & C_1 & 0 & 0 \\
   0 & 0 & 0 & 0 & C_1 & C_1 & 0 \\
   0 & 0 & 0 & 0 & 0 & C_1 & C_1 \\
   0 & 0 & 0 & 0 & 0 & 0 & C_1 \\
   0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}
\end{bmatrix}
\begin{bmatrix}
   w(0) \\
   w(1) \\
   w(2) \\
   w(3) \\
   w(4) \\
   w(5) \\
   w(6) \\
   w(7)
\end{bmatrix}
\]

\(x(0) + x(7) + x(3) + x(4) + x(1) + x(2) + x(5) + x(6)\)

\(x(0) + x(7) + x(3) + x(4) - x(1) - x(2) - x(5) - x(6)\)

\(x(0) + x(7) - x(3) - x(4)\)

\(x(1) + x(6) - x(2) - x(5)\)

\(x(0) - x(7)\)

\(x(1) - x(6)\)

\(x(2) - x(5)\)

\(x(3) - x(4)\)

The introduction of zero terms within the matrix corresponds to a reduction of the number of multiplications required from 48 to 20 at the expense of a few extra additions. The rotator product technique \[7\] is furthermore applied to reduce the number of multiplications from 20 to 12. The rotator product technique is the method of changing the redundancy of multiplication matrix from 4 to 3, and can be described by:

\[
\begin{bmatrix}
   c_6 & -c_2 \\
   c_2 & c_6
\end{bmatrix}
\begin{bmatrix}
   w_1 \\
   w_2
\end{bmatrix}
= \begin{bmatrix}
   1 & -1 & 0 \\
   0 & 1 & 1
\end{bmatrix}
\begin{bmatrix}
   (c_6 + c_2)w_3 \\
   c_6w_2 + w_1 \\
   (c_6 - c_2)w_2
\end{bmatrix}
\]

After rotator product, the multiplication values of each block are following:

\[X_1 = c_2 - c_6\]

\[X_2 = c_6\]

\[X_3 = c_2 + c_6\]

\[X_4 = c_1 - c_3 + c_4 + c_7\]

\[X_5 = c_3 - c_4\]

\[X_6 = c_1 - c_3 + c_4 - c_7\]

\[X_7 = c_1 - c_3\]

\[X_8 = c_3\]

\[X_9 = c_1 - c_3\]

\[X_{10} = c_3 + c_4 - c_7\]

\[X_{11} = c_3 - c_4\]

\[X_{12} = c_1 + c_3 + c_4 + c_7\]

The technique can be used for hardware architecture of 1D discrete cosine transformation. As you can see from the Fig. 2., each step of calculation may use different times of calculation. Therefore, we have to put the time delay (D block) to ensure the synchronization between signals. The circuit, composed of 12 multiplications and 32 additions, is connected in term of common cell matrix. Bit serial adders and subtractor cells have been used and the multipliers have been implemented. The 8 bits input data and 12 bits output data are selected to be 2’s complement. The coefficients are constant and defined within hardware.

4 Implementation using Field Programmable Gate Array

FPGA is developed and applied for different types of work, for example in part of calculation, image processing, etc. The advantage of using FPGA is that it can be reprogrammed at a logic gate level for each different specific application. There are different series of Xilinx chips provided for different works. For example, the 4000 series \[8\] is one that has high efficiency and has a large amount of gates, up to 18,0000 equivalent gates.
The FPGA’s structure is composed of 2 main parts. The first one is Input/Output Block (IOB) which can be used for IC’s pin connected to other peripherals and Configurable Logic Block (CLB). It is used for the programming in order to make them work as our designed. For example, XC4010 contains 400 CLBs, 160 IOBs, and 10,000 equivalent gates.

## 5 DCT Simulation by Computer

From the experiment, we duplicated the 2D discrete cosine transformation by using the software. According to the mentioned method compared to the direct transformation which calculated from the equation(1). The Fig. 4. shows the result of the experiment by (a) We have the 256x256 pixel picture. First of all, we cut this picture into 8x8 pixel block. Then we transform it by using both method of 2D discrete cosine. The result is the same as the forth picture (b). If we do inverse discrete cosine transform, we will get the original image back.
6 Conclusion

The 8x8 2D DCT can be deduced to the case of 8x1 1D DCT. The implementation of 1D DCT requires only 12 multiplications, and thus feasible to be programmed on a single FPGA chip.

The next step is to implement 2D DCT by using hardware according to the architecture that is designed on FPGA. The complexity of implementation is the multiplier. However, using distributed arithmetic technique can solve this problem.

![Original image](image1.jpg) ![2D DCT transformed](image2.jpg)

Fig. 4. Result (a) Original image (b) 2D DCT transformed

References:


