

УДК 004.031.6

В.В. Ключеня, Н.А. Петровский

**БЫСТРОЕ ПРОТОТИПИРОВАНИЕ ВСТРАИВАЕМЫХ ПРОГРАММИРУЕМЫХ СИСТЕМ НА ПЛИС ДЛЯ МУЛЬТИМЕДИЙНЫХ ПРИЛОЖЕНИЙ**

*Рассматривается процесс быстрого прототипирования мобильных мультимедийных систем трансформационного кодирования изображений по схеме L2L (lossless-to-lossy): сжатие и восстановление изображения как без потерь, так и с контролируемым внесением артефактов. За основу берется отладочный модуль Xilinx ML-401, в качестве основного управляющего ядра используется встроенный софт-процессор MicroBlaze. Процессоры дискретного косинусного преобразования (ДКП) представляются в виде модулей на языке описания аппаратуры VHDL, которые подключаются к софт-процессору MicroBlaze в виде сопроцессора, результат обработки изображений выводится через VGA-интерфейс на экран монитора. Предлагается архитектура рекурсивного процессора вычисления ДКП – обратного ДКП, которая может рассматриваться как базовая при построении кодеров класса L2L.*

**Введение**

Интенсивное развитие информационных технологий в современном мире делает особенно важным решение проблемы быстрой и качественной передачи различного рода информации по цифровым линиям связи. В качестве передаваемой информации часто выступают такие мультимедиаданные, как изображения, видео-, аудио- и речевые сигналы. Алгоритм ДКП играет ключевую роль в приложениях компрессии как аудиоданных [1], так и изображений [2]. Несмотря на большое количество предложений аппаратной реализации алгоритма ДКП, специфические требования прикладных приложений компрессии мультимедиаданных обуславливают актуальность вопроса реализации процессора ДКП, удовлетворяющего ограничениям соответствующего приложения. Большинство решений направлено на получение заказных СБИС для специфических алгоритмов кодирования. Например, в работе [3] показана подсистема обработки мультимедиаданных на основе гибридной схемы «цифровой процессор обработки сигналов – программируемая логическая интегральная схема (ПЛИС)» (DSP-FPGA), реализуемой на печатной плате, которая гарантирует пользователю высокую гибкость настройки аппаратуры на соответствующий алгоритм обработки. Однако недостатком такого решения является повышенная потребляемая мощность (выше, чем в интегральном исполнении), а также то, что производительность системы может быть ниже.

Современные ПЛИС с архитектурой FPGA (Field Programmable Gate Array) обеспечивают уровень интеграции, сопоставимый с текущими специализированными интегральными схемами, и достаточно быстры. Данные характеристики FPGA позволяют сделать доступной реализацию нескольких встроенных процессоров, которые могут быть эффективно задействованы для получения достаточно высокой скорости обработки мультимедиаданных. Это делает реальной возможность реализации программируемой системы-на-кристалле ПЛИС (Field-Programmable System-on-Chip, FPSoC) для компрессии мультимедиаданных. Как и любая система-на-кристалле, FPSoC представляет собой один или сочетание нескольких микропроцессоров, некоторое встроенное программное обеспечение в памяти и другую аппаратную логику, которые с помощью определенных шин связи объединены на кристалле. Для получения эффективной FPSoC при решении соответствующего приложения необходимо такое распределение аппаратно-программных ресурсов, которое бы приводило к высокой производительности системы при решении прикладной задачи [4].

Традиционно к проектированию FPSoC привлекаются многочисленные группы инженеров, включая разработчиков алгоритма, программистов, проектировщиков аппаратной части и группу реализации проекта, причем каждая группа разработчиков задействована только на определенном этапе. Этот традиционный процесс разработки занимает много времени, потому что проектировщики алгоритма часто не имеют доступа к аппаратным средствам, которые фак-

тически уже созданы. Процесс быстрого прототипирования объединяет фазы проектирования, устраняя потенциально узкие места, дает возможность инженерам видеть результаты и быстро повторять решения без использования дорогих аппаратных средств.

Целью настоящей статьи является демонстрация процесса быстрого прототипирования на примере создания процессора ДКП – встраиваемого процессора в программируемую систему-на-кристалле ПЛИС реального времени на основе отладочного модуля Xilinx ML-401 с использованием ПЛИС типа FPGA семейства Virtex 4 (XC4VLX25-FF668-10B). Функциональное назначение FPSoC – компрессия изображений по схеме L2L (lossless-to-lossy) [5, 6]: сжатие и восстановление мультимедиаданных как без потерь, так и с контролируемым внесением артефактов. Система состоит из управляющего процессора, в качестве которого используется встроенный софт-процессор MicroBlaze, выполняющий квантование и энтропийное кодирование, и сопроцессора ДКП, выполняющего преобразование исходного изображения. Основными приложениями данной FPSoC могут быть, например, аэрокосмическое наблюдение и съемка, для которых характерны строгие ограничения по пропускной способности и рассеиваемой мощности. При выполнении подобных проектов по созданию подсистем компрессии изображений во внимание принимаются такие характеристики, как диапазон компрессии; минимальное число кадров, обрабатываемых в секунду; количество пикселей в кадре изображения; обеспечение возможности «легкого апгрейда» алгоритма кодирования.

### **1. Аппаратная платформа для быстрого прототипирования программируемых систем-на-кристалле ПЛИС**

Представители всех серий нового поколения кристаллов программируемой логики могут эффективно применяться для реализации встраиваемых микропроцессорных систем, разрабатываемых на основе конфигурируемых 32-разрядных ядер семейства MicroBlaze фирмы Xilinx или NIOS компании Altera. В рамках решаемой задачи можно использовать отладочную плату Xilinx ML-401, где управляющим процессором выбирается софт-процессор MicroBlaze с широким спектром настраиваемых интерфейсов, периферийных устройств и пользовательских компонентов интеллектуальной собственности (Intellectual Property Core – IP-core), далее IP-компонентов. Встроенное в ПЛИС 32-разрядное софт-процессорное ядро MicroBlaze с RISC-архитектурой оптимизировано для создания FPSoC. Структура ядра софт-процессора MicroBlaze (рис. 1) построена по принципу гарвардской архитектуры, в основе которой лежит концепция использования отдельных шин для данных и команд [7–10]. Организация магистралей микропроцессора согласно этой концепции обеспечивает достижение высоких скоростей выполнения операций. Серым цветом на рисунке отмечены опциональные блоки, наличие которых определяется при конфигурировании системы. Для конфигурации доступны различные параметры софт-процессора: размер кеша, длина конвейера (трех- или пятиуровневый), встроенная периферия, блок управления памятью, шинные интерфейсы и т. д. Кроме того, ключевые компоненты процессора, которые редко используются, но при этом требуют значительных аппаратных затрат (т. е. блоки умножения, деления, операций с плавающей запятой и т. п.), могут быть выборочно добавлены или удалены.

Встроенный в ПЛИС софт-процессор MicroBlaze имеет несколько интерфейсных шин: шину ввода-вывода Core Connect PLB (Processor Local Bus) – универсальное средство связи с периферийными устройствами; специальную шину LMB для доступа к внутренней памяти ПЛИС (Block RAM), позволяющую снизить нагрузку на другие шины. Большинство IP-компонентов как от Xilinx, так и от других производителей подключаются напрямую к PLB или через мост между шинами PLB и OPB (On-Chip Peripheral Bus). Подключение встраиваемого процессора – сопроцессора для решения прикладной задачи возможно средствами специального соединения FIFO – FSL (Fast Simplex Link). Функциональная схема подключения пользовательских IP-блоков показана на рис. 2.

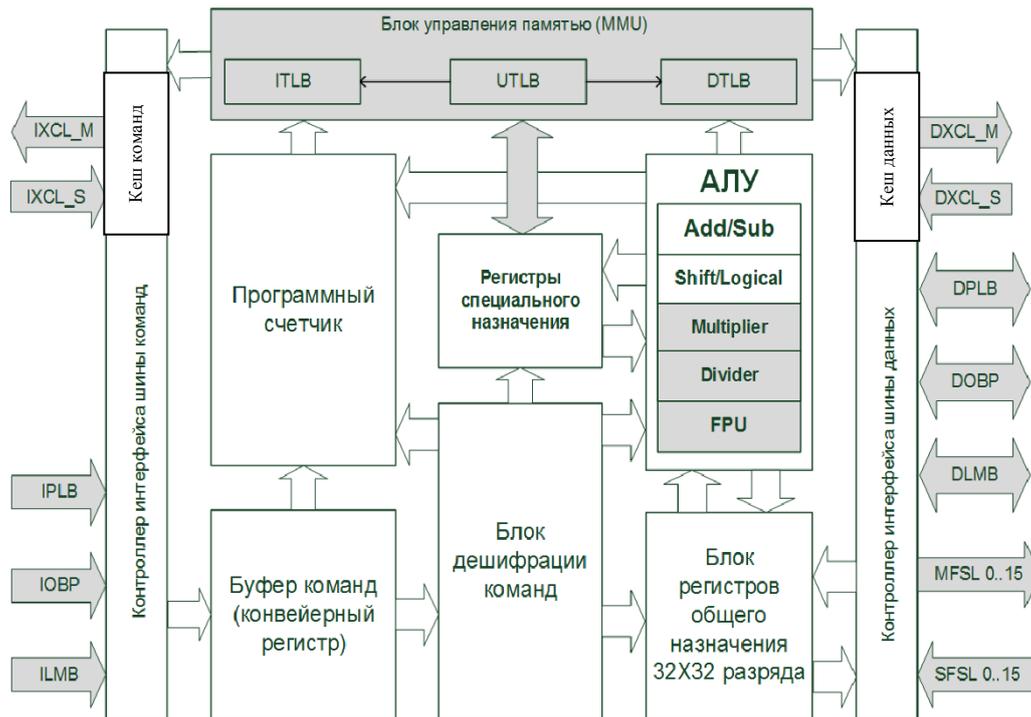


Рис. 1. Обобщенная структура микропроцессорного ядра MicroBlaze [8]

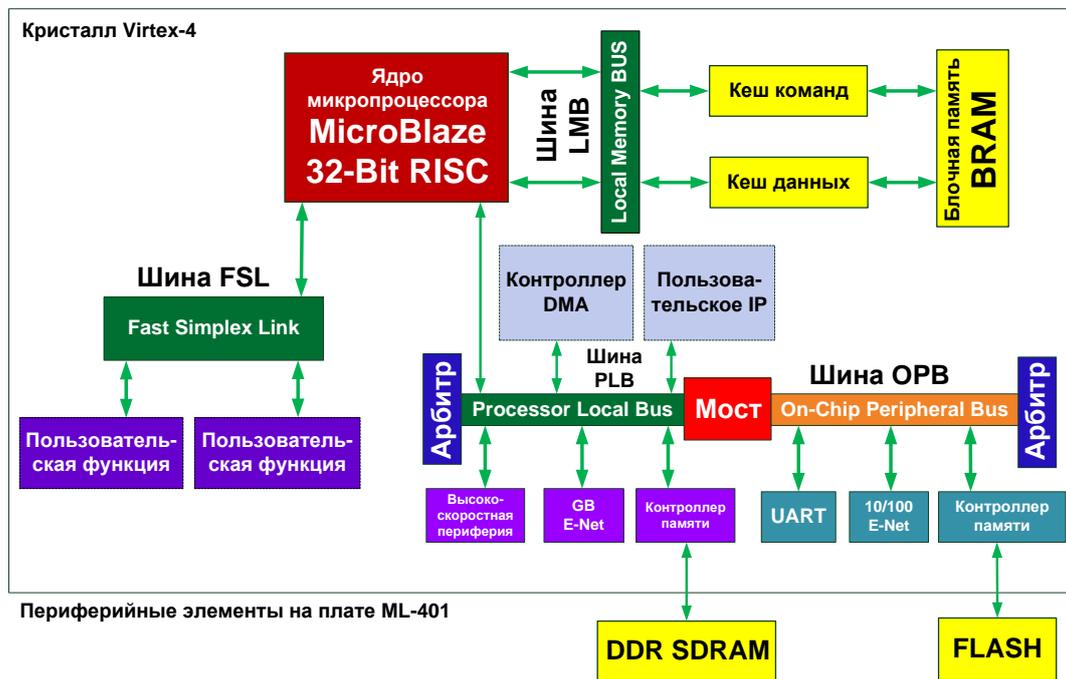


Рис. 2. Функциональная схема подключения периферийных устройств к MicroBlaze

## 2. Этапы быстрого прототипирования встраиваемых FPGAs

Существует много подходов совместного проектирования аппаратного и программного обеспечения. В данном проекте в качестве методологии для быстрого прототипирования программируемых систем-на-кристалле ПЛИС используется функционально-архитектурное проектирование на основе спиралевидной модели проектирования (рис. 3), которая позволяет осуществлять возврат и корректирование предыдущих этапов процесса проектирования.

Функционально-архитектурное проектирование встраиваемых FPSoC включает в себя следующие уровни абстракции [11, 12]:

- 1) задание спецификации или требований на FPSoC, в которые кроме функциональных требований включаются, например, требования реального масштаба времени (частотный диапазон), максимальная потребляемая мощность, ограничение максимальной стоимости и т. д.;
- 2) моделирование, тестирование и верификацию алгоритмического обеспечения;
- 3) структурный синтез, связанный с выбором и оценкой эффективной архитектуры процессора, соответствующей алгоритмическому обеспечению и требованиям спецификации;
- 4) аппаратную реализацию на выбранной ПЛИС с архитектурой FPGA.

На рис. 3 показаны этапы создания прототипа встраиваемых программируемых систем-на-кристалле ПЛИС, а также инструментальные среды, которые используются для проектирования таких систем. Подобная концепция быстрого прототипирования всей FPSoC на архитектурном уровне является неотъемлемой частью всех современных маршрутов проектирования [9]. Например, при разработке архитектуры FPSoC для определенного приложения может возникнуть необходимость в полной или частичной реализации какого-то прикладного IP-компонента: оценке его аппаратных затрат, потребляемой мощности, производительности, по результатам которой архитектура FPSoC может быть частично или полностью модифицирована.



Рис. 3. Этапы быстрого прототипирования программируемых систем-на-кристалле ПЛИС

### 3. Проектирование встраиваемого процессора ДКП в FPSoC компрессии изображений по схеме L2L

#### 3.1. Требования к аппаратной реализации алгоритма ДКП

Преобразование, используемое в кодерах изображений по схеме L2L, должно обладать следующими свойствами [13]: обеспечивать перфективную реконструкцию исходного изображения и линейную фазово-частотную характеристику (ФЧХ), иметь конечную импульсную характеристику, быть максимально децимированным и равнополосным. Основопологающими факторами при проектировании преобразователя систем кодирования изображений FPSoC на

ПЛИС с архитектурой FPGA для достижения высокой производительности при программной и аппаратной реализациях являются модульность и регулярность (обеспечение параллельных вычислений), малая емкость буферной памяти и низкая потребляемая мощность. При этом прямое и обратное преобразования осуществляются по схеме обработки «целое в целое» с сохранением перфективной реконструкции исходного изображения (коэффициенты представляются целыми или двоичными рациональными числами; число операций умножения минимально, по возможности они исключаются из алгоритма).

Объективная оценка качества реконструкции исходного изображения осуществляется при помощи часто используемых в области сжатия изображений метрик [5] – среднеквадратической ошибки MSE и пикового отношения сигнал-шум PSNR, что позволяет сравнивать полученные результаты с результатами других систем кодирования:

$$MSE = \frac{\sum_{x,y \in A} (s_{x,y} - s'_{x,y})^2}{A}, \quad PSNR = 20 \lg \frac{255}{\sqrt{MSE}} \text{ dB},$$

где  $s_{x,y}$  – значение  $(x, y)$ -го пиксела оригинального изображения,  $s'_{x,y}$  соответствует значению  $(x, y)$ -го пиксела реконструированного изображения,  $A$  – общее количество пикселов в изображении.

Основным достоинством данной метрики является охват всего сигнала (в данном случае изображения) при подсчете значения оценки, а не конкретной локальной области (частотной или временной), что важно для выбора метрики при сжатии изображений.

### 3.2. Моделирование в программной среде MATLAB: выбор преобразования

Целочисленные ДКП (IntDCT) [14], которые реализованы на основе лестничной схемной параметризации, были предложены для кодирования изображений по схеме L2L. Однако известные IntDCT генерируют в восстановленном исходном изображении такой артефакт, как «шахматная доска» при представлении коэффициентов преобразования в формате фиксированной запятой с малой разрядностью слова. В работе [14] предложено преобразование для системы трансформационного кодирования на основе блочной лестничной структурной параметризации прямого и обратного ДКП (ДКП-ОДКП), где блоками лестничных ступеней являются матрицы прямого или обратного ДКП. Математическое описание прямого (рис. 4, а) и обратного (рис. 4, б) преобразований на основе ДКП-ОДКП следующее [14]:

– прямого

$$\begin{bmatrix} \mathbf{C} & \mathbf{0} \\ \mathbf{0} & \mathbf{D} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{C} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & -\mathbf{D} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{C} & \mathbf{I} \end{bmatrix};$$

– обратного

$$\begin{bmatrix} \mathbf{C} & \mathbf{0} \\ \mathbf{0} & \mathbf{D} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{C} \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{C} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{D} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ -\mathbf{C} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{0} & -\mathbf{I} \\ \mathbf{I} & \mathbf{0} \end{bmatrix}.$$

Прямое и обратное ДКП обозначены как матрицы  $\mathbf{C}$  и  $\mathbf{D}$  соответственно и функционально описываются как ДКП второго типа и ДКП третьего типа [14]:

$$[\mathbf{C}]_{m,n} = \sqrt{\frac{2}{M}} \cdot c_m \cos\left(\frac{m(n+\frac{1}{2})\pi}{M}\right), \quad [\mathbf{D}]_{m,n} = \sqrt{\frac{2}{M}} \cdot c_n \cos\left(\frac{n(m+\frac{1}{2})\pi}{M}\right),$$

где  $\mathbf{I}$  – единичная матрица;  $\mathbf{D} = \mathbf{C}^{-1} = \mathbf{C}^T$ ,  $0 \leq m, n \leq M-1$ ;  $[\mathbf{C}]_{m,n}$  – матрица прямого ДКП второго типа, а  $[\mathbf{D}]_{m,n}$  – матрица ДКП третьего типа, являющаяся обратной для ДКП второго типа;

$m$  – количество столбцов,  $n$  – количество строк;  $M = 2^n (n \in N)$ ,  $c_m = \begin{cases} \frac{1}{\sqrt{2}} (m=0) \\ 1 (m \neq 1) \end{cases}$ ,  $c_n = \begin{cases} \frac{1}{\sqrt{2}} (n=0) \\ 1 (n \neq 1) \end{cases}$ .

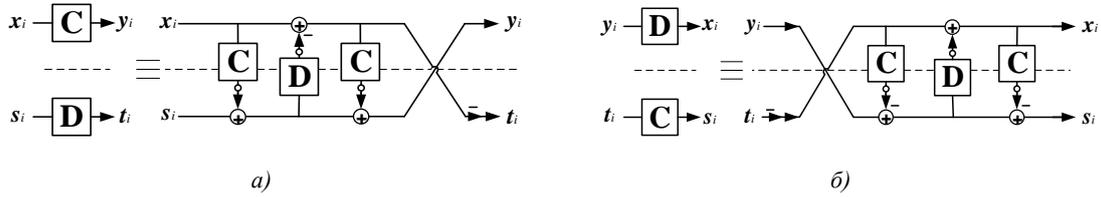


Рис. 4. Прямое (а) и обратное (б) преобразования для системы трансформационного кодирования на основе блочной лестничной структурной параметризации ДКП-ОДКП. Пустые кружки на схеме обозначают округление

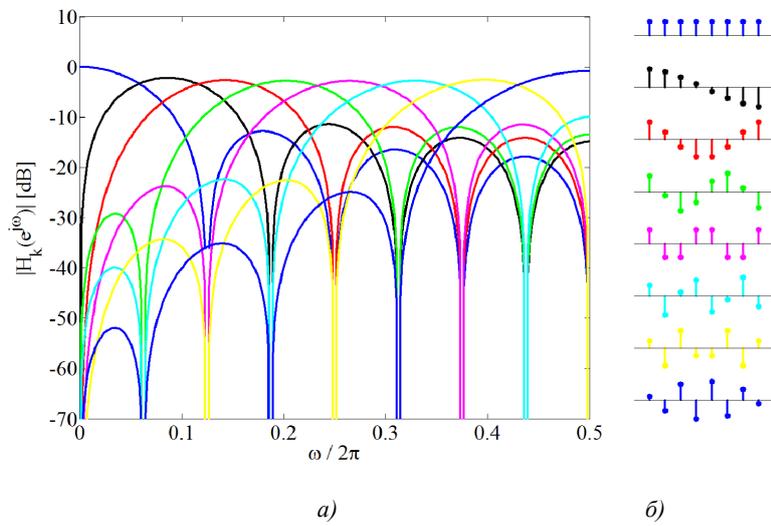


Рис. 5. Амплитудно-частотные (а) и импульсные (б) характеристики прямого ДКП

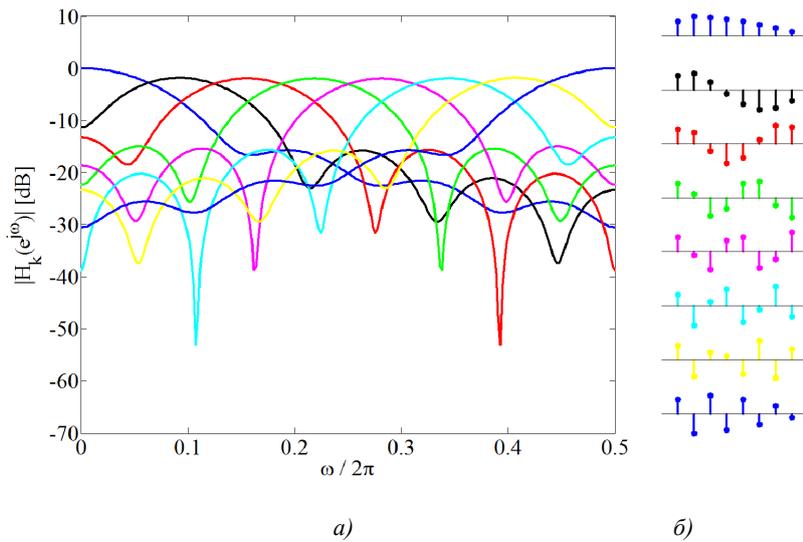


Рис. 6. Амплитудно-частотные (а) и импульсные (б) характеристики обратного ДКП

Амплитудно-частотные и импульсные характеристики восьмиточечного прямого и обратного ДКП представлены на рис. 5 и 6 соответственно. Анализ рисунков показывает, что

преобразование ДКП является равнополосным максимально децимированным банком фильтров с линейной ФЧХ (импульсные характеристики обладают свойством симметрии), постоянная составляющая локализована первым фильтром банка. Однако все фильтры ОДКП имеют постоянную составляющую (см. рис. 6, *a*), т. е. присутствует так называемый эффект DC-leakages.

В программной среде MATLAB с помощью инструментального модуля Fixed Point для моделирования вычислений на арифметике с фиксированной запятой был осуществлен расчет амплитудно-частотных характеристик (АЧХ) системы анализа-синтеза прямого и обратного преобразований ДКП-ОДКП с блочной лестничной структурной параметризацией и преобразований стандарта JPEG [13] для разрядности данных 12 бит, из которых 1 бит знаковый, 1 бит – целая часть слова, оставшиеся 10 бит представляют дробную часть слова. Анализ АЧХ данных преобразований (рис. 7) показывает, что АЧХ системы анализа-синтеза преобразования ДКП-ОДКП с блочной лестничной структурной параметризацией (рис. 7, *a*) представляет собой прямолинейную АЧХ, т. е. система анализа-синтеза данного преобразования вносит только задержку. Между тем АЧХ системы анализа-синтеза прямого и обратного преобразований стандарта JPEG [13] соответствует нелинейной функции с максимальной неравномерностью, равной  $-0,03$  дБ. Таким образом, в отличие от алгоритма преобразования, используемого в стандарте сжатия изображений JPEG, преобразование на основе блочной лестничной структурной параметризации прямого и обратного ДКП-ОДКП обладает свойством перфективной реконструкции исходного изображения.

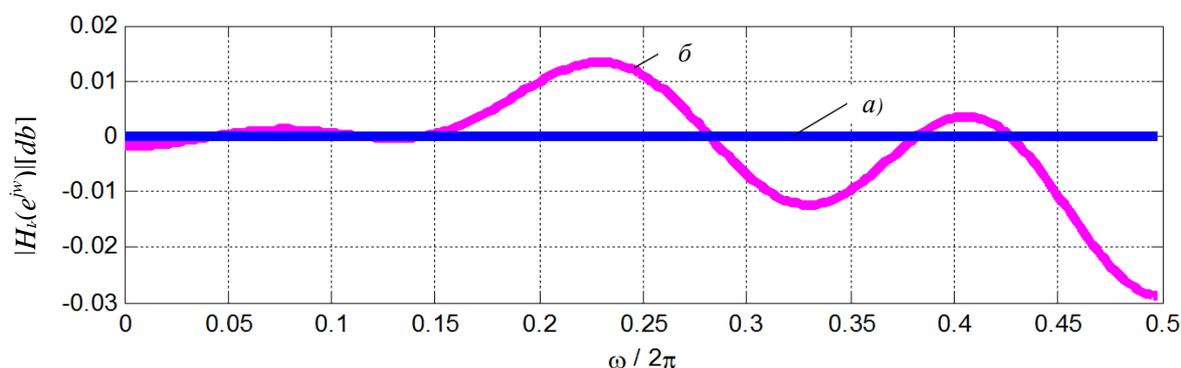


Рис. 7. АЧХ системы анализа-синтеза: *a*) преобразования ДКП-ОДКП с блочной лестничной структурной параметризацией; *б*) преобразования стандарта JPEG [9]

Проявление эффекта DC leakages в системе преобразования ДКП-ОДКП с блочной лестничной структурной параметризацией, обусловленного наличием постоянной составляющей во всех фильтрах ОДКП (см. рис. 6, *a*), что приводит к потере свойства регулярности первого рода<sup>1</sup> [6] банком фильтров ДКП-ОДКП, и в реконструированном исходном изображении проявляется эффект «шахматной доски». Для устранения данного недостатка в работе [14] предлагается преобразователь системы трансформационного кодирования по схеме L2L на основе диагональных матриц ДКП-ОДКП с использованием дополнительного информационного SIB-блока  $S_i$  размера  $M \times M$ . Формулы для прямого и обратного преобразований имеют следующий вид:

$$\begin{bmatrix} y_0 \\ \vdots \\ y_{n-1} \\ s_n \end{bmatrix} = \begin{bmatrix} \mathbf{C} & \cdots & \cdots & 0 \\ \vdots & \ddots & & \vdots \\ \vdots & & \mathbf{C} & \vdots \\ 0 & \cdots & \cdots & \mathbf{D}^n \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ \vdots \\ x_{n-1} \\ s_0 \end{bmatrix}, \quad \begin{bmatrix} x_0 \\ \vdots \\ x_{n-1} \\ s_0 \end{bmatrix} = \begin{bmatrix} \mathbf{D} & \cdots & \cdots & 0 \\ \vdots & \ddots & & \vdots \\ \vdots & & \mathbf{D} & \vdots \\ 0 & \cdots & \cdots & \mathbf{C}^n \end{bmatrix} \cdot \begin{bmatrix} y_0 \\ \vdots \\ y_{n-1} \\ s_n \end{bmatrix},$$

<sup>1</sup> Система отвечает условию регулярности первого рода, когда значения амплитудно-частотных характеристик всех канальных фильтров, кроме первого, равны нулю на нулевой частоте.

где  $C$  и  $D$  – матрицы прямого и обратного ДКП. Процесс обработки изображения на основе преобразования ДКП-ОДКП с использованием дополнительного информационного SIB-блока изображен на рис. 8. Блоки, показанные пунктирной линией, могут быть факторизованы согласно блочной лестничной структурной параметризации на основе ДКП-ОДКП (см. рис. 4).

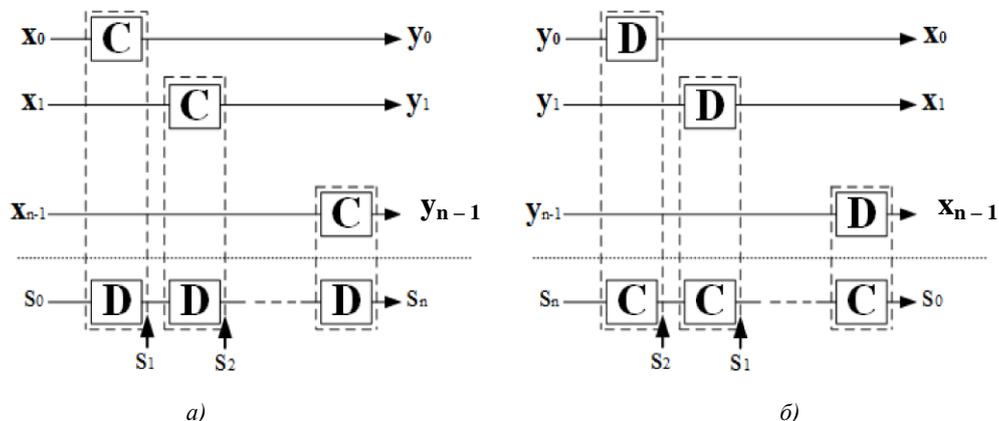


Рис. 8. Прямое (а) и обратное (б) преобразования с использованием SIB-блоков

Моделирование системы анализа-синтеза на основе блочной лестничной параметризации ДКП-ОДКП в программной среде MATLAB с помощью инструментального модуля Fixed Point с использованием арифметики с фиксированной запятой для исходного изображения Lena  $512 \times 512$  пикселей (рис. 9, а) показало, что без применения информационного SIB-блока  $S_n$  наблюдается эффект шахматной доски: PSNR=84,6834 дБ, MSE =  $2,2118 \times 10^{-4}$  (рис. 9, б), а с обработкой по схеме с SIB-блоком  $S_n$  PSNR=128,888 дБ, MSE =  $8,4001 \times 10^{-9}$ , т. е. исходное изображение восстанавливается без потерь (рис. 9, в).



Рис. 9. Реконструкция изображения Lena  $512 \times 512$  пикселей системой анализа-синтеза преобразования ДКП-ОДКП с блочной лестничной структурной параметризацией

### 3.3. Выбор архитектурного решения процессора преобразования ДКП-ОДКП с использованием SIB-блоков

#### 3.3.1. Универсальный рекурсивный процессор ДКП-ОДКП

Реализация преобразователя на основе ДКП-ОДКП с использованием дополнительного информационного SIB-блока для системы трансформационного кодирования изображений по схеме L2L предполагает применение рекурсивной структуры процессора (рис. 10), которая позволит за три процессорных цикла в режиме кодирования без потерь выполнить прямое и обратное преобразования блочной лестничной структурной параметризации ДКП-ОДКП согласно рис. 4. В режиме кодирования изображения с потерями используются результаты только первого цикла обработки.

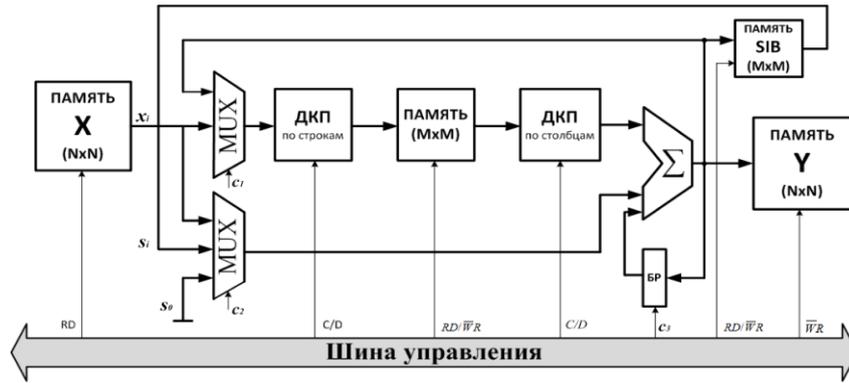


Рис. 10. Архитектура модуля преобразователя на основе ДКП-ОДКП для компрессии изображения по схеме L2L

Вычисление ДКП-ОДКП блока  $\mathbf{x}_i$   $M \times M$  входного изображения осуществляется по схеме разделимого преобразования столбцов и строк соответственно для прямого  $\mathbf{C}_{2D}\mathbf{x}_i \triangleq (\mathbf{C}(\mathbf{C}\mathbf{x}_i)^T)^T = \mathbf{C}\mathbf{x}_i\mathbf{C}^T$  и обратного  $\mathbf{D}_{2D}\mathbf{x}_i \triangleq (\mathbf{D}(\mathbf{D}\mathbf{x}_i)^T)^T = \mathbf{D}\mathbf{x}_i\mathbf{D}^T$  преобразований. Для этого используются два последовательно включенных через буфер памяти размера  $M \times M$  процессора ДКП, настраиваемых на выполнение прямого и обратного преобразований [15, 16]. Исходное изображение  $\mathbf{X}$  размера  $N \times N$  (память  $X$  исходного изображения) разделяется на блоки  $\mathbf{x}_i$  ( $0 \leq i \leq n-1$ ,  $n = (N/M)^2$ ); для каждого блока  $\mathbf{x}_i$  вычисляется ДКП  $\mathbf{y}_i = \mathbf{C}_{2D}\mathbf{x}_i$ ; параллельно с процессом обработки блоков  $\mathbf{x}_i$  выполняется ОДКП SIB-блока  $\mathbf{s}_i$ : для  $i=0$  SIB-блок  $\mathbf{s}_0$  задается нулевой матрицей, а для  $i = 1, 2, \dots, n-1$  SIB-блок  $\mathbf{s}_i$  будет являться итерационным преобразованием от  $\mathbf{s}_0$ :  $\mathbf{s}_i = \mathbf{D}_{2D}\mathbf{s}_{i-1}$  (см. рис. 8, а). Следует заметить, что SIB-блок  $\mathbf{s}_n \neq 0$  из-за ошибки округления на каждом шаге преобразования. Реконструкция исходного изображения осуществляется из всего потока ДКП коэффициентов  $\mathbf{y}_n$  (память  $Y$  размером  $N \times N$ ) и SIB-блока  $\mathbf{s}_n$ . Каждый блок  $\mathbf{x}_i$  и SIB-блок  $\mathbf{s}_i$  восстанавливаются на основе преобразования  $\mathbf{x}_i = \mathbf{D}_{2D}\mathbf{y}_i$  и  $\mathbf{s}_i = \mathbf{C}_{2D}\mathbf{s}_{i+1}$  соответственно без потерь (см. рис. 8, б). В режиме кодирования с потерями исходное изображение реконструируется на основе потока коэффициентов ДКП  $\mathbf{y}_n$  без SIB-блока  $\mathbf{s}_n$ .

### 3.3.2. Модифицированная схема Лофлера ДКП: лестничная схемная параметризация

Сокращение вычислительной сложности алгоритма ДКП очень важно для встраиваемых мультимедиа систем реального времени. Поэтому актуальной задачей является разработка структурных и схемотехнических решений реализации алгоритмов ДКП, которые позволяли бы уменьшить число операций умножения или избавиться от них вовсе [17]. Одним из наиболее эффективных способов реализации алгоритма ДКП является схема Лофлера [18], на основе которой алгоритмы 8-точечного или 16-точечного ДКП могут быть представлены с минимальным количеством операций умножения (рис. 11).

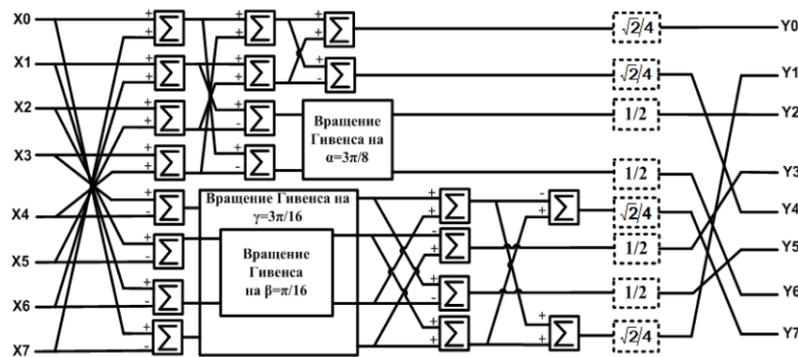


Рис. 11. Схема Лофлера 8-точечного ДКП второго типа

Основной операцией, которая вызывает трудности при аппаратной реализации, является операция умножения на поворачивающиеся множители (вращение Гивенса) – нетривиальные углы вращения. На основе лестничной схемной параметризации матрица вращения Гивенса представляется в виде следующей факторизации матриц:

$$\begin{bmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{bmatrix} = \begin{bmatrix} 1 & \tan(\alpha) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -\sin(2\alpha)/2 & 1 \end{bmatrix} \begin{bmatrix} 1/\cos(\alpha) & 0 \\ 0 & \cos(\alpha) \end{bmatrix}.$$

Например, для вращения на угол  $\alpha = 3\pi/8$  (рис. 12) коэффициенты данных матриц являются рациональными числами, которые можно аппроксимировать в виде суммы степеней двойки, что позволит заменить операции умножения на операции сдвига и сложения:

$$\begin{bmatrix} 1 & 77/32 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -45/128 & 1 \end{bmatrix} \begin{bmatrix} 128/49 & 0 \\ 0 & 49/128 \end{bmatrix} = \\ = \begin{bmatrix} 1 & (2^1 + 2^{-2} + 2^{-3} + 2^{-5}) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -(2^{-2} + 2^{-4} + 2^{-5} + 2^{-7}) & 1 \end{bmatrix} \begin{bmatrix} (2^1 + 2^{-1} + 2^{-4} + 2^{-5} + 2^{-6}) & 0 \\ 0 & (2^{-2} + 2^{-3} + 2^{-7}) \end{bmatrix}.$$

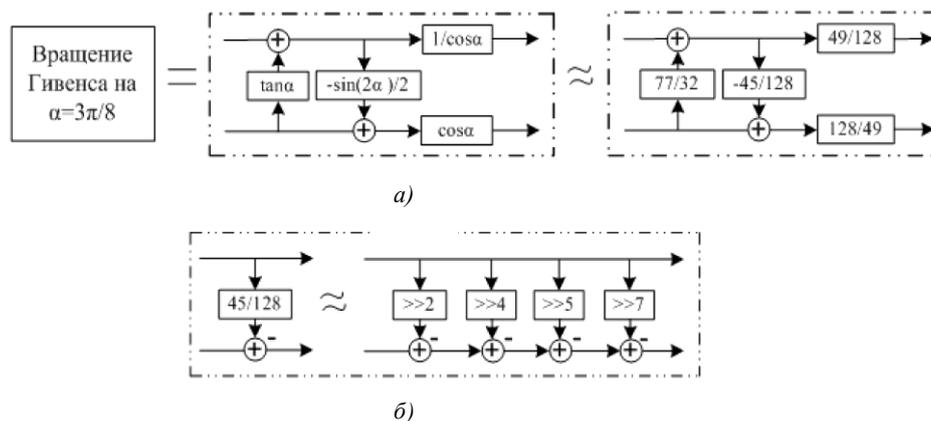


Рис. 12. Двухступенчатая лестничная схемная параметризация матрицы вращения Гивенса для угла  $\alpha = 3\pi/8$ :  
 а) представление коэффициентов лестничной схемы рациональными числами;  
 б) пример реализации ступени лестничной схемы на основе регистров сдвига

На рис. 13 показана модифицированная схема Лофлера алгоритма ДКП второго типа с лестничной схемной параметризацией на целочисленной арифметике [19]. Матрица  $[C]_{m,n}$  прямого 8-точечного ДКП второго типа с учетом шагов лестничной схемной параметризации вращений Гивенса представляется в виде произведения семи разреженных матриц [11]  $[C]_{m,n} = P \cdot G \cdot F \cdot E \cdot D \cdot C \cdot B \cdot A$ , каждая из которых описывает структуры ступеней модифицированной схемы Лофлера ДКП. Операции с матрицами **G** и **P** в схеме ДКП могут быть объединены и вынесены в блок квантования, что позволит уменьшить вычислительную сложность алгоритма ДКП (см. рис. 11). Для достижения высокой производительности процессора 8-точечного ДКП второго типа на основе лестничных структур предлагается использовать линейную многоступенчатую конвейерную схему. Каждой матрице **A**, **B**, **C**, **D**, **E**, **F**, **G** и **P** соответствует своя ступень процессора (рис. 13) [11]. В целом архитектура каждой ступени содержит два набора регистров и арифметические модули для вычисления согласно лестничной схемной параметризации матриц вращения Гивенса соответствующих ступеней модифицированной схемы Лофлера. Каждая ступень имеет восемь входов и восемь выходов, а также свои управляющие сигналы (рис. 13, б). Каждая ступень в VHDL-описании представлена в виде конечного автомата. В свою очередь, ступень – это отдельный компонент, выход которого подается на вход следующего компонента.

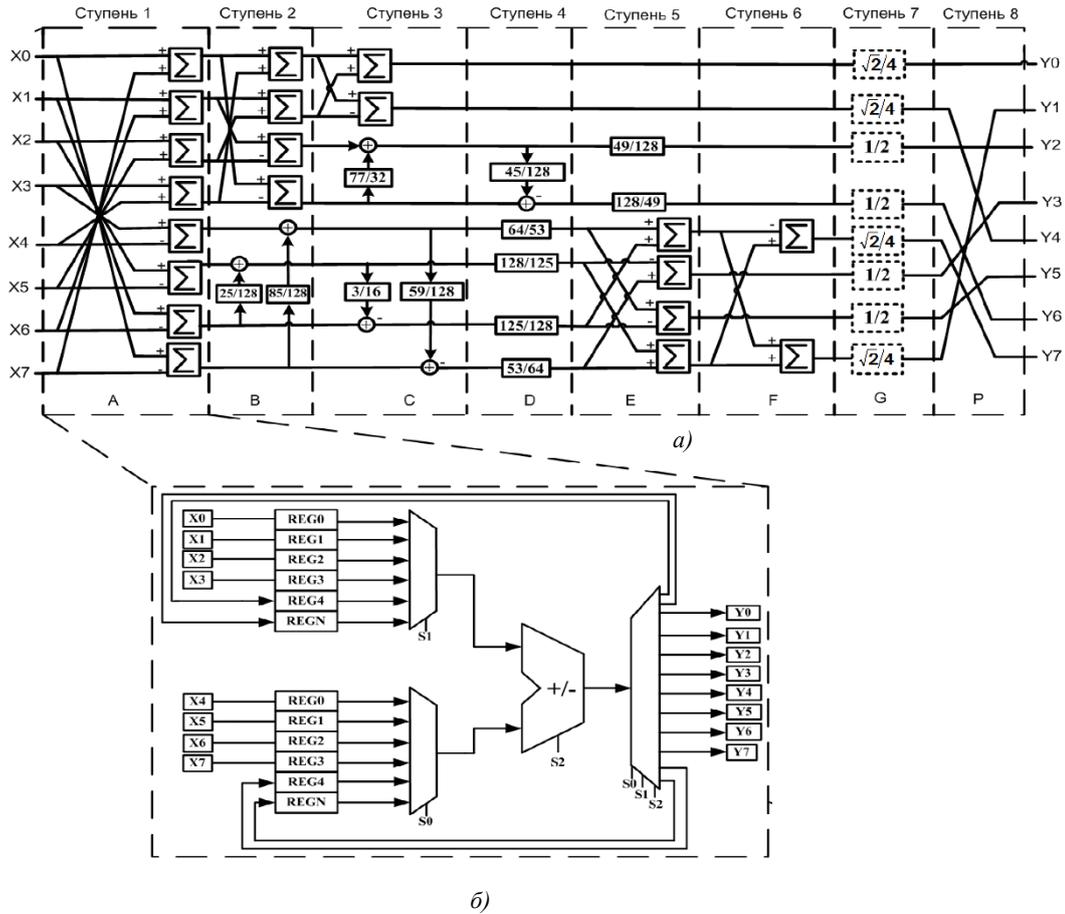


Рис. 13. Модифицированная схема Лофлера 8-точечного ДКП второго типа с лестничной схемной параметризацией на целочисленной арифметике: а) схема алгоритма (лестничная схемная параметризация матриц вращения Гивенса и представление коэффициентов лестничной схемы в виде рациональных коэффициентов); б) структура первой ступени

### 3.3.3. Реализация оборотов Гивенса на распределенной арифметике

Другим альтернативным решением при построении процессора ДКП, позволяющим исключить операции умножения, является применение распределенной арифметики для реализации вращения Гивенса на нетривиальный угол [20]. Прямой механизм распределенной арифметики предполагает вычисление выражения [20]

$$y = \sum_{k=1}^K a_k x_k,$$

где  $a_k$  – постоянные коэффициенты;  $x_k$  – входные нормированные переменные  $|x_k| \leq 1$ , которые в двоичной системе счисления в дополнительном коде представляются следующим образом:  $x_k = -b_{k0} + \sum_{l=1}^{B-1} b_{kl} 2^{-l}$ . Здесь  $B$  – разрядность слова;  $b_{kl} \in 0,1$ ,  $b_{k0}$  – бит знака переменной  $x_k$ ;  $b_{kB-1}$  – наименьший значащий бит. Принимая во внимание двоичное представление входных переменных  $x_k$ , выражение для  $y$  можно записать следующим образом:

$$y = \sum_{k=1}^K a_k \left[ -b_{k0} + \sum_{l=1}^{B-1} b_{kl} 2^{-l} \right] = \sum_{k=1}^K a_k (-b_{k0}) + \sum_{l=1}^{B-1} \left[ \sum_{k=1}^K a_k b_{kl} \right] 2^{-l}.$$

Таким образом, сумма  $\sum_{k=1}^K a_k x_k$  имеет  $2^K$  возможных значения, которые сохраняются в памяти. Двоичные коды входных переменных  $x_k$  могут быть использованы для прямой адресации

памяти. После  $B$  итераций в сумматоре-аккумуляторе будет результат выражения для  $y$ . Сокращение объема памяти в два раза ( $2^{K-1}$ ) можно получить, если входные переменные представить в смещенном двоичном дополнительном коде следующим образом [20]:

$x_k = \frac{1}{2} \left[ \sum_{l=0}^{B-1} c_{kl} 2^{-l} - 2^{-(B-1)} \right]$ , где  $c_{kl} = b_{kl} - \bar{b}_{kl}$ ,  $c_{k0} = -(b_{k0} - \bar{b}_{k0})$ ,  $\bar{b}_{kn}$  – инверсные значения битов  $b_{kn}$ ,  $c_{kl} \in -1, 1$ . Следовательно, результат  $\mathbf{y} = [y_1, y_2]^T$  вычисления операции вращения Гивенса на угол  $\alpha$   $y_1 = cx_1 + sx_2$ ,  $y_2 = -sx_1 + cx_2$ , где  $c = \cos(\alpha)$ ,  $s = \sin(\alpha)$ , с помощью распределенной арифметики можно представить в следующем виде:

$$y_1 = \sum_{l=0}^{B-1} (cc_{1l} + sc_{2l}) 2^{l-1} - (c+s) 2^{-B};$$

$$y_2 = \sum_{l=0}^{B-1} (-sc_{1l} + cc_{2l}) 2^{l-1} - (c-s) 2^{-B}.$$

Анализ данного выражения показывает, что для вычисления вектора  $\mathbf{y} = [y_1, y_2]^T$  необходимы два модуля памяти (два постоянных запоминающих устройства (ПЗУ)) на два слова каждый, а для схемы Лофлера, где происходят три вращения Гивенса на нетривиальный угол, понадобится шесть ПЗУ, которые могут быть реализованы на регистрах. При последовательном побитном приеме входных данных требуется  $B$  итераций. Однако в процессе вычисления формируются все компоненты  $y_1, y_2$  вектора  $\mathbf{y}$ , поэтому вычисление с использованием механизма распределенной арифметики является более эффективным по временным затратам, чем применение в этих же целях параллельного умножителя.

Поскольку поступление переменных  $x_k$  происходит по одному биту за один такт, такие схемы называются 1-ВААТ (one-bit-at-a-time) [20]. Распределенная арифметика позволяет распараллеливать процесс обработки, когда анализируется  $L$  бит за такт. С использованием данного подхода, исходя из условия кратности  $L$  и разрядности  $B$  представления переменных  $x_k$ , можно синтезировать, к примеру, варианты схем 1-ВААТ, 2-ВААТ, 4-ВААТ, ...,  $B$ -ВААТ, среди которых можно выбрать оптимальную реализацию одним из методов нелинейного программирования, задав определенные ограничения целевой функции, например минимум потребляемого тока, при выбранной постоянной частоте дискретизации [21, 22].

### 3.3.4. Анализ эффективности реализации ДКП на FPGA Virtex4 xc4vlx25-10ff668

На ПЛИС FPGA Virtex4 xc4vlx25-10ff668 была осуществлена реализация трех архитектурных решений процессора ДКП:

- модифицированной схемы Лофлера 8-точечного ДКП второго типа с лестничной схемой параметризацией на целочисленной арифметике;
- модифицированной схемы Лофлера 8-точечного ДКП второго типа с применением распределенной арифметики для реализации вращения Гивенса на нетривиальный угол (1ВААТ);
- процессора ДКП-ОДКП с использованием дополнительного информационного SIB-блока (блочной лестничной факторизации) – универсального рекурсивного процессора ДКП-ОДКП для компрессии изображений по схеме L2L. Архитектурные решения реализованы на языке описания аппаратуры VHDL [23].

В таблице приведены результаты синтеза структур ДКП на FPGA Virtex4 xc4vlx25-10ff668, анализ которых показывает, что конвейерная схема процессора ДКП имеет высокую пропускную способность, однако аппаратные затраты выше по сравнению с другими решениями. Это обусловлено тем, что конвейер процессора состоит из шести ступеней, каждая из которых является отдельным компонентом при синтезе процессора ДКП.

Преимущество реализации архитектуры процессора ДКП на распределенной арифметике (вариант 1-ВААТ) заключается в уменьшении аппаратных затрат, но при этом снижается скорость вычислений. Как отмечалось выше, распределенная арифметика позволяет распараллели-

вать процесс обработки (варианты L-ВААТ). При этом будет наблюдаться рост аппаратных затрат, зато можно искать компромиссные решения.

Результаты синтеза структур ДКП на FPGA Virtex4 xc4vlx25-10ff668

Архитектура процессора ДКП	Количество регистров	Количество четырех-входовых таблиц LUT	Максимальная тактовая частота, МГц	Среднеквадратическая ошибка	Производительность процессора, количество 8-точечных ДКП в секунду
Конвейерная схема (лестничная схемная параметризация)	1054	1594	345,781	$1,16 \times 10^{-5}$	$6,65 \times 10^6$
Распределенная арифметика (1ВААТ)	152	507	210,615	$3,43 \times 10^{-5}$	$9,57 \times 10^6$
Процессор ДКП-ОДКП и SIB-блок (блочная лестничная факторизация)	463	619	201,375	$8,40 \times 10^{-9}$	$7,76 \times 10^6$

Реализация универсального рекурсивного процессора ДКП-ОДКП для компрессии изображений по схеме L2L с использованием дополнительного информационного SIB-блока (блочная лестничная факторизация) требует меньше ресурсов кристалла FPGA, чем конвейерная схема процессора ДКП. Отличительной особенностью этого процессора является применение операционного блока, выполняющего операции как прямого, так и обратного ДКП [16], а также то, что данную схему можно использовать для обработки исходных изображений в двух режимах: с потерями и без потерь данных.

### 3.4. Реализация прототипа FPSoC-кодера изображения на основе софт-процессора MicroBlaze и процессора ДКП-ОДКП

Для реализации прототипа устройства используется пакет программ Xilinx EDK (Embedded Development Kit), применяемый для разработки встроенных систем на основе софт-процессора MicroBlaze. Пакет EDK состоит из двух независимых сред разработки – XPS и SDK. Разработчик использует XPS (Xilinx Platform Studio) для создания и конфигурирования аппаратной спецификации встроенной системы (процессорного ядра, управления памятью, интерфейсов ввода-вывода и др.). Пакет Xilinx SDK (Software Development Kit), основанный на Eclipse, предназначен для разработки и отладки программ на C/C++. Имеется специальный симулятор ISS (Instruction Set Simulator), благодаря которому можно тестировать программу либо в режиме симулятора, либо на основе отладочного модуля Xilinx ML-401 с ПЛИС XC4VLX25-FF668-10.

Процессор ДКП-ОДКП реализован на языке описания аппаратуры VHDL. VHDL-файлы описывают архитектуру преобразования в виде периферийного устройства, которое подключается к ядру софт-процессора MicroBlaze по быстродействующей шине FSL (вариант I) или PLB (вариант II: обмен данными под управлением контроллера DMA) (рис. 14, а) [24, 25]. В структуру FPSoC также включены контроллеры многопортовой памяти, сети Ethernet, дисплея VGA (640×480 пикселей), прерываний, клавиатуры, последовательной передачи данных (UART). Основная задача управляющего софт-процессора MicroBlaze заключается в вводе исходных данных по сети, выполнении этапа квантования и кодирования коэффициентов преобразования

ДКП-ОДКП, программа которого написана на языке C, выводе результатов преобразования на VGA-дисплей, а также в передаче их на персональный компьютер для последующего анализа.

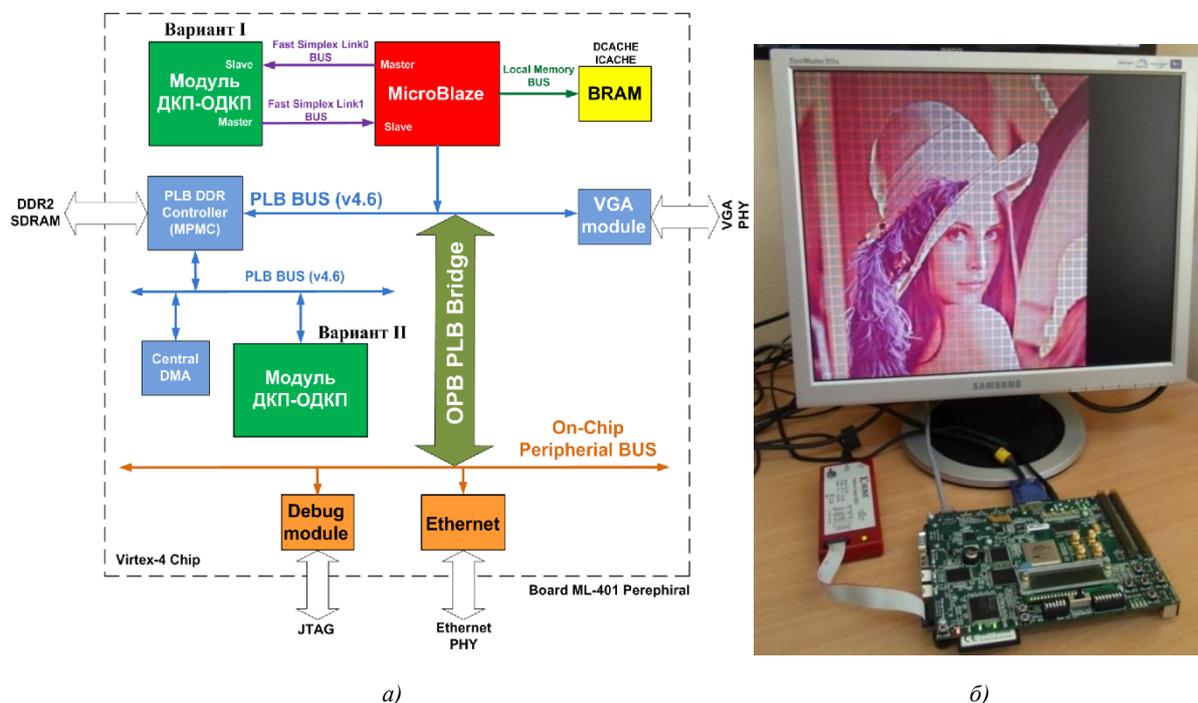


Рис. 14. FPSoC-компрессии изображений на ПЛИС XC4 VLX25-FF668-10 (а) и инструментальный стенд на Xilinx ML-401 (б)

В отладочный модуль Xilinx ML-401 загружается проект FPSoC, подготовленный в виде bit-файла с помощью JTAG-интерфейса. После загрузки bit-файла в ПЛИС стартует софт-процессор под управлением программы, записанной в блочную память. В данном случае там записан стандартный загрузчик, ожидающий управляющую программу через JTAG-интерфейс. Такой подход исключает необходимость разработки собственного загрузчика из флеш-памяти или другого источника, что ускоряет процесс отладки прототипа системы. Управляющая программа, разработанная на языке C, выполняет инициализацию периферийных устройств и ввод исходных данных по сети. Для работы с сетью не требуется полноценный стек TCP/IP (Transmission Control Protocol/Internet Protocol), достаточно реализации транспортного UDP/IP (User Datagram Protocol/Internet Protocol). При этом со стороны персонального компьютера требуется лишь соответствующая запись в таблице ARP (Address Resolution Protocol) с адресом MAC (Media Access Control) платы Xilinx ML-401. Со стороны отладочного модуля Ethernet-фрейм формируется полностью программным образом и для корректной доставки данных потребуется лишь правильный формат и подсчет контрольной суммы UDP-пакета. Таким образом, данное сетевое соединение, несмотря на свои ограничения как по функциональности, так и по скорости, позволяет работать с пользовательскими программами на персональном компьютере и требует меньше усилий по сравнению с полноценной сетевой системой для софт-процессора MicroBlaze. Результаты кодирования изображения передаются по сети и выводятся в буфер динамической памяти, из которого через контроллер TFT передаются в VGA-интерфейс для отображения на экране монитора (рис. 14, б).

### Заключение

С помощью функционально-архитектурного проектирования в статье по шагам описана реализация системы быстрого прототипирования процессоров дискретного косинусного преобразования для трансформационного кодирования изображений по схеме L2L на ПЛИС с архитектурой FPGA. Предложен стенд для исследования и быстрого создания прототипа подобных

систем, где разрабатываемые архитектуры процессоров ДКП подключаются в виде модулей периферийных устройств или сопроцессоров и результат преобразования выводится на экран через VGA-интерфейс. Данная технология обеспечивает гибкость в поиске компромиссных решений при реализации программируемых систем на кристалле FPSoC реального времени по таким параметрам, как аппаратная сложность, производительность и точность вычислений. Проектировщик может исследовать ряд архитектурных решений, например, преобразователя трансформационного кодирования изображений по схеме L2L и выбрать более подходящее требованиям спецификации. Предложена архитектура универсального рекурсивного процессора ДКП-ОДКП FPSoC реального времени, работающего в режимах сжатия изображений с потерями и без потерь, которая может рассматриваться как базовая при построении кодеров класса L2L. Данная методология быстрого прототипирования встраиваемых программируемых систем на ПЛИС для мультимедийных приложений реализована как высокоуровневая инструментальная система, которая позволяет получать IP-компоненты пользователя.

### Список литературы

1. Spanias, A. Audio Signal Processing and Coding / A. Spanias, T. Painter, V. Atti. – Hoboken : Wiley, 2007. – 486 p.
2. Гонсалес, Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс. – М. : Техносфера, 2005. – 1072 с.
3. A real time image processing subsystem: GEZGIN / N. Ismailoglu [et al.] // Proc. of the 16th Annual AIAA/USU Conf. on Small Satellites. – Logan, 2002. – Vol. SSC02-VIII-3. – P. 10.
4. Using FPGA and JAVA in rapid prototyping of a real-time H.264/AVC decoder / A. Petrovsky [et al.] // Zeszyty Naukowe Telekomunikacja i Elektrotechnika. – 2009. – Vol. 2009, № 12. – P. 43–55.
5. Suzuki, T. Realization of lossless-to-lossy image coding compatible with JPEG standard by direct-lifting of DCT-IDCT / T. Suzuki, M. Ikehara // 17th IEEE Intern. Conf. on Image Processing (ICIP). – Hong Kong, 2010. – P. 389–392.
6. Vaidyanathan, P.P. The role of lossless systems in modern digital signal processing: A tutorial / P.P. Vaidyanathan, Z. Doğanata // IEEE Trans. Educ. – 1989. – Vol. 32, № 3. – P. 181–197.
7. Зотов, В.Ю. Проектирование встраиваемых микропроцессорных систем на основе ПЛИС фирмы Xilinx / В.Ю. Зотов. – М. : Горячая линия – Телеком, 2006. – 520 с.
8. MicroBlaze Processor Reference Guide Embedded Development Kit EDK 14.5 UG081 (v14.5) [Электронный ресурс]. – 2015. – Режим доступа : [http://www.xilinx.com/support/documentation/sw\\_manuals/mb\\_ref\\_guide.pdf](http://www.xilinx.com/support/documentation/sw_manuals/mb_ref_guide.pdf). – Дата доступа : 20.06.2015.
9. Немудров, В. Система-на-кристалле. Проектирование и развитие / В. Немудров, Г. Мартин. – М. : Техносфера, 2004. – 216 с.
10. Иванюк, А.А. Проектирование встраиваемых цифровых устройств и систем / А.А. Иванюк. – Минск : Бестпринт, 2012. – 337 с.
11. Петровский, А.А. Быстрое проектирование систем мультимедиа от прототипа / А.А. Петровский, А.В. Станкевич. – Минск : Бестпринт, 2011. – 412 с.
12. FPGA-based Implementation of Signal Processing Systems / R. Woods [et al.]. – Chichester : Wiley, 2008. – 364 p.
13. Rao, K. Techniques and Standards for Image, Video, and Audio Coding / K. Rao, J. Hwang. – Upper Saddle Rive : Prentice Hall, 1996. – 563 p.
14. Suzuki, T. Integer DCT based on direct-lifting of DCT-IDCT for lossless-to-lossy image coding / T. Suzuki, M. Ikehara // IEEE Trans. Image Process. – 2010. – Vol. 19, № 11. – P. 2958–2965.
15. A high-throughput and area-efficient video transform core with a time division strategy / Y.-H. Chen [et al.] // IEEE Trans. VLSI Syst. – 2013. – Vol. PP, № 99. – P. 10.
16. Ключеня, В.В. Модуль вычисления прямого и обратного ДКП для систем сжатия данных / В.В. Ключеня, А.А. Петровский // Труды 16-й Междунар. конф. «Цифровая обработка сигналов и ее применение» (DSPA'2014). – М., 2014. – Т. 2. – С. 642–645.

17. Миано, Д. Форматы и алгоритмы сжатия изображений в действии / Д. Миано. – М. : Триумф, 2003. – 336 с.
18. Loeffler, C. Practical fast 1-D DCT algorithms with 11 multiplications / C. Loeffler, A. Lightenberg, G. Moschytz // Proc. IEEE Intern. Conf. Acoust., Speech, Signal Process. (ICASSP). – Glasgow, 1989. – Vol. 2. – P. 988–991.
19. Kliuchenia, V. FPGA architecture of DCT processor based lifting steps for real-time embedded applications / V. Kliuchenia // Proc. 11th Intern. Conf. Pattern recognition and information processing (PRIP'2011). – Minsk, 2011. – P. 259–262.
20. White, S.A. Applications of distributed arithmetic to digital signal processing: A tutorial review / S.A. White // IEEE ASSP Mag. – 1989. – Vol. 6, № 3. – P. 4–19.
21. Fox, T.W. Rapid prototyping of field programmable gate array-based discrete cosine transform approximations / T.W. Fox, L.E. Turner // EURASIP Journal on Advances in Signal Processing. – 2003. – Vol. 2003, № 6. – P. 543–554.
22. Ключеня, В.В. Выбор оптимальной реализации структурного решения процессора ДКП на распределенной арифметике / В.В. Ключеня, А.А. Петровский // Доклады БГУИР. – 2010. – № 7. – С. 66–75.
23. Бибило, П.Н. Основы языка VHDL : учеб. пособие / П.Н. Бибило. – 6-е изд. – М. : ЛИБРОКОМ, 2014. – 328 с.
24. Rosinger, H.-P. Connecting customized IP to the microblaze soft processor using the fast simplex link (FSL) channel : Tech. Rep. XAPP 529 / H.-P. Rosinger // [Electronic resource]. – 2004. – Mode of access : [http://www.xilinx.com/support/documentation/application\\_notes/xapp529.pdf](http://www.xilinx.com/support/documentation/application_notes/xapp529.pdf). – Date of access : 29.06.2015.
25. Петровский, Н.А. Процессор обработки изображения на многополосном вейвлет-преобразовании в алгебре кватернионов / Н.А. Петровский // Доклады БГУИР. – 2011. – Т. 61, № 7. – С. 76–82.

Поступила 05.06.2015

*Белорусский государственный университет  
информатики и радиоэлектроники,  
Минск, П. Бровки, 6  
e-mail: vitaly.kliuchenia@gmail.com,  
nick@petrovsky.eu*

**V.V. Kliuchenia, N.A. Petrovsky**

## **RAPID PROTOTYPING OF EMBEDDED PROGRAMMABLE SYSTEMS ON FPGA FOR MULTIMEDIA APPLICATIONS**

This paper illustrates rapid prototyping method applied for mobile multimedia systems with L2L (lossless-to-lossy) compression scheme. Debug module Xilinx ML-401 based on the FPGA (Filed Programmable Gate Array) is taken as the basis. Microblaze soft-processor is used as the main control unit. Proposed DCT-IDCT cores are developed using VHDL hardware description language as FSL (Fast Simplex Link) coprocessors for Microblaze or standalone PLB (Processor Logical Bus) acceleration peripheral. Result of image processing is displayed on the screen connected via VGA interface. Proposed recursive architecture of DCT-IDCT core can be used as a basis of L2L coder.