

## КВАНТОВАНИЕ И УПАКОВКА КОЭФФИЦИЕНТОВ СУБПОЛОС В КОДЕРЕ ИЗОБРАЖЕНИЙ НА ОСНОВЕ ПАРАУНИТАРНОГО БАНКА ФИЛЬТРОВ

Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь

Арабей К. В.

Петровский А. А. – д-р. техн. наук, профессор

В последнее время банки фильтров рассматриваются как одна из наиболее эффективных техник компрессии мультимедиа данных. Банки фильтров широко применяется в области кодирования видео и аудио сигналов а также изображений.

Существующие форматы кодирования изображений, например JPEG 2000, могут осуществлять компрессию данных, как с потерями, так и без потерь. При этом для каждого типа компрессии необходимо проводить отдельную процедуру трансформационного преобразования, что требует дополнительных вычислительных затрат. Кодер изображений на основе параунитарного банка фильтров, позволит использовать одно трансформационное преобразование как для компрессии с потерями, так и для компрессии без потерь.

Банк фильтров – цифровая система, состоящая из банка фильтров анализа и банка фильтров синтеза (рис. 1). Входной сигнал  $x(n)$ , который представляется последовательностью отсчетов, разбивается на  $M$  субполосных составляющих при помощи фильтров блока анализа  $H_k(z)$  ( $k = 0, 1, \dots, M-1$ ). В идеальном случае эти составляющие в частотной области не перекрываются. Операции, выполняемые блоком синтеза, являются обратными операциями блока анализа. Подобранным соответствующим образом набор фильтров блока синтеза  $F_k(z)$  ( $k = 0, 1, \dots, M-1$ ), можно восстановить исходный сигнал  $y(n)$  из его субполосных компонент.

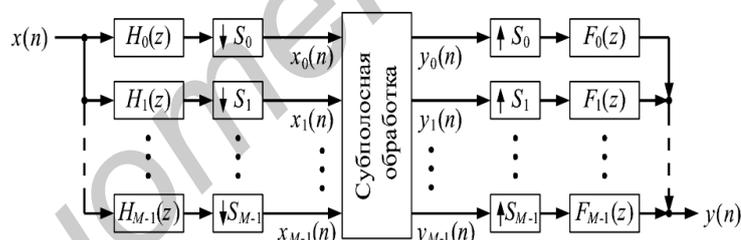


Рис. 1 – Банк фильтров: система анализа/синтеза сигнала

Параунитарный (ортогональный) банк фильтров (ПУБФ) – банк фильтров, у которого передаточные функции анализирующих и синтезирующих фильтров и их соответственно смещенные версии ортогональны друг другу. Фильтры синтеза в параунитарных банках являются транспонированными версиями фильтров анализа:

$$F_k z = H_k^T z^{-1}$$

При соблюдении этого условия обеспечивается возможность перфективной реконструкции банком фильтров входного сигнала  $x(n)$ .

Преобразование при помощи параунитарного банка фильтров не осуществляет сжатия данных, оно необходимо для подготовки информации к этапам компрессии с потерями упаковки. При работе с изображениями преобразование идет в два этапа: сначала по строкам, потом по столбцам. После преобразования основная часть информации концентрируется в низкочастотных субполосах, высокочастотные полосы представляют матрица, в которой многие коэффициенты либо близки, либо равны нулю. Кроме того, благодаря несовершенству человеческого зрения можно аппроксимировать коэффициенты более грубо без заметной потери качества изображения. Для этого используется квантование коэффициентов. В самом простом случае - это арифметический побитовый сдвиг вправо. При этом преобразовании теряется часть информации, но может достигаться большая степень сжатия.

Обычно в алгоритмах кодирования изображений используется скалярное квантование. Числовая прямая разбивается на отрезки равной длины. Далее, квантуемый элемент, попадающий в какой-то отрезок, заменяется центром этого отрезка. Для каждой субполосы выбирается свой коэффициент квантования, что позволяет регулировать степень сжатия изображения. Как правило, их значения для субполос растут по направлению слева направо и сверху вниз. Операция квантования является единственной фазой, где

происходит потеря информации.

Далее матрицы коэффициентов субполос переводится в вектор при помощи "зигзаг"-сканирования. Полученный вектор свертывается с помощью алгоритма группового кодирования (RLE). Алгоритм RLE заключается в следующем: любой последовательности повторяющихся входных символов ставится в соответствие набор из двух выходных символов: первый байт, определяющий длину входной последовательности, второй — сам входной символ.

На следующем этапе проводится Кодирование по Хаффману. В основе алгоритма кодирования Хаффмана лежит довольно простой принцип: символы заменяются кодовыми последовательностями различной длины. Чем чаще используется символ, тем короче должна быть кодовая последовательность. Кодовая таблица строится на основе статистического анализа имеющейся информации. Код Хаффмана обладает свойствами префиксных кодов и легко может быть свернут обратно в последовательность длин серий.

Сжимая файл по алгоритму Хаффмана, первое, что нужно сделать - подсчитать сколько раз встречается каждый символ. После подсчета частоты вхождения каждого символа, необходимо построить таблицу кодов и сформировать мнимую компоновку между кодами по убыванию. Для восстановления первоначального файла, нужно иметь декодирующую таблицу, так как они будут различны для разных файлов. Следовательно, необходимо сохранять таблицу вместе с файлом.

Для реализации вышеописанных алгоритмов был использован язык технических вычислений MatLab. Таким образом, была разработана модель кодера изображений на основе параунитарного банка фильтров в ряде случаев превосходящая по скорости вычислений и степени сжатия уже существующие кодеры.

Список использованных источников:

1. Петровский, А. А. Цифровые банки фильтров: анализ, синтез и применение в мультимедиа системах : учебно-методическое пособие / А.А. Петровский [и др.]. – Минск, БГУИР, 2006. – 82 с.
2. Парфенюк, М. Параунитарные банки фильтров на основе алгебры кватернионов: теория и применение/ Парфенюк М., Петровский А.А - Минск, БГУИР, 2008. – 14с.
3. Гонсалес, Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс. – М. : Техносфера, 2005. – 1072 с.

## ОПИСАНИЕ ПРОЦЕССОРНОГО ЯДРА MOTOROLA MC6800 НА ЯЗЫКЕ ОПИСАНИЯ АППАРАТУРЫ VHDL

*Белорусский государственный университет информатики и радиоэлектроники  
г. Минск, Республика Беларусь*

*Винокуров В. В.*

*Герасимович В. Ю. – аспирант, профессор*

Рассмотрен один из методов проектирования процессорного ядра Motorola 6800 с использованием языка описания аппаратуры, показаны основные проблемы на различных этапах проектирования.

### **Введение**

Микропроцессоры представляют собой сложную структуру связанных между собой различных компонентов цифровой техники. Проектирование процессорного ядра является очень трудоемкой задачей, которая занимает колоссальное количество времени и человеческого труда. Без соответствующих САПР, этот процесс занимало бы не один год и увеличивало себестоимость в несколько раз, что в современном темпе развития техники, делает продукт полностью не конкурентоспособным. Одним из главных методов проектирования аппаратуры, является проектирование с использованием языков описания интегральных схем.

Основная цель моей работы – на практике, опробовать VHDL в качестве инструмента для проектирования микропроцессоров, ознакомиться с основными проблемами, с которыми сталкиваются начинающие системотехники. Проектируемая модель в точности повторяет архитектуру известного микропроцессора Motorola 6800. Motorola 6800 — микропроцессор, разработан и выпущен компанией Motorola в 1974 году. Это был первый микропроцессор с индексным регистром.

### **Основная часть**

Перед началом реализации процессора на VHDL, требовалось как можно точнее изучить внутреннюю архитектуру и принципы работы ядра MC6800. В качестве основного источника информации использовалась официальная документация на микропроцессор, в которой были описаны основные режимы работы процессора, а также в общих чертах показана его структура. Вот тут и появляется первая проблема, для молодых специалистов.

Любая крупная IT фирма всегда утаивает ключевые аспекты построения своей архитектуры. Motorola не исключение. В частности, отсутствовала какая-либо информация об организации внутренних шин ядра и взаимодействии между собой основных компонентов. В таком случае проектирование происходит на основе