New Architecture for High hroughput-Rate Real-Time 2-D DCT and the VLSI Design

Jen-Shiun Chiang and Hsiang-Chou Huang
Department of Electrical Engineering
Tamkang University
Tamsui, Taipei Taiwan

Abstract — The discrete cosine transform(DCT) has been widely used as the core of digital image and video signal compression. However, its computation is so intensive and is of great necessity to meet the requirement of high speed. In this paper, a new architecture for the VLSI design of 2-D DCT has been developed. This architecture contains the following features: (1) using the programmable logic array (PLA) to replace multipliers, (2) overlapped row-column operations and pipeline structure to reduce the total computation time, and (3) highly modular and regular structure for the efficient VLSI implementation. The architecture is implemented to a 8×8 2-D DCT. The circuit is designed by UMC's 0.8 μm spdm CMOS process and the cell library is provided by ITRI CCL. The simulation is shown that the speed of the data processing for this DCT is more than 50 MHz. It performs equivalently 800 million multiplication and accumulations per second.

I. INTRODUCTION

The discrete cosine transform (DCT)[1] is an orthogonal transform which consists of a set of sampled cosine functions in vector forms. The practical application of the DCT is a 2-D DCT. The 2-D DCT can be represented as $Z = C^t X C$, where X is a 2-D input data matrix, and C^t is the transposition matrix of the transform coefficient matrix C. A N-th order DCT coefficient matrix C is defined as follows

$$C_{kl} = \begin{cases} \frac{1}{\sqrt{N}} & for \quad l = 0, 0 \le k \le N-1 \\ \sqrt{\frac{2}{N}} \cos[\frac{\pi(2k+1)l}{2N}] & for \quad 1 \le l \le N-1, 0 \le k \le N-1 \end{cases} \quad .....(1)$$

A direct implementation of the 2-D DCT is of intensive matrix computations. Due to the dense computation requirement, the real application of the DCT should be a high performance chip. In recent designs [2-6], some of them use the Distributed Architecture (DA) with memory look-up table to implement the DCT chip [2-4]. Others apply row-column overlapped operations with multipliers [5], and matrix re-permutation with memory look-up tables [6].

The DA architecture [2-4] would save area that compared to other designs, but due to the input buffer requirement it could not achieve fully pipeline performance, and the total computation time is very long. The two designs of [5] and [6] contain the multiplier and transposition RAM in the architecture, and that would cause reduction of computation speed and long latency in total computation cycles.

A new architecture is proposed to combine the merits of the above two designs, matrix re-permutation with memory look-up tables and row-column overlapped operations. The matrix re-permutation simplifies the connections between the accumulator and the PLA look-up tables. The elimination of multipliers reduces the computation delay significantly, and the row-column overlapped operation does not only save the total computation time, but also simplify the hardware.

II. THE NEW ARCHITECTURE

The 2-D DCT can be expressed as two cascaded 1-D DCT as shown in Fig.1. It is implemented by the row-column decomposition technique. We compute the 8×1 DCT of each column of the input data matrix X to yield $X^t C$. The 8×1 DCT of each column of $C^t X$ is computed to find the desired 8×8 DCT. From (1) the two stages of DCT coefficient matrix can be re-written as follows.

$$C_1 = \begin{cases} \frac{1}{\sqrt{2}} & for \quad l_1 = 0 \\ \sqrt{2} \cos\frac{\pi(2m_1+1)l_1}{2N} & \end{cases}$$

$$for \quad l_1 \ne 0, m_1 = 1,2...N \quad .....(2)$$

$$C_2 = \begin{cases} \frac{1}{\sqrt{2}} & for \quad l_2 = 0 \\ \cos[\frac{\pi(2m_2+1)l_2}{2N}] & \end{cases}$$

$$for \quad l_2 \ne 0, m_2 = 1,2...N \quad .....(3)$$

By the following two steps, the 2-D DCT is therefore obtained.

0-7803-3302-0/96 $5.00 © 1996 IEEE

Step 1 :

$$Y(l_1,m_2)=\sum f(m_1,m_2)\bullet C_1(l_1,m_1) \quad for \quad l_1=1....N \ .....(4)$$

Step 2 :

$$Z(l_1,l_2)=\sum Y(l_1,m_2)\bullet C_2(l_2,m_2) \quad for \quad l_2=1....N \ .....(5)$$

Notice that (4) and (5) have independent indices, therefore, they can be done in parallel for all their own indices. The major difference between the two equations, (4) and (5) is the sequence order of input data. In (4), the input data are in the sequence order of m_1, and every $Y(l_1,m_2)$ will be computed. Thus, the regular multiplication and accumulation can be implemented in (4). In (5), the data are also supplied in the sequence order of m_2 and all $Z(l_1,l_2)$ will be computed in the same way as (4). Finally the operations of transposition and overlapped row-column computation is realized. For the 8×8 2-D DCT, there are only seven kinds of coefficients in the DCT coefficient matrix. The matrix form can be expressed as follow :

$$C_1=\begin{bmatrix} 1 & c & a & -d & 1 & e & b & f \\ 1 & d & b & -f & -1 & c & -a & -e \\ 1 & e & -b & -c & -1 & f & a & d \\ 1 & f & -a & -e & 1 & d & -b & -c \\ 1 & -f & -a & e & 1 & -d & -b & c \\ 1 & -e & -b & c & -1 & -f & a & -d \\ 1 & -d & b & f & -1 & -c & -a & e \\ 1 & -c & a & d & 1 & -e & b & -f \end{bmatrix}$$

where

$a=(\sqrt{2})\cos(\frac{2\pi}{16}), b=(\sqrt{2})\cos(\frac{6\pi}{16}), c=(\sqrt{2})\cos(\frac{\pi}{16}), d=(\sqrt{2})\cos(\frac{3\pi}{16}),$
$e=(\sqrt{2})\cos(\frac{5\pi}{16}), f=(\sqrt{2})\cos(\frac{7\pi}{16})$

In order to find a more regular and modular structure, a permutation matrix P [6] can be used to re-permute the column order of matrix C as follows.

$$C\bullet P=\begin{bmatrix} 1 & 1 & a & b & c & e & d & f \\ 1 & -1 & b & -a & d & -c & -f & -e \\ 1 & -1 & -b & a & e & f & -c & d \\ 1 & 1 & -a & -b & f & d & -e & -c \\ 1 & 1 & -a & -b & -f & -d & e & c \\ 1 & -1 & -b & a & -e & -f & c & -d \\ 1 & -1 & b & -a & -d & c & f & e \\ 1 & 1 & a & b & -c & -e & -d & -f \end{bmatrix}$$

where $P=\begin{bmatrix} 1&0&0&0&0&0&0&0 \\ 0&0&0&0&1&0&0&0 \\ 0&0&1&0&0&0&0&0 \\ 0&0&0&0&0&0&1&0 \\ 0&1&0&0&0&0&0&0 \\ 0&0&0&0&0&1&0&0 \\ 0&0&0&1&0&0&0&0 \\ 0&0&0&0&0&0&0&1 \end{bmatrix}$

After the re-permutation all rows contain the same eight coefficients that can be multiplied by every input data. According to the above matrix, the new architecture of the DCT is shown in Fig. 3.

III. THE OVERALL OPERATIONS

A. The PLA Look-up Tables

Because all the DCT coefficients are fixed numbers, the product patterns of all the input data and coefficients can be pre-stored in a PLA table. Here we multiply the input data to a constant to find the product. The data size of the input is 8-bit wide, and the PLA size for each indices is 256. Actually the PLA size can be reduced by some arrangement. Let us partition the input data into two parts, upper four bits and lower four bits. The upper four bits can be multiplied to the constant indices to find the partial product and be stored in a PLA table; the lower four bits can be multiplied to the same constant indices to find the other partial product and be stored in the other PLA table. The total product is equal to the sum of the two partial product. By this method the hardware can be saved almost half of the original approach. The block diagram of the PLA look-up table is shown in Fig. 2.

B. 1-D DCT of the 1st Stage

The block diagram of the architecture of DCT is shown in Fig. 3. In Fig. 3, there are multiplexers, accumulation adders, and auxiliary registers which are connected together to implement the summation and transposition with overlapped row-column operations. The inputs of the multiplexer are from the output of different PLA look-up tables. The connections of all the multiplexers and PLA look-up tables are due to permuted coefficient matrix. Through the multiplexer, the data from PLA look-up table are loaded to the accumulation adder, and are added to the output of auxiliary registers. Thereafter, the sum of the accumulation adder will be "latched" by the auxiliary register which are waiting for the next data from PLA look-up tables. The summation of the matrix computation is recursive, and the result of the accumulator will be put to the output register (as shown in Fig. 3) by every cycle of recursion, and the auxiliary register is reset every N cycles of recursion.

C. 1-D DCT of the 2nd Stage

The 1-D DCT of the 2nd stage is different from the 1st stage in the number of registers. The registers in the 2nd stage are seven times more than that in the 1st stage. In this stage, the data matrix $\Sigma Y(l_1, m_2)C_2(l_2, m_2)$ are loaded to the accumulation adder one by one in the order of m_2 (column-wise) in each row. A row of $\Sigma Y(l_1, m_2)C_2(l_2, m_2)$ will be added to the N auxiliary registers. Each column of the resultant matrix $Z(l_1, l_2)$ will be stored in the N auxiliary registers respectively. Then each column in the N auxiliary registers is downloaded to each of the N output registers. Afterwards, each column of the N output registers will be shifted out serially in the order of l_1 (row-wise), and complete the transposition and overlapped row-column operations.

Let us consider the total clock cycles in computing the N× N 2-D DCT, there are (N× 1+3) cycles for the 1-D DCT of the 1st stage and (N× N+3) cycles for the 2nd stage. For N× N 2-D DCT, the total computation time is N^2+N+6 cycles. The methods shown in [2-6] require N^2+2N to $2N^2$ clock cycles. Obviously, the total computation time has been effectively reduced compared to any of the former and conventional approaches.

IV. THE DESIGN OF A 8×8 2-D DCT AND THE SIMULATION RESULT

According to the modular and regular structure, the chip design is straightforward. The hierarchical modules are designed by Verilog HDL, and the logical functions are simulated by Cadence's Verilog-XL simulator. The Verilog HDL program is synthesized by Synopsys tools and *CCL* cell library, and the circuit lay-out is accomplished by Cadence's Cell Ensemble tool to automatically place and route. The look-up tables are made by the single clock dynamic CMOS PLA to save power dissipation and data output latency. The precision in the internal arithmetic are also well considered. The input data is eight bits, and the intermediate results (after 1-D DCT) are of 12-bit precision. The final result of the 2-D DCT is 14-bit precision.

Design rule of this chip is 0.8 μm spdm CMOS, and the cell library is provided by CCL. The total transistor count is over 140,000. The synthesized netlists are taken to simulate with the time view models of the cell library by Verilog, and we find the maximum computation speed is 55.6 MHz. The critical path is in the adder and the register stages, and we use carry look-ahead adder and simple latch circuit to overcome the difficulties. Fig. 4 shows the simulation result. Table I summarizes the design characteristics based on the simulation, and the VLSI layout of the DCT is shown in Fig. 5.

V. CONCLUSION

A new architecture of the 8×8 2-D DCT is presented which efficiently combined the merits of replacing multipliers with PLA, elimination of transposition memory, and overlapped row-column operation. The total computation time is reduced by the overlapped row-column operation in 1-D DCT, and the data rate speeds up due to accumulator and PLA look-up table improvement. This approach is feasible for the VLSI implementation and is suitable for high speed application such as HDTV.

The transistor count of the designed circuit is over 140,000. Since the design is finished by the automatic synthesis and auto-place-auto-route. If the layout is accomplished by the fully customer design, we expect the area of the implementation chip can be reduced significantly, and the speed can thus be even higher. These aspects may be the further research attempt.

VI. REFERENCE

[1] N. Ahmed, T. Natarajan, K. R. Rao, "Discrete Cosine Transform", IEEE Trans. Comput., vol. C-23, pp. 90-93, Jan.1974.

[2] M. T. Sun, L. Wu, and M. L. Liou, "A Concurrent Architecture for VLSI Implementation of Discrete Cosine Transform", IEEE Trans. on Circuit and System, vol. CAS-34, No. 8, Aug. 1987

[3] J. C. Carlach, P. Penard, J. L. Sicre, "TCAD: a 27 MHz 8×8 Discrete Cosine Transform Chip", Proceedings of ICASSP'89, pp.2429-2432, 1989

[4] U. Sjostrom, I. Defilipps, M. Ansorge, and F. Pellandini, "Discrete Cosine Transform Chip for Real-Time Video Application", Proc. ISCAS, 1990, pp. 1620-1623

[5] S. P. Kim and D. K. Pan, "Highly Modular and Concurrent 2-D DCT Chip",Proceedings of ISCAS, pp. 1081-1084, 1992

[6] M. H. Sheu, J. Y. Lee, J. F. Wang, A. N. Suen, and L. Y. Liu, "A High Throughput-Rate Architecture for 8× 8 2-D DCT", Proceedings of ICASSP'93, pp. 1587-1590, 1993

[7] G. M. Blair, "PLA Design for Single-Clock CMOS", IEEE JSSC, vol. 27, No. 8, Aug. 1992

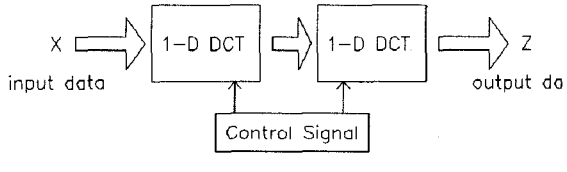

Fig. 1 Block diagram of 2-D DCT

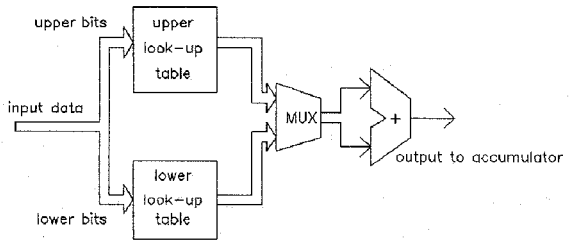

Fig. 2 Block diagram of PLA look-up table

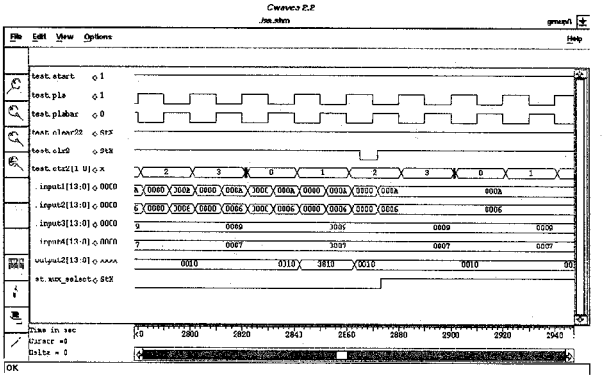

Fig. 4 The result of simulation with time view models of cell library

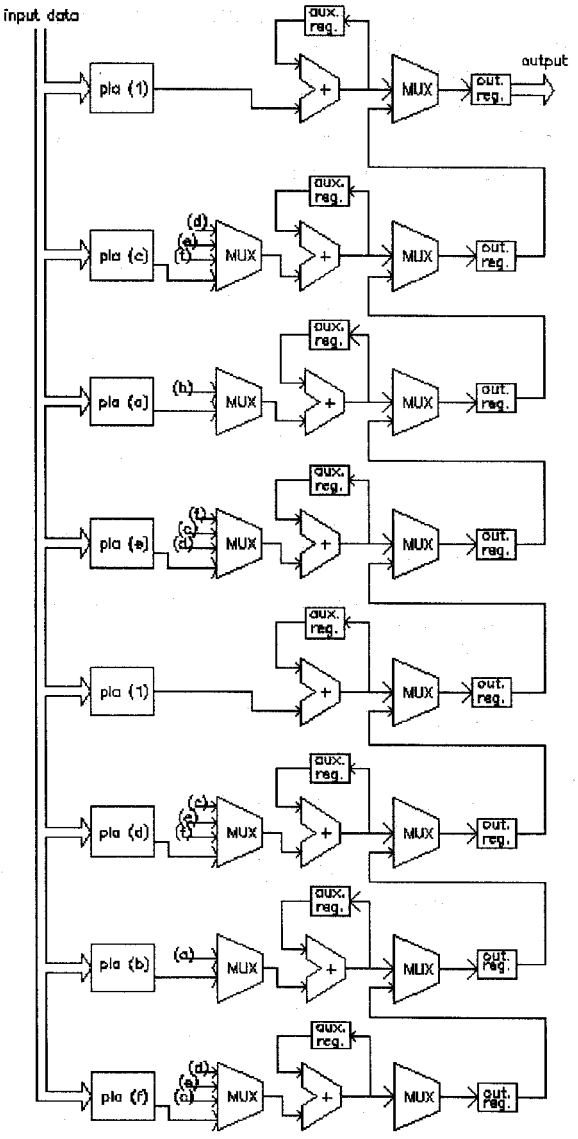

Fig. 3 The new architecture of 1-D DCT

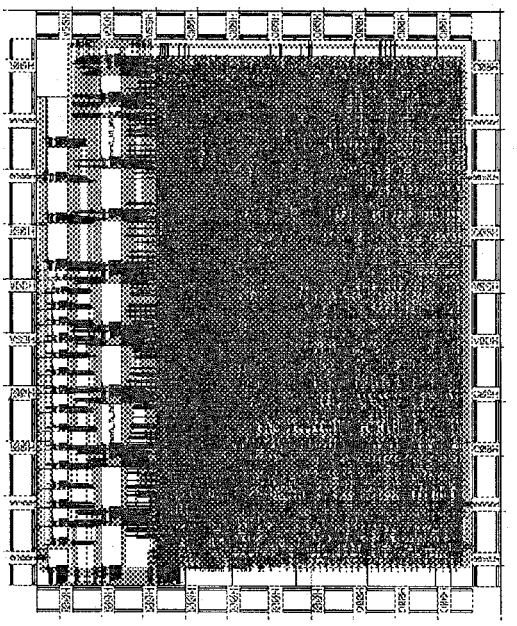

Fig. 5 The VLSI layout of the 2-D DCT

Design Rule (CMOS)	0.8 um spdm CMOS
Cell Library	ITRI CCL's TSMC08
Clock Rate	50 MHz (time view model simulated)
Latency	78 clock cycles
Transister Count	147,839
Layout Area	6337.8 um × 7715.9 um
Pipeline	100%

Table I : Summary of the designed DCT chip