

Метод цифрового волновода впервые был использован в первом цифровом синтезаторе "Yamaha - VL1" 1994 года выпуска. С момента своего открытия и до сих пор метод используется в современных синтезаторах передовых компаний, таких как "Yamaha", Korg, Sakewalkи т.д.

Список использованных источников:

1. David A. Jaffe and Julius O. Smith Extensions of the Karplus-Strong Plucked-String Algorithm – Computer Music Journal, Vol.7, No.2 (Summer, 1983), pp. 56-59
2. Kevin Karplus and Alex Strong, Digital Synthesis of Plucked-String and Drum Timbres – Computer Music Journal, Vol.7, No.2 (Summer, 1983), pp.43-55
3. Блок-схема улучшенного алгоритма Карплуса-Стронга [Электронный ресурс]. – Режим доступа: [https://ccrma.stanford.edu/realsimple/faust\\_strings/Extended\\_Karplus\\_Strong\\_Algorithm.html](https://ccrma.stanford.edu/realsimple/faust_strings/Extended_Karplus_Strong_Algorithm.html)

## ПРИМЕНЕНИЕ ОТЛАДОЧНОЙ ПЛАТЫ ZEDBOARD ДЛЯ РЕАЛИЗАЦИИ 8-ТОЧЕЧНОГО БПФ НА ОСНОВЕ ЛЕСТНИЧНОЙ СХЕМЫ

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

Качинский В.М.

Станкевич А.В. – к.т.н., доцент

В настоящее время среди различных применений встраиваемых вычислительных систем реального времени важное место занимают системы цифровой обработки сигналов (ЦОС). Они широко используются при проектировании различных компонентов систем мультимедиа, таких как кодирование аудио- и речевых сигналов, шумоподавление, а также в ряде других приложений.

Одним из основных алгоритмов, используемых при построении встраиваемых вычислительных систем обработки мультимедиа данных, является алгоритм дискретного преобразования Фурье (ДПФ). Однако реализация ДПФ требует выполнения большого количества операций. Так, при длине выборки сигнала  $N$  прямое вычисление ДПФ требует  $N^2$  комплексных умножений и такого же количества сложений. Поэтому на практике используют различные алгоритмы быстрого вычисления ДПФ, например, алгоритм быстрого преобразования Фурье (БПФ), которые позволяют существенно уменьшить вычислительную сложность реализации ДПФ. Для вычисления ДПФ были разработаны различные эффективные схемы, использующие преимущества свойств симметрии и периодичности значений  $e^{j2\pi k/N}$ , такие как БПФ по основанию 2, БПФ по основанию 4 и БПФ по расщепленному основанию [1].

В настоящее время одним из основных компонентов при построении встраиваемых вычислительных систем обработки мультимедиа данных являются программируемые логические схемы с архитектурой FPGA, которые обеспечивают высокую производительность и гибкость структурной организации при реализации алгоритмов ЦОС [2].

При проектировании встраиваемых цифровых систем, работающих в реальном масштабе времени, особое внимание уделяется таким показателям как скорость вычислений, аппаратные затраты, сложность алгоритмов выполнения арифметических операций. Поэтому большинство систем такого класса выполняется на основе арифметики с фиксированной запятой, которая позволяет получить хорошие значения приведенных выше показателей. Во многих приложениях используется свойство обратимости ДПФ для того, чтобы можно было выполнять полную реконструкцию сигнала. Однако использование фиксированной запятой приводит к нарушению свойства обратимости, так как выполняется квантование коэффициентов ДПФ. Такая проблема отсутствует при вычислениях с плавающей запятой, однако операции в арифметике с плавающей запятой сложны в реализации [2, 3].

Одним из путей решения данной проблемы является использование лестничных структур при реализации алгоритмов БПФ в системах мультимедиа [2, 3]. Использование лестничных структур позволяет при использовании фиксированной запятой (целочисленной арифметики) выполнять восстановление сигнала без потерь, и кроме того уменьшить необходимое для реализации алгоритмов количество арифметических операций. Пример лестничной схемы для выполнения операции комплексного умножения представлен на рисунке 1 [3].

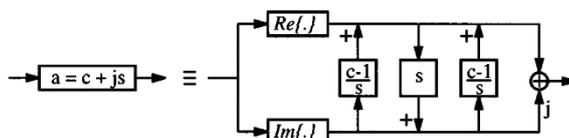


Рис.1 – Пример лестничной схемы для выполнения операции комплексного умножения

В докладе рассматриваются вопросы реализации 8-точечного БПФ с расщепленным основанием на основе лестничной схемы. Вопросы реализации моделировались с помощью системы Matlab. Для аппаратной реализации данного алгоритма использовалась отладочная плата Zedboard фирмы Avnet. Zedboard реализована на системе-на-кристалле Zynq-7020 фирмы Xilinx, которая включает в себя программируемую логическую схему с архитектурой FPGA Artix-7, а также двухъядерный процессор ARM Cortex-A9. FPGA содержит 13300 секций, каждая из которых состоит из 4 шестивходовых элементов LUT и восьми триггеров, 140 блоков RAM по 36 Кб и 220 специализированных арифметических элементов DSP48E1s. Каждый элемент DSP48E1s состоит из умножителя 18x25 в дополнительном коде и 48-битного аккумулятора, которые могут работать на частоте до 741 МГц [4].

Список использованных источников:

1. Опленгейм, А.В. Цифровая обработка сигналов / А.В. Опленгейм, Р.В. Шафер. – М.: Связь, 1979. – 200 с.
2. Петровский, Ал.А. Быстрое проектирование систем мультимедиа от прототипа / Ал.А. Петровский, А.В. Станкевич, А.А. Петровский. – Минск : Бестпринт, 2011. – 412 с.
3. Oraintara, S. Integer Fast Fourier Transform / S. Oraintara, Y. Chen, T. Q. Nguyen // IEEE Transactions on Signal Processing. - Vol. 50, no. 3, March 2002. - pp. 607–618.
4. Crockett L.H., Elliot R.A., Enderwitz M.A., Stewart R.W., *The Zynq Book: Embedded Processing with the ARM CortexA9 on the Xilinx Zynq-7000 All Programmable SoC*, First Edition / Strathclyde Academic Media, 2014. – 460 p.

## СИСТЕМА КОМПРЕССИИ ИЗОБРАЖЕНИЙ С ТЕКСТОВЫМ НАПОЛНЕНИЕМ

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

*Власенко А.А.*

*Азаров И.С. – д.т.н., доцент*

В настоящее время множество книг хранится в электронном виде. Большая часть из них хранится в виде изображений с текстом, которые занимают большое количество дискового пространства. Большинство алгоритмов, которые эффективны для полутоновых и полноцветных изображений, таких как портреты, пейзажи не эффективны для изображений текста.

Лучшие алгоритмы для компрессии изображений текста основаны на разбиении изображений на символы и их классификации. Символы разбиваются на классы так, чтобы одинаковые символы были записаны в один класс. Идея заключается в том, что достаточно хранить только один элемент класса. Также необходимо иметь карту размещения символов для восстановления изображения.

- 1) В лучшем случае должно одновременно выполнялись два условия:
  - 2) В каждом классе находятся изображения только одного и того же символа;
- Все изображения символа находятся в одном классе.

Очень редко эти условия одновременно выполняются. Это происходит потому, что сканируют страницы с шумами, которые возникают при печати страницы. Влияние шумов необходимо минимизировать.

Для этого можно использовать алгоритм, который учитывает влияние шумов при печати. Чтобы разделить изображения символов на классы нужно:

- 1) Выбрать меру отличий двух символов
- 2) Выбрать алгоритм разбиения на классы.

Для второго пункта можно использовать алгоритм «просеивания». Он состоит в том, что к не классифицированному элементу присоединяются не классифицированные элементы, которые близки к нему согласно мере отличия двух символов. Элемент и все к нему присоединенные образуют класс. Алгоритм заканчивает работу, когда все элементы оказываются классифицированными.

Мера отличия изображений  $S_1$  и  $S_2$  после наложения изображений символов друг на друга определяется как отношение:

$$\varepsilon(S_1, S_2) = \frac{R(S_1, S_2)}{D(S_1, S_2)} \cdot 100\%,$$

где  $R(S_1, S_2)$  – функция несовпадающих черных точек, которые не являются смежными для общих черных точек,  $D(S_1, S_2)$  – количество общих черных точек. Функция  $R(S_1, S_2)$  рассчитывается с учетом веса каждой точки. Вес точки в функции  $R(S_1, S_2)$  возрастает при увеличении у данной точки таких же смежных точек.

Список использованных источников:

1. Классификационные методы сжатия изображений оцифрованного текста. Часть I / В.Г. Иванов, Ю.В. Ломоносов, М.Г. Любарский // К.: Научно-технический журнал «Системы обработки информации», 2013, №2. С. 36 – 43.
2. Иванов В.Г. Сжатие изображения текста на основе выделения символов и их классификации / В.Г. Иванов, М.Г. Любарский, Ю.В. Ломоносов // К.: Международный научно-технический журнал «Проблемы управления и информатики», 2010, №6. С. 111 – 122.