

АРХИТЕКТУРА МОДУЛЯ ВЫЧИСЛЕНИЯ ДКП-ОДКП С ПЕРФЕКТИВНОЙ РЕКОНСТРУКЦИЕЙ ДЛЯ ОБРАБОТКИ ИЗОБРАЖЕНИЙ БЕЗ ПОТЕРЬ

В.В. Ключеня

Кафедра электронных вычислительных средств,
Белорусский государственный университет информатики и радиоэлектроники
Минск, Республика Беларусь
E-mail: vitaly.kliuchenia@gmail.com

В данной работе предлагается структурное решение вычислительного ядра дискретного косинусного преобразования (ДКП) и обратного (ОДКП) в виде единого модуля для систем сжатия изображений без потерь. Для сжатия данных без потерь или с совершенной реконструкцией будет использоваться лестничная структура, блоками которой будут являться ДКП и ОДКП.

ВВЕДЕНИЕ

Благодаря свойству «уплотнения» энергии ДКП стало популярным ядром многих стандартов сжатия, но при аппаратном декодировании данных появляются артефакты на изображении в виде эффекта "шахматной доски". Поэтому для того, чтобы применить ДКП для систем кодирования данных без потерь воспользуемся лестничной структурной параметризацией, предложенной в статье [1], одним из свойств которых является полное восстановление данных. Лестничная параметризация представляет собой блочную ступенчатую структуру, в свою очередь блоки представляют собой прямое и обратное ДКП, которые описаны как ДКП-II типа (блок С) и ДКП-III типа (блок D) (рис.1).

I. АППАРАТНЫЙ МОДУЛЬ ДКП-ОДКП

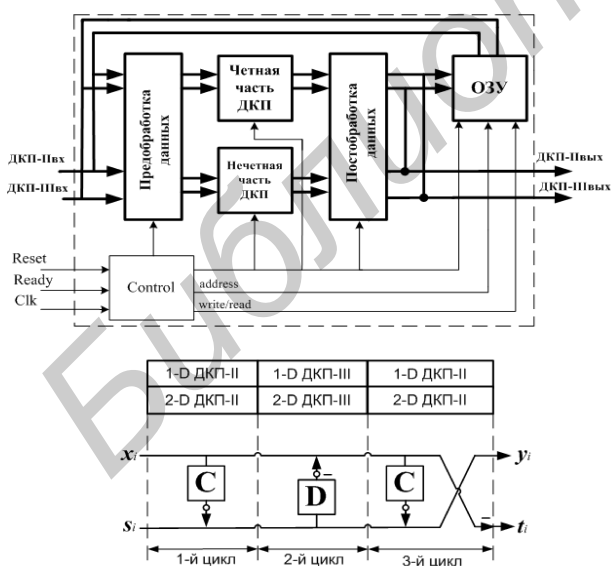


Рис. 1 – Архитектура модуля ДКП-ОДКП

Так как в качестве лестничных ступеней в схеме L2L (lossless-to-lossy)[1] используется несколько подряд стоящих блоков вычисления ДКП, то целесообразно было бы объединить их в один модуль. Поэтому для реализации бло-

ков ДКП и ОДКП для L2L схемы предлагается использовать универсальный модуль (рис. 1), который позволяет вычислять ДКП и ОДКП как одновременно, так и последовательно в разных режимах. За основу берется двумерное 2-D (two dimensional) ДКП размерностью 8x8. Вычисление будет производиться сначала по строкам, сохраняя промежуточные результаты в памяти (ОЗУ), затем по столбцам как для прямого так и для обратного ДКП. В режиме без потерь вычисление производится по схеме 2-D ДКП/2-D ОДКП/2-D ДКП (рис.1) за три цикла. В режиме с потерями данных используется только первый цикл. Вектор входных и выходных значений представляется следующим образом соответственно:

$$\mathbf{x} = [x_0 \ x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6 \ x_7]^T;$$

$$\mathbf{Z} = [Z_0 \ Z_1 \ Z_2 \ Z_3 \ Z_4 \ Z_5 \ Z_6 \ Z_7]^T;$$

Для того, чтобы воспользоваться свойством симметрии ДКП нужно предварительно подготовить данные по формулам:

$$\mathbf{a} = \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} x_0 + x_7 \\ x_1 + x_6 \\ x_2 + x_5 \\ x_3 + x_4 \end{bmatrix}, \quad \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} = \frac{1}{2} [\mathbf{a} + \mathbf{b}],$$

$$\mathbf{b} = \begin{bmatrix} b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} x_0 - x_7 \\ x_1 - x_6 \\ x_2 - x_5 \\ x_3 - x_4 \end{bmatrix}, \quad \begin{bmatrix} x_7 \\ x_6 \\ x_5 \\ x_4 \end{bmatrix} = \frac{1}{2} [\mathbf{a} - \mathbf{b}].$$

Для вычисления значений векторов \mathbf{a} и \mathbf{b} используется модуль предварительной обработки данных. Входные значения подаются через мультиплексоры в буфер памяти, далее через сумматор-вычитатель поступают на демультимплексор, где они разбиваются на четные и нечетные части прямого ДКП. Также на вход данного модуля подаются рассчитанные значения уже из предыдущего цикла Z_0-Z_7 , которые накапливаются в буфере (LIFO Buffers) и далее также

разбиваются на четные и нечетные части. Аналогичная ситуация происходит в модуле постобработки данных, который реализует вычисление значений вектора x . Более подробно каждая часть модуля описана в статье [2].

После предварительной обработки, данные поступают в модуль вычисления четной и нечетной части ДКП и ОДКП. Четная часть прямого и обратного ДКП выражается формулами:

$$\begin{bmatrix} Z_0 \\ Z_4 \\ d_0 \\ d_1 \end{bmatrix} = c_4 \begin{bmatrix} a_0 + a_3 + a_1 + a_2 \\ a_0 + a_3 - (a_1 + a_2) \\ Z_0 + Z_4 \\ Z_0 - Z_4 \end{bmatrix},$$

$$\begin{bmatrix} d_2 \\ d_3 \\ Z_2 \\ Z_6 \end{bmatrix} = \begin{bmatrix} Z_2 & Z_6 \\ -Z_6 & Z_2 \\ a_0 - a_3 & a_1 - a_2 \\ -(a_1 - a_2) & a_0 - a_3 \end{bmatrix} \begin{bmatrix} c_2 \\ c_6 \end{bmatrix},$$

$$\begin{bmatrix} d_0 \\ d_1 \\ d_2 \\ d_3 \end{bmatrix} = \frac{1}{4} \begin{bmatrix} a_0 + a_3 \\ a_1 + a_2 \\ a_0 - a_3 \\ a_1 - a_2 \end{bmatrix}, \frac{1}{2} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} d_0 + d_2 \\ d_1 + d_3 \\ d_1 - d_3 \\ d_0 - d_2 \end{bmatrix},$$

где $c_i = \cos(\frac{i\pi}{16})$ - постоянные коэффициенты ДКП и ОДКП.

Структура четной части ДКП представлена тремя блоками [2]:

- предварительного вычисления значений множителя, которая состоит из сумматора-вычитателя и двух наборов регистров, где хранятся предварительно рассчитанные значения или входные данные;
- умножения на коэффициенты c_2, c_4 и c_6 , реализуются при помощи метода распределенной арифметики, так как необходимо получить перфективную реконструкцию, то используется схема, где данные обрабатываются по одному биту за такт (1-BAAT – one-bit-at-a-time), предварительные значения сумм коэффициентов c_2 и c_6 и отдельно коэффициента c_4 хранятся в памяти (ПЗУ). Все значения коэффициентов представлены в дополнительном коде;
- третья часть для постобработки произведений.

Нечетная часть прямого и обратного ДКП выражается формулами:

$$\begin{bmatrix} Z_7 \\ Z_5 \\ Z_1 \\ Z_3 \end{bmatrix} = \begin{bmatrix} -b_3 & b_2 & b_0 & -b_1 \\ -b_1 & b_3 & b_2 & b_0 \\ b_0 & b_1 & b_3 & b_2 \\ -b_2 & b_0 & -b_1 & -b_3 \end{bmatrix} \begin{bmatrix} c_1 \\ c_3 \\ c_7 \\ c_5 \end{bmatrix},$$

$$\frac{1}{2} \begin{bmatrix} b_0 \\ b_1 \\ b_3 \\ b_2 \end{bmatrix} = \begin{bmatrix} Z_1 & Z_3 & Z_7 & Z_5 \\ -Z_5 & Z_1 & -Z_3 & -Z_7 \\ -Z_7 & Z_5 & Z_1 & -Z_3 \\ -Z_3 & Z_7 & Z_5 & Z_1 \end{bmatrix} \begin{bmatrix} c_1 \\ c_3 \\ c_7 \\ c_5 \end{bmatrix}.$$

Структура нечетной части ДКП состоит из модулей распределенной арифметики, все возможные значения множителей представлены в виде суммы коэффициентов c_1, c_3, c_7 и c_5 , которые хранятся в ПЗУ [2].

II. ЭКСПЕРИМЕНТАЛЬНЫЕ РЕЗУЛЬТАТЫ

Сравним результат синтеза предлагаемой реализации с уже существующими на ПЛИС типа FPGA Spartan-2 XC2VP30.

Таблица 1 – Реализации архитектур ДКП и ДКП-ОДКП на Spartan-2 XC2VP30

Архитектура	[3]	[4]	[5]	[6]	[7]	Предлагаемая
Функция	ДКП	ДКП	ДКП	ДКП	ДКП-ОДКП	ДКП-ОДКП
LUTs	2990	990	10310	2618	2237	1109
Slices	1872	612	5729	2823	1352	627
Flip Flops	1837	652	3736	3431	1170	887
Freq., MHz	99	110	149	107	168	120
Delay, ns	10.1	9.1	6.7	9.3	6.2	8.3

Если посмотреть на таблицу, то видно, что предлагаемая реализация требует в два раза меньше аппаратных затрат в сравнении с аналогичной, описанной в [7], и в несколько раз меньше занимает площади кристалла в сравнении с [3], [5] и [6], где представлена реализация только одного прямого ДКП. Как видно из результатов синтеза, мы получили значительную экономию в ресурсах FPGA.

1. Suzuki, T. Integer DCT Based on Direct-Lifting of DCT-IDCT for Lossless-to-Lossy Image Coding/ T. Suzuki, M. Ikehara, // IEEE TRANSACTIONS ON IMAGE PROCESSING, VOL. 19, NO. 11, NOVEMBER 2010, PP. 2958-2965
2. Kliuchenia V. Architecture of DCT-IDCT processor for lossless scheme coding/ V. Kliuchenia, A. Petrovsky // Pattern Recognition and Information Processing, PRIP'2014, Proceedings of the 12th international conference 28-30 May 2014, Minsk, Belarus, pp.119-123.
3. Chen, Y. H. High throughput DA-based DCT with high accuracy error-compensated adder tree / Y. H. Chen, T. Y. Chang, and C. Y. Li // IEEE Trans.VLSI Syst., vol. 19, no. 4, pp. 709–714, Apr. 2011.
4. Chen, Y. H. A high performance video transform engine by using space-time scheduling strategy/ Y. H. Chen, T. Y. Chang // IEEE Trans. VLSI Syst., vol. 20, no. 4, pp. 655–664, Apr. 2012.
5. Sun, C. C. Low-complexity multi-purpose IP core for quantized discrete cosine and integer transform, // C. C. Sun, P. Donner, J. Gotze, // in Proc. IEEE Int. Symp. Circuits Syst., 2009, pp. 3014-30.17.
6. Tumeo, A. A pipelined fast 2D-DCT accelerator for FPGA-based SoCs/ A. Tumeo, M. Monchiero, G. Palermo, F. Ferrandi, and D. Sciuto, // in Proc. IEEE Comput. SoC. Annu. Symp.VLSI., 2007, pp. 331-336
7. Chen, Y.-H. A High-Throughput and Area-Efficient Video Transform Core with a Time Division Strategy/ Y.-H. Chen, R.-Y. Jou, T.-Y. Chang, C.-W. Lu // IEEE Transactions on Very Scale Integration Systems, 2013.