

ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ

3/2001

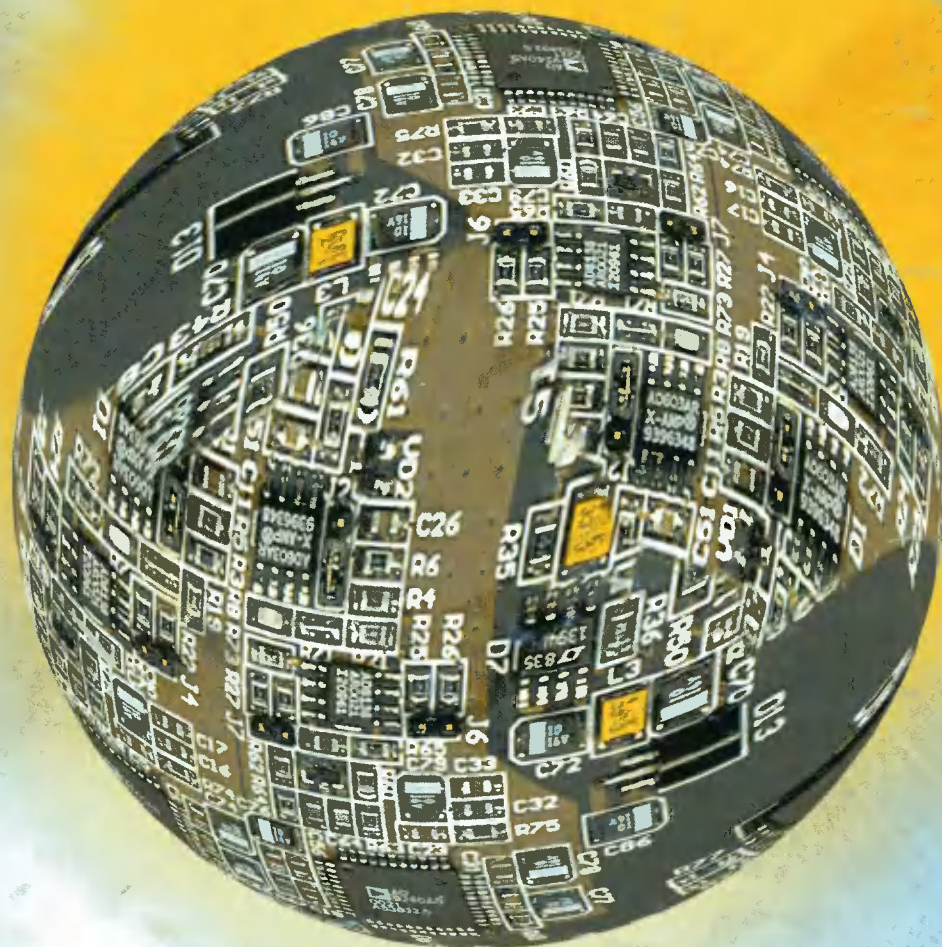
НАУЧНО - ТЕХНИЧЕСКИЙ ЖУРНАЛ

ЦОС в мобильных системах

Вейвлет-анализ

Новые разработки DSP

ПЛИС обработки сигналов



ЦИФРОВАЯ ОБРАБОТКА СИГНАЛОВ

№ 3/2001



ГЛАВНЫЙ РЕДАКТОР

Ю.Б. ЗУБАРЕВ

ЗАМЕСТИТЕЛИ ГЛАВНОГО РЕДАКТОРА:

В.В. ВИТЯЗЕВ, В.П. ДВОРКОВИЧ

ОТВЕТСТВЕННЫЙ РЕДАКТОР

В.Н. Вяльцев

РЕДАКЦИОННАЯ КОЛЛЕГИЯ:

П.А. Арутюнов, Б.А. Бабаян, Ю.И. Борисов,
С.А. Грибачев, Г.Н. Егоров, Г.В. Зайцев,
Е.П. Зелевич, Р.В. Зубарев,
А.П. Кирпичников, М.И. Кривошеев,
Н.А. Кузнецов, М.С. Куприянов, А.А. Ланнэ,
В.Г. Мистюков, С.Л. Мишенков, А.А. Петровский,
Ю.Н. Прохоров, Ю.Г. Сосулин,
Н.Г. Харатишвили, В.В. Шахгильдян,
Ю.С. Шинаков

Адрес редакции:

103064 Москва, ул. Казакова, 16, к. 1202

Тел./факс: (095) 267-0662

E-mail: pl@com2com.ru

Цифровая обработка сигналов

№ 3/2001, с. 1 – 64

Издатель: ООО "ЛТИС"

Генеральный директор

П.Л. Серебрянников

Дизайн и верстка С.Г.Тюканова,
Л.Н. Сыроватская

Корректура Е.В. Карасева
Н.Н. Сергеева

Издание зарегистрировано в Министерстве
Российской Федерации
по делам печати, телерадиовещания и средств
массовых коммуникаций
Свидетельство о регистрации
ПИ № 77-1488 от 14.01.00

В НОМЕРЕ:

*А.В. Гармонов, Ю.Е. Карпитский,
В.Б. Манелис, А.Ю. Савинков*
**Алгоритмы квазикогерентного приема
фазоманипулированных сигналов
в канале с быстрым федингом** 2

С.Н. Кириллов, С.В. Зорин
**Алгоритмы биортогонального вейвлет-
анализа на основе рекурсивных фильтров** 9

А.В. Дворкович
**Еще об одном методе расчета эффективных
оконных функций, используемых при
гармоническом анализе с помощью ДПФ** 13

*Р.В. Белянкин, В.В. Риман, А.И. Селянин,
В.А. Фин, А.А. Чернышев*
**Медицинское портативное ультразвуковое
диагностическое устройство
на базе модуля ЦОС типа ADP62/6701E** 19

*Ю.Н. Александров, А.А. Беляев, А.В. Глушков,
Ю.Н. Грибов, А.В. Никольский,
Я.Я. Петричкович, Т.В. Солохина*
**Новая отечественная платформа
СБИС "МУЛЬТИКОР" для высокоточной
скоростной обработки информации и
управления объектами** 25

В.А. Кашкаров
**Организация параллельных вычислений
преобразования Адамара на процессоре
NM6403 (L1879BM1)** 39

В.Е. Чернов
**Согласование последовательных АЦП с
высокопроизводительными сигнальными
процессорами Texas Instruments** 46

Analog Devices: новые разработки DSP 49

В.Б. Стешенко
**Система проектирования ПЛИС
фирмы ALTERA** 57

Алгоритмы квазикогерентного приема фазоманипулированных сигналов в канале с быстрым федингом

Рассматриваются различные алгоритмы квазикогерентного приема MPSK-сигналов с прерывистым пилот-сигналом в условиях быстрого фединга: алгоритм максимального правдоподобия (АМП), алгоритмы с решающей обратной связью (РОС), итеративный алгоритм (ИА). Алгоритмы не требуют знания автокорреляционной функции (АКФ) канала. Они основаны на параметризации изменяющейся во времени комплексной огибающей сигнала путем разложения ее в ряд по некоторой системе функций и оценке этих параметров. Для условий быстрого Релеевского фединга, соответствующего модели Джейкса, методом математического моделирования выполнен анализ представленных алгоритмов, оптимизированы их параметры, произведено сравнение точностных характеристик алгоритмов.

Введение

Одним из требований к системам мобильной связи 3-го поколения, зафиксированных в соответствующих стандартах (UMTS, cdma 2000 и др.), является поддержание связи с абонентами, перемещающимися с высокой скоростью. Как следствие, мобильная система связи должна эффективно функционировать в условиях быстрого (высокочастотного) фединга. В настоящее время получение высокой емкости сотовых систем связи и высокой скорости передачи информации связывается с использованием когерентных видов модуляции, таких, как MPSK. В системах связи CDMA 2-го и 3-го поколения наибольшее распространение получили BPSK- и QPSK-виды модуляции.

Потенциальные преимущества этих видов модуляции могут быть реализованы на практике только при использовании квазикогерентных демодуляторов, которые в присутствии шумов с высокой точностью оценивают комплексную огибающую принимаемого сигнала. Для такой оценки полезный сигнал обычно содержит пилот-компоненту. Возможны два основных типа пилот-компоненты, различающиеся по временной структуре. Первый – это так называемый непрерывный пилот-сигнал, передаваемый параллельно вместе с информационной компонентой сигнала. Пилот и информационная компоненты при этом являются ортогональными. Данный тип пилот-компоненты имеет место в стандартах IS-95 и cdma 2000. Вопросам организации квазикогерентного приема при наличии непрерывного пилот-сигнала посвящено значительное количество публикаций [1] – [5]. Получены хорошие результаты, в том числе для случая быстрого фединга, и, по-видимому, задачу создания эффективных алгоритмов оценки комплексной огибающей сигнала для данного типа пилот-компоненты можно считать в основном решенной.

Второй тип пилот-компоненты – пилотные символы, мультиплексированные во времени с информационными символами. То есть пилот-компонента представляет собой известную пилот-последовательность, периодически вставляемую в сообщение. Такой тип пилот-компоненты имеет

место в проекте стандарта UMTS и, возможно, будет перспективным для систем связи 4-го поколения. Задаче оценки комплексной огибающей и организации квазикогерентного приема при прерывистом пилот-сигнале посвящено большое число публикаций [4] – [14]. Заметим, что в условиях высокочастотного фединга для широкого диапазона значений скорости передачи информации эта задача становится нетривиальной. Здесь многие известные методы восстановления несущей становятся неработоспособными или неэффективными.

Так, метод, использующий гауссовскую интерполяцию 1 – 2 порядка по усредненным значениям пилот-групп [6], является работоспособным лишь при низкочастотном фединге. Когда изменения комплексной огибающей становятся существенными в пределах слота, качество данного метода – неудовлетворительное. В [10] реализована идея РОС, когда для восстановления комплексной огибающей на интервале оцениваемого символа используются оценки предыдущих информационных символов. Для оценки фазы полезного сигнала используется адаптивный фильтр. Адаптация осуществляется по обучающей пилот-выборке, которая предшествует информационным символам. Предложенный относительно простой алгоритм показал хорошее качество при отсутствии замираний и медленном изменении фазы сигнала. Однако, как показано в [13], в случае замираний подобный алгоритм обладает недостаточно высокой помехоустойчивостью. В ряде работ [4], [5], [8], как и в [6], восстановление несущей осуществляется только по пилот-символам. Хорошего качества алгоритмов при таком подходе можно добиться лишь за счет увеличения мощности сигнала при относительно низкочастотном фединге. Между тем информация о комплексной огибающей полезного сигнала содержится и на интервалах информационных символов. Учет этой информации в алгоритме квазикогерентного приема позволит повысить его эффективность и снизить необходимую мощность полезного сигнала.

В этой связи следует отметить алгоритмы квазикогерентного приема, предложенные в [9], где для



оценки комплексной огибающей используются отсчеты информационной части сигнала. При частотах фединга, не превышающих 108 Гц, получены очень хорошие характеристики. Однако метод интерполяции, предполагаемый здесь, требует, как и в алгоритме [7], знания АКФ-канала. Необходимость оценки АКФ усложняет алгоритм обработки. Кроме того, ошибки в определении АКФ, неизбежные в условиях низких отношений сигнал-шум, вносят дополнительную погрешность в оценку комплексной огибающей.

В [11], [12] для упрощения процедуры оценки изменяющегося во времени канала и устойчивости этой оценки применена полиномиальная модель. Эта модель опосредованно используется для определения весовых коэффициентов фильтра, с помощью которого формируется оценка комплексной огибающей сигнала. Предложенный эвристический подход не требует знания статистики канала. Однако ввиду небольшой величины окна фильтра точность представленного алгоритма оценки комплексной огибающей относительно невелика.

В данной статье комплексная огибающая входного сигнала трактуется, как некоторая неизвестная гладкая функция, подлежащая параметризации путем разложения в ряд по некоторой системе функций. Рассматриваются различные алгоритмы квазикогерентного приема MPSK-сигналов с прерывистым пилот-сигналом в условиях быстрого фединга. В разделе 2 формулируется математическая постановка задачи. В разделе 3 выполнены синтез АМП-оценки информационных параметров MPSK-сигнала и его анализ методом математического моделирования. В разделе 4 описываются алгоритмы с РОС, оптимизируются значения параметров этих алгоритмов. Раздел 5 посвящен ИА. В заключение 6 проводится сравнительный анализ представленных алгоритмов.

Постановка задачи

Положим, что на интервале $[0, T]$ наблюдается реализация суммы полезного сигнала и помехи

$$x(t) = s(t, \mathbf{A}, \psi, \mathbf{a}, \mathbf{b}) + n(t), \quad (1)$$

где $n(t)$ – белый гауссовский шум с односторонней спектральной плотностью N_0 , $\langle n(t_1)n(t_2) \rangle = (N_0/2)(t_1 - t_2)$, $\delta(\cdot)$ – дельта-функция, $s(t, \mathbf{A}, \psi, \mathbf{a}, \mathbf{b})$ – полезный двухкомпонентный сигнал

$$s(t, \mathbf{A}, \psi, \mathbf{a}, \mathbf{b}) = \sum_{j=1}^N A_j g_j(t) [b_j \cos(\omega_0 t - \psi_j) + a_j \sin(\omega_0 t - \psi_j)]. \quad (2)$$

Здесь ω_0 – несущая частота, $g_j(t)$ – псевдослучайная последовательность, известная на интервале каждого элементарного символа Δ_j , длина каждого интервала равна Δ ; $\mathbf{A} = \{A_j\}$, $\psi = \{\psi_j\}$, $j = \overline{1, N}$ – априори неизвестные амплитуда и фаза полезного сигнала, приближенно постоянные на интервалах Δ_j ; $\mathbf{a} = \{a_j\}$, $\mathbf{b} = \{b_j\}$, $j = \overline{1, N}$ – векторы параметров, каждый из которых содержит N_d неизвестных информационных параметров (соответствующие

векторы обозначим $\mathbf{a}_d, \mathbf{b}_d$), а также N_p известных параметров пилот-символов (соответствующие векторы – $\mathbf{a}_p, \mathbf{b}_p$), $N_d + N_p = N$; $N = T/\Delta$ – количество интервалов, укладываемых на $[0, T]$. Количество и взаимное расположение информационных и пилот-символов в векторах \mathbf{a} и \mathbf{b} априори известны. Все элементы векторов $\mathbf{a}_p, \mathbf{b}_p$ имеют значение 1. Для информационных параметров, например для QPSK-модуляции, $a_j, b_j = \pm 1$.

Необходимо по наблюдаемым данным (1) вынести решение о значении информационных параметров $\mathbf{a}_d, \mathbf{b}_d$.

Рассмотрим различные подходы к решению данной задачи.

Алгоритм максимального правдоподобия

Синтез алгоритма выполним в предположении достаточной гладкости компонент комплексной огибающей полезного сигнала и их независимости [15]. Это предположение позволяет параметризовать задачу, т. е. будем считать, что дискретные компоненты комплексной огибающей $A_{cj} = A_j \cos \varphi_j$, $A_{sj} = A_j \sin \varphi_j$ на рассматриваемом интервале $[0, T]$ хорошо описывается в некотором базисе $\{f_m(t)\}$, $m = \overline{0, K}$, $t \in [0, T]$

$$A_{cj} = \sum_{m=0}^K c_m f_m(t_j), \quad A_{sj} = \sum_{m=0}^K d_m f_m(t_j), \quad (3)$$

где t_j – временное положение элементарного интервала Δ_j . Параметры $\mathbf{c} = \{c_m\}$, $\mathbf{d} = \{d_m\}$, $m = \overline{0, K}$, определяющие комплексную огибающую сигнала на анализируемом интервале $[0, T]$, априори неизвестны.

В соответствии с оценкой максимального правдоподобия информационных параметров $\mathbf{a}_d, \mathbf{b}_d$ решение принимается в пользу таких значений параметров, при которых функционал отношения правдоподобия или его монотонная функция принимает наибольшее значение. С точностью до несущественных слагаемых логарифм функционала отношения правдоподобия (ЛФОП) принимаемого квазидетерминированного сигнала имеет вид

$$M(\mathbf{a}_d, \mathbf{b}_d, \mathbf{c}, \mathbf{d}) = \frac{2}{N_0} \int_0^T x(t) \cdot s(t, \mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}) dt - \frac{1}{N_0} \int_0^T s^2(t, \mathbf{a}, \mathbf{b}, \mathbf{c}, \mathbf{d}) dt. \quad (4)$$

Подставляя (2) в (4), после необходимых преобразований с учетом (3) получаем в векторно-матричной форме

$$M(\mathbf{a}_d, \mathbf{b}_d, \mathbf{c}, \mathbf{d}) = \frac{2}{N_0} (\alpha^T \mathbf{c} + \beta^T \mathbf{d} - \frac{1}{2} \mathbf{c}^T \Gamma \mathbf{c} - \frac{1}{2} \mathbf{d}^T \Gamma \mathbf{d}), \quad (5)$$

где обозначено

$$\alpha = \left\{ \sum_{j=1}^N f_m(t_j) p_j \right\}, \quad \beta = \left\{ \sum_{j=1}^N f_m(t_j) q_j \right\}, \quad m = \overline{0, K},$$

$$\Gamma = \left\{ \Delta \sum_{j=1}^N f_m(t_j) f_n(t_j) \right\}, \quad m, n = \overline{0, K}, \quad (6)$$

$$p_j = b_j X_j + a_j Y_j, \quad q_j = b_j Y_j - a_j X_j, \quad j = \overline{1, N},$$

$$X_j = \int_{\Delta_j} x(t) g_j(t) \cos(\omega_0 t) dt, \quad Y_j = \int_{\Delta_j} x(t) g_j(t) \sin(\omega_0 t) dt.$$

Учет дискретизации входной реализации осуществляется заменой интегралов в выражениях для X_j, Y_j (6) конечной суммой отсчетов в пределах соответствующих интервалов Δ_j .

Параметры сигнала $\mathbf{a}_d, \mathbf{b}_d$ являются оцениваемыми, а неизвестные параметры \mathbf{c}, \mathbf{d} – сопровождающими. При наличии сопровождающих параметров в соответствии с методом максимального правдоподобия необходимо формировать функционал $M(\mathbf{a}_d, \mathbf{b}_d) = \max_{\mathbf{c}, \mathbf{d}} M(\mathbf{a}_d, \mathbf{b}_d, \mathbf{c}, \mathbf{d})$,

в котором неизвестные сопровождающие параметры \mathbf{c}, \mathbf{d} берутся такими, чтобы выходной сигнал $M(\mathbf{a}_d, \mathbf{b}_d)$ был максимален. Оценка максимального правдоподобия параметров $\mathbf{a}_d, \mathbf{b}_d$ находится по положению абсолютного максимума функционала $M(\mathbf{a}_d, \mathbf{b}_d)$

$$(\hat{\mathbf{a}}_d, \hat{\mathbf{b}}_d) = \arg \sup M(\mathbf{a}_d, \mathbf{b}_d). \quad (7)$$

В результате максимизации (5) по сопровождающим параметрам \mathbf{c}, \mathbf{d} получаем

$$M(\mathbf{a}_d, \mathbf{b}_d) = \frac{1}{N_0} (\alpha \Gamma \Gamma^{-1} \alpha + \beta \Gamma \Gamma^{-1} \beta). \quad (8)$$

Отбрасывая в (8) слагаемые, не зависящие от оцениваемых параметров, окончательно получаем решающую функцию в следующем виде

$$M(\mathbf{a}_d, \mathbf{b}_d) = \frac{2}{N_0} \sum_{i=1}^{N-1} \sum_{j=i+1}^N Q_{ij} (p_i p_j + q_i q_j) \chi_{ij}, \quad (9)$$

$$\text{где } Q_{ij} = \sum_{m=0}^K \sum_{n=0}^K \Theta_{mn} f_m(t_i) f_n(t_j), \Theta = \{\Theta_{mn}\}, m, n = \overline{0, K},$$

$\Theta = \Gamma^{-1}$; $\chi_{ij} = 0$ – если оба индекса i и j соответствуют пилот-символам, в противном случае $\chi_{ij} = 1$. Элементы $Q_{ij}, i, j = \overline{1, N}$ не зависят ни от наблюдаемых данных, ни от оцениваемых параметров и могут быть рассчитаны заранее. В (9) войдут известные векторы пилот-символов $\mathbf{a}_p, \mathbf{b}_p$. Их присутствие делает оценку $\hat{\mathbf{a}}_d, \hat{\mathbf{b}}_d$ (7) состоятельной.

Таким образом, для оценки информационных параметров $\mathbf{a}_d, \mathbf{b}_d$ необходимо рассчитать значение решающей функции (9) для всех возможных комбинаций значений векторов $\mathbf{a}_d, \mathbf{b}_d$ и найти положение абсолютного максимума (9) в соответствии с (7). Для MPSK-модуляции число таких комбинаций равно M^{N_d} . Видно, что вычислительная сложность АМП резко возрастает с увеличением количества информационных символов N_d на интервале анализа. Кроме того, при больших величинах N_d , когда интервал $[0, T]$ соизмерим или больше интервала корреляции комплексной огибающей входного сигнала, возникает проблема точности представления (3). Если для сохранения необходимой точности представления комплексной огибающей сигнала увеличить число элементов ряда (3), то возрастет шумовая составляющая оценки комплексной огибающей ввиду неизбежных ошибок при оценке векторов \mathbf{c}, \mathbf{d} (из-за роста числа шумовых слагаемых). Таким образом, суммарная погрешность оценки комплексной огибающей, обусловленная шу-

мами и неточностью представления, увеличивается, что приводит к ухудшению точности демодуляции.

Заметим также, что сложность алгоритма в значительной степени зависит от вида модуляции (параметр M). Так, при переходе от BPSK-модуляции ($M = 2$) к QPSK-модуляции ($M = 4$) вычислительная сложность синтезированного алгоритма возрастает в 2^{N_d} раз.

Поэтому одним из возможных вариантов применения АМП является разбиение непрерывного потока символов на блоки, число N_d информационных символов в которых относительно невелико, и использование полученных выражений (7), (9) для независимой оценки информационных символов этих блоков.

Данный подход был применен для демодуляции сигнала, информационная структура которого представлена на рис. 1. Предполагалось, что данные передаются слотами по 10 комплексных символов каждый. Первые 4 символа каждого слота являются пилот-символами (пилот-группа), следующие 6 символов слота – информационные. Каждый символ содержит по два независимых бита, которые для пилот-символов равны 1, а для информационных символов могут принимать значения ± 1 (QPSK-модуляция).

На рис. 2 показаны зависимости вероятности битовой ошибки до декодера АМП от величины отношения сигнал – шум на бит, полученные с помощью компьютерного моделирования для различных порядков K полиномов, используемых для аппроксимации комплексной амплитуды сигнала. В качестве базиса был выбран степенной полином

$$\{f_m(t) = (t - T/2)^m\}, m = \overline{0, K}, t \in [0, T]. \quad (10)$$

Все временные интервалы имели одинаковую длину, равную сумме длительностей слота и пилот-группы следующего слота ($N = 14$). Каждые два соседних интервала перекрывались на интервале длительности пилот-группы. Длительность слота составляла 0.625 мс. При моделировании было внесено постоянное рассогласование частот 150 Гц между входным и опорным сигналами, которое имитировало максимальную ошибку системы автоподстройки частоты. Моделирование было выполнено для частоты фединга $f_f = 880$ Гц (модель Джейкса [15]), которая при частоте несущей 2 ГГц соответствует движению абонента со скоростью 500 км/ч. Указанная скорость движения приведена в требованиях на проектирование аппаратуры систем сотовой связи 3-го поколения [16] и соответствует наиболее жестким условиям приема.

При частоте фединга 880 Гц комплексная огибающая сигнала существенно изменяется в течение

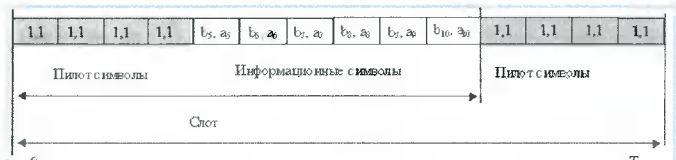


Рис. 1. Информационная структура сигнала

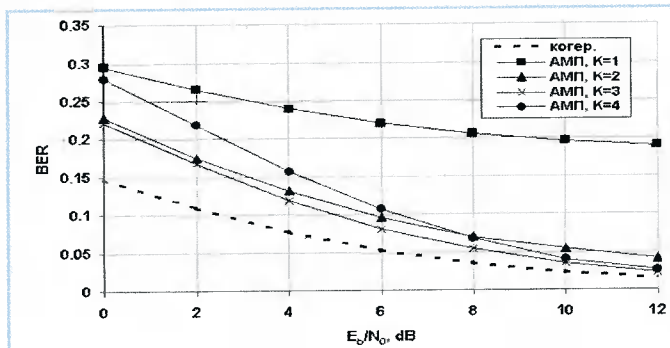


Рис.2. Зависимость вероятности битовой ошибки от отношения сигнал – шум на бит для АМП при частоте фединга 880 Гц (модель Джейкса) для различных порядков K аппроксимирующих полиномов

слота. При выборе порядка аппроксимирующего полинома K необходимо найти компромисс между точностью описания комплексной огибающей (расчет с увеличением K) и уменьшением шумовой компоненты ее оценки, которая с увеличением K также растет. Как следует из кривых рис.2, использование полинома третьего порядка $K=3$ обеспечивает наилучшую помехоустойчивость в рабочей области современных систем связи ($BER \approx 0.1$). С учетом последующего декодирования этот уровень BER обеспечивает необходимое качество передачи информации.

Кривая “когерентный прием” на рис.2 соответствует приему при априори известной комплексной огибающей сигнала. Данную кривую можно использовать в качестве нижней границы алгоритмов квазикогерентного приема. Видно, что проигрыш АМП (при $K=3$) по отношению к когерентному приему составляет немногим более 2 дБ.

АМП реализует идею приема “в целом”, которая и обуславливает его сложность. В этой связи актуальной является разработка более простых квазиоптимальных алгоритмов, характеристики которых не сильно отличались бы от характеристик АМП.

Алгоритмы с решающей обратной связью

Гораздо более простым по отношению к приему в “целом” является посимвольный прием. Осуществить переход к посимвольному приему можно, используя РОС. В этом случае для оценки текущих символов используются оценки предыдущих информационных символов.

а) Квазиоптимальный алгоритм с решающей обратной связью (КАРОС)

Одной из возможных реализаций идеи РОС применительно к рассматриваемой задаче является следующий квазиоптимальный алгоритм (будем обозначать его КАРОС). Пусть имеется блок из L символов, неизвестным в котором является только последний L -й символ (a_L, b_L). Предыдущие $L-1$ символы либо известны (как пилот-символы), либо оценены на предыдущих этапах работы алгоритма.

Оценка неизвестного символа основана на использовании формул АМП (7), (9). Отбрасывая в (9) слабые, не зависящие от оцениваемого L -го символа, после необходимых преобразований получаем

$$M(a_L, b_L) = \frac{2}{N_0} [b_L(X_L u_L + Y_L v_L) + a_L(Y_L u_L - X_L v_L)], \quad (11)$$

где $u_L = \sum_{i=1}^{L-1} Q_{iL} p_i$, $v_L = \sum_{i=1}^{L-1} Q_{iL} q_i$, а величины X_L, Y_L определены в (6). Находя значения (\tilde{a}_L, \tilde{b}_L), максимизирующие (11), получаем оценку КАРОС. Для относительно простых видов модуляции несложно получить явные оценки информационных параметров. Так, для QPSK-модуляции имеем

$$\tilde{b}_L = \text{sgn}(X_L u_L + Y_L v_L), \quad \tilde{a}_L = \text{sgn}(Y_L u_L - X_L v_L). \quad (12)$$

Аналогично демодулируется следующий $(L+1)$ -й символ и т. д.

На рис. 3 показаны зависимости вероятности битовой ошибки КАРОС от величины отношения сигнал – шум на бит, полученные методом математического моделирования для различных значений размера окна (блока) L и порядка K полиномов, используемых для аппроксимации комплексной огибающей сигнала. Все параметры и условия приема при моделировании КАРОС такие же, как и для АМП. Из рис. 3 следует, что наилучшие результаты в рабочей области BER обеспечиваются при значениях параметров КАРОС $L=5, K=1$.

б) Алгоритм с решающей обратной связью и экстраполяцией (АРОСЭ)

Упрощенной модификацией описанного выше КАРОС является алгоритм с решающей обратной связью и с экстраполяцией (АРОСЭ). В АРОСЭ оценка комплексной амплитуды на интервале текущего $(L+1)$ -го символа выносится по значениям комплексной амплитуды L предыдущих символов (либо пилот-символы, либо оцененные ранее информационные символы). Эти значения используются следующим образом. Комплексная амплитуда представ-

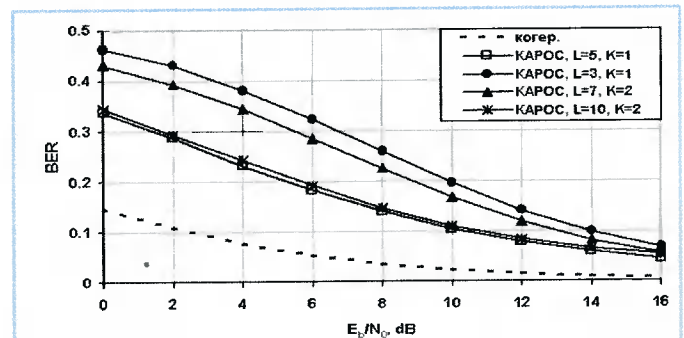


Рис.3. Зависимость вероятности битовой ошибки от отношения сигнал – шум на бит для КАРОС при частоте фединга 880 Гц (модель Джейкса) для различных величин длины окна L и порядка K аппроксимирующего полинома

ляется в виде (3) в полиномиальном базисе (10). Производится оценка векторов \mathbf{c} и \mathbf{d} по методу наименьших квадратов в соответствии с условием

$$\sum_{j=1}^L (\tilde{A}_{cj} - \sum_{m=0}^K c_m (t_j - t/2)^m)^2 = \min,$$

$$\sum_{j=1}^L (\tilde{A}_{sj} - \sum_{m=0}^K d_m (t_j - t/2)^m)^2 = \min, \quad (13)$$

где

$$\tilde{A}_{cj} = \frac{\tilde{b}_j X_j + \tilde{a}_j Y_j}{\Delta}, \quad \tilde{A}_{sj} = \frac{\tilde{b}_j Y_j - \tilde{a}_j X_j}{\Delta}. \quad (14)$$

Для информационных символов величины \tilde{b}_j, \tilde{a}_j в (14) представляют собой полученные ранее оценки, для пилот-символов $\tilde{b}_j = \tilde{a}_j = 1$. Условие (13) приводит к системе линейных уравнений, решение которой относительно векторов \mathbf{c}, \mathbf{d} не представляет затруднений.

Далее в предположении справедливости полученной модели на интервале текущего оцениваемого $(L+1)$ -го символа осуществляется экстраполяция (\hat{A}_c, \hat{A}_s) его комплексной огибающей по формулам

$$\hat{A}_c = \sum_{m=0}^K \tilde{c}_m (t_{L+1} - T/2)^m, \quad \hat{A}_s = \sum_{m=0}^K \tilde{d}_m (t_{L+1} - T/2)^m, \quad (15)$$

где t_{L+1} – временное положение оцениваемого символа, $\tilde{c}_m, \tilde{d}_m, m=0, K$ – оценки, удовлетворяющие (13). В соответствии с прогнозом (15) производится демодуляция $(L+1)$ -го символа, а также оценка комплексной огибающей этого символа по формуле (14). Последняя вместе со значениями комплексной огибающей $L-1$ предыдущих символов используется описанным выше образом для демодуляции и оценки комплексной огибающей на интервале следующего информационного символа и т. д. (рис. 4).

АРОСЭ характеризуется длиной окна L и порядком полинома K . В начале приема при демодуляции слота окно включает лишь пилот-символы, и поэтому длина окна ограничена числом пилот-символов в слоте. Заметим, что, если в КАРОС окно символов, по которому формируется прогноз комплексной огибающей демодулируемого символа, включает этот символ, то в АРОСЭ демодулируемый символ не включается в окно.

На рис. 5 показаны зависимости вероятности битовой ошибки АРОСЭ от величины отношения сигнал – шум на бит, полученные методом математического моделирования для различных параметров K, L аппроксимации комплексной огибающей сигнала. Демодуляция различных слотов проводилась независимо. Остальные параметры и условия приема при моделировании АРОСЭ та-

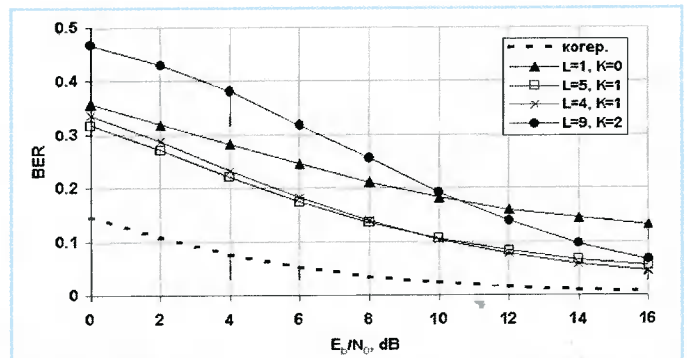


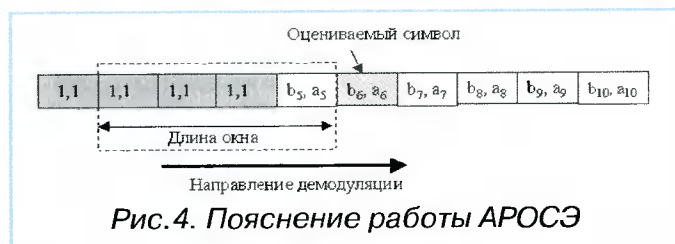
Рис. 5. Зависимость вероятности битовой ошибки от отношения сигнал – шум на бит для АРОСЭ при частоте Фединга 880 Гц (модель Джейкса) для различных величин длины окна L и порядка K аппроксимирующего полинома

кие же, как и для других рассмотренных алгоритмов. Из рис. 5 следует, что наилучшие результаты в рабочей области BER обеспечиваются при значениях параметров АРОСЭ $L=5, K=1$.

с) Двухпроходный алгоритм с решающей обратной связью и экстраполяцией (ДАРОСЭ)

Относительно невысокая помехоустойчивость АРОСЭ и КАРОС связана с тем, что оценка каждого информационного символа опирается на уже полученные оценки предыдущих символов. При этом ошибочно оцененные предыдущие символы (вследствие ненулевой вероятности ошибки) увеличивают вероятность ошибки при оценке текущего символа. При удалении от пилот-символов вероятность ошибки увеличивается и появляется тенденция к срыву приема. Поэтому для увеличения помехоустойчивости необходимо минимизировать количество информационных символов, “привязанных” к пилот-группе. Эта идея реализована в двухпроходном алгоритме с решающей обратной связью (ДАРОСЭ), где поздняя часть слота демодулируется, начиная с последнего информационного символа, по пилотным отсчетам следующего слота, то есть процесс демодуляции информационной части слота идет как с начала, так и с конца слота к его середине (рис. 6) [13].

Граничный символ слота (тот символ, который делит слот на две независимо демодулируемые части) здесь может выбираться по-разному: либо по середине слота; либо по символу с минимальной энергией, где имеет место сильное замирание полезного сигнала и вероятность ошибочного приема символа наиболее велика. Очевидно, ДАРОСЭ требует временной задержки и более сложен в реализации, чем КАРОС или АРОСЭ. Однако, как следует из кривых рис. 7 (граничный символ – в середине слота) и рис. 8 (граничный символ – символ с минимальной энергией), он позволяет получить лучшие точностные характеристики. Лучший набор параметров для ДАРОСЭ: $L=5, K=1$. Указанное значение длины окна приблизительно соответствует $(0.275/f_T)$ с. Заме-



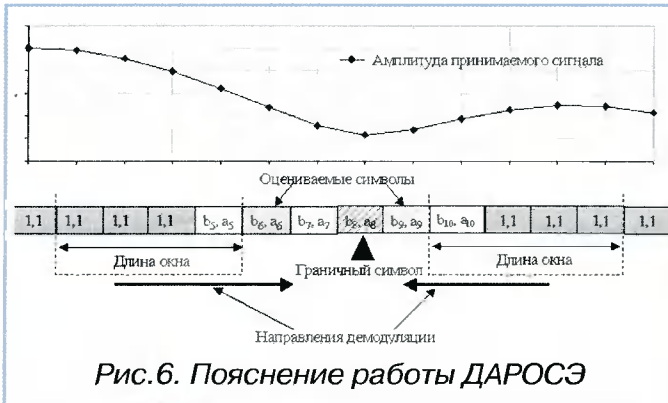


Рис. 6. Пояснение работы ДАРОСЭ

тим, что выбор граничного символа по минимуму энергии позволяет улучшить характеристики алгоритма в рабочей области примерно на 1 дБ по сравнению со случаем граничного символа в се-

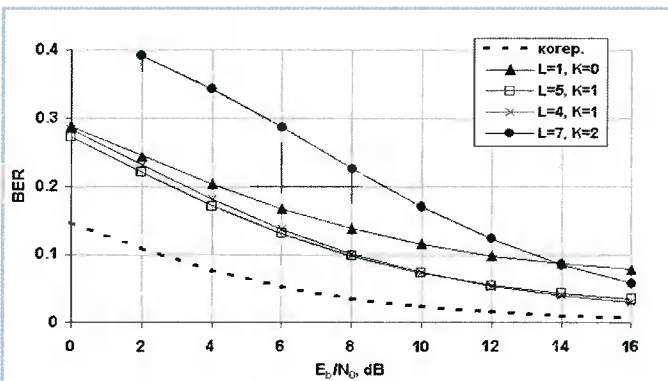


Рис. 7. Зависимость вероятности битовой ошибки от отношения сигнал – шум на бит для ДАРОСЭ при частоте фединга 880 Гц (модель Джейкса) для различных величин длины окна L и порядка K аппроксимирующего полинома (граничный символ – в середине слота)

редине слота. Тем не менее из-за отмеченных выше недостатков, свойственных алгоритмам посимвольной демодуляции с РОС, ДАРОСЭ существенно уступают в точности АМП.

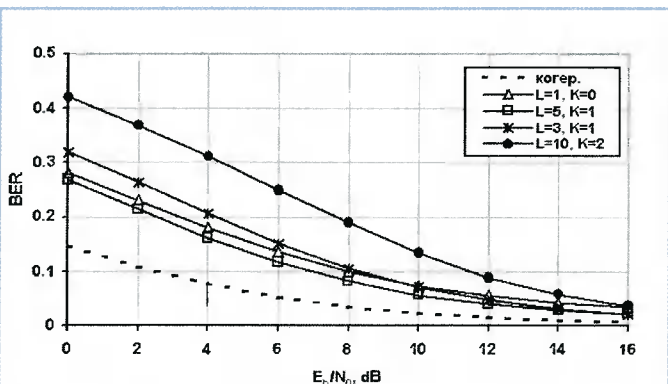


Рис. 8. Зависимость вероятности битовой ошибки от отношения сигнал – шум на бит для ДАРОСЭ при частоте фединга 880 Гц (модель Джейкса) для различных величин длины окна L и порядка K аппроксимирующего полинома (граничный символ – символ с минимальной энергией)

Итеративный алгоритм

Очевидно, точность оценок комплексной огибающей и информационных параметров взаимозависимы. Итеративный алгоритм (ИА) [14] основан на последовательном улучшении оценки информационных параметров за счет улучшения оценки несущей. Последнее, в свою очередь, обусловлено последовательным увеличением точности демодуляции.

ИА, так же, как и АМП, и алгоритмы квазикогерентного приема с РОС, предполагает некую полиномиальную аппроксимацию (3) комплексной огибающей сигнала в пределах слота. Первоначально формируется первое приближение оценок информационных параметров слота, например используя описанный выше ДАРОСЭ. Затем на основании этих оценок по методу наименьших квадратов (13) выносятся оценка параметров полинома \mathbf{c} , \mathbf{d} , т. е. осуществляется восстановление комплексной огибающей, в соответствии с которым проводится демодуляция информационной части слота. В результате формируется второе приближение оценки информационных параметров. Это второе приближение используется для уточнения комплексной огибающей символов слота по формуле (14), находятся новые значения векторов \mathbf{c} , \mathbf{d} и т. д. Использование нескольких итераций позволяет увеличить точность окончательной демодуляции. Максимальное количество итераций ИА определяется скоростью сходимости алгоритма. В случае равенства оценок информационных параметров двух соседних итераций работа ИА должна быть досрочно остановлена. Это несколько повышает быстродействие алгоритма

На рис.9 показаны зависимости вероятности битовой ошибки ИА от величины отношения сигнал – шум на бит, полученные методом математического моделирования при использовании ДАРОСЭ с лучшими параметрами длины окна и порядка полинома и различных значений порядка полинома K , используемого для описания комплексной амплитуды в итеративной процедуре. Из рис.9 следует, что оптимальным значением порядка полинома K , как и для АМП, является $K=3$. Отмечена достаточно быстрая сходимость ИА. Уже после 2 – 3 итераций характеристики практически достигают своих предельных значений. Некоторое увеличение вычислительной сложности из-за итеративной процедуры позволяет получить выигрыш 1 – 2 дБ по сравнению с ДАРОСЭ.

Заключение

В данной статье представлены различные по сложности реализации алгоритмы квазикогерентного приема MPSK-сигналов. Эти алгоритмы основываются на параметризации изменяющейся во времени комплексной огибающей сигнала путем разложения ее в ряд по некоторой системе функций. Анализ алгоритмов выполнен в условиях быстрого фединга с частотой 880 Гц в соответствии с моделью Джейкса, когда изменение комплексной

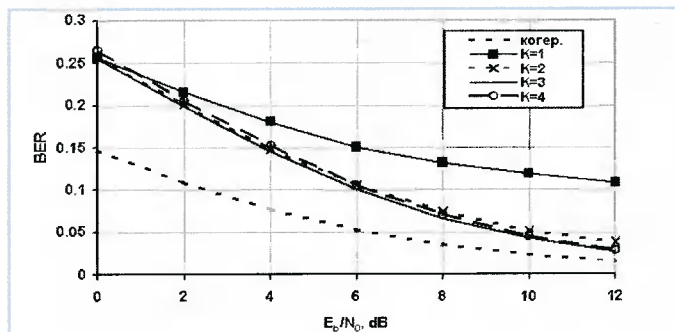


Рис. 9. Зависимость вероятности битовой ошибки от отношения сигнал – шум на бит для ИА при частоте фединга 880 Гц (модель Джейкса) для различных порядков K аппроксимирующего полинома

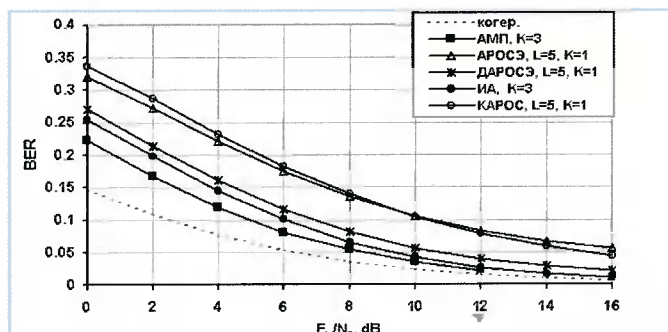


Рис. 10. Зависимость вероятности битовой ошибки от отношения сигнал – шум на бит для различных алгоритмов с наилучшими параметрами длины окна L и порядка K аппроксимирующего полинома при частоте фединга 880 Гц (модель Джейкса)

огibaющей от символа к символу является существенным. Сравнительная точность рассмотренных алгоритмов представлена на рис. 10.

Были выбраны наилучшие параметры для каждого алгоритма для рассмотренных условий приема. Аналогично могут быть получены оптимальные параметры алгоритмов для других типовых условий приема. Из кривых рис. 10 следует, что наилучшей точностью из относительно легко реализуемых алгоритмов обладают ДАРОСЭ и ИА. Последний уступает сложному для реализации АМП не более 1 дБ.

Таким образом, полученные результаты позволяют обоснованно выбрать алгоритм квазикогерентного приема MPSK-сигналов в условиях быстрого фединга в зависимости от требований к точности и простоте его реализации.

Литература

1. Toskala A., et al. FRAMES FMA2 Wideband CDMA for UMTS // Europ. Trans. Telecommun. - July/Aug. 1998. – Vol. 9. – №4. – P.325 – 336.
2. Gorazza G. E. and Gaudenzi R. Pilot-Aided Coherent Uplink for Mobile Satellite CDMA Networks // IEEE Trans. Commun. – May 1999. – Vol. 47. – №5. – P.773 – 784.
3. Квазикогерентный прием фазоманипулированного сигнала в системах CDMA/ Гармонов А. В., Карпитский Ю.Е., Каюков И. В., Манелис В. Б. //5-я международная научно-техническая конференция «Радиолокация, навигация, связь». – Воронеж, 20 – 23 апреля 1999. – Т. 1. – С. 305 – 313.
4. Abeta S., Sawahashi M., and Adachi F. Performance comparison between time-multiplexed pilot channel and parallel pilot channel for coherent rake combining in DS-SS-CDMA mobile radio // IEICE Trans. Commun. – July 1998. – Vol. E81-B. – №7. – P.1417 – 1425.
5. Abeta S., Sawahashi M., and Adachi F. Adaptive channel estimation for coherent DS-SS-CDMA mobile radio using time-multiplexed pilot and parallel pilot structures // IEICE Trans. Commun. – Sept. 1999. – Vol. E82-B. – №9. – P.1505 – 1513.
6. Sampei S. and Sunaga T. Rayleigh fading compensation for QAM in land mobile radio communications //

IEEE Trans. Veh. Technol. – May 1993. – Vol. 42. – P.137 – 146.

7. Lodge J. H. and Moher M. J. Maximum-likelihood sequence estimation of CPM signals transmitted over Rayleigh flat-fading channels // IEEE Trans. Commun. – June. 1990. – Vol. 38. – №6. – P.787 – 794.
8. Ling F. Optimal reception, performance bound, and cutoff rate analysis of references-assisted coherent CDMA communications with applications // IEEE Trans. Commun. – Oct. 1999. – Vol. 47. – №10. – P.1583 – 1592.
9. Genko K., Nam L., Sakaniwa K. Improvement of extended symbol-aided estimation for Rayleigh fading channels // IEICE Trans. Fundam. – Oct. 1998. – Vol. E81-A. – №10. – P.2073 – 2082.
10. Kam P. Y., Sihna P. and Kan A. M. C. Adaptive digital Coherent Receiver for MPSK // Electronics Letters. – Oct. 1992. – Vol. 28. – №22. – P.2099 – 2101.
11. Borah D. K. and Hart B. D. A robust receiver structure for time-varying frequency-flat, Rayleigh fading channels // IEEE Trans. Commun. – March. 1999. – Vol. 47. – № 3. – P.360 – 364.
12. Borah D. K. and Hart B. D. Frequency-selective fading channel estimation with a polynomial time-varying channel model // IEEE Trans. Commun. – June. 1999. – Vol. 47. – № 6. – P.862 – 873.
13. Гармонов А. В., Карпитский Ю.Е., Манелис В.Б., Савинков А.Ю. Алгоритмы квазикогерентного приема фазоманипулированного сигнала при наличии прерывистого пилот-сигнала в канале с быстрыми замираниями // Синтез, передача и прием сигналов управления и связи. – Воронеж, ВГТУ, 1999. – № 6. – С.46 – 53.
14. Итеративный квазикогерентный многолучевой прием с линейной регрессией/ Гармонов А.В., Савинков А.Ю., Карпитский Ю.Е.//5-я международная научно-техническая конференция «Радиолокация, навигация, связь». – Воронеж, 20 – 23 апреля 1999. – Т. 1. – С. 546 – 550.
15. Microwave Mobile Communications. / W. Jakes, Ed. – N. Y.: IEEE Press, 1993, reissue.
16. Universal Mobile Telecommunication System (UMTS); Selection procedures for the choice of radio transmission technologies for the UMTS (UMTS 30.03 version 3.2.0), TR 101 112 v3.2.0 (1998-04), ETSI, 1998.



С.Н. Кириллов, С.В. Зорин

Алгоритмы биортогонального вейвлет-анализа на основе рекурсивных фильтров

Введение

В современных системах передачи, обработки и хранения нестационарных сигналов используются алгоритмы вейвлет-анализа (ВА) [1], представляющие собой разложение сигнала по сдвигам, сжатиям и растяжениям по времени одной и той же функции, называемой вейвлетной функцией (ВФ). Требования, предъявляемые к ВФ, имеют различный характер, обусловленный спецификой решаемой задачи. В случае обработки и сжатия изображений [2] необходима минимальная длительность по времени ВФ при фиксированном количестве нулевых моментов, а требования, накладываемые на форму спектра, невысокие. В системах сжатия речевых сигналов [3] в силу свойства их локальной стационарности и особенностей слухового восприятия человека существенно повышаются требования к компактности спектра ВФ при отсутствии жестких ограничений к длительности ВФ.

Реализация алгоритмов ВА возможна на основе ортогональных [4] либо биортогональных [5] ВФ. В некоторых случаях биортогональные ВФ по сравнению с ортогональными при одних и тех же вычислительных затратах обеспечивают меньшую среднеквадратическую ошибку восстановления случайного процесса при фиксированном коэффициенте сжатия. Одним из недостатков теории ВА, использующей биортогональные ВФ, является отсутствие соотношений, позволяющих оценивать взаимное влияние корреляционной функции (КФ) случайного сигнала и КФ дуальной и первичной ВФ на качество восстановленного сигнала. Реализация алгоритмов ВА на основе ортогональных и биортогональных ВФ обычно осуществляется в классе нерекурсивных цифровых фильтров. Использование в алгоритмах ВА рекурсивных фильтров по сравнению с нерекурсивными позволит уменьшить ошибку восстановления сигнала при одних и тех же вычислительных затратах.

Цель данной статьи – получение расчетных соотношений для ошибки восстановления сигнала, а также разработка эффективных алгоритмов биортогонального ВА на основе схемы лифтинга, реализованной на рекурсивных фильтрах.

Вывод расчетных соотношений

Дискретный стационарный в широком смысле сигнал s_i : $i = 0..N-1$ может быть представлен как

$$s_i = \sum_{j=0}^{N/2-1} \gamma_j \phi_{ji} + \sum_{j=0}^{N/2-1} \lambda_j \varphi_{ji}, \quad (1)$$

где γ_j, λ_j – коэффициенты разложения, определяющие низкочастотные (НЧ) и высокочастотные (ВЧ) компоненты сигнала, а $\phi_{ji} = \phi(i-2j)$ и $\varphi_{ji} = \varphi(i-2j)$ сдвиги первичных скейлинг функции (СФ) и ВФ соответственно. Индекс i определяет отсчеты времени, а индекс j – номер или сдвиг СФ и ВФ по времени. При сжатии исходного сигнала в 2 раза и отбрасывании при этом всех ВЧ-компонент сигнала λ_j оценка сигнала может быть записана в виде

$$\hat{s}_i = \sum_{j=0}^{N/2-1} \gamma_j \phi_{ji}. \quad \text{Тогда сигнал ошибки определяется}$$

выражением

$$d_i = s_i - \hat{s}_i = \sum_{j=0}^{N/2-1} \lambda_j \varphi_{ji} = \sum_{j=0}^{N/2-1} \lambda_j \varphi(i-2j), \quad (2)$$

а сумма квадратов ошибки восстановленного сигнала по индексу i будет иметь вид

$$d = \sum_{i=0}^{N-1} d_i^2 = \sum_{j_1, j_2=0}^{N/2-1} \lambda_{j_1} \lambda_{j_2} \sum_{i=0}^{N-1} \varphi(i-2j_1) \varphi(i-2j_2) = \sum_{j_1, j_2=0}^{N/2-1} \lambda_{j_1} \lambda_{j_2} K_{\varphi}(2(j_1-j_2)), \quad (3)$$

где $K_{\varphi}(i)$ – КФ первичной ВФ. Дисперсию ошибки восстановленного сигнала получим усреднением d по ансамблю реализаций

$$D = M[d] = \sum_{j_1, j_2=0}^{N/2-1} M[\lambda_{j_1} \lambda_{j_2}] K_{\varphi}(2(j_1-j_2)). \quad (4)$$

Из условия стационарности сигнала можно заключить, что последовательность отсчетов λ_j – тоже стационарна, и представить: $M[\lambda_{j_1} \lambda_{j_2}] = R^{\lambda}_{|j_1-j_2|}$, где $R^{\lambda}_{|j|}$ – КФ ВЧ-компонент сигнала λ_j . Тогда выражение (4) можно преобразовать к виду

$$D = \sum_{j_1, j_2=0}^{N/2-1} R^{\lambda}_{|j_1-j_2|} K_{\varphi}(2(j_1-j_2)). \quad (5)$$

Учитывая, что матрица $A = \{R^{\lambda}_{|j_1-j_2|} K_{\varphi}(2(j_1-j_2))\}$ – симметричная и теплицева, перейдем от двумерного суммирования в (5) к одномерному

$$D = \sum_{i=-N/2+1}^{N/2-1} \left(\frac{1-|i|}{N} \right) R_i^{\lambda} K_{\varphi}(2j). \quad (6)$$

Пусть $2m$ – длина фильтра, реализующего алгоритмы ВА, тогда $K_{\varphi}(2j) = 0$, для всех $i \geq m$. Кроме того, обычно $m \ll N$ и можно полагать,

что $\left(\frac{1-|i|}{2}\right) \approx \frac{1}{2}$. С учетом этого имеем

$$D = \frac{1}{2} \sum_{i=m+1}^{m-1} R_i^\lambda K_\varphi(2i) = R_0^\lambda \frac{K_\varphi(0)}{2} + \sum_{k=0}^{m-1} R_k^\lambda K_\varphi(2k). \quad (7)$$

Определим зависимость, связывающую КФ R_k^λ с КФ входного сигнала R_k^S . Известно, что

$$\lambda_j = \sum_{i=0}^{N-1} s_i \tilde{\varphi}(i-2j), \text{ тогда выражение для } R^\lambda_{|j_1-j_2|}$$

имеет вид

$$R^\lambda_{|j_1-j_2|} = \sum_{i_1, i_2=0}^{N-1} M[s_{i_1} s_{i_2}] \tilde{\varphi}(i_1-2j_1) \tilde{\varphi}(i_2-2j_2), \quad (8)$$

где $\tilde{\varphi}(i)$ – дуальная СФ. Если учесть, что

$$R^\lambda_{|j_1-j_2|} = R^\lambda_{|j_1-j_2|}, M[s_{i_1} s_{i_2}] = R^S_{j_1-j_2}$$

и положить $j_1=0$, а $j_2=k$. То выражение (8) преобразуется следующим образом

$$R_k^\lambda = \sum_{i_1, i_2=0}^{N-1} R^S_{|i_1-i_2|} \tilde{\varphi}(i_1) \tilde{\varphi}(i_2-2k) = \sum_d R_d^S K_{\tilde{\varphi}}(d-2k), \quad (9)$$

где $K_{\tilde{\varphi}}(d)$ – КФ дуальной ВФ.

С учетом выражения (9), дисперсия ошибки восстановленного сигнала при отбрасывании всех ВЧ-компонент равна

$$D_1 = \frac{1}{2} \sum_{k=-m+1}^{m-1} \sum_d R_d^S K_\varphi(2k) K_{\tilde{\varphi}}(2k-d) = \frac{1}{2} \sum_d R_d^S \sum_k K_\varphi(2k) K_{\tilde{\varphi}}(2k-d). \quad (10)$$

Аналогичное выражение можно получить для дисперсии ошибки восстановленного сигнала при отбрасывании его НЧ-компонент:

$$D_2 = \frac{1}{2} \sum_{k=-m+1}^{m-1} \sum_d R_d^S K_\varphi(2k) K_{\tilde{\varphi}}(2k-d) = \frac{1}{2} \sum_d R_d^S \sum_k K_\varphi(2k) K_{\tilde{\varphi}}(2k-d), \quad (11)$$

где $K_\varphi(k)$ и $K_{\tilde{\varphi}}(k)$ – ковариационные функции

первичной и дуальной СФ.

Схема лифтинга

Вейвлет-разложение (ВР) первого уровня может быть осуществлено по схеме подполосной фильтрации, предложенной Маллатом [6], в виде рис. 1, где $h(z) = \sum_i h_i z^{-i}$, $g(z) = \sum_i g_i z^{-i}$,

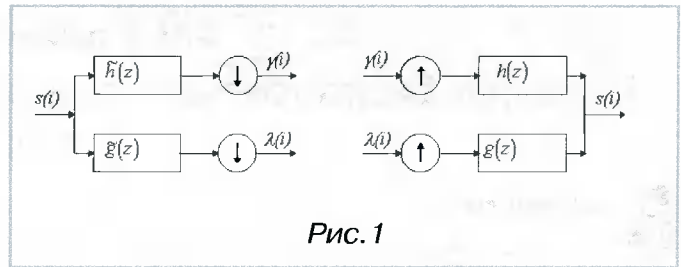


Рис. 1

$$\tilde{h}(z) = \sum_i \tilde{h}_i z^{-i}, \tilde{g}(z) = \sum_i \tilde{g}_i z^{-i} \text{ z-преобразования}$$

импульсных характеристик $h_i, g_i, \tilde{h}_i, \tilde{g}_i$ первичных НЧ- и ВЧ-фильтров, а также дуальных НЧ- и ВЧ-фильтров соответственно, $i \in Z, Z$ – множество целых чисел.

В этом случае условия для точного восстановления обрабатываемого сигнала записываются в виде [5]

$$h(z)\tilde{h}(z^{-1}) + g(z)\tilde{g}(z^{-1}) = 2,$$

$$h(z)\tilde{h}(-z^{-1}) + g(z)\tilde{g}(-z^{-1}) = 0.$$

Для повышения вычислительной эффективности алгоритмов ВА на основе схемы подполосной фильтрации Маллата была предложена схема лифтинга [7]. Для математической записи ВР в соответствии со схемой лифтинга необходимо ввести матрицы модуляции [8]

$$M(z) = \begin{bmatrix} h(z) & h(-z) \\ g(z) & g(-z) \end{bmatrix} \text{ и } \tilde{M}(z) = \begin{bmatrix} \tilde{h}(z) & \tilde{h}(-z) \\ \tilde{g}(z) & \tilde{g}(-z) \end{bmatrix}, \text{ при этом}$$

условие точного восстановления запишется в виде: $\tilde{M}(z^{-1})^T M(z) = 2I$, где T – символ, означающий транспонировку, а I – единичная матрица. Для выявления структуры матриц модуляции воспользуемся полифазным представлением фильтра h , которое задается соотношением

$$h(z) = h_e(z^2) + z^{-1}h_o(z^2), \text{ где } h_e(z) = \sum_k h_{2k} z^{-k} -$$

содержит четные коэффициенты, а

$$h_o(z) = \sum_k h_{2k+1} z^{-k} - \text{нечетные. Для дальнейшего построения}$$

схемы лифтинга введем полифазную матрицу

$$P(z) = \begin{bmatrix} h_e(z) & g_e(z) \\ h_o(z) & g_o(z) \end{bmatrix},$$

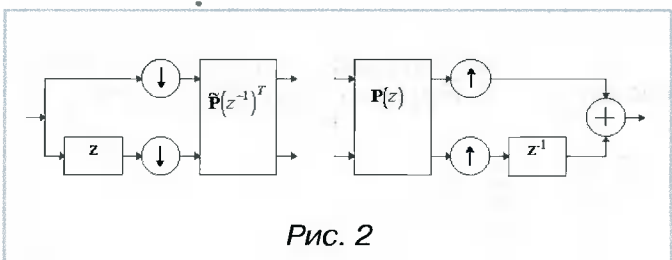


Рис. 2

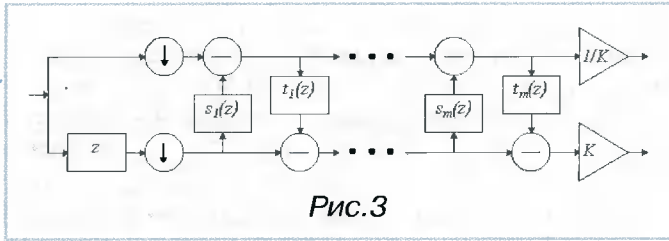


Рис.3

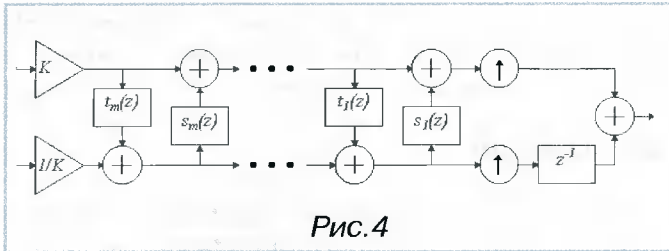


Рис.4

аналогичным образом определяется матрица $\tilde{P}(z)$. Тогда условие точного восстановления обрабатываемого сигнала задается равенством: $P(z)\tilde{P}(z^{-1})^T=I$, а реализация ВР через полифазные матрицы схематически представлена на рис. 2.

Известно [8], что полифазные матрицы, имеющие определитель равный единице, могут быть факторизованы следующим образом

$$P(z) = \prod_{i=1}^m \begin{bmatrix} 1 & s_i(z) \\ 0 & t_i(z) \end{bmatrix} \begin{bmatrix} 1 & 0 \\ t_i(z) & 1 \end{bmatrix} \begin{bmatrix} K & 0 \\ 0 & 1/K \end{bmatrix},$$

$$\tilde{P}(z) = \prod_{i=1}^m \begin{bmatrix} 1 & -s_i(z^{-1}) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ -t_i(z^{-1}) & 1 \end{bmatrix} \begin{bmatrix} 1/K & 0 \\ 0 & K \end{bmatrix}.$$

В соответствии с таким представлением полифазных матриц прямое и обратное вейвлет-преобразования могут быть произведены по схемам лифтинга, приведенным на рис.3 и 4, где $s_i(z), t_i(z)$ – фильтры коррекции и предсказания.

Оптимизация параметров фильтров коррекции и предсказания

Для оптимизации параметров фильтров коррекции и предсказания необходимо ввести понятия эквивалентных ВЧ- и НЧ-фильтров и обосновать критерии синтеза. Обозначим

$$K_g(z) = g(z)g(z^{-1}), K_{\tilde{g}}(z) = \tilde{g}(z)\tilde{g}(z^{-1}),$$

$$K_h(z) = h(z)h(z^{-1}), K_{\tilde{h}}(z) = \tilde{h}(z)\tilde{h}(z^{-1}), K(z)$$

как z-преобразования КФ импульсных характеристик первичного и дуального ВЧ- и НЧ-фильтров, а также КФ произвольного стационарного случайного сигнала соответственно. Учтем, что $\phi(i)=h_i, \phi(i)=g_i, \tilde{\phi}(i)=\tilde{h}_i$ и $\tilde{\phi}(i)=\tilde{g}_i$, тогда дисперсия ошибки восстановления обрабатываемого сигнала, при передаче только НЧ-компонент, имеет в соответствии с выражением (10) вид

$$D_1 = \frac{1}{2\pi_0} \int_{-\pi_0}^{\pi_0} K(e^{i\omega})K_{\tilde{g}}(e^{i\omega})d\omega,$$

$$\text{где } K_{g\tilde{g}}(z) = \frac{K_g(z)+K_g(-z)}{4} K_{\tilde{g}}(z),$$

а в случае передачи только ВЧ-компонент сигнала в соответствии с выражением (11) – вид

$$D_2 = \frac{1}{2\pi_0} \int_{-\pi_0}^{\pi_0} K(e^{i\omega})K_{\tilde{h}}(e^{i\omega})d\omega,$$

$$\text{где } K_{h\tilde{h}}(z) = \frac{K_h(z)+K_h(-z)}{4} K_{\tilde{h}}(z).$$

Анализ выражений для дисперсий ошибок восстановления D_1 и D_2 позволяет рассматривать $K_{g\tilde{g}}(z)$ и $K_{h\tilde{h}}(z)$ как z-преобразования КФ импульсных характеристик некоторых эквивалентных фильтров, на выходе которых сигнал имеет дисперсию, равную дисперсии ошибки восстановления исходного обрабатываемого сигнала в случае отбрасывания ВЧ- и НЧ-компонент соответственно. Введение понятия эквивалентных фильтров необходимо для сравнения различных биортогональных и ортогональных ВР и для определения критериев для оптимизации параметров фильтров коррекции и предсказания.

Идеология ВА построена на разложении обрабатываемого сигнала на ВЧ- и НЧ-компоненты при условии локальности базисных функций. Таким образом, необходимо использовать такие ВР, чтобы соответствующие эквивалентные фильтры были как можно ближе по какому-либо критерию к идеальным НЧ- и ВЧ-фильтрам. В качестве критерия оптимизации параметров фильтров коррекции и предсказания будем использовать минимум значения дисперсии ошибки D_1 в случае сжатия НЧ-случайного сигнала с прямоугольным спектром: $S(\omega)=rect(\omega)$,

$$\omega \in \left[0, \frac{\pi}{2}\right] \cup \left[\frac{3\pi}{2}, 2\pi\right],$$

при дополнительном условии $D_2=D_1$ в случае сжатия ВЧ-случайного сигнала также с прямоугольным спектром.

Для параметризации биортогональных ВР, которым соответствует фиксированная вычислительная эффективность, необходимо задать количество звеньев схемы лифтинга и структуру фильтров коррекции и предсказания. Таким образом, задача сводится к такой оптимизации параметров фильтров коррекции и предсказания, чтобы соответствующие им эквивалентные ВЧ- и НЧ-фильтры имели бы форму, близкую к идеальной по заданному выше критерию.

Необходимо отметить, что широко применяемые на практике биортогональные вейвлеты Когена – Добеши – Фово (2,2),(4,2) и (6,2) [5] в основном приспособлены для сжатия НЧ-сигналов, большей частью изображений, а информация о ВЧ-компонентах передается посредством фиксации тех участков сигнала, где имеются сингулярности (неоднородности или резкие переходы). Для этих ВР амплитудно-частотная характеристика (АЧХ) $|Q(\omega)|$ эквивалент-

ного ВЧ-фильтра, определяющего ошибку восстановления НЧ-случайного сигнала, гораздо ближе по форме к идеальной, чем АЧХ эквивалентного НЧ-фильтра. В случае ортогональных ВР эквивалентные фильтры совпадают с НЧ- и ВЧ-фильтрами (Н и G), соответственно их АЧХ имеют одну и ту же форму относительно половины частоты дискретизации (т.е. ортогональные ВР могут с одинаковым качеством использоваться как для обработки ВЧ-, так и НЧ-сигналов). По этой причине в вейвлетно-пакетных преобразованиях наиболее часто применяют ортогональные ВФ, так как необходимо разбивать весь частотный диапазон с одинаковым качеством.

Исследуем схему лифтинга с одним корректором и предсказателем, где фильтры коррекции и предсказания имеют z-преобразование своих импульсных характеристик следующего вида

$$s(z) = \frac{\sum_i a_i z^{-i}}{1 - \sum_i b_i z^{-i}}, \quad t(z) = \frac{\sum_i c_i z^{-i}}{1 - \sum_i d_i z^{-i}}.$$

Реализация корректора и предсказателя на основе рекурсивных фильтров не нарушает требования локализации ВФ, используемых в биортогональных ВР. Экспериментальные исследования показали целесообразность использования корректора и предсказателя, z-преобразования импульсных характеристик которых имеют вид

$$s(z) = \frac{a_0 + a_1 z^{-1} + a_2 z^{-2}}{1 - b_1 z^{-1}}, \quad t(z) = \frac{c_0 + c_1 z^{-1} + c_2 z^{-2}}{1 - d_1 z^{-1}}.$$

Минимизацией дисперсий ошибок D_1 и D_2 при дополнительном условии равенства $D_2 = D_1$ были получены следующие значения коэффициентов фильтров предсказания и коррекции: $a_0 = -0.671$, $a_1 = -1.067$, $a_2 = -0.325$, $b_1 = -0.706$, $c_0 = 0.471$, $c_1 = 0.242$, $c_2 = -0.125$, $d_1 = -0.455$. На рис.5 приведены зависимости $y(\omega) = |Q(\omega)|^2$ эквивалентных НЧ- и ВЧ-фильтров предложенного биортогонального ВР (кривые 1,2). При одних и тех же вычислительных затратах использование рекурсивной части фильтров коррекции и предсказания в предложенном биортогональном ВР позволяет получить дисперсию ошибки восстановления случайных НЧ- и ВЧ-сигналов, име-

ющих прямоугольный спектр, $D_1 = D_2 = 5.85\%$, в то время как ортогональные ВР Добеши (4 – нулевых момента) обеспечивает $D_1 = D_2 = 9.055\%$ (рис. 5, кривые 3,4), а биортогональное вейвлет ВР Когена – Добеши – Фово (2,6) $D_1 = 10.357\%$, $D_2 = 8.213\%$ (рис.5, кривые 5,6). Необходимо отметить, что примерно такие же дисперсии ошибок, как у предложенного биортогонального ВР-будут при использовании ВР Добеши с 20 нулевыми моментами и ВР Когена – Добеши – Фово (4,16) при возрастании вычислительных затрат в 2.5 раза.

Заключение

Получено выражение для дисперсии ошибки восстановления случайного стационарного сигнала при отбрасывании либо ВЧ, либо НЧ его компонентов в случае биортогональных ВР. Предложена реализация биортогональных ВР на основе схемы лифтинга, в которой применяются в качестве корректоров и предсказателей рекурсивные фильтры, и произведена оптимизация коэффициентов рекурсивных фильтров предсказания и коррекции схемы лифтинга. При этом по сравнению с алгоритмами на основе биортогональных ВФ Когена – Добеши – Фово (2,6) и ортогональных ВФ Добеши (4 нулевых момента) обеспечивается в 1.5 раза меньшая дисперсия ошибки восстановления сигнала с прямоугольным спектром при одних и тех же вычислительных затратах.

Литература

1. Кравченко В.Ф., Рвачев В.А. Wavelet-системы и их применение в обработке сигналов. – Зарубежная радиоэлектроника, 1996, №4.
2. M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies. Image coding using the wavelet transform – IEEE Trans. on Image Processing, 1992 vol.1.
3. M.V. Wickerhauser, Acoustic signal compression with wavelet packets. – In Charles K. Chui, editor, Wavelets – A Tutorial in Theory and Applications, Academic Press, Boston, 1992.
4. I. Daubechies. Orthonormal bases of compactly supported wavelets. Comm. Pure Appl. Math., 41:909-996, 1988.
5. A. Cohen, I. Daubechies, and J. Feauveau. Biorthogonal bases of compactly supported wavelets. Comm. Pure Appl. Math., 45:485-560, 1992.
6. Mallat S.G. A theory for multiresolution signal decomposition: The wavelet representation – IEEE Trans. Patt. Anal. And Mach. Intell. 1989. 11. P.674.
7. W. Sweldens and P. S. Schroder. Building your own wavelets at home. In Wavelets in Computers Graphics, pages 15-87. ACM SIGGRAPH Course note, 1996. <http://cm.belllab.com/who/wim/papers/papers.html#athome>.
8. I. Daubechies and W. Sweldens, Factoring Wavelets Transforms into Lifting Steps, Program for Applied and Computational Mathematics, Princeton University, Princeton NJ 08544.

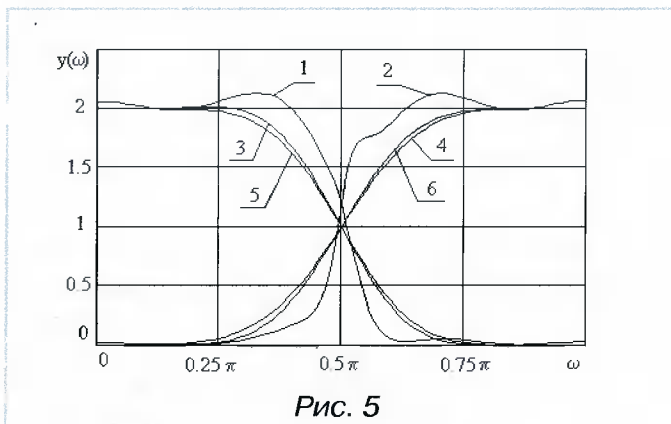


Рис. 5

А.В. Дворкович

Еще об одном методе расчета эффективных оконных функций, используемых при гармоническом анализе с помощью ДПФ

Предлагается еще один новый метод создания эффективных окон, основанный на расчете формы специальных ограниченных на заданном временном интервале функций, совпадающих с заданной точностью с формой их спектра. Анализируются основные параметры таких оконных функций.

В статье [1] рассмотрен метод расчета оконных функций, основанный на минимизации мощности их спектральных компонент вне пределов заданного интервала частот. При этом предполагается, что такие функции симметричны относительно середины временного интервала $-T/2 \leq t \leq T/2$, ограничены по длительности этим интервалом и представимы в виде

$$u(x) = \frac{1 + 2 \sum_{m=1}^M a_m \cos(2\pi m x)}{1 + 2 \sum_{m=1}^M a_m}, \quad -1/2 \leq x \leq 1/2, \quad (1)$$

где $x = t/T$ – нормированный временной интервал. Нормированный спектр такой функции

$$F(y) = \text{sinc}(\pi y) + \sum_{m=1}^M a_m [\text{sinc}(\pi(y+m)) + \text{sinc}(\pi(y-m))], \quad |y| < \infty, \quad (2)$$

где $\text{sinc}(z) = \frac{\sin(z)}{z}$, $y = \omega T/2\pi = fT$ – нормированная частота, содержит $(2M + 1)$ компонент, центральный из которых ($\text{sinc}(\pi y)$) имеет равный 1 максимум при $y = 0$, а остальные смещены от центрального на относительные частотные интервалы $-M, -(M-1), \dots, -2, -1, 1, 2, \dots, (M-1), M$.

Таким образом, функция $F(y)$ всегда равна нулю в точках, кратных целым числам и смещенных относительно нуля на интервал $(M+1)$.

Преобразуем аргумент функции (1) так, чтобы временной интервал совпадал с относительным временным отрезком $[-(M+1), (M+1)]$

$$u(y) = \frac{1 + 2 \sum_{m=1}^M a_m \cos(\pi m y / (M+1))}{1 + 2 \sum_{m=1}^M a_m}, \quad -(M+1) \leq y \leq (M+1). \quad (3)$$

Вне пределов указанного интервала $u(y)$ тождественно равна нулю.

Если выполняется соотношение

$$1 + 2 \sum_{m=1}^M (-1)^m a_m = 0, \quad (4)$$

то функции $F(y)$ и $u(y)$ совпадают в точках $y = -(M+1)$; $y = 0$; $y = (M+1)$.

Это соотношение определяет равенство нулю оконной функции на границах временного интервала T . Учитывая, что и производная функции (3) на границах интервала $[-(M+1), (M+1)]$ также равна нулю, все оконные функции, определяемые соотношениями (3) и (4), обладают скоростью спада боковых лепестков (ΔW) не менее 18 дБ на октаву [2].

Функция $F(y)$, близко совпадающая с окном $u(y)$ и обладающая минимальным размером боковых лепестков (в среднеквадратичном смысле), может быть рассчитана при выполнении условия

$$\int_{-\infty}^{\infty} [F(y) - u(y)]^2 dy \rightarrow \min.$$

Возможно использование более простого варианта расчета функций $u(y)$, близких по форме к их спектру $F(y)$, который связан с решением системы уравнений [3]

$$a_k = \frac{1 + 2 \sum_{m=1}^M a_m \cos(\pi m k / (M+1))}{1 + 2 \sum_{m=1}^M a_m}, \quad 1 \leq k \leq M. \quad (5)$$

Численный анализ этой системы уравнений показывает, что

$$1) \quad 1 + 2 \sum_{m=1}^M a_m = \sqrt{2(M+1)}, \quad M \geq 1 \quad (6)$$

(эта величина определяет коэффициент усиления оконной функции $CA = 1/\sqrt{2(M+1)}$);
2) при $M > 3$ система (5) содержит $n = \lfloor \frac{M-1}{2} \rfloor$ независимых уравнений ($\lfloor \cdot \rfloor$ – целая часть).

Таким образом, если в системе (5) исключить n уравнений, тем не менее функции $u(y)$ и $F(y)$ будут совпадать во всех $(2M + 3)$ точках оси абсцисс.

Для вычисления всех коэффициентов оконной функции (1) систему можно дополнить уравнениями либо равенства $u(y)$ и $F(y)$ при нецелых значениях y , либо равенства нулю высших производных $u(y)$ и/или $F(y)$ при $y = M + 1$, что, в свою очередь, может привести к возрастанию скорости спада боковых лепестков (ΔW) Фурье-спектра оконной функции. Ниже приводятся результаты расчета оконных

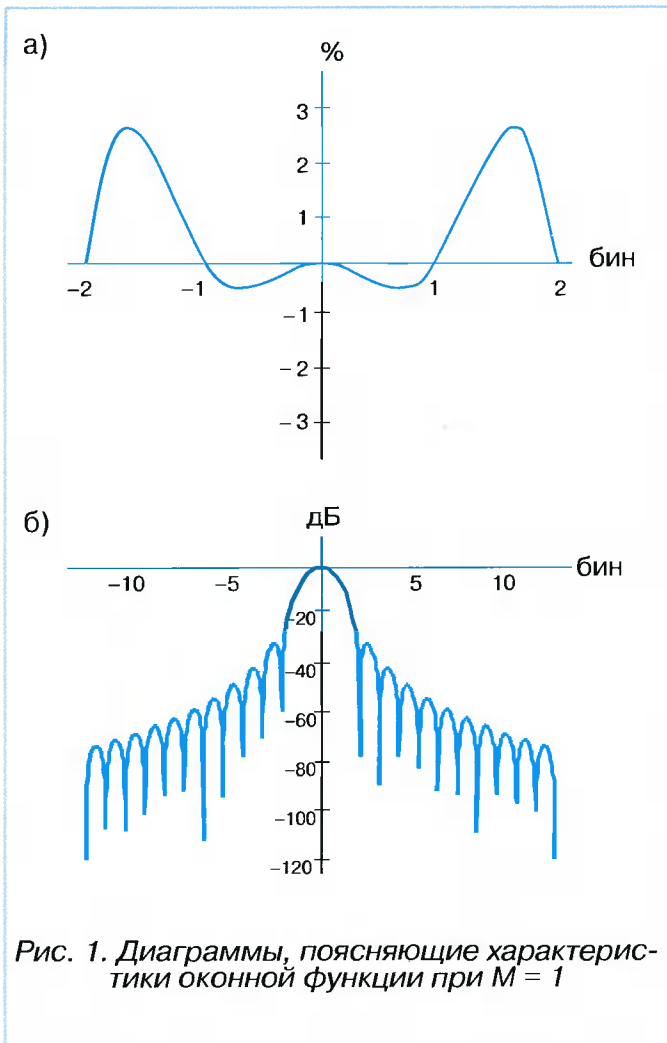


Рис. 1. Диаграммы, поясняющие характеристики оконной функции при $M = 1$

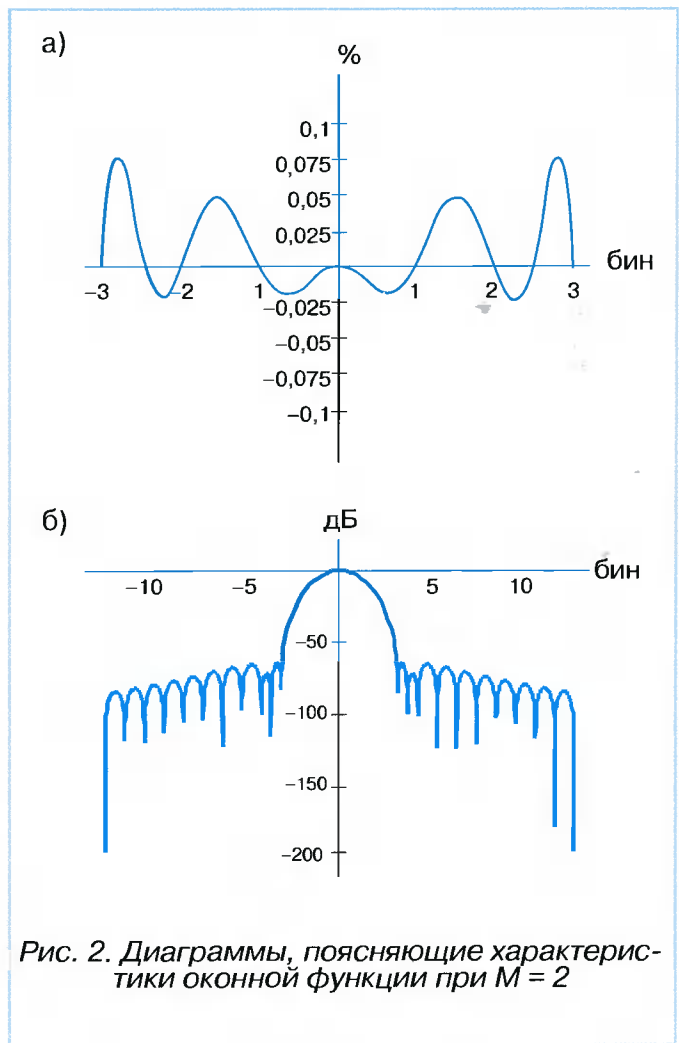


Рис. 2. Диаграммы, поясняющие характеристики оконной функции при $M = 2$

функций в соответствии с предлагаемой методикой.

I. $M = 1$. В этом случае $a_1 = 0,5$, при котором реализуется оконная функция Хэннинга. $CA = 0,5$.

На рис. 1а приведена кривая, определяющая отклонение функций $F(u)$ и $u(u)$ на интервале $[-2, 2]$ в процентах относительно их максимального значения, равного 1 при $u = 0$. Максимальное отклонение составляет более 2,5%.

На рис. 1б изображен логарифм модуля преобразования Фурье для этой функции. Ее параметры даны в табл. 1. Следует заметить, что максимальный боковой лепесток вне пределов интервала $[-2, 2]$ оказывается равным по величине максимальному отклонению функций $F(u)$ и $u(u)$ внутри этого интервала.

II. $M = 2$. В этом случае $a_1 = \frac{1}{2}\sqrt{\frac{3}{2}}$; $a_2 = \frac{1}{2}(\sqrt{\frac{3}{2}} - 1)$; $CA = \frac{1}{\sqrt{6}}$.

Характеристики и параметры такой оконной функции приведены на рис. 2 и в табл. 1. Максимальное отклонение функций $F(u)$ и $u(u)$ внутри интервала $[-3, 3]$ и величина максимального бокового лепестка составляют в данном случае более 0,05%.

III. $M = 3$. В данном случае возможна реализация нескольких вариантов, поскольку возможно использование дополнительного усло-

вия. Например, если в качестве дополнительного условия использовать равенство нулю производной функции $F(u)$ при $u = M + 1 = 4$, то коэффициенты, определяющие оконную функцию, соответственно будут равны: $a_1 = 5(20\sqrt{2+7})/256$; $a_2 = (\sqrt{2}-1)/2$; $a_3 = 7(4\sqrt{2-5})/256$. $CA = 1/2\sqrt{2}$. Параметры такой оконной функции приведены в графе 3.1 табл. 1.

Однако лучшие параметры окна можно получить при использовании условия $F(u) = u(u)$ в точке $u = 3,68$. В этом случае $a_1 = 0,6882754385$; $a_2 = 0,2071067812$; $a_3 = 0,0188313426$, параметры приведены в графе 3.2 табл. 1, а характеристики поясняются рис. 3. Максимальное отклонение функций $F(u)$ и $u(u)$ внутри интервала $[-4, 4]$ и величина максимального бокового лепестка составляют в данном случае 0,0025%. Следует заметить, что точка $u = 3,68$ выбрана таким образом, что максимальные отклонения функций в положительную и отрицательную стороны примерно равны и минимальны по величине.

IV. $M = 4$. При дополнительном условии равенства нулю производной функции $F(u)$ в точке $u = M + 1 = 5$ коэффициенты, определяющие оконную функцию, соответственно равны

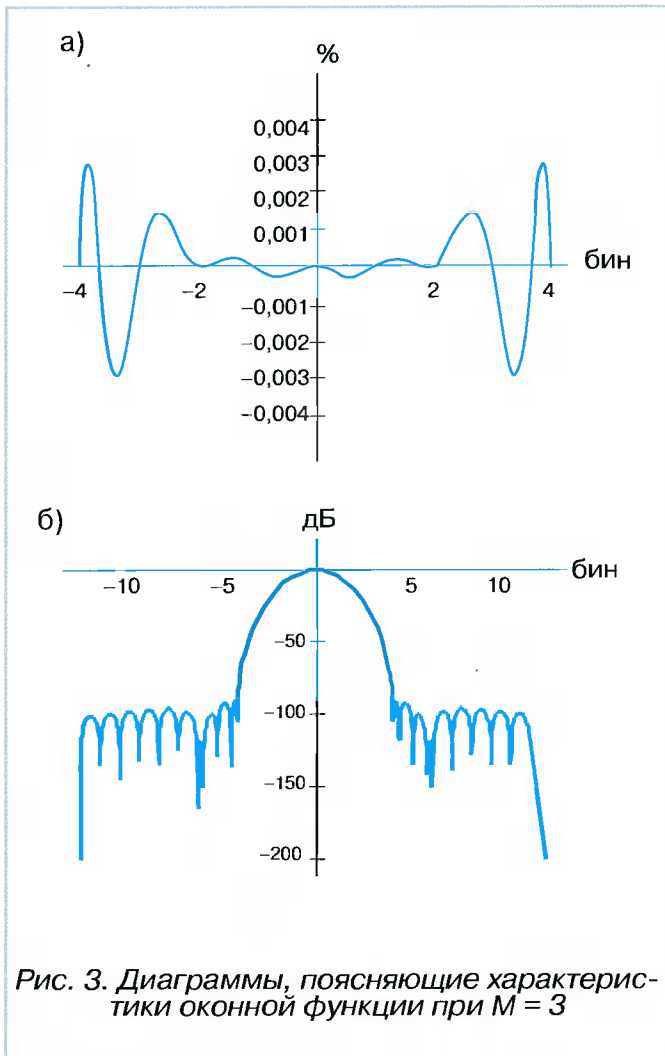


Рис. 3. Диаграммы, поясняющие характеристики оконной функции при $M = 3$

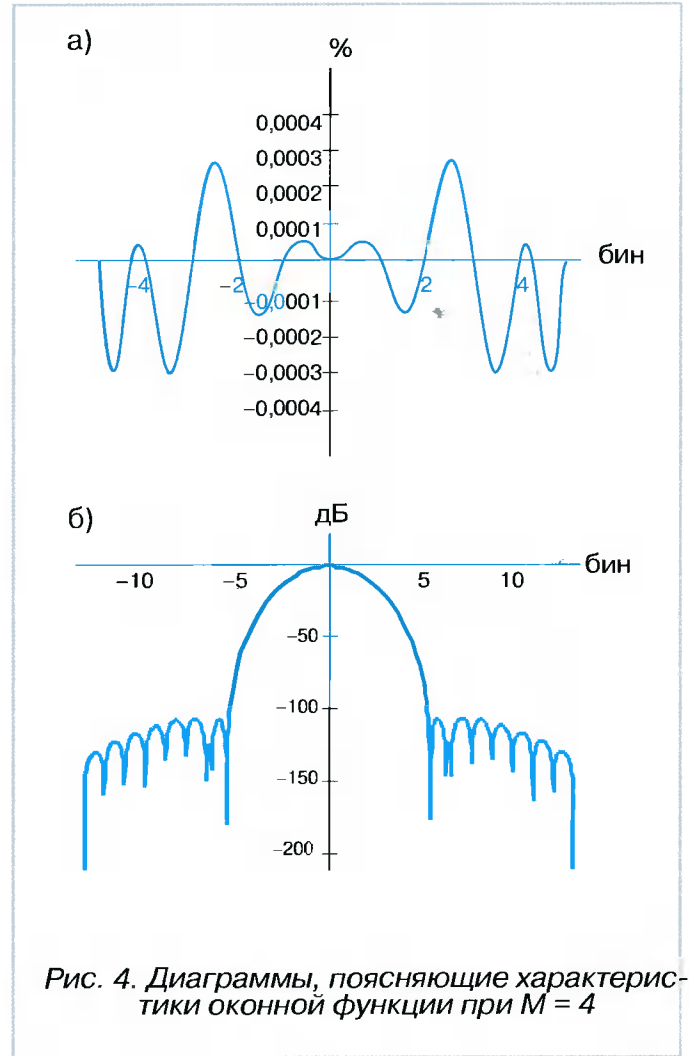


Рис. 4. Диаграммы, поясняющие характеристики оконной функции при $M = 4$

$$a_1 = \frac{992\sqrt{2} + 375\sqrt{5} - 800\sqrt{10}}{50(85\sqrt{2} - 128)};$$

$$a_2 = \frac{7(1024 - 587\sqrt{2} + 500\sqrt{5} - 425\sqrt{10})}{100(85\sqrt{2} - 128)};$$

$$a_3 = \frac{-496\sqrt{2} + 875\sqrt{5} - 400\sqrt{10}}{25(85\sqrt{2} - 128)};$$

$$a_4 = \frac{3(-256 - 47\sqrt{2} + 250\sqrt{5} - 75\sqrt{10})}{100(85\sqrt{2} - 128)};$$

Параметры оконной функции приведены в графе 4 табл. 1, а характеристики поясняются рис. 4. Максимальное отклонение функций $F(y)$ и $u(y)$ внутри интервала $[-5, 5]$ и величина максимального бокового лепестка составляют в данном случае менее 0,0003%.

V. $M = 5$. Наилучший результат получается при дополнительном условии равенства нулю производной функции $F(y)$ в точке $y = M + 1 = 6$ и при равенстве функций $F(y)$ и $u(y)$ в точке $y = 5,5$. Коэффициенты, определяющие оконную функцию, соответственно равны:

$$a_1 = 0,7769281169765806; a_2 = 0,351002835937277;$$

$$a_3 = 0,0888108108173183; a_4 = 0,0099251201907108;$$

$$a_5 = 0,000286475990536.$$

Параметры оконной функции приведены в графе 5 табл. 1, а характеристики поясняются рис. 5. Максимальное отклонение функций $F(y)$ и $u(y)$ внутри интервала $[-6, 6]$ и величина максимального бокового лепестка составляют в данном случае менее 0,00001%.

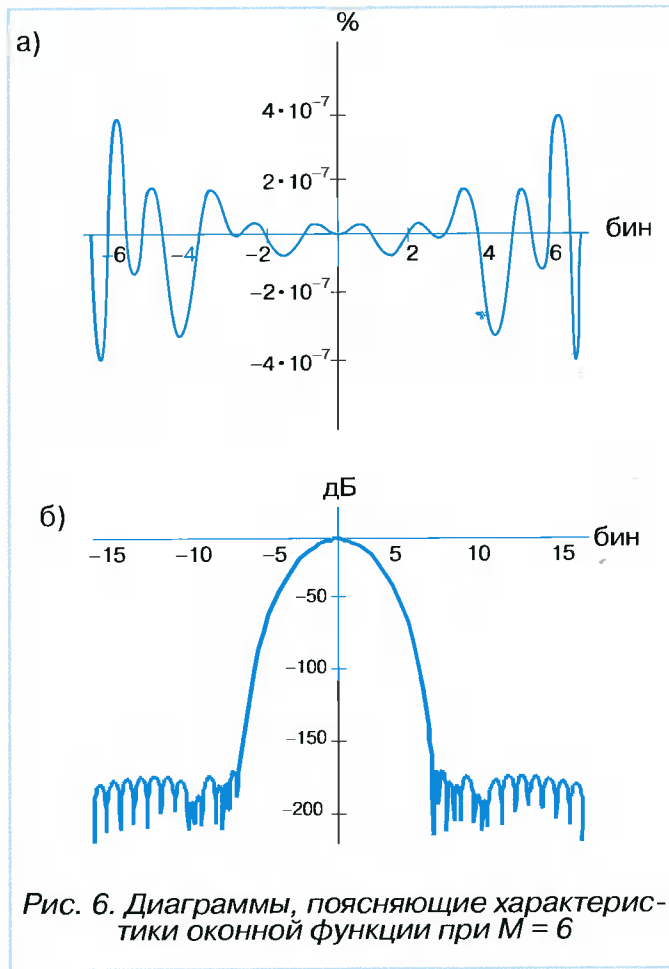
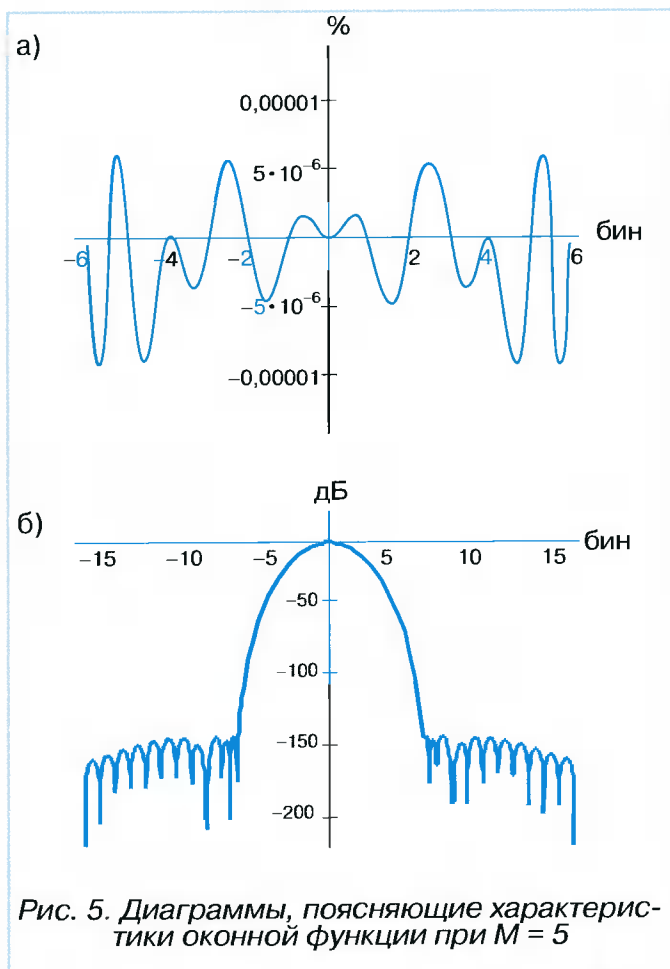
VI. $M = 6$. Наилучший результат получается при дополнительном условии равенства нулю производной функции $F(y)$ в точке $y = M + 1 = 7$ и при равенстве функций $F(y)$ и $u(y)$ в точке $y = 6,58$. Коэффициенты, определяющие оконную функцию, соответственно равны:

$$a_1 = 0,8045531907379879; a_2 = 0,413151892914697;$$

$$a_3 = 0,1291637830851933; a_4 = 0,0222305222584773;$$

$$a_5 = 0,001697372870304; a_6 = 0,0000319311940148.$$

Параметры оконной функции приведены в графе 6 табл. 1, а характеристики поясняются рис. 6. Максимальное отклонение функций $F(y)$ и $u(y)$ внутри интервала $[-7, 7]$ и величина максимального бокового лепестка составляют в данном случае менее $4 \cdot 10^{-7}$ %.



VII. $M = 7$. Наилучший результат получается при дополнительных условиях равенства функций $F(y)$ и $u(y)$ в точках $y = 6,5$, $y = 7,5$ и $y = 7,8$. Коэффициенты, определяющие оконную функцию, соответственно равны:

$$a_1=0,8259742277366686; a_2=0,461278708395833;$$

$$a_3=0,1691587106274119; a_4=0,0384454662766642;$$

$$a_5=0,004863024608423; a_6=0,0002758253275029;$$

$$a_7=0,0000040370274967.$$

Параметры оконной функции приведены в графе 7 табл. 1, а характеристики поясняются рис. 7. Максимальное отклонение функций $F(y)$ и $u(y)$ внутри интервала $[-8, 8]$ и величина максимального бокового лепестка составляют в данном случае менее $2 \cdot 10^{-8}\%$.

VIII. $M = 8$. Наилучший результат получается при дополнительных условиях равенства функций $F(y)$ и $u(y)$ в точках $y = 7,5$, $y = 8,5$ и $y = 8,85$. Коэффициенты, определяющие оконную функцию, соответственно равны:

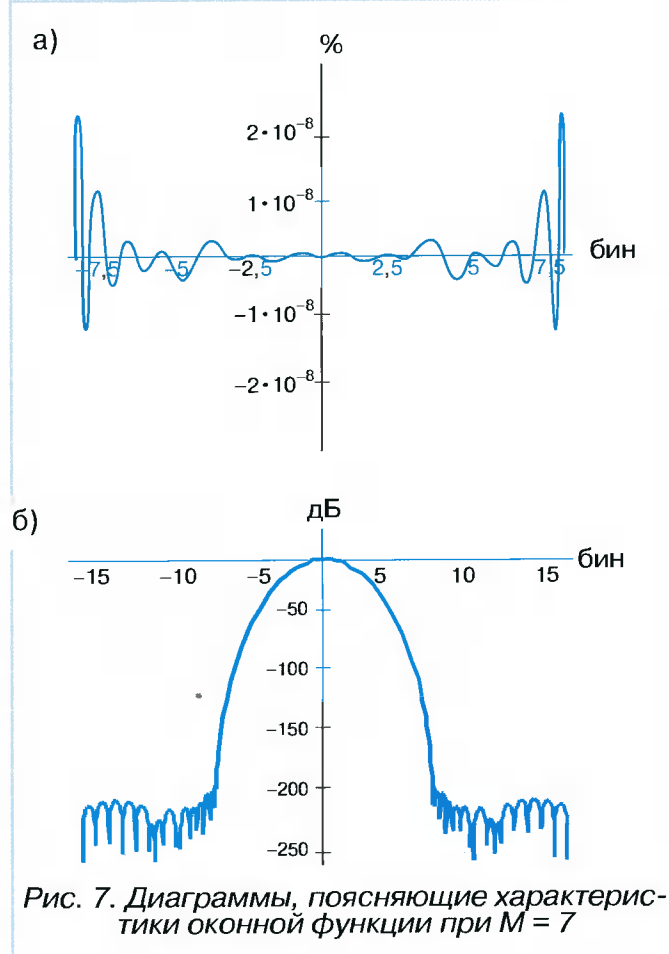
$$a_1=0,8432792099766920; a_2=0,502555969824581;$$

$$a_3=0,2074980193316718; a_4=0,0571592359599415;$$

$$a_5=0,009842608422934; a_6=0,0009445324360745;$$

$$a_7=0,0000403340485263; a_8=0,000000433559220.$$

Параметры оконной функции приведены в графе 8 табл. 1, а характеристики поясняются



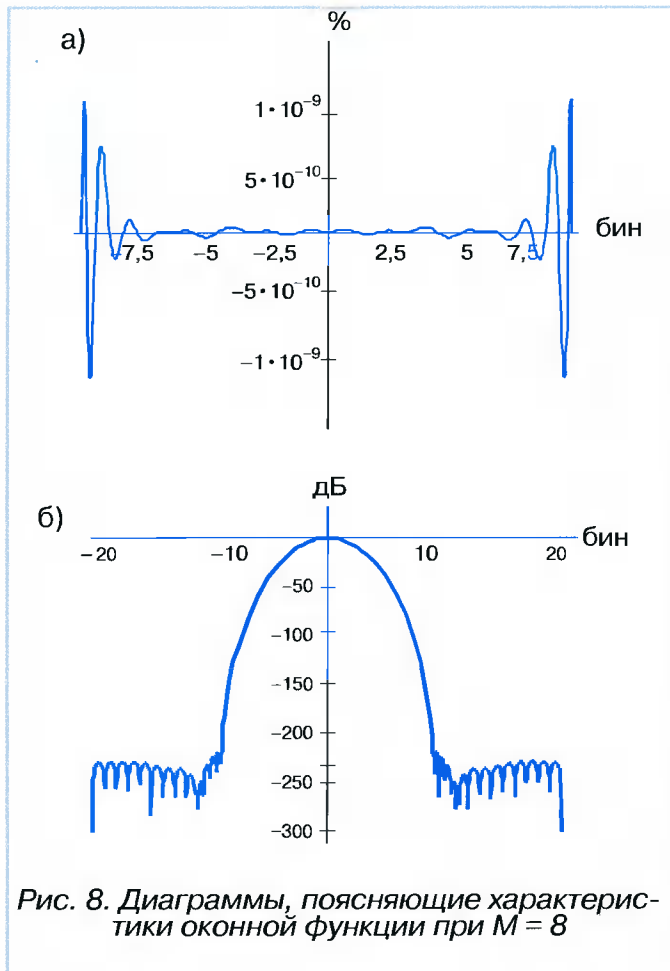


Рис. 8. Диаграммы, поясняющие характеристики оконной функции при $M = 8$

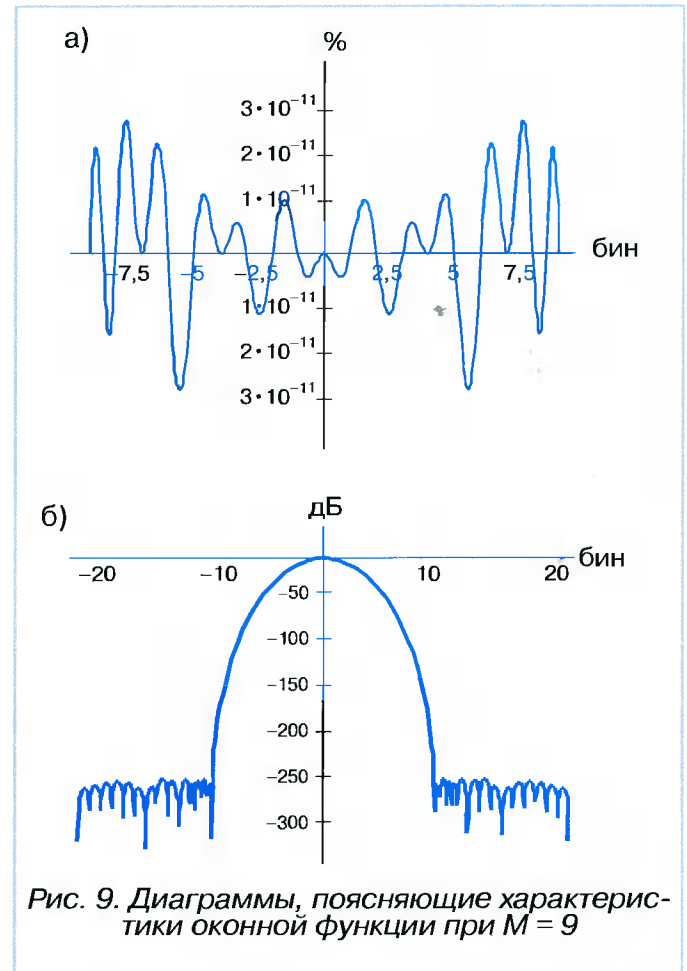


Рис. 9. Диаграммы, поясняющие характеристики оконной функции при $M = 9$

рис. 8. Максимальное отклонение функций $F(y)$ и $u(y)$ внутри интервала $[-9, 9]$ и величина максимального бокового лепестка составляют $10^{-9}\%$.

IX. $M = 9$. Наилучший результат получается при дополнительных условиях равенства функций $F(y)$ и $u(y)$ в точках $y = 8,5$, $y = 9,5$ и $y = 9,75$ и равенства нулю производной функции $F(y)$ при $y = 10$. Коэффициенты, определяющие оконную функцию, соответственно равны:

$$a_1=0,8576328844007747; a_2=0,538450117111417; \\ a_3=0,2437902969544110; a_4=0,0774107971864791; \\ a_5=0,016456005075136; a_6=0,0021685042704880; \\ a_7=0,0001547749600201; a_8=0,000004570181510; \\ a_9=0,0000000273595535.$$

Параметры оконной функции приведены в графе 9 табл. 1, а характеристики поясняются рис. 8. Максимальное отклонение функций $F(y)$ и $u(y)$ внутри интервала $[-10, 10]$ и величина максимального бокового лепестка составляют менее $3 \cdot 10^{-11}\%$.

Об эффективности рассмотренного метода расчета оконных функций можно судить по следующим данным.

1. Крутизна зависимости максимального уровня боковых лепестков W_{max} от изменения эквивалентной шумовой полосы $\Delta F_{ш}$ составляет 130 дБ/бин, в то время как этот параметр

для наиболее широко используемых окон Барсилона – Темеша, Кайзера – Бесселя и Дольфа – Чебышева соответственно равен 72 , 84 и 88 дБ/бин.

2. Сравнение оконных функций по крутизне зависимости максимального уровня боковых лепестков W_{max} от максимальных потерь преобразования LT также говорит о преимуществах функций, указанных в табл. 1, – на каждый децибел прироста величины LT уменьшение W_{max} составляет 101 дБ, в то время как для окон Барсилона – Темеша, Кайзера – Бесселя и Дольфа – Чебышева эта величина соответственно равна 60 , 67 и 83 дБ.

Заключение

Широкое внедрение цифровых технологий выдвигает новые требования к обработке сигналов. В частности, при применении много-разрядных АЦП для анализа спектральных компонент сигнала с помощью БПФ требуется применение оконных функций с максимальным уровнем боковых лепестков, существенно меньшим – 100 дБ. Предложен эффективный метод создания окон, основанный на расчете формы специальных ограниченных на заданном временном интервале функций, совпадающих с заданной точностью с формой их спектра. Этот метод позволяет рассчитать необхо-

Таблица 1

М	W_{max} , дБ	СА	$\Delta F_{ш}$, бин	$\Delta F_{3дБ}$, бин	δ , %	$\Delta F_{6дБ}$, бин	РАМ, дБ	ЛТ, дБ	Cor(r), %	
									r=0,75	r=0,5
1	-31,5	0,5000	1,5000	1,4406	4,12	2,0000	1,4236	3,1845	65,9	16,7
2	-63,3	0,4082	1,7753	1,6886	5,13	2,3632	1,0417	3,5343	54,8	7,8
3.1	-78,8	0,3536	2,0363	1,9308	5,46	2,7092	0,7984	3,8868	44,9	3,3
3.2	-91,1	0,3536	2,0339	1,9275	5,52	2,7058	0,8016	3,8850	44,9	3,4
4	-110,8	0,3162	2,2671	2,1452	5,68	3,0153	0,6480	4,2027	36,8	1,4
5	-140,5	0,2887	2,4768	2,3404	5,83	3,2935	0,5452	4,4841	30,1	0,6
6	-168,0	0,2673	2,6704	2,5205	5,95	3,5503	0,4704	4,7361	24,7	0,3
7	-194,0	0,2500	2,8503	2,6886	6,01	3,7890	0,4140	4,9628	20,3	0,1
8	-219,9	0,2357	3,0202	2,8474	6,07	4,0146	0,3693	5,1697	16,6	0,05
9	-250,4	0,2236	3,1823	2,9997	6,09	4,2300	0,3329	5,3603	13,6	–

Примечания:

W_{max} – максимальный уровень боковых лепестков в дБ по отношению к уровню основного лепестка модуля преобразования Фурье;

СА – когерентное усиление оконной функции;

$\Delta F_{ш}$ – эквивалентная шумовая полоса;

$\Delta F_{3дБ}$ – ширина полосы основного лепестка по уровню –3 дБ;

$\delta = 100 \cdot (\Delta F_{ш} - \Delta F_{3дБ}) / \Delta F_{3дБ}$, % – показатель качества окна;

$\Delta F_{6дБ}$ – ширина полосы основного лепестка по уровню –6 дБ;

РАМ – паразитная амплитудная модуляция;

ЛТ – максимальные потери преобразования;

Cor(r) – корреляция случайных компонент сигнала в преобразованиях двух соседних участков последовательности при относительном перекрытии этих участков, равном r.

димые оконные функции. Проведен анализ основных параметров таких оконных функций. Результаты этого анализа показали, что разработанные оконные функции имеют значительные преимущества по сравнению со стандартными окнами.

Литература

1. Дворкович А. В. Новый метод расчета эффективных оконных функций, используемых

при гармоническом анализе с помощью ДПФ // Цифровая обработка сигналов, 2001, №

2. Хэррис Ф. Дж. Использование окон при гармоническом анализе методом дискретного преобразования Фурье. – ТИИЭР, 1978, т.6, №1, с. 60 – 96.

3. Дворкович В. П. Способ измерения линейных характеристик канала связи / Патент России №1518924 от 18 мая 1993.



Р.В. Белянкин, В.В. Риман, А.И. Селянин, В.А. Фин, А.А. Чернышев

Медицинское портативное ультразвуковое диагностическое устройство на базе модуля ЦОС типа ADP62/6701E

Введение

Портативное ультразвуковое диагностическое устройство (ПДУ) – новая оригинальная конверсионная разработка НИИ точных приборов Росавиакосмоса. ПДУ предназначено для дву- и трехмерной визуализации внутренних органов, мягких тканей и кровеносных сосудов с высокой пространственной разрешающей способностью и возможностью определения скорости кровотока различными методами.

Актуальность разработки ПДУ обуславливается социальной значимостью проекта. Она выражается в большой потребности отечественных медицинских учреждений в портативных УЗ-диагностических устройствах. Эти устройства, с одной стороны, должны обладать диагностическими возможностями на уровне современных зарубежных элитных стационарных диагностических устройств, а с другой – быть доступными по цене. После дефолта в августе 1998 г. покупка зарубежных УЗ-диагностических устройств для большинства отечественных медицинских учреждений стала практически невозможной. В связи с этим фактом особую актуальность приобрела задача ускоренной разработки современных отечественных УЗ-диагностических устройств.

Разработка ПДУ базируется на предыдущих работах НИИ точных приборов в области создания стационарной УЗ-диагностической аппаратуры, в частности визуализатора предстательной железы и других органов. В процессе разработки стационарных устройств было преодолено большое количество технических трудностей. ПДУ использует научно-технический задел предыдущих разработок в части математического обеспечения, системотехники, схемотехники и клинического опыта.

В основе новых разработок (в части схемотехнической реализации) лежит идея адаптивного ультразвукового локатора. При этом для уменьшения коммерческого риска, повышения экономической устойчивости, привлечения дополнительных финансовых и трудовых ресурсов, унификации и удешевления проводимых работ одновременно с ПДУ разрабаты-

вается портативный акустический компьютерный дефектоскоп (ПАКД). Это устройство позволит получать УЗ-изображения интересных сечений анализируемых объектов, а не только осциллографические кривые отраженных сигналов, как в существующих дефектоскопах. ПАКД сможет найти применение в промышленности, на транспорте, при эксплуатации вооружений и пр. ПДУ и ПАКД используют практически единые (за исключением некоторых УЗ-датчиков) аппаратные средства и большое количество общих программных модулей. Разработка ПАКД, в свою очередь, обогащает разработку ПДУ и других УЗ-медицинских диагностических устройств. Единая техническая политика в части разработки семейств медицинских УЗ-диагностических устройств и дефектоскопов позволяет во много раз уменьшить затраты на разработку при одновременном сокращении сроков разработок и повышении качества устройств.

Так же, как и в стационарных аппаратах, в ПДУ для получения изображений используется метод синтеза апертуры, который дает возможность улучшить разрешающую способность в поперечном направлении и одновременно уменьшить объем аппаратных средств и сложность УЗ-датчиков, а следовательно, себестоимость диагностических УЗ-аппаратов.

Разработка эффективных медицинских ПДУ с функциональными возможностями стационарных офисных устройств опирается на последние схемотехнические решения АО "Инструментальные системы", реализованные в их автономном модуле ЦОС ADP62/6701E [1]. Модуль ADP62/6701E является функциональным аналогом модуля ADP6201, используемого в стационарных аппаратах. Оба модуля содержат одинаковые цифровые сигнальные процессоры (ЦСП) TMS320C6201 (или TMS320C67). Поэтому переход от стационарного варианта к портативному практически не требует переделки программного обеспечения ЦСП. Приемопередающие устройства стационарных и портативной модификаций также одинаковые. ПДУ реализуется на ноутбуке. Модуль ЦОС связан с ноутбуком посредством

высокоскоростного интерфейса IEEE 1394. Питание ПДУ может осуществляться от промышленной сети, автомобильной сети 12 В или аккумуляторов.

Принципы и режимы работы ПДУ

Прежде чем перейти к описанию принципов и режимов работы ПДУ, отметим основные медико-технические требования, которым должны удовлетворять разрабатываемые диагностические устройства (ПДУ должно обеспечивать работу в режимах формирования дву- и трехмерных УЗ-изображений, а также доплерограмм (спектрограмм) и карт кровотока, их обработки и анализа):

1. Двумерная визуализация должна выполняться посредством УЗ-датчиков, имеющих центральную частоту, лежащую в диапазоне 3,5 – 7,5 МГц.
2. Двумерные УЗ-изображения должны формироваться в режиме реального времени.
3. Время синтеза и отображения на экране трехмерного УЗ-изображения объектов исследования должно быть не более 60 с.
4. Программными средствами должны обеспечиваться возможности выделения на “замороженных” изображениях поверхностей отдельных структур и геометрических измерений (определение по УЗ-изображениям длин, периметров, площадей и объемов).
5. Прибор должен позволять проводить доплерографические исследования в следующих режимах:
 - звукового доплера (ЗД) (для мобильного использования ПДУ);
 - спектрального доплера (СД);
 - цветового доплеровского картирования (ЦДК);

– энергетического доплеровского картирования (ЭДК).

6. Формат экрана дисплея – не хуже 800*600 точек.

7. Вычислительно-отображающая подсистема ПДУ должна быть построена на основе портативного ПК с необходимыми периферийными устройствами.

8. Оператору должны быть обеспечены возможности:

- управления прибором в диалоговом режиме;
- технической диагностики;

- отображения на экране дисплея двумерных изображений в реальном времени;

- синтеза и отображения на экране дисплея псевдообъемного трехмерного изображения;

- определения по изображениям геометрических параметров (длин, площадей, периметров, объемов);

- выделения поверхностей отдельных структур;

- поворотов трехмерных изображений на требуемый угол;

- формирования и воспроизведения на экране дисплея трех произвольных взаимно перпендикулярных сечений и изометрической проекции;

- цветового кодирования элементов исследований;

- нанесения служебного текста и меток;

- запоминания и хранения информации на магнитных носителях;

- интерактивной работы с комплексом программ;

- распечатки на принтере УЗ-изображений и протоколов обследования.

Основными режимами работы ПДУ являются: синтез и визуализация УЗ-изображе-

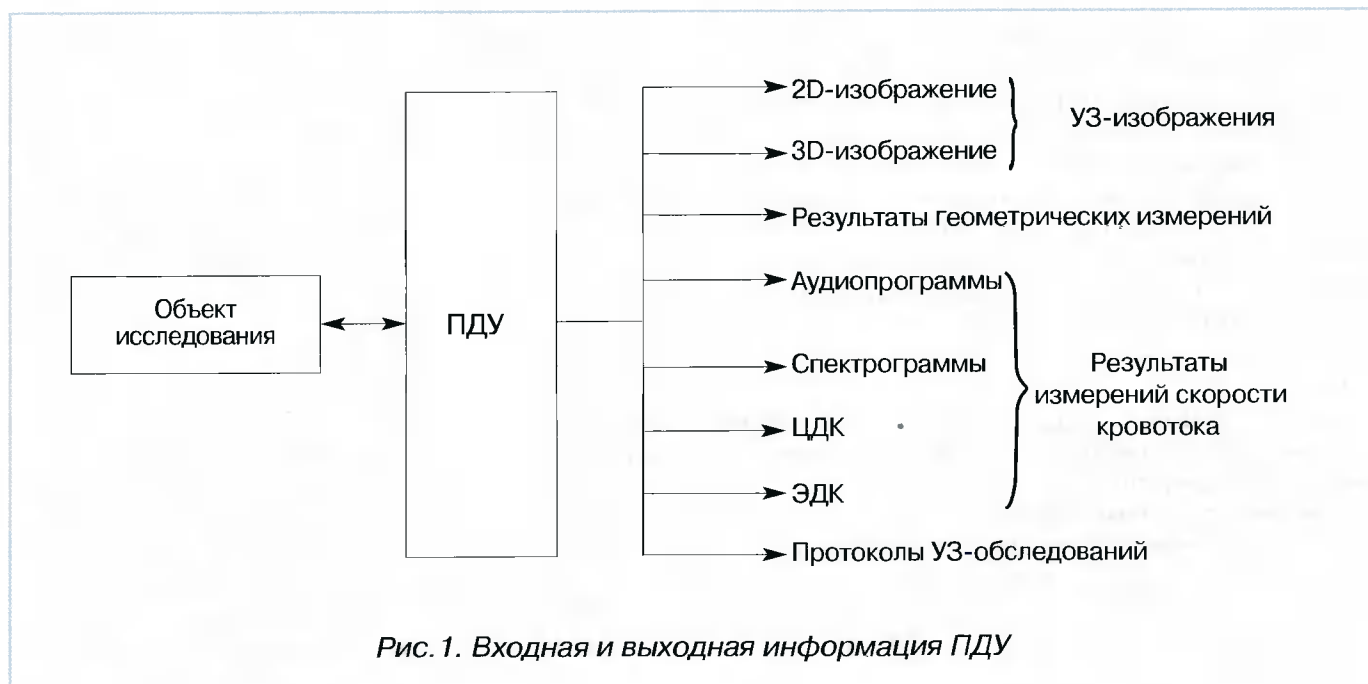


Рис. 1. Входная и выходная информация ПДУ



ний, измерение и визуализация скорости кровотока.

На рис. 1 показана входная и выходная информация ПДУ. Входной информацией являются эхо-сигналы, отраженные от внутренних неоднородностей исследуемой области, а выходной – 2D- и 3D-изображения, результаты геометрических измерений по “замороженным” изображениям, результаты измерения скорости кровотока и протоколы УЗ-обследований.

Для выполнения предъявленных к ПДУ технических требований по получению однородных по качеству 2D- и 3D-изображений с высоким разрешением во всей зоне обзора при минимальных габаритах прибора предлагается реализовать следующие основные технические решения, определяющие принцип действия и конструктивное выполнение прибора:

- использование линейных и конвексных решеток пьезоэлементов (ПЭ) с электронным сканированием луча для формирования изображения двумерных В-эхограмм;
- применение алгоритмов синтеза апертуры в целях минимизации количества приемопередающих каналов при сохранении высокого поперечного разрешения;
- использование короткого (0,5 – 1,0 мкс) зондирующего импульса для обеспечения разрешения по глубине;
- применение электромагнитного датчика положения для координированного набора 2D-срезов при формировании 3D-массивов;
- синтез 3D-изображения путем последовательной записи в память ПК-изображений двумерных В-эхограмм, полученных в реальном времени при свободном перемещении УЗ-датчика рукой, а не посредством электромеханического позиционера, как в предыдущих разработках.

На рис. 2 показана обобщенная структурная схема УЗ-диагностических устройств.

В состав всех УЗ-диагностических устройств входят:

- персональный компьютер (ПК);
- многоэлементные УЗ-датчики (с решеткой ПЭ);
- программно-управляемое приемопередающее устройство;
- плата с цифровым сигнальным процессором (ЦСП), используемая для синтеза 2D УЗ-изображений и управления аппаратными средствами ПДУ.

Эта же обобщенная структурная схема справедлива и для всех отладочных стендов и большого класса прочих устройств, разрабатываемых в НИИТП. Отличия могут быть в количестве плат с ЦСП и АЦП.

Приемопередающий модуль (ППМ) содержит фазирующие цепи, позволяющие запитывать ПЭ УЗ-датчика напряжениями с определенными фазами. Это позволяет формировать диаграммы направленности (ДН) УЗ-датчика требуемой формы. Электрические УЗ-колебания, вырабатываемые в ПДУ, преобразуются посредством ПЭ УЗ-датчика в акустические колебания, направляемые в объект исследования. После отражения от неоднородностей среды акустические колебания принимаются ПЭ, преобразуются в электрические колебания, усиливаются и обрабатываются. Затем синтезируются и визуализируются УЗ-изображения и/или измеряется скорость кровотока и визуализируется карта этих скоростей.

Принцип работы ПДУ в режиме синтеза изображений состоит в использовании методов синтеза апертуры. Подробно эта процедура для радиолокаторов СВЧ-диапазона рассматривается в работах [2, 3]. Эти методы и алгоритмы существенно отличаются от известных классических алгоритмов восстановления изображений по проекциям, базирующихся на преобразовании Радона и используемых в задачах компьютерной томографии [4, 5] и др.

Принимаемые отраженные сигналы преобразуются в цифровой код. Каждый отсчет отраженного сигнала соответствует определенной полоске дальности (ПД), которая представляет собой геометрическое место точек, находящихся на одинаковом расстоянии от УЗ-датчика. Один отсчет представляет собой суммарный сигнал от всей полоски дальности. Задача программного формирования (синтеза) УЗ-изображения состоит в том, чтобы для каждой полоски дальности выделить парциальные сигналы, соответствующие внутренним точкам этой полоски дальности. Это реализуется посредством процедуры синтеза изображения, одним из основных этапов которой является спектральный анализ сигналов от одноименных полосок дальности.

Принцип работы ПДУ в режимах измерения и визуализации скорости кровотока состоит в селекции движущихся целей, фильтрации шумов и помех (в частности, колебаний стенок сосудов) и измерении набега фазы (для фазового метода) между отраженными сигналами, возникающего за период времени, равный периоду повторения зондирующих импульсов. Этот набег фазы пропорционален радиальной составляющей скорости движения форменных элементов крови (в основном эритроцитов) в кровеносных сосудах (для неподвижных структур этот набег фазы равен нулю). Знак набега фазы определяет направление радиальной проекции скорости кровотока. На картах кровотока это направление кодируется цветом: красным к датчику, синим – от датчика.

Спектральный метод измерения скорости кровотока в заданной точке интересующей области состоит в цифровом спектральном анализе дискретной функции (последовательности отсчетов) эхо-сигналов, отраженных от этой точки. Для измерения скорости кровотока формируется узкий луч, поперечные размеры которого должны равняться величине поперечной разрешающей способности. При визуализации карты кровотока производится совмещение (наложение) этой карты на УЗ-изображение.

Автономный модуль ЦОС, реализуемый на одном сигнальном процессоре типа TMS320C6201, выполняет как функции управления работой аппаратных средств прибора, так и функции обработки информации, т.е. формирования изображения. При этом сначала производится съемка (регистрация) голограмм одного кадра, затем их обработка. Предполагается, что в течение регистрации кадра специализированное устройство DDC (Digital Down Converter), входящее в состав печатной платы с ЦСП, будет выполнять фильтрацию оцифрованных сигналов и формирование квадратур без участия ЦСП TMS320C6201.

Это существенно сократит нагрузку на ЦСП и облегчит программирование. После завершения обработки одного кадра информация передается в ОЗУ ПК для визуализации. Таким образом, частота кадров определяется суммой времен регистрации кадра и выполнения программы синтеза изображения в ЦСП. Перезапись информации в ОЗУ ПК при использовании интерфейса IEEE 1394 (FireWire) занимает незначительную часть времени кадра и не скажется на частоте кадров.

Данная циклограмма повторяется неограниченное количество раз, пока врач не задаст новый режим работы прибора с панели управления.

В настоящее время на АВТ отработан режим 2D с УЗ-датчиком, дающим зону обзора из 112 линий сканирования (56 мм) и 256 отсчетов в глубину (100 мм). Время регистрации кадра Трег составило 17 мс, суммарное время регистрации и обработки кадра составило 50 мс.

Аппаратные средства и программное обеспечение ПДУ

Укрупненная структурная схема ПДУ показана на *рис. 3*. В состав ПДУ входят: ПК со вставленной в него печатной платой с цифровым сигнальным процессором (ЦСП), программно-управляемое приемопередающее устройство и УЗ-датчик.

Рассмотрим состав и работу ПДУ по его структурной схеме (*рис. 3*).

В состав платы с ЦСП, в свою очередь, входят:

- собственно ЦСП типа TMS320C6201;
- цифроаналоговый преобразователь (ЦАП), вырабатывающий сигналы радиоимпульса (РИ) и временной автоматической регулировки усиления (ВАРУ);
- модуль АЦП, оцифровывающий отраженные эхо-сигналы.

В состав приемно-передающего устройства (ППУ) входят следующие блоки (*рис. 3*):

- мультиплексирования ПЭ;
- усилителей мощности зондирующих импульсов (ЗИ);
- формирования УЗ-луча;
- коммутации и попарного суммирования входных сигналов,
- выходных усилителей приемного канала, а также интерфейсный блок (плата), служащий в основном для создания гальванической развязки между ППУ, с одной стороны, и платой с ЦСП и ПК – с другой.

УЗ-датчик соединен с электромагнитным датчиком пространственных координат, который используется в режиме 3D.

Работа ПДУ происходит следующим образом. В соответствии с выбранным режи-

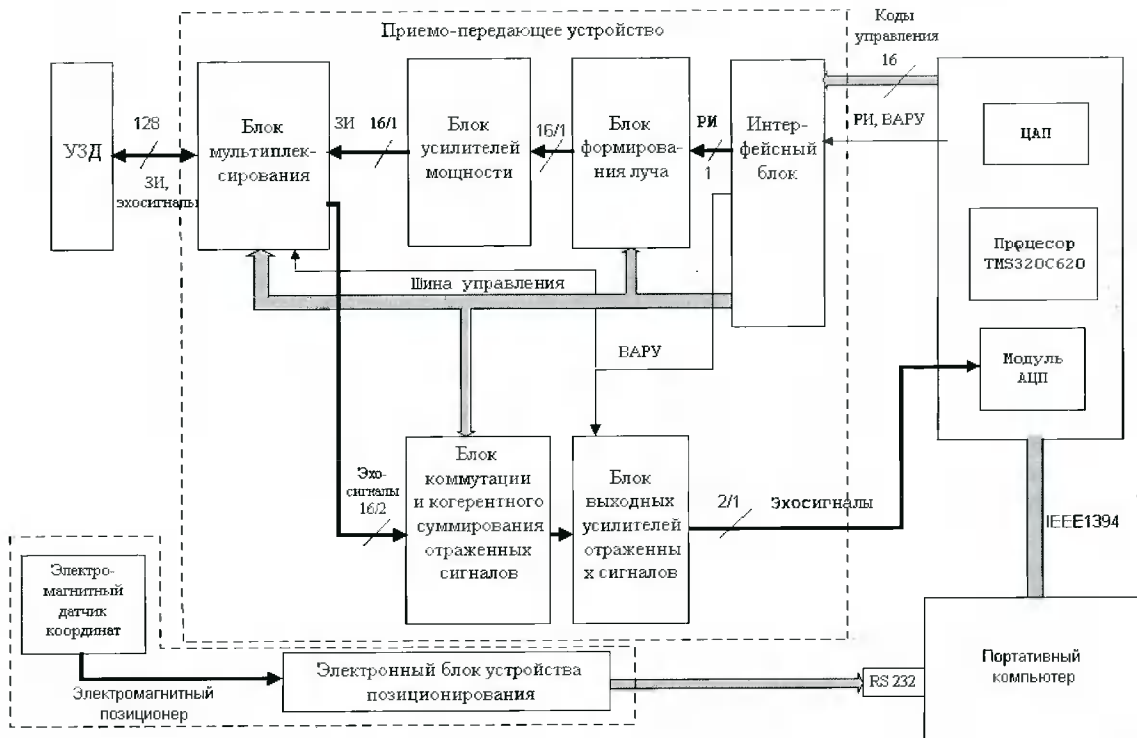


Рис.3. Увеличенная структурная схема ПДУ

мом и параметрами работы из ПК в ЦСП загружаются уставки. ЦСП вырабатывает коды управления аппаратными средствами, поступающими через интерфейсную плату на блоки ППУ.

В блоке формирования УЗ-луча формируются радиоимпульсы возбуждения ПЭ, они усиливаются усилителями мощности и через блок мультиплексирования поступают на ПЭ. Отраженные эхо-сигналы через блоки мультиплексирования ПЭ, коммутации и попарного суммирования поступают на блок выходных усилителей приемного канала и затем на плату с ЦСП, где они преобразуются в цифровой код и обрабатываются. ЦСП синтезирует двумерные изображения отдельных сканов, массивы которых переписываются в память ПК. В режиме 3D из отдельных сканов синтезируются 3D изображения.

Программное обеспечение (ПО) медицинских УЗ-локаторов с синтезированием апертуры – это большой комплекс программ, листинги которых содержат многие десятки тысяч строк программного кода.

Рассматриваемое ПО укрупненно подразделяется на три класса: прикладное, инструментальное и системное. Прикладное ПО, разрабатываемое в НИИ точных приборов специально для медицинских УЗ-локаторов с синтезированием апертуры, является резидентным (находится в составе диагностических устройств) и предназначено для:

- формирования зондирующего сигнала;
- управления аппаратной частью прибора, включающего в себя управление излучением, формированием зондирующего луча, регулировкой усиления приемно-усилительного тракта, выбором режима работы, электронным сканированием луча;
- первичной и вторичной обработки принимаемого прибором сигнала;
- визуализации двумерного ультразвукового изображения;
- проведения геометрических измерений исследуемых структур на их “замороженных” изображениях;
- статистического анализа параметров исследуемых структур;
- регистрации серии УЗ-срезов, предназначенных для формирования трехмерного массива, оперативного хранения в ОЗУ портативного ПК ПДУ зарегистрированных серий изображений и их просмотр;
- синтеза трехмерного массива УЗ-данных из серий УЗ-изображений;
- интерактивного выделения поверхностей (сегментации) исследуемых внутренних органов;
- трехмерной визуализации, обеспечивающей:
 - визуализацию сечений трехмерного массива;
 - представление трехмерного массива УЗ-данных в градациях серого (в том числе с

заданием порога прозрачности и с представлением структур в режиме полупрозрачности в зависимости от уровня их яркости);

- представление поверхностей выделенных (сегментированных) внутренних органов в псевдоцветах с яркостью, зависящей от глубины расположения объекта (для придания им ощущения объемности) и с возможностью совмещения представления трехмерного массива в градациях серого и сегментированных объектов в псевдоцветах;
 - повороты трехмерного массива и выделенных объектов на заданный пользователем угол, отображение сечений по взаимно перпендикулярным плоскостям и формирование выреза ближнего к наблюдателю угла из массива данных;
 - отображение сечений трехмерного массива произвольной плоскостью;
 - быстрое объемное отображение сегментированных объектов (быстрая 3D-визуализация)
- архивирования и документирования результатов исследований, заключающихся в проведении операций, обеспечивающих поддержание и ведение баз данных.

Инструментальное ПО служит для автоматизации разработки, в частности отладки и тестирования прикладных программ. Подразделяется на инструментальное ПО собственной разработки НИИ точных приборов и промышленное ПО, разрабатываемое специализированными фирмами, в основном зарубежными.

Системное ПО подразделяется на операционные системы, служащие для обеспечения функционирования вычислительных средств медицинских диагностических устройств, и системы программирования, предназначенные для разработки остального ПО.

Предлагаемые для построения ПДУ вычислительные средства (портативный ПК и ЦСП типа TMS320C6201) и системное программное обеспечение (ОС MS Windows) позволяют строить программы управления устройством и обработки информации на современном уровне. При этом реализуется удобный графический интерфейс с пользователем с возможностью многооконного представления результатов обработки ультразвуковых данных.

Разработка ПО проводилась с использованием так называемых виртуальных приборов и инструментов пакета LabVIEW фирмы National Instruments, представляющего собой язык графического программирования для создания программ в виде блок-схем. Этот пакет используется для создания ПО экранного интерфейса. Модули обработки написаны на языке Си, а

подпрограммы ресурсоемких алгоритмов обработки данных реализованы на языке ассемблера ЦСП (для увеличения скорости обработки).

Структурно ПО включает ядро на базе системных программ ПК и модули, обеспечивающие выполнение различных функций системы, а также модуль графического интерфейса с пользователем. Ядро управляет взаимодействием модулей, через него производится обмен информацией между модулями. Модуль графического интерфейса обеспечивает организацию диалогового взаимодействия (экранного интерфейса) пользователя, а модуль управления аппаратными средствами – взаимодействие ПО с техническими средствами прибора, вырабатывая управляющие сигналы для работы устройства коммутации каналов, сигналы кодов ВАРУ для входного усилителя и т.д.

Модуль синтеза и визуализации 2D-массива содержит в своем составе программные средства обработки данных в реальном времени и служит для формирования 2D-изображения на экране компьютера. В модуле обеспечивается проведение геометрических измерений на плоскости. Модуль синтеза 3D-массива и 3D-визуализации проводит расчеты координат элементов 3D-массивов УЗ-данных и последующей визуализации на экране монитора средствами графического интерфейса. В модуле обеспечивается проведение геометрических измерений в пространстве.

Для архивирования и документирования данных в систему включен модуль базы данных. Этот модуль обеспечивает выполнение минимально необходимого набора функций базы данных (запись, поиск, уничтожение, сортировка записей) для алфавитно-цифровой информации и архивацию УЗ-изображений.

Литература

1. Модуль ЦОС ADP6701E. Руководство пользователя. ОАО “Инструментальные системы”. М., 2000.
2. Радиолокационные станции с цифровым синтезированием апертуры антенны. /Под ред. Горяинова В.Т. М.: Радио и связь, 1988.
3. Радиолокационные станции обзора Земли. М.: Радио и связь, 1983.
4. Хермен Г. Восстановление изображений по проекциям. (Основы реконструктивной томографии). /Пер. с англ. М., 1983.
5. Наттерер Ф. Математические аспекты компьютерной томографии. /Пер. с англ. М.: Мир, 1990.



Ю.Н. Александров, А.А. Беляев, А.В. Глушков, Ю.Н. Грибов, А.В. Никольский,
Я.Я. Петричкович, Т.В. Солохина

Новая отечественная платформа СБИС “МУЛЬТИКОР” для высокоточной скоростной обработки информации и управления объектами

Рассматриваются аспекты новой открытой отечественной платформы для проектирования современных сверхбольших интегральных схем (СБИС) “МУЛЬТИКОР”. Платформа разработана в ГУП НПЦ ЭЛВИС и предназначена для проектирования широкого спектра СБИС для коммерческих, военных и космических применений, которые различаются по вычислительной мощности, стоимости, функциональным возможностям и области применения.

На базе СБИС супермикроконтроллеров (СМК) платформы “МУЛЬТИКОР” могут быть спроектированы радарные наземные и бортовые авиационные и космические комплексы, гидроакустические системы, системы связи и видеонаблюдения. Мультимедийные и коммуникационные СБИС “МУЛЬТИКОР” ориентированы на создание мультимедийных и коммуникационных систем нового поколения от аудиодекодеров и связанных процессоров с низкой степенью интеграции до высокопроизводительных мультимедийных или коммуникационных процессоров 2.5 и 3G типа.

1. Отличительные особенности новой отечественной платформы проектирования СБИС “МУЛЬТИКОР”

Основными отличительными особенностями платформы “МУЛЬТИКОР” по сравнению с прежними подходами к проектированию СБИС являются:

- ядерная технология проектирования СБИС (“See-of the-cores”, что означает “море ядер”) на базе набора ядер, разработанных на основе субмикронных библиотек отечественного изготовителя (ОАО “АНГСТРЕМ”);
- однокристалльная двухпроцессорная архитектура на базе процессорных ядер разработки НПЦ ЭЛВИС: стандартного RISC-ядра и оригинального ядра цифрового сигнального процессора – DSP-ядра (DSP – Digital Signal Processing);
- RISC-ядро, разработанное как функциональный аналог ИМС R3000 MIPS1 архитектуры, обеспечивает совместимость с программными средствами для RISC-процессоров линии “БАГЕТ”;
- программируемые DSP-ядра с фиксированной и плавающей точкой (обеспечивается аппаратная поддержка специальных приложений и масштабируемость в 16 раз);
- реализация параллельных и многопроцессорных наращиваемых систем с миллиардной производительностью и с оптимальными показателями по массогабаритам и энергопотреблению;
- развитое инструментальное программное обеспечение для эффективной разработки новых приложений на базе открытой платформы “МУЛЬТИКОР”.

2. Процессорные ядра платформы

Платформа “МУЛЬТИКОР” содержит на настоящий момент в базе САПР аналог MIPS1 RISC-ядро и два масштабируемых оригинальных DSP-ядра: с фиксированной точкой (“ЭЛКОР-1”) и со стандартной плавающей точкой (“ЭЛКОР-2”).

RISC-ядро по своим функциональным возможностям аналогично 32-разрядному микропроцессору MIPS R3000 и реализует систему команд MIPS-I (за исключением команд деления, процедура умножения поддерживается аппаратно).

DSP-ядра имеют стандартную для большинства современных цифровых сигнальных процессоров (ЦСП) гарвардскую программируемую архитектуру с внутренним параллелизмом по потокам обрабатываемых данных. Они предназначены для высокоскоростной обработки информации (сигналов, изображений, управляющей информации и т.д.) с производительностью, равной нескольким миллиардам операций в секунду для данных, представленных в формате с фиксированной точкой, и с производительностью, равной нескольким сотням миллионов операций с плавающей точкой, в случае особо точной обработки.

Оба ядра могут работать с различными типами форматов данных: как с фиксированной, так и с плавающей точкой. Так, ядро “ЭЛКОР-1” способно обрабатывать данные как в 8/16/32-разрядных форматах с фиксированной точкой (и блочным порядком), так и программным способом 64-разрядные данные и данные в формате 32E16 с плавающей точкой, причем с высокой производительностью, а ядро “ЭЛКОР-2” работает в стандартном формате IEEE754 с плавающей точкой и в 8/16/32-разрядных форматах с фиксированной точкой.

3. Масштабируемость архитектуры СБИС

DSP-ядра расширяют возможности RISC-ядра по обработке сигналов и изображений и поддерживают в качестве основной операции MAC-процедуру (Multiplying-Accumulation – умножение с накоплением), лежащую в основе сверточных и матричных вычислений. Для более эффективного использования внутренних операционных ресурсов DSP-ядра и поддержки обработки комплексных данных использован формат параллельной инструкции, а для достижения высокой производительности – внутреннее распараллеливание потоков обработки по SIMD-типу (Single Instruction, Multiple Data – один поток инструкций – множественные потоки данных).

Система команд любого DSP-ядра, являясь стандартной с точки зрения традиционных ЦСП, позволяет простое программирование всех базовых процедур для указанных приложений и не уступает по полноте традиционным ЦСП (к примеру, DSP56600 фирмы Motorola).

Основные особенности DSP-ядер с фиксированной и плавающей точкой

- Полная программируемость ядер с использованием 32- и 64-разрядных VLIW-подобных инструкций переменной длины; совмещение процедур обработки и пересылок;
- SIMD-механизм наращивания производительности путем распараллеливания арифметических трактов на уровне микроархитектуры в одном DSP-ядре;
- 12 вычислительных операций в течение цикла при частоте 100 МГц (для “ЭЛКОР-1” P=1200 млн.оп./с, 16-битные данные) и 50 МГц (для “ЭЛКОР-2” P=600 млн. оп./с для данных в стандартном формате с плавающей точкой IEEE-754 или MFlops) для четырехкратной SIMD-конфигурации ядер (4SIMD) и КМОП 0.52-мкм технологии реализации;
- простые механизмы для наращивания числа DSP-ядер до четырех в пределах одной СБИС с увеличением производительности свыше 7200 млн.оп./с (16 бит) для ядер типа “ЭЛКОР-1” и свыше 3600 MFlops в стандарте IEEE-754 для ядер типа “ЭЛКОР-2” для 4SIMD-конфигурации ядер и КМОП 0.35-мкм технологии реализации;
- расширенные возможности по динамическому диапазону – компромиссный выбор “точность/производительность”.

Использование ядерного подхода в проектировании СБИС на базе платформы “МУЛЬТИКОР”, а также DSP-ядер с масштабируемой архитектурой позволяет легко создавать спектр СБИС, отличающихся по вычислительной мощности, используемым интерфейсам, объему внутренней памяти, потребляемой мощности, стоимости и т.д.

Все СБИС на базе разработанной платформы можно разделить условно на три класса в соответствии с их вычислительной мощностью.

1. СБИС “мини-архитектуры” (рис. 1), характеризующиеся минимальной вычислительной мощностью (сотни миллионов операций) и содержащие в своем составе RISC-ядро и SISD DSP-ядро. Используемые интерфейсы и размер внутренней памяти формируются по требованию заказчика. Как пример “мини-архитектуры” может быть названа СБИС программируемого аудиodeкодера нового поколения, “мягкое ядро” которого разработано НПЦ ЭЛВИС в 2000 г.

На базе “мини-архитектуры” при обеспечении соответствующих интерфейсов возможно создание коммуникационного процессора нового поколения, имеющего минимальные стоимостные и мощностные характеристики. Такой процессор не только не будет уступать, к примеру, СБИС TMS320C5x фирмы TI, но и превосходить его по функциональным возможностям.

2. СБИС “миди-архитектуры”. Они характеризуются средней вычислительной мощностью (единицы миллиардов операций) и содержат в составе RISC-ядро и 4 SIMD DSP-ядро с фиксированной (“ЭЛКОР-1”) или с плавающей (“ЭЛКОР-2”) точкой. Используемые интерфейсы и размер внутренней памяти формируются по требованию заказчика.

В качестве примера “миди-архитектуры” могут быть названы СБИС супермикроконтроллеров (СМК) с фиксированной “Мультикор_01” и с плавающей “Мультикор_02” точкой (рис. 2), разрабатываемые по специальному заказу. СБИС супермикроконтроллера “Мультикор_01” планируется к запуску на ОАО “Ангстрем” в конце 2001 г.

Используются интерфейсные ядра 64-разрядного порта памяти, стандартного PCI-порта (PCI Local Bus Specification Rev.2.2), ядра тетрадных параллельно-последовательных портов обменов (линки), совместимых с SHARC ЦПС

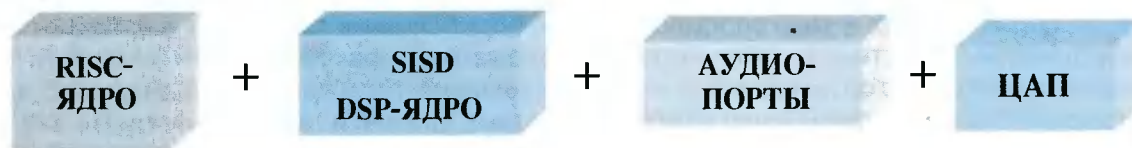


Рис. 1. Пример “мини-архитектуры” платформы “МУЛЬТИКОР”. СБИС аудиodeкодера. ЦАП-ядро цифроаналогового преобразователя (к примеру, «ТАКТ-AV011» разработки фирмы “АНГСТРЕМ-СБИС”)

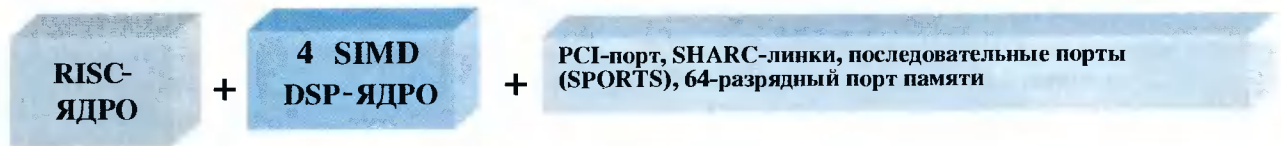


Рис.2. Пример “миди-архитектуры” платформы “МУЛЬТИКОР”. СБИС супермикроконтроллеров “Мультикор_01/02”

фирмы ADI ADSP21060, а также ядра последовательных портов (SPORTS).

Другим примером “миди-архитектуры” платформы “МУЛЬТИКОР” является перспективная СБИС мультимедийного (коммуникационного) процессора “Мультикор_01М” (рис. 3).

Совместно с RISC- и DSP-ядрами использованы ядра мультимедийных интерфейсов, обеспечивающие связь с аудио/видеопортами СБИС (аудиокодеком и видеочамерой), специальными аналоговыми радиотрактами (для беспроводных персональных интерфейсов типа Bluetooth или для мобильной беспроводной связи и мобильного Интернета – Air Interface), а также интерфейсные ядра для связи с ПЭВМ (инфракрасный порт IrDA). Интерфейсное ядро порта MMC(MultiMedia Card Controller)/SD (Secure Digital) процессора позволяет подключать к СБИС MC_01М стандартные FLASH-карты и другую периферию для хранения больших объемов мультимедийных данных, таких, как аудио, цифровое видео, карты и цифровые фотоснимки.

СБИС “Мультикор_01М” поддерживает все необходимые интерфейсы и режимы функционирования для обеспечения самых современных приложений в области беспроводной связи 2.5 и 3G типа. Она обслуживает такие приложения, как мобильный Интернет, потоковое аудио/видео (MPEG4, видеоконференции), плееры (MP3), игры 2D/3D, GPS-навигация, защитные биометрические и идентификационные средства, охранные системы и системы защиты информации, бизнес-приложения, и т.д.

3. СБИС “макси-архитектуры”. К ним могут быть отнесены перспективные супермикроконтроллеры, условно обозначенные ниже, как “Мультикор_03/04” с фиксированной и/или плавающей точкой. Они характеризуются вычислительной мощностью, сопоставимой с супер-ЭВМ 80 – 90-х гг. которая превышает уровень производительности,

равный 10 млрд. оп./с с фиксированной точкой или нескольким млрд. оп./с с плавающей точкой.

Следует отметить, что СБИС “Мультикор_03/04” отличаются от супермикроконтроллеров MC_01/02, во-первых, использованием в своем составе не одного, а до четырех DSP-ядер, включенных по MIMD-схеме, каждое из которых имеет 4SIMD-расширение. Во-вторых, наличием внешних портов со значительно более высокой пропускной способностью, а в-третьих, большей внутренней памятью данных и программ, объем которой приближается к максимальным размерам, определенным архитектурой. Реализация таких СБИС возможна на технологии 0.25 – 0.35-мкм, что придает этим супермикроконтроллерам статус перспективных.

Для поддержки разработок планируется поставка базового инструментального ПО (Multicore Composer Studio – MCS), включая ассемблерные средства, Си-компиляторы, прикладные пакеты, а также программные симуляторы и другие аппаратно-программные инструментальные средства подготовки программ и их отладки (на базе XIL-, INX-прототипов, JTAG-отладчиков и др.).

Потребительские свойства DSP-ядра “ЭЛКОР-1” (4SIMD)

- ❑ Высокая производительность (1200 млн. оп./с, 16 бит/ 3200 млн. оп./с, 8 бит/ 150 MFlops (программная плавающая точка 32E16)/240 MFlops (программная плавающая точка 16E16);
- ❑ обработка данных в 8/16/32-разрядных форматах с фиксированной точкой, аппаратная блочная плавающая точка, программная плавающая точка –32E16 и 16E16;
- ❑ полностью параллельная процедура “умножение – сложение – сдвиг”;

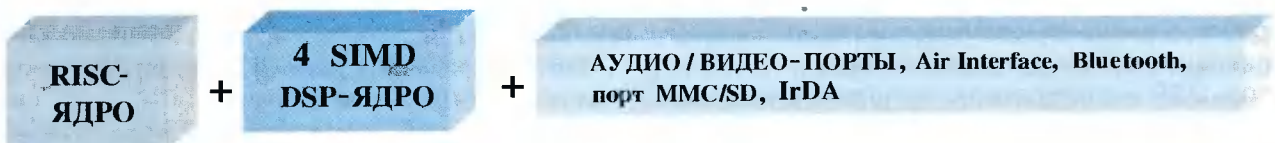


Рис.3. Пример “миди-архитектуры” платформы “МУЛЬТИКОР”. СБИС мультимедийного (коммуникационного) процессора “Мультикор_01М” (MC_01М)

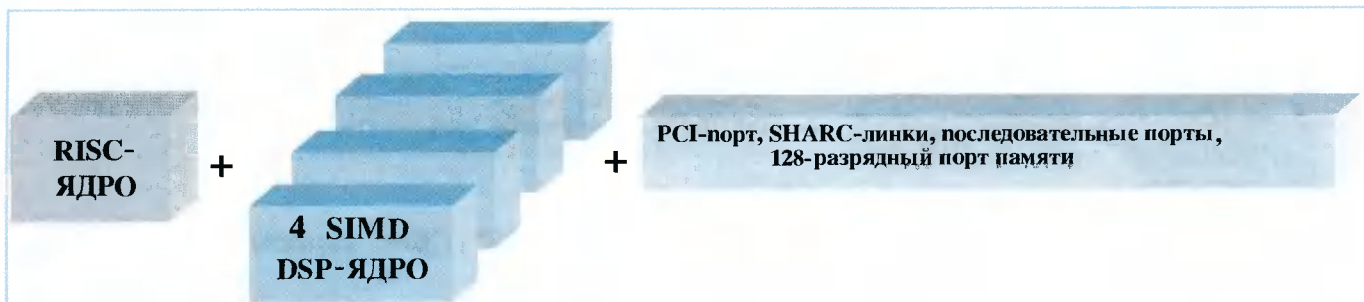


Рис.4. Пример “макси-архитектуры” платформы “МУЛЬТИКОР”: СБИС супермикроконтроллеров с числом RISC и/или DSP-ядер более одного

- специальная аппаратная поддержка мультимедийных и связанных приложений на уровне системы инструкций (к примеру, декодера Витерби);
- уникальные режимы ЦСП-адресации; аппаратная поддержка циклов (Nested hardware DO loops); аппаратный системный стек и стек циклов;
- память программ – до 16 К 64-разрядных слов; память данных – до 8 Мбит; ПЗУ для констант, объемом не менее 256w*32b (слов по 32 бита);
- низкая, управляемая программно, мощность потребления (сотни мВт) при напряжении питания 3.3/5, особенно для DSP-ядра SISD-типа;
- возможность подключения к диагностическому и отладочному порту.

4. Супермикроконтроллеры “Мультикор_01/02” на базе платформы “МУЛЬТИКОР” для скоростной высокоточной обработки информации и управления объектами

Супермикроконтроллеры (СМК) с фиксированной точкой (“Мультикор_01” или MC_01) и фиксированной/плавающей точкой (“Мультикор_02” или MC_02) реализуются как однокристальные двухпроцессорные микроконтроллеры на базе RISC-ядра и ядра DSP-акселератора с фиксированной и/или плавающей точкой.

32-разрядное RISC-ядро имеет архитектуру, функционально совместимую с MIPS R3000, который дополнен КЭШ инструкций и памятью данных.

RISC-ядро выполняет основной алгоритм обработки данных и обеспечивает:

- взаимодействие с системной шиной;
- настройку DSP-ядра, загрузку в него программ и данных;
- настройку контроллера прямого доступа (DMA);
- обработку внешних и внутренних прерываний;
- управление всеми потоками данных.

Ядро DSP-акселератора выполняет высокоскоростную обработку информации (сигналов, изображений, управляющей информации и т.д.) с производительностью, равной нескольким миллиардам операций в секунду для данных, представленных в формате с фиксированной точкой

(8/16/32 разряда) – ядро “ЭЛКОР_01”, и с производительностью, равной нескольким сотням миллионов операций с плавающей точкой, в случае особо точной обработки – ядро “ЭЛКОР_02”. Архитектура ядра масштабируется в 16 раз от варианта «мини» (SISD), «миди» (4SIMD) до конфигурации «макси» 4 SIMD-ядра в составе СБИС по 4SIMD каждое.

Основные технические параметры супермикроконтроллеров MC_01/02

Характеристики представлены для реализации СБИС с четвертью от максимальной конфигурации архитектуры (т.е. с одним из четырех возможных в максимальной конфигурации DSP-ядер 4 SIMD архитектуры). Архитектура СМК масштабируется в 16 раз при переходе от модификаций СБИС мини-архитектуры к модификациям макси-архитектуры.

Технология проектирования – заказная КМОП СБИС на базе параметризуемых библиотек ОАО «АНГСТРЕМ», 0.52-мкм (технология уточняется на тестовых кристаллах).

Частота синхронизации – входная – 33 МГц, рабочие – 33/66/132/100/50 МГц.

Пиковая производительность, млн. оп./с с фиксированной точкой – МОП/с или млн. оп./с с плавающей точкой – Mflops (при номинальных параметрах температуры и напряжения), рабочая частота MC_01/02 – 100/50 МГц.

Для СБИС “МУЛЬТИКОР_01”:

3200/1600 МОП/с в 8-разрядном комплексном формате;

1200/600 МОП/с в 16-разрядном комплексном формате;

200/100 МОП/с в 32-разрядном действительном формате;

не менее 150/75 Mflops (программно, формат 32E16);

100/50 МОП/с для RISC-ядра в 32-разрядном действительном формате.

Для СБИС “МУЛЬТИКОР_02”:

Для 8/16-разрядных форматов с фиксированной точкой (соответствуют параметрам производительности для СБИС “МУЛЬТИКОР_01” – см. выше);



1200/600 МОП/с в 32-разрядном действительном формате;

600/300 Mflops – в формате плавающей точки 24Е8.

Объем внутреннего ОЗУ скалярных данных RISC-ядра (уточняется в процессе разработки топологии СБИС) – не менее 256 Кбит для серийных образцов, (для первых тестовых образцов может быть меньше);

Суммарный объем внутреннего ОЗУ векторных/скалярных данных акселератора (уточняется в процессе разработки топологии СБИС) – не менее 1,5 Мбит для серийных образцов; (для первых тестовых образцов может быть меньше).

Внешняя память (на порте памяти) – до 16GB;

Внешние интерфейсы (порты):

системный – PCI-совместимый;

линки – от 4 до 6 SHARC-совместимых байтовых портов, для тестового образца – тетрадные;

последовательные порты – 2 (SHARC-совместимые);

порт памяти SDRAM/SRAM/DRAM/FlashROM, 64-разрядный.

Пропускная способность портов при номинальных параметрах температуры и напряжения, МГц, не ниже:

порт памяти	100/66
PCI-порт	33
линки (SHARC)	33
последовательные порты (SHARC)	33

JTAG-отладочный порт, совместимый с IEEE 1149.1 (IEEE Standard Test Access Port and Boundary-Scan Architecture).

DMA – 14 каналов, возможности взаимной синхронизации с обоими процессорными ядрами.

Система инструкций – для RISC-ядра и СБИС в целом – MIPS-I (за исключением деления); ядро акселератора имеет систему инструкций традиционного ЦСП, кроме того, выполнена поддержка фильтровых процедур, коммуникационных и мультимедийных функций, используются эффективные параллельные VLIW-подобные инструкции.

Потребляемая мощность (3.3V) предварительно, Вт: ~1.6 для MC_01, ~2.0 для MC_02.

Кроме того, обеспечены режимы пониженного энергопотребления.

Напряжение питания – 2.5, 3.3, 5 В.

Корпус – от 240 до 304 выводов (уточняется)

Условия эксплуатации – от -40 до +125 °С.

Программное обеспечение. На базе разработанного RISC-ядра MIPS1 архитектуры для СБИС СМК под WINDOWS и LINUX создается ПО, которое включает интегрированную среду программирования, несколько аппаратно-программных отладчиков, прикладные библиотеки, предполагается создание операционной системы реального времени (OS PV).

СБИС “Мультикор_01” размещается в 240 – 304-выводном корпусе (в зависимости от конфигурации интерфейсов). Тип корпуса СБИС будет определен в ближайшее время. Рассматривается возможность размещения первой реализации СБИС в 304-контактном корпусе типа RQFP, либо в перспективном корпусе отечественного производства. Назначение выводов СБИС MC_01 приведено в табл. 1.

На рис. 5 представлена структурная схема СБИС “Мультикор_01/02”. В состав СБИС СМК “МУЛЬТИКОР_01/02” входят следующие основные узлы и компоненты:

RISC_ЯДРО RISC-ядро, функциональный аналог MIPS-1 (R3000) архитектуры;

DSP_ЯДРО ядро ЦСП-акселератора с фиксированной точкой (“ЭЛКОР-1”) для MC_01 или фиксированной/плавающей для СМК MC_02 точкой (“ЭЛКОР-2”)*;

КГДП контроллер прямого доступа в память;

ОЗУ_R память RISC_ядра (не менее 256 Кбит для серийных образцов, для первых тестовых образцов может быть меньше);

КЭШ кэш программ RISC-ядра, не менее 512w*32b;

СП системный порт (PCI-совместимый, стандарт PCI Local Bus Specification Rev.2.2);

ПВП 64-разрядный порт внешней памяти (поддержка нескольких типов памяти, в т.ч. SDRAM);

ЛП четыре SHARC-совместимых линки (ADSP21060, ф.ADI) для параллельных систем обработки на базе СМК, предварительно тетрадные; могут использоваться как поле программируемых флагов ввода/вывода;

ПП два порта обмена последовательным кодом (SHARC-совместимые, ADSP21060, ф.ADI);

КПР контроллер прерывания; обслуживает 6 входов прерывания: 2 – внешние, 1 – контроллер DMA (14-канальный), одно – DSP-ядро, 1 – таймер, 1 – PCI-совместимый системный порт. Внешние прерывания синхронизируются по внутренней частоте СМК;

ТАЙМЕР 32-разрядный таймер;

JTAG JTAG-порт для доступа к встроенным средствам отладки программ, контроллер стандартного диагностического JTAG-порта;

АГ адресный генератор;

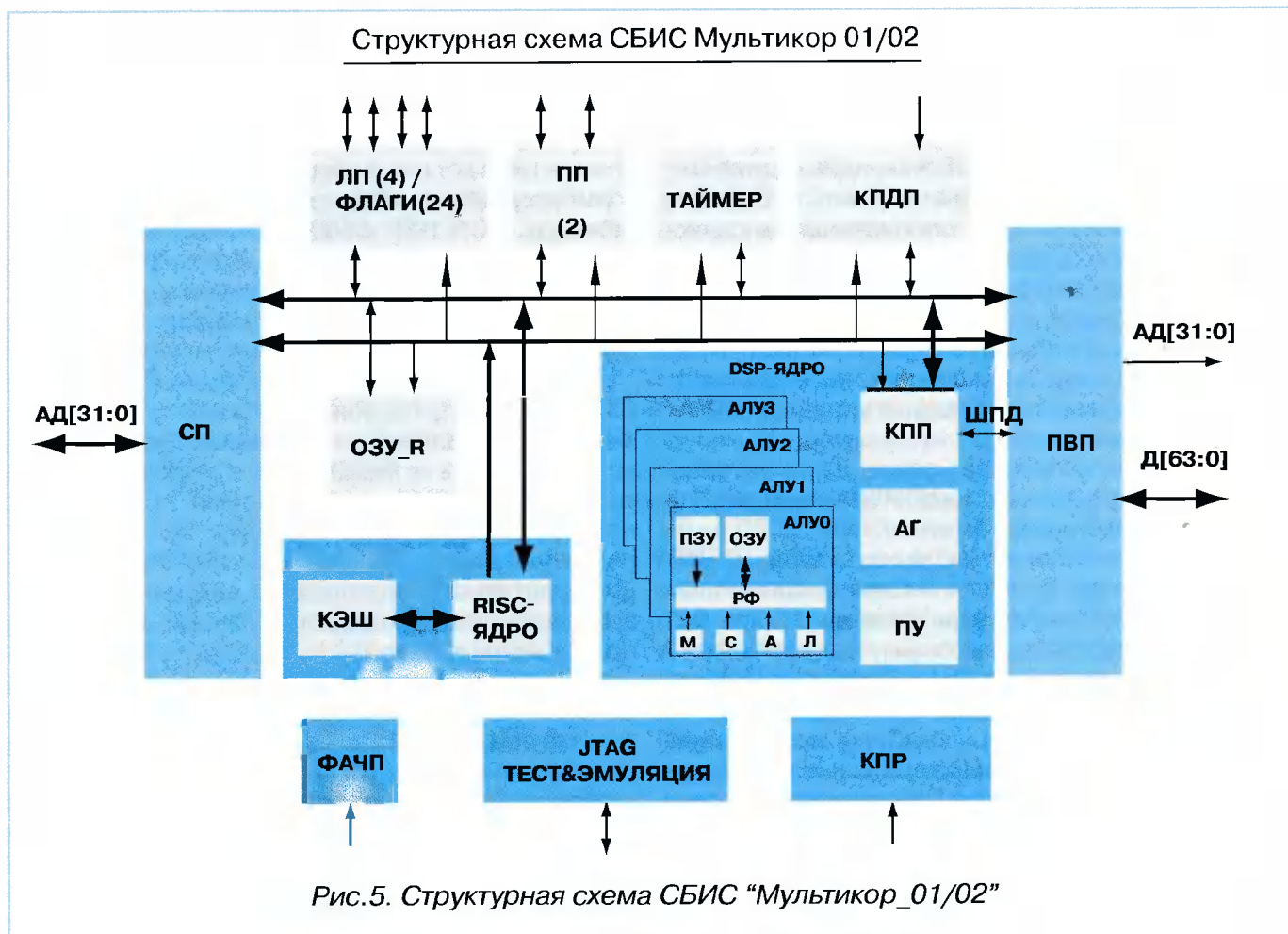
КШ коммутатор шин;

ПУ программное управление;

A(31:0) внешняя шина адреса MC_01(порт памяти);

Таблица 1. Назначение выводов ИМС "Мультикор_01" (см. рис.5)

Сигнал	Количество	Тип	Назначение
Системный PCI-порт			
AD[31:0]	32	IO	Адрес/Данные
nC/BE[3:0]	4	IO	Команда/ выбор байта
nFRAME	1	IO	Признак выполнения операции передачи данных
nIRDY	1	IO	Готовность задатчика
nTRDY	1	IO	Готовность исполнителя
nSTOP	1	IO	Признак остановки передачи данных
PAR	1	IO	Дополнение до четности количества единиц на шинах AD и nC/BE
nDEVSEL	1	O	Подтверждения выборки
IDSEL	1	I	Выборка при доступе к конфигурационным регистрам
nREQ	1	O	Запрос захвата шины
nGNT	1	I	Разрешение захвата шины
CLKIN	1	I	Тактовая частота работы шины PCI. Входная частота умножителя частоты
nRST	1	I	Сигнал установки исходного состояния
Порт памяти (ПВП)			
A[31:0]	32	O	Адрес внешней памяти. Младшие 2 разряда используются, если 6 банк памяти является 8-разрядным
D[63:0]	64	IO	Данные внешней памяти
A12_0, A12_1	2	O	12 разряды внешней синхронной памяти SDRAM
nWR[1:0]	2	O	Запись асинхронной памяти
nRD[1:0]	2	O	Чтение асинхронной памяти
nACK	1	I	Готовность асинхронной памяти
BE0[3:0]	4	O	Разрешения выборки байтов асинхронной памяти
nCS[4:0]	5	O	Разрешение выборки банков асинхронной памяти
nRAS0[1:0] nRAS1[1:0]	4	O	Строб адреса строки SDRAM
nCAS0[1:0] nCAS1[1:0]	4	O	Строб адреса колонки SDRAM
nWE0[1:0] nWE1[1:0]	4	O	Разрешение записи SDRAM
DQM0[3:0] DQM1[3:0]	8	O	Разрешение выборки байтов SDRAM
SCLK0[1:0] SCLK1[1:0]	4	O	Тактовая частота работы SDRAM
nFLYBY	1	O	Признак режима Flyby
nOE	1	O	Разрешение перевода выходных буферов внешнего устройства в режим Flyby (передача из внешнего устройства во внешнюю память)
nDMAR(4:0)	5	I	Запросы DMA
nIRQ(3:0)	4	I	Запросы прерывания
CLK_SEL(1:0)	2	I	Выбор коэффициента умножения частоты CLKIN: "0" – 1; "1" – 2; "2" – 3; "3" – 4
BYTE	1	I	Разрядность шины данных 6 банка внешней памяти: 0 – 32 разряда; 1 – 8 разрядов
JTAG-отладочный порт			
TCK	1	I	Тестовый тактовый сигнал (JTAG)
TRST	1	I	Установка исходного состояния (JTAG)
TMS	1	I	Выбор режима теста (JTAG)
TDI	1	I	Вход данных теста (JTAG)
TDO	1	O	Выход данных теста (JTAG)
Порты обмена последовательным кодом (ПП, два)			
DT	1	O	Передаваемые данные
DR	1	I	Принимаемые данные
TCLK	1	IO	Частота передачи
RCLK	1	IO	Частота приема
TFS	1	IO	Синхронизация передачи
RFS	1	IO	Синхронизация приема
Линковые порты (ЛП, четыре)			
LDAT	4	IO	Шина данных
LCLK	1	IO	Синхронизация
LACK	1	IO	Подтверждение
Электропитание			
VCC	32	I	Электропитание логики
VCC PLL	1	I	Электропитание PLL
GND	32	I	Земля



Д(64:0) внешняя шина данных МС_01(порт памяти);
 АД(31:0) шина адреса и данных системной PCI-совместимой шины;
 ФАПЧ узел фазовой автоподстройки частоты.

*DSP-ядро имеет внутреннюю микро-архитектуру – 4SIMD (4 ALU) на основе регистрового файла $32w \times 16b / 16w \times 32b$, арифметические блоки умножения (М)/сдвига(С)/арифметики (А)/логических(Л) операций, блок управления программой ПУ, ПЗУ-констант и память данных ОЗУ; суммарный объем внутреннего ОЗУ векторных/скалярных данных акселератора – максимально 8 Мбит, но не менее 1 Мбит для серийных образцов (четыре двухпортовых банка внутренней памяти, обеспечивающей параллельный ввод данных и обработку их в DSP-ядре);

Каждое из двух процессорных ядер (RISC и DSP) функционирует под управлением своей программы, хранящейся в собственной памяти программ. МК имеют удобную и очень эффективную RISC-ориен-

тированную систему инструкций, как для управляющего ядра, так и для ядра акселератора.

Включение обеих процессорных ядер в интегральную систему не является равноправным с точки зрения их возможностей в системе.

RISC-ядро осуществляет:

- выполнение основного алгоритма работы МК;
- взаимодействие с системной шиной;
- настройку DSP-ядра, загрузку в него программ и данных;
- настройку контроллера DMA;
- обработку внешних и внутренних прерываний;
- управление всеми потоками данных.

Ядро акселератора МК (DSP-ядро) реализует следующие функции:

- выполняет процедуры цифровой обработки информации (сигналов, изображений и т.д.);
- формирует признаки выполнения команд;
- формирует прерывание в CPU.

RISC-ядро взаимодействует с DSP-ядром посредством команд LOAD/STORE и прерываний.

DSP-ядро поддерживает полностью параллельную процедуру умножения/сложения/сдвига, лежащую в основе процедуры фильтрации и базовой операции преобразования Фурье, а также имеет:

- набор инструкций, совмещающий процедуры обработки и пересылки данных;

- режимы адресации, оптимизированные для задач ЦОС (цифровой обработки сигналов);
- аппаратную поддержку вложенных циклов;
- аппаратный стек;
- форматы данных (для ядра “ЭЛКОР-1”): 32- и 16-разрядные целые; 16-разрядные дробные комплексные; поддержка обработки байтных форматов с помощью специальных инструкций; поддержка программно форматов плавающей точки 32E16 (~150MOPS) и 64-разрядных форматов;
- параллельный множитель-аккумулятор 16x16. RISC-ядро выполняет контроль над всеми процессами обработки в DSP-ядре и управляет DMA-блоком.

DMA-блок обеспечивает обмен данными между всеми основными узлами СМК, а также между внутренней и внешней памятью.

Важной особенностью архитектуры СБИС MC_01 в отличие от большинства современных процессоров сигналов является возможность выполнения быстрого управления потоками информации с помощью канала DMA. Помимо стандартного подхода к синхронизации потоков информации (основанного, к примеру, на прерывании RISC-ядра от DMA или DSP-ядра) взаимная самосинхронизация DSP-ядра и DMA здесь осуществляется при минимальном участии RISC-ядра, что позволяет значительно увеличить производительность обработки и ввода-вывода, в частности, вложенных массивов данных. Этот механизм дает возможность для большинства задач, особенно обработки изображений, приблизить реальную производительность к пиковой, т.к. ввод/вывод информации будет выполняться преимущественно параллельно с ее обработкой, не ухудшая параметры реального времени.

DSP-ядро имеет четырехкратно распараллеленные тракты обработки данных и может работать в скалярном или SIMD-режиме. DSP-ядро использует программную модель с распараллеливанием арифметических трактов SIMD-типа. Это означает, что, учитывая массивный характер области обработки сигналов, для организации одинаковой обработки над четырьмя разными потоками данных DSP содержит четыре одинаковые вычислительные секции, состоящие из арифметико-логического устройства, регистрового файла и памяти данных.

В остальном DSP-ядро имеет традиционную RISC-ориентированную однопоточную для большинства инструкций архитектуру, за исключением некоторых особенностей, которые делают его очень удобным для использования в ЦСП-приложениях:

- экономичный переменный формат данных и команд (в том числе 64-разрядные VLIW-подобные инструкции);
- специальные возможности работы с битовыми полями, аппаратно-поддержанные циклы, эффективно реализованные стеки и т.д.

Разработанное масштабируемое DSP-ядро в составе СБИС СМК MC_01 поддерживает обработку 8-/16-/32-разрядных форматов данных с фиксированной точкой, а также обеспечивает программным способом обработку в 64-разрядном и в плавающем форматах. DSP-ядро не только не уступает по полноте системы инструкций, к примеру, ЦСП DSP 56600 фирмы Motorola или СБИС серии 218x фирмы ADI (из класса 16-разрядных процессоров с фиксированной точкой), но имеет значительно более широкие возможности по программированию.

Операционные блоки версии DSP-ядра для СМК Мультикор_02 дополнительно поддерживают обработку данных с плавающей точкой в стандартном формате IEEE754 с пиковой производительностью 600Mflops. Выполнена эффективная поддержка сверточных процедур, FFT, DCT, DWT, ряда коммуникационных и мультимедийных функций; доступно использование эффективных для арифметического устройства параллельных VLIW-подобных инструкций.

5. Программное обеспечение СБИС платформы “МУЛЬТИКОР”

Для разработки эффективного программного обеспечения требуется использование всех возможностей целевой аппаратуры. Оценки показывают, что примерно 80% усилий разработчиков и 80% сложности встроенных систем реального времени падает на программное обеспечение.

Параллельно, не дожидаясь появления первого кристалла новой платформы супермикроконтроллера MC_01 MC и учитывая важность и приоритетность создания программного обеспечения для СБИС, архитектура которой содержит два программируемых процессорных ядра, ГУП НПЦ ЭЛВИС разрабатывает в этом году первую версию базового программного обеспечения платформы “МУЛЬТИКОР”, которая включает инструментальное ПО для написания и отладки программ для обоих ядер и в целом СБИС платформы “МУЛЬТИКОР” (МультиКОР_СТУДИЯ или сокращенно MCS);

МультиКОР_СТУДИЯ (MultiCore Studio – MCS) – это интегрированная среда, предоставляющая пользователю доступ ко всем инструментам из одного интерфейса и обеспечивающая полный цикл разработки и отладки программ для платформы “МУЛЬТИКОР”.

MCS включает в себя среду разработки и среду отладки.

Среда разработки

Среда разработки состоит из менеджера проекта, редактора, компиляторов и утилит подготовки кода.

Менеджер проекта обеспечивает управление файлами проекта. Структура проекта может быть представлена в окне проекта в виде дерева, в ко-

тором файлы размещаются по функциональным категориям, например, файлы программ RISC-ядра на Си и на ассемблере, заголовочные файлы, файлы программ DSP-ядра. Менеджер позволяет создавать проект, добавлять и удалять файлы из проекта.

Синтаксически зависимый Редактор используется для написания программ на Си и на ассемблере для RISC-ядра и DSP-ядра. Для выделения синтаксических элементов (ключевые слова, зарезервированные слова, комментарии и т.п.) используется выделение цветом, что уменьшает количество ошибок и облегчает их поиск. Имеются операции поиска и замены.

В среде предоставляется доступ к компиляторам RISC-ядра (C, ассемблер) и DSP-ядра (ассемблер).

Среда отладки

Включает символьный отладчик, позволяющий проводить отладку программы в тексте языка Си, и двоичный отладчик, позволяющий проводить отладку программы на ассемблере.

Отладчик позволяет устанавливать точки останова по тексту программы, по обращениям к переменным или регистрам. Точки останова могут быть постоянными или разовыми. Используются возможности исполнения в пошаговом режиме, с заходом в функцию, с пропуском функции, с выходом из функции. Обеспечивается наблюдение в процессе выполнения программы за значением переменных, регистров, ячеек памяти. Отладка может выполняться в интерактивном режиме и по командным файлам.

Кроме того, в среде отладки имеется Интерактивный профилировщик, который используется для оценки времени выполнения кода во время

разработки. Оцениваться может какой-либо участок кода по времени использования процессора или некоторые другие события, такие, как число обращений к подпрограммам, число переходов, число прерываний, и т.д.

Интегрированная среда позволяет подключать дополнительные инструменты для визуализации данных в отдельных окнах во время выполнения программы. Такие инструменты представляют информацию в естественном формате для удобства интерпретации анализа и проверки. Это позволяет "видеть" процесс преобразования информации в процессе отладки программы.

Отладочные средства разделяются на две группы – программные и программно-аппаратные.

Программная среда отладки в MCS (Симулятор СБИС Мультикор_01) разрабатывается для LINUX/sunOS и для MSWindows, включая полную интерпретацию всей системы команд RISC- и DSP-ядер, а также интерфейсов СБИС. Она строится на базе бинарных отладчиков и дополняется символьным отладчиком программ на языке Си для RISC-ядра. Наличие данного продукта позволяет потенциальным заказчикам быстро знакомиться с возможностями аппаратуры и проводить отладку программного проекта на этапе рабочего проектирования.

Пример окон MCS (MSWindows) представлен на рис.6 и иллюстрирует возможности параллельной отладки программ как для RISC-ядра, так и для DSP-ядра.

Программно-аппаратные отладчики реализуются в разных вариантах и предназначены для ускорения процесса отладки в условиях реального исполнения программы. Схема их реализации практически тождественна: символьный отладчик

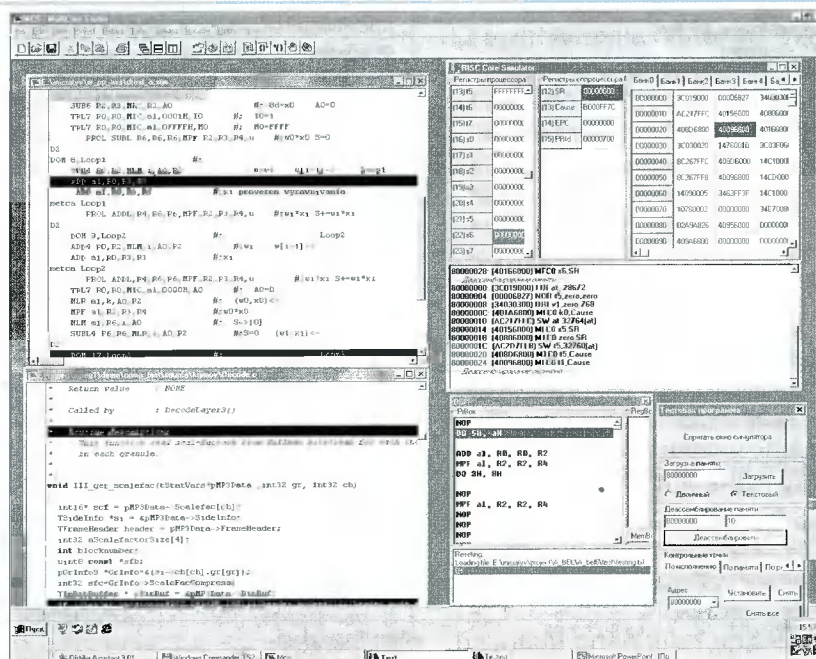


Рис.6. Пример окон в MCS для СБИС платформы МУЛЬТИКОР

располагается на хосте (к примеру, ПЭВМ), который соединен с прототипом СБИС через какой-либо канал. С другой стороны, на прототипе размещается отладочный агент – программа или аппаратура, реализующая команды отладчика. Такой программно-аппаратный отладчик СБИС MC_01 предполагается реализовать к началу 2002 г. на ее FPGA-прототипе серии Virtex.

Среда исполнения прикладных программ

Системы реального времени требуют поддержки обязательного набора сервисов для обеспечения планирования выполнения задач, управления ресурсами (памятью), управления вводом/выводом и сбора информации о выполняемых задачах. Как правило, эти возможности обеспечиваются ядром операционной системы или на более низком уровне.

Кроме того, для отладки систем реального времени в среде используются дополнительные средства, обеспечивающие анализ выполнения программ реального времени без использования точек останова. Для этого в ОСРВ предусматривается режим, при котором осуществляется сбор целевой информации о ходе выполнения задач или результаты вычислений. Для этих целей резервируется память под временное хранение собранной информации и часть вычислительной мощности. К примеру, в DSO/BIOS TI (TI) такие затраты оцениваются как приблизительно 2 тысячи слов целевой памяти и производительность, равная 1 миллиону операций в секунду.

При этом необходимые действия указываются в анализируемой программе путем обращения к

библиотечным программам ядра или с помощью опций. Собранная в процессе выполнения на целевой аппаратуре информация периодически передается на хост, где обрабатывается и визуализируется. Таким образом обеспечивается непрерывное наблюдение за выполнением задачи в реальном времени, что особенно важно для построения высокопроизводительных систем реального времени на базе СБИС платформы "МУЛЬТИКОР".

Для того чтобы проанализировать ход выполнения приложения в реальном времени, необходимо одновременно представить во времени взаимодействие между задачами и потоки информации. Для этого информация, собранная в процессе выполнения на целевой системе, передается на хост через специальный канал. Последовательность событий графически представляется с помощью программы-анализатора, входящей в состав MCS.

Планируемый способ программирования прикладной задачи для СБИС на базе платформы МУЛЬТИКОР основывается прежде всего на языке Си, а далее выполняется профилирование затрат времени с помощью программных отладчиков на базе симуляторов RISC-ядра. После чего возможна замена длинных регулярных вычислительных процедур специальными функциями из библиотеки прикладных программ для DSP-ядра, либо замена длинных нерегулярных вычислительных процедур специализированными подпрограммами, разработанными на базе ассемблерных средств программирования DSP-ядра.

В качестве примера можно отметить уже имеющийся опыт программирования СБИС ми-

ПРИМЕР № 1 ПРОГРАММИРОВАНИЯ DSP-ЯДРА

Свертка (КИХ-фильтр)

Производительность: 1,25 такта на отвод

Разрядность данных и коэффициентов – 16 бит

Разрядность аккумулятора – 32 разряда

Производительность: 4 такта на базовую операцию в одной SIMD-секции

Особенности алгоритма: ((16x16→32)+32)+64
(прореживание 16:1)

```
ELCORE GAS Prim1.s      page 1

1 0000 020078A1          c1r1 R8
2 0001 01144321          c1r1 R4      (A0)+,R10
3 0002 0198C321          c1r1 R6      (A1)+,R12
4 0003 00000080          mpf R10, R12,R0  c1r1 R2
4 00802021
4 5018411D
5 0006 0000410D          do #4,Lp
5 00000000
6 0008 00848024          mpf R11,R13,R0  addl R0,R2,R2
6 581A431D
7 000a 00848024          mpf R14,R16,R0  addl R0,R2,R2
7 7020431D
8 000c 00848024          mpf R15,R17,R0  addl R0,R2,R2
8 7822431D
9 000e 00848024          Lp: mpf R10,R12,R0  addl R0,R2,R2
9 5018431D
10 0010 110878A4          addl R2,R4,R4
11 0011 418C78A4          addl R8,R6,R6
12
```

ПРИМЕР № 2 ПРОГРАММИРОВАНИЯ DSP-ЯДРА

Базовая операция FFT

Разрядность данных и коэффициентов – 16 бит

Производительность: 4 такта на базовую операцию

Особенности алгоритма: основание 2,
прореживание по частоте,
блочная плавающая точка

```
ELCORE GAS FFTbt1.s      page 1

1                                ;FFT butterfly; radix 2, complex data, time decimation
2                                .text
3 0000 00001000          tr R31,R31      (A1)+1,R2 (AT)+1T,R0
3 FFC16306
4 0002 190278FD          mpf R3,R1,R4
5 0003 198078FD          mpf R3,R0,R6
6 0004 120278FD          mpf R2,R1,R8
7 0005 00000080          mpf R2,R0,R10  addlR R6,R6,R8 (A1)+1,R2 (AT)+1T,R0
7 320C102A
7 1281631D
8 0008 5248703D          mpf R3,R1,R4  sublrR R10,R4,R9 (A0)+10,R14
8 1902611D
9 000a 429D0827          mpf R3,R0,R6  addxs R8,R14,R10
9 1980001D
10 000c 8000418F          do R16, LPb
11 000d 00000080          mpf R2,R1,R8  subxs R8,R14,R12 R10, (A2)+12
11 431D503B
11 1202641D
12 0010 320C102A          mpf R2,R0,R10  addlR R6,R6,R8 (A1)+11,R2
12 1280631D
13 0012 5248703D          mpf R3,R1,R4  sublrR R10,R4,R9 (A0)+10,R14
13 1902611D
14 0014 429D6027          LPb: mpf R3,R0,R6  addxs R8,R14,R10 R12,(A3)+13
14 1980661D
15                                .end
16
```

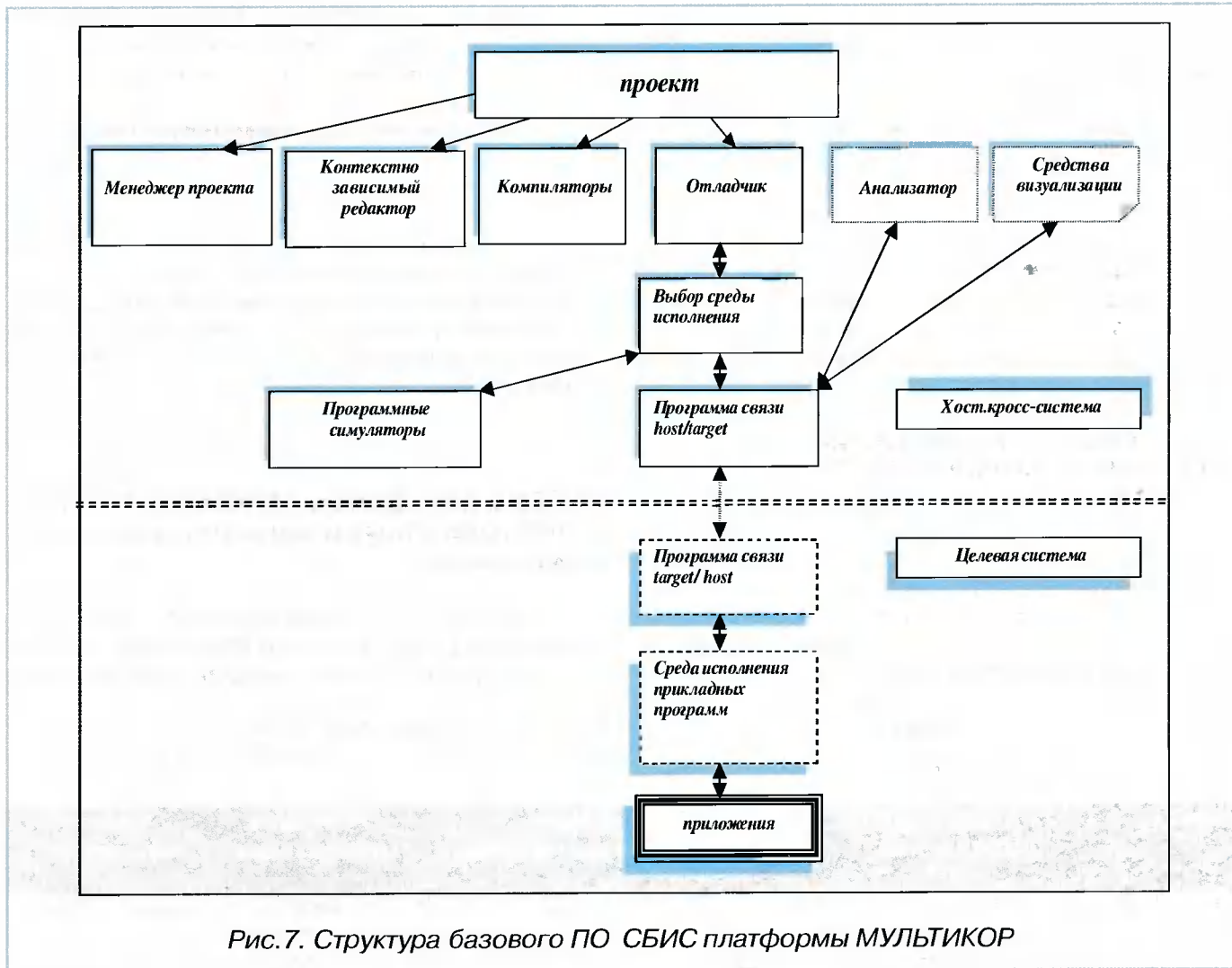



Рис. 7. Структура базового ПО СБИС платформы МУЛЬТИКОР

ни-конфигурации программируемого аудиодекодера на базе платформы "МУЛЬТИКОР", выполненный в ЭЛВИС для задачи MP3 аудиокодирования. Результат программирования этой прикладной задачи состоит в том, что объем программы, подготовленный первоначально на Си для фильтровых процедур MPEG-декодера (~80% программы), при переносе от RISC-ядра в DSP-ядро сокращается приблизительно в 32 раза. При этом время выполнения таких подпрограмм уменьшается почти на порядок.

Выше приведены два примера программ DSP-ядра для КИХ-фильтрации и БПФ, из которых видно, что ассемблерные программы обработки сигналов достаточно компактны.

На рис. 7 представлена общая иерархия ПО для СБИС платформы "МУЛЬТИКОР".

Состояние и перспективы разработки ПО для СБИС MC_01.

К концу года планируется разработка первой версии инструментального программного обеспечения для СБИС на базе платформы МУЛЬТИКОР, включая:

- программный симулятор СБИС Мультитор_01 под Win98,
- Си-компилятор для RISC-ядра,

- ассемблерные средства программирования DSP-ядра "ЭЛКОР-1",
- базовую прикладную библиотеку для DSP-ядра.

Планируется разработка аппаратно-программного отладчика СБИС на базе его FPGA-прототипа.

Базовая рабочая версия инструментального программного обеспечения будет разработана в 2002 г.

Инструментальное ПО будет включать ассемблерные средства программирования обоих ядер и Си-компилятор MIPS-ядра (III кв), программный симулятор кристалла MC_01 (включая симуляторы ядер и устройств ввода-вывода – IV кв., а также кристалла в целом), базовый пакет прикладных программ (IV кв), а также аппаратно-программный отладчик на базе XILINX FPGA прототипа СБИС MC_01 (I кв. 2002 г.).

Создана и поставлена на тестирование альфа-версия ассемблерных средств программирования DSP-ядра СБИС MC_01. Планируется в ближайшем времени начало разработки ПО в среде LINUX/sunOS, а также ядра ОС PV.

Разработка всех указанных средств осуществляется в ГУП НПЦ ЭЛВИС по договору с заказчиком.

Тем не менее может быть рассмотрен вопрос о поставке всего перечня или отдельных продуктов из базовых программных средств платфор-

мы “МУЛЬТИКОР” и другим заказывающим фирмам в IV кв. 2001 г. по соответствующей договорной цене.

Предполагается, что работа над программным симулятором RISC-ядра, входящего в состав наших кристаллов, будет завершена в IV кв. 2001 г. Поэтому потенциальные пользователи СБИС MC_01 в конце 2001 г. смогут перенести свои алгоритмы, представленные на Си, на RISC-ядро и тем самым отладить на нем свое прикладное ПО. Параллельно появится возможность в конце года оценить все критические места для прикладных задач пользователей на базе программного симулятора DSP-ядра.

6. Основные приложения СБИС на базе платформы “МУЛЬТИКОР”

К числу наиболее перспективных применений СБИС “МУЛЬТИКОР 01/02” относятся:

1. Высокопроизводительные комплексы бортовой радарной обработки в целях экологического мониторинга земной поверхности на борту малых спутников и в авиационных системах;

2. Подводные и надводные гидроакустические комплексы, системы береговой охраны;
3. Коммуникационные комплексы (модемы с пропускной способностью свыше 150 Мбит/с, кабельные однокристалльные модемы 10 Мбит/с и т.д.), цифровое вещание, однокристалльные системы спутниковой навигации);
4. Интеллектуальные терминалы доступа (IAT) с доступом в Интернет (HOME LAN) и системы спутникового Интернета;
5. Интеллектуальные проводные и беспроводные охранные системы аудио- и видеонаблюдения (кодеры и декодеры в стандарте MPEG4 и JPEG2000);
6. Мультимедийные системы (игры, проигрыватели и т.д.).

7. СБИС платформы “МУЛЬТИКОР” для мультимедийных и коммуникационных применений

На базе платформы “МУЛЬТИКОР” может быть разработан большой спектр СБИС для телекоммуникационных и мультимедийных В качестве

Таблица 2. Характеристики производительности СБИС “МУЛЬТИКОР_01” по типовым процедурам сигнальной обработки (миди-конфигурация)

№	Алгоритм выполнения	Время, производительность	Реальная реализация	Формат млн. опер./с
1	Нерекурсивный фильтр, действительные 16-разрядные данные	2,5 нс (на отвод)	800	16x16→32+32
2	Нерекурсивный фильтр, комплексные данные (8+j8)	2,5 нс (на отвод)	3200	8x8→16+16
3	FFT-1024, комплексные данные (16+j16), коэффициенты (16+j16), блочная плавающая точка	51,2 мкс (на преобразование)	1000 1200 (с масштабированием)	
4	FFT-1024, комплексные данные (16+j16), коэффициенты (8+j8), блочная плавающая точка	25,6 мкс (на преобразование)	2000 2400 (с масштабированием)	

Пример характеристик бортовой система синтеза изображения на базе СБИС “МУЛЬТИКОР”

Характеристики типовой космической РСА-системы, к примеру, в целях экологического мониторинга земной поверхности на базе СБИС “Мультикор_01” представлены ниже.

ПАРАМЕТР	ЗНАЧЕНИЕ
Разрешающая способность	2 м
Размер кадра	2 x 2 км
Периодичность съемки	10 с
Число некогерентных накоплений	4
Средняя интенсивность обрабатываемого потока	23 Мбайт/с
Компрессия синтезированного изображения	коэффициент сжатия – 4 – 10 раз
Число процессорных унимодулей на базе СБИС MC_01 (конструктивного исполнения РС-104)	2
Объем (с холодным резервированием) *	менее 5 л
Энергопотребление	менее 15 Вт

* Система обеспечивает 2-кратное резервирование за счет использования одномодульной обработки с увеличением периодичности съемки в 1,6 раза.



примера приведем две СБИС: коммуникационный и мультимедийный процессор нового поколения для беспроводных применений 2.5- и 3G-типа “Мультикор_01М” и СБИС “Аудиодекодер”, которые следует рассматривать как перспективные СБИС, проектируемые на базе платформы МУЛЬТИКОР.

Архитектура СБИС для коммуникационных и мультимедийных применений “Мультикор_01М” (рис. 3) реализуется на базе внутренней микроархитектуры СБИС супермикроконтроллера “Мультикор_01” и является ее мультимедийным расширением. СБИС поддерживает все необходимые интерфейсы и режимы функционирования для обеспечения самых современных приложений в области беспроводной связи 2.5- и 3G-типа. Эти приложения охватывают мобильный Интернет, потоковое аудио/видео (MPEG4, видеоконференции), плееры (MP3), игры 2D/3D, GPS-навигацию, защитные биометрические и идентификационные средства, охранные системы и системы защиты информации, бизнес приложения и т.д.

ПЕРСПЕКТИВНЫЕ ПРИЛОЖЕНИЯ МУЛЬТИМЕДИЙНОГО ПРОЦЕССОРА, ОБЪЕДИНЯЮЩЕГО В ОДНОМ КРИСТАЛЛЕ ИНТЕРНЕТ И БЕСПРОВОДНУЮ СВЯЗЬ 2.5- И 3 G-ТИПА

- **Аудио приложения:** MP3 кодер/декодер для телефонов, плееров, игр; перспективная обработка речи, (включая “текст – в – речь”, распознавание речи, AMR, высококачественное аудио)
- **Видеоприложения** (MPEG-2/4 (JPEG) кодер/декодер для видеоконференций, видеотелефонов, систем видеонаблюдений, игр и цифровых фотоаппаратов)
- **Мобильная спутниковая GPS-навигация** (для мобильных телефонов, бортовых компьютеров автомобилей и т.д.)
- **Игры – 2D, 3D**
- **Беспроводные модемы 2.5- и 3G-типа** (стандарты IMT-2000 DS-WCDMA, IMT-2000 MC-WCDMA Multi-Carrier 3X RTT, Blue Tooth, мобильный Интернет в реальном времени и т.д.)
- **Защитные технологии** (биометрические системы, идентификация пользователей, защита информации и т.д.)
- **Бизнес-технологии** (электронная коммерция, персональные помощники (PDA), голосовая навигация в информации)

Вторым СБИС коммерческого назначения, “мягкое ядро” которого разработано в ГУП НПЦ ЭЛВИС, является СБИС аудиодекодера (рис. 1).

Характеристики аудиодекодера нового поколения:

- Обеспечение стандарта MPEG-1/audio Layer 1, 2 и 3 (MP3)
- 1 или 2 канала (в том числе стереозвук)
- Частоты отсчетов на каждый канал (32, 44.1 или 48 кГц)

- Битовые скорости на канал: от 32 до 224 кбит/с (коэффициент сжатия: 2.3 – 24)
- Управляющий код для MP3, не уступающий лучшим в мире аналогам (15 с при реализации на 25 МГц плате R3041)
- Программирование любых мировых аудио-стандартов: (MPEG-2, AAC, ATRAC3/Sony и т.д.) на базе RISC-и DSP-ядер
- Наращиваемость архитектуры для поддержки видеообработки путем масштабирования DSP-ядра;
- Возможность обеспечения новой технологии защиты музыкальных файлов – Secure Digital Music Initiative (SDMI)
- Возможность синтеза СБИС аудиодекодера на 0,52-мкм библиотеках ОАО «Ангстрем».

8. Модули на базе СБИС “Мультикор_01/02”

НПЦ ЭЛВИС приступил к разработке модулей на базе СБИС “Мультикор_01/02”, которые предназначены, во-первых, для работы в составе аппаратно-программного отладчика инструментального и прикладного ПО (в конструктивном исполнении PC), так и в стандарте **PC-104Plus** для реализации в составе систем пользователей.

Предполагается, что каждый из модулей будет сначала реализован на базе Virtex прототипа СБИС “Мультикор_01” (I квартал. 2002 г.), а затем изготовлен с использованием кристалла первого отечественного супермикроконтроллера MC_01 (предварительно IV квартал 2002 г.).

На базе модулей супермикроконтроллеров в стандарте **PC-104Plus** пользователи смогут разрабатывать свои модули в другом конструктивном исполнении.

В качестве примера на рис. 8 приведен пример структуры модуля обработки сигналов (МОС) для производственных систем, имеющего габариты 233*160 мм (типоразмер печатной платы “Евро-механика” 6U) и VME-шину. МОС содержит 4 вычислительных элемента (ВЭ), спроектированных на базе СБИС MC_01 и конструктивно оформленных в виде модулей в стандарте PC-104Plus, устанавливаемых на VME плату мезонинным способом. Внутри шины ВЭ объединены в мультипроцессорную систему на шине PCI, образуя также кластерную конфигурацию посредством объединения SHARC-совместимых линков.

Модуль обработки сигналов (МОС), показанный на рис. 8, представляет собой вычислительное устройство с симметричной внутренней архитектурой, содержащее четыре вычислительных элемента, и может использоваться как основное вычислительное устройство бортового комплекса обработки информации, совмещающее функции обработки данных с функциями контроля и управления объектами.

Модуль обеспечивает пиковую производительность 12,8 млрд. оп./с с фиксированной точ-

кой и дополнительно 600MFlops операций в стандарте IEEE754 при реализации на СБИС MC_01/02.

Входные данные (от аналогово-цифровых преобразователей) поступают в буфер данных программируемого мультиплексора по 32-разрядной параллельной шине или по высокоскоростным оптическим каналам и «на проходе» распределяются между ВЭ обработки сигналов. В данном примере ввод оцифрованных потоков данных с АЦП в СБИС осуществляется через порты памяти ВЭ. В других примерах реализации модулей ВЭ в конструктивном исполнении PC-104Plus данные могут быть введены в МОС, к примеру, через линки, а обработанные результаты могут выводиться через PCI-шину и далее из модуля через мост PCI/VME.

Алгоритм распределения данных задается программно. В результате каждый ВЭ модуля обработки сигналов получает предназначенный ему набор данных в локальную очередь (FIFO). Обработка информации в МОС может вестись как после прихода всей совокупности данных, так и «на проходе» по мере их поступления. При этом микроконтроллер ВЭ работает, как правило, со своей локальной оперативной памятью по индивидуальной программе. Для обмена данными с другими ВЭ используются 4 линковых канала с суммарной пропускной способностью 132 Мбайт/с.

В данном примере реализации PC-104Plus модуля в состав ВЭ входят:

- супермикроконтроллер MC_01;
- локальное ОЗУ емкостью 64 Мбайт;

- локальное буферное ОЗУ входных данных типа FIFO емкостью 128Кх 32-разрядных слов;
- энергонезависимое ПЗУ для хранения программы начального тестирования и загрузки;
- электрически перепрограммируемое энергонезависимое ЗУ для хранения параметров модуля.

Заключение

Запуск в изготовление опытных тестовых образцов СБИС «МУЛЬТИКОР_01» миди-конфигурации по технологии 0.5-мкм запланирован ОАО «АНГСТРЕМ» в декабре 2001 г. На настоящий момент ОАО «АНГСТРЕМ» ведет разработку топологии СБИС.

К концу года планируется получение первой версии инструментального программного обеспечения для СБИС на базе платформы «МУЛЬТИКОР», включая: программный симулятор СБИС «Мультикор_01» под Win98, Си-компилятор для MIPS RISC-ядра, ассемблерные средства программирования DSP-ядра «ЭЛКОР-1», базовую прикладную библиотеку для DSP-ядра. Базовая рабочая версия инструментального программного обеспечения будет разработана в 2002 г.

НПЦ ЭЛВИС приступил к разработке модулей на базе СБИС «Мультикор_01/02», которые предназначены, во-первых, для работы в составе аппаратно-программного отладчика инструментального и прикладного ПО (в конструктивном исполнении PC), так и в стандарте PC-104Plus для реализации в составе систем пользователей.

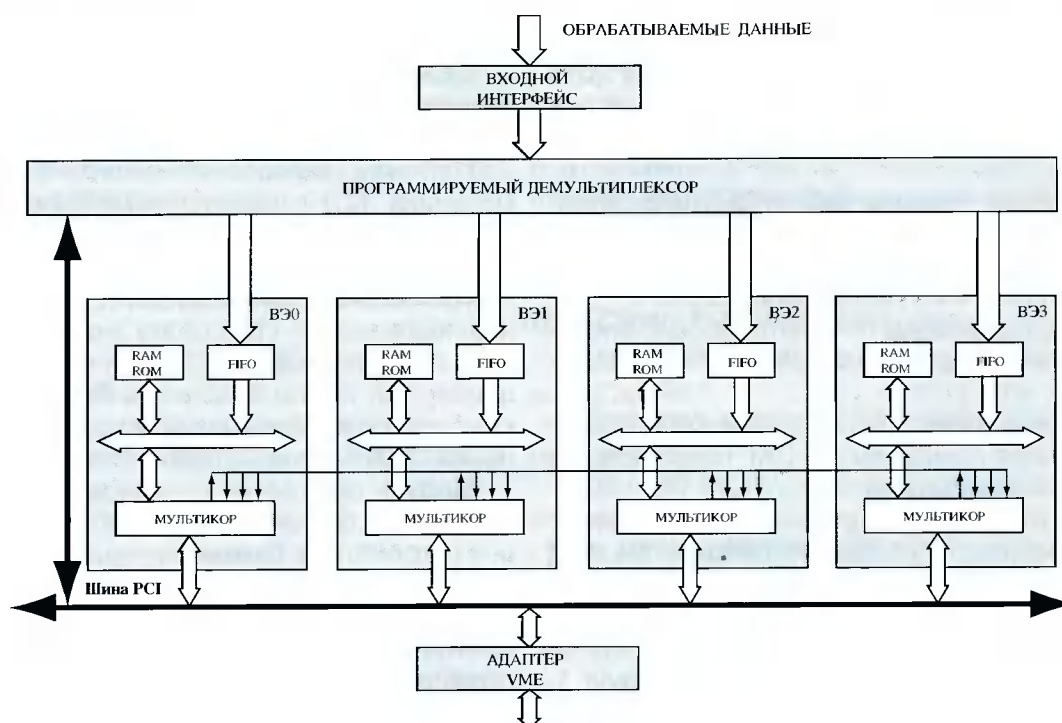


Рис.8. Структура типового МОС на базе СБИС MC_01

В.А. Кашкаров

Организация параллельных вычислений преобразования Адамара на процессоре NM6403 (L1879VM1)

Введение

Статья продолжает серию публикаций, посвященных вопросам эффективной реализации типовых алгоритмов ЦОС на архитектуре процессора NeuroMatrix® NM6403. Особенностью этого процессора, созданного коллективом НТЦ «Модуль», является наличие векторного ядра для выполнения векторно-матричных операций над данными произвольной разрядности до 64 бит [1, 4].

Операционное устройство векторного ядра – регулярная матричная структура 64×64 ячейки с 64-разрядными входами \bar{X} , \bar{Y} и выходом OUT . Матрица может быть произвольно разделена на столбцы и строки. В образовавшиеся после разделения ячейки загружаются весовые коэффициенты W_{ij} . На вход матрицы подаются входные данные x_1, x_2, \dots, x_n , упакованные в 64-разрядный вектор $\bar{X} = (x_1, x_2, \dots, x_n)$. Каждому элементу вектора \bar{X} соответствует строка матрицы. Ширина строки (в битах) – разрядность данного элемента входных данных. В ячейках происходит умножение элемента вектора входных данных на весовой коэффициент и сложение со значением верхней ячейки (либо значений входов Y_i). Таким образом, для каждого столбца вычисляется скалярное произведение $OUT_i = Y_i + \sum_j W_{ij} X_j$. Для снижения раз-

рядности выходных данных и защиты от арифметического переполнения используется программируемая функция насыщения.

Все операции в матрице выполняются параллельно за один такт. Загрузка весовых коэффициентов W_{ij} происходит за 32 такта. В векторном сопроцессоре есть «теневая» матрица, в которую весовые коэффициенты можно загружать в фоновом режиме. Переключение «теневого» и рабочей матриц занимает один такт.

Благодаря векторному ядру на частоте 40 МГц процессор NM6403 эффективно выполняет большинство алгоритмов ЦОС. В данной статье показана реализация преобразования Адамара со следующими условиями:

- разрядность входных/выходных данных – 8/16 бит (числа со знаком),
- количество шагов – 8,
- размер входного/выходного вектора – 256 элементов.

На решение этой задачи процессор расходует 349 тактов, включая затраты на вызовы Си-функций. Таким образом, эффективная скорость выполнения такого преобразования составляет около 6 арифметических операций за один такт или

240 млн. арифметических операций в секунду при тактовой частоте 40 МГц.

В данном документе рассматривается реализация алгоритма преобразования Адамара. Особое внимание уделяется работе с векторным узлом процессора.

Описание преобразования Адамара

Преобразование Адамара осуществляется над целыми числами. В качестве базисных функций используются двузначные функции Уолша, принимающие значения 1, –1. Поэтому преобразование Адамара не имеет других операций кроме сложения и вычитания.

Преобразование Адамара можно выразить в виде произведения матрицы на вектор. Строки и столбцы матрицы преобразования ортогональны и состоят из элементов +1 и –1. Матрицу Адамара можно определить рекурсивно:

$$H_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \text{ и } H_{2N} = \frac{1}{\sqrt{2}} \begin{bmatrix} H_N & H_N \\ H_N & -H_N \end{bmatrix}.$$

Например, матрица восьмого порядка выглядит следующим образом:

$$H_8 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ 1 & -1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & -1 \end{bmatrix}$$

Рис. 1. Матрица Адамара 8-го порядка

Если на входе имеется вектор x длиной N ($N=2^n$), то вектор y , полученный в результате преобразования Адамара, равен $y = H_N x$.

Преобразование, описанное выше, называется дискретным преобразованием Адамара (ДПА). Однако на практике чаще используется быстрое преобразование Адамара (БПА). Его суть может быть описана табл. 1.

Из табл. 1 видно, что элементы третьего шага преобразования могут быть выражены через элементы входного массива следующим образом:

Таблица 1. Шаги быстрого преобразования Адамара

Входные данные	Первый шаг (1)	Второй шаг (2)	Третий шаг (3)
a_1	$b_1 = a_1 + a_2$	$c_1 = b_1 + b_3$	$d_1 = c_1 + c_5$
a_2	$b_2 = a_1 - a_2$	$c_2 = b_2 + b_4$	$d_1 = c_2 + c_6$
a_3	$b_3 = a_3 + a_4$	$c_3 = b_1 - b_3$	$d_1 = c_3 + c_7$
a_4	$b_4 = a_3 - a_4$	$c_4 = b_2 - b_4$	$d_1 = c_4 + c_8$
a_5	$b_5 = a_5 + a_6$	$c_5 = b_5 + b_7$	$d_1 = c_1 - c_5$
a_6	$b_6 = a_5 - a_6$	$c_6 = b_6 + b_8$	$d_1 = c_2 - c_6$
a_7	$b_7 = a_7 + a_8$	$c_5 = b_5 - b_7$	$d_1 = c_3 - c_7$
a_8	$b_8 = a_7 - a_8$	$c_6 = b_6 - b_8$	$d_1 = c_4 - c_8$

$$\begin{aligned}
 d_1 &= a_1 + a_2 + a_3 + a_4 + a_5 + a_6 + a_7 + a_8 \\
 d_2 &= a_1 - a_2 + a_3 - a_4 + a_5 - a_6 + a_7 - a_8 \\
 d_3 &= a_1 + a_2 - a_3 - a_4 + a_5 + a_6 - a_7 - a_8 \\
 d_4 &= a_1 - a_2 - a_3 + a_4 + a_5 - a_6 - a_7 + a_8 \\
 d_5 &= a_1 + a_2 + a_3 + a_4 - a_5 - a_6 - a_7 - a_8 \\
 d_6 &= a_1 - a_2 + a_3 - a_4 - a_5 + a_6 - a_7 + a_8 \\
 d_7 &= a_1 + a_2 - a_3 - a_4 - a_5 - a_6 + a_7 + a_8 \\
 d_8 &= a_1 - a_2 - a_3 + a_4 - a_5 + a_6 + a_7 - a_8
 \end{aligned}$$

$$\begin{aligned}
 b_1 &= a_1 + a_2 \\
 b_2 &= a_1 - a_2
 \end{aligned}$$

Поэтому вычисление значений пар элементов $(b_1, b_2), (b_3, b_4), \dots (b_{2^{k-1}}, b_{2^k})$ можно вести независимо. И так на каждом шаге вплоть до последнего.

Реализация алгоритма на процессоре NM6403

Если обратиться к рис. 1, то станет очевидно, что коэффициенты 1 и -1 перед элементами входного вектора в точности соответствуют матрице Адамара восьмого порядка.

В общем виде преобразование на k -ом шаге определяет взаимодействие 2^k элементов входного вектора. При этом каждый из первых 2^{k-1} элементов заполняется суммой значений элемента текущей ячейки и ячейки, отстоящей на 2^{k-1} элементов, а каждый из вторых 2^{k-1} элементов – разностями указанных ячеек (рис. 2).

В результате преобразования каждый элемент выходного вектора выражается через все элементы входного. Однако если отступить на один шаг преобразования назад, то легко заметить, что существуют два блока данных, в каждом из которых наблюдается полная взаимосвязь между элементами, тогда как в формуле преобразования элементов одного блока не участвует ни один элемент другого, то есть блоки являются абсолютно независимыми друг от друга. Например, на первом шаге каждые два элемента являются независимыми от соседних пар, поскольку вычисляются по формуле вида:

Процессор NM6403 позволяет программно изменять разрядность обрабатываемых данных. Это значит, что задав соответствующее разбиение его рабочей матрицы, участвующей в вычислениях, можно в течение нескольких шагов выполнять преобразования над 8-разрядными данными, а когда теоретически рассчитанная разрядность результатов потребует выхода за 8 бит, преобразовать данные к 16-разрядному виду и продолжить вычисления, и т.д. Максимально возможная разрядность накопителя, реализованная в процессоре NM6403, составляет 64 бита. То есть при разрядности начальных данных 8 бит возможно выполнить 56 шагов преобразования Адамара (при этом потребуются $2^{56}(\sim 10^{17})$ 64-разрядных слов памяти для размещения вектора результата).

Наличие у процессора матричного вычислительного узла позволяет вести вычисление результата сразу пяти шагов преобразования. Конечно, каждый шаг преобразования может быть реализован отдельно, однако эффективность параллельной обработки нескольких слоев будет несравнимо выше.



Рис. 2. Преобразование на k -ом шаге

Далее более подробно приводится описание выполнения каждой из перечисленных выше операций.

Загрузка рабочей матрицы

Схема вычислений на Матричном Операционном Узле (МОУ) процессора NM6403 приведена на рис. 4.

В узлах матрицы расположены весовые коэффициенты, предварительно загруженные из памяти. Обрабатываемый поток данных поступает на вход **X**. Входом и выходом при обработке данных являются два буфера, организованных по принципу FIFO, каждый глубиной в 32 64-разрядных слова. В качестве источника данных для входа **X** могут использоваться: внешняя память (data), небольшой буфер внутренней памяти (ram) или аккумулирующее FIFO результатов (afifo). В качестве приемника результата всегда выступает afifo (неявный параметр всех векторно-матричных операций).

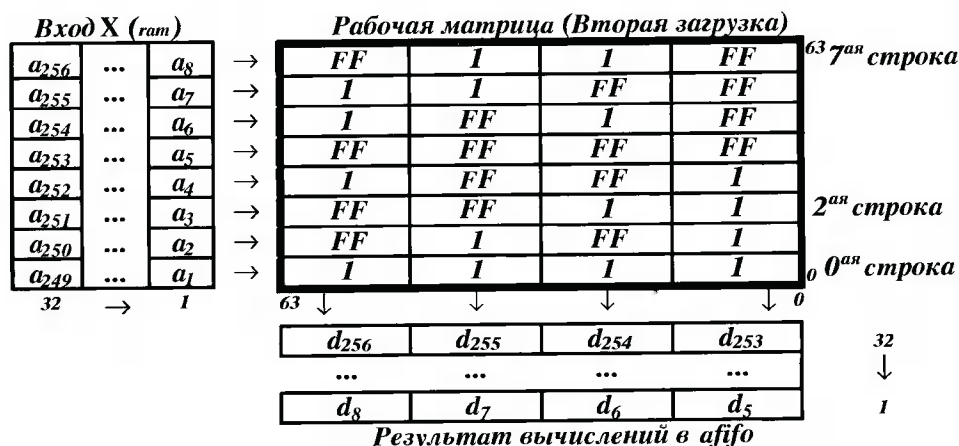
Весовые коэффициенты заносятся в рабочую матрицу из внешней памяти, доступной процессору. В памяти они хранятся в виде массива 64 векторов. При загрузке матрицы в нее попадет столько слов, каково ее разбиение по строкам, заданное в регистре sb2. Каждая строка задается отдельным 64-разрядным словом. Слово массива с нулевым индексом попадает в нулевую строку матрицы. Как видно из рис. 4, строки рабочей матрицы нумеруются снизу вверх. Такой способ нумерации обусловлен нумерацией битов в слове (нумерация ведется справа налево, младший бит находится справа). В том же направлении ведется увеличение адресов памяти.

Поскольку в нашем случае рабочая матрица процессора NM6403 разбивается на 4 столбца, младшие 16 разрядов нулевого слова массива попадают в нулевой столбец, следующие 16 бит в первый столбец, и т.д.

Весовые коэффициенты читаются из памяти в специальную внутреннюю память процессора, называемую wfifo. Этот блок внутренней памяти также работает по принципу FIFO. Он



а) Первая стадия вычислений (нечетные слова результата)



б) Вторая стадия вычислений (четные слова результата)

Рис. 4. Обработка данных в Матричном Операционном Узле

используется только для загрузки данных в матрицу. Память wfifo обладает отличительной особенностью. Она может заполняться постепенно, за несколько операций.

Загрузка данных из памяти в рабочую матрицу описывается следующими командами на языке ассемблера:

```
begin ".text"
<Steps_1_3>
.branch; // установка бита, разрешающего параллельное выполнение
// векторных команд.
nb1 = 80808080h; // разбиение матрицы на столбцы.
sb = 03030303h; // разбиение матрицы на строки.

ar0 = M_1_3; // адрес массива весов
rep 8 wfifo = [ar0+], ftw, wtw; // загрузка весов
...
end ".text";
```

сразу после присваивания им новых значений, а только после выполнения команды wtw, которая скопирует содержимое теневой матрицы в рабочую.

Регистры nb1 и sb являются 64-разрядными. Если они инициализируются 32-разрядной константой, то процессор копирует это значение в старшие 32 бита, то есть получается, что регистр nb1 инициализирован длинной константой 8080808080808080h. То же верно и для регистра sb. Более подробно работа с регистрами nb1 и sb (sb1, sb2) описана в [3].

Итак, определено будущее разбиение рабочей матрицы.

В адресный регистр заносится адрес массива весовых коэффициентов, а затем данные загружаются из памяти в блок wfifo.

По команде ftw данные из блока wfifo попадают в теневую матрицу. Эта передача занимает всегда 32 такта, независимо от того, сколько слов загружается в матрицу. Например, в случае загрузки матрицы Адамара вось-

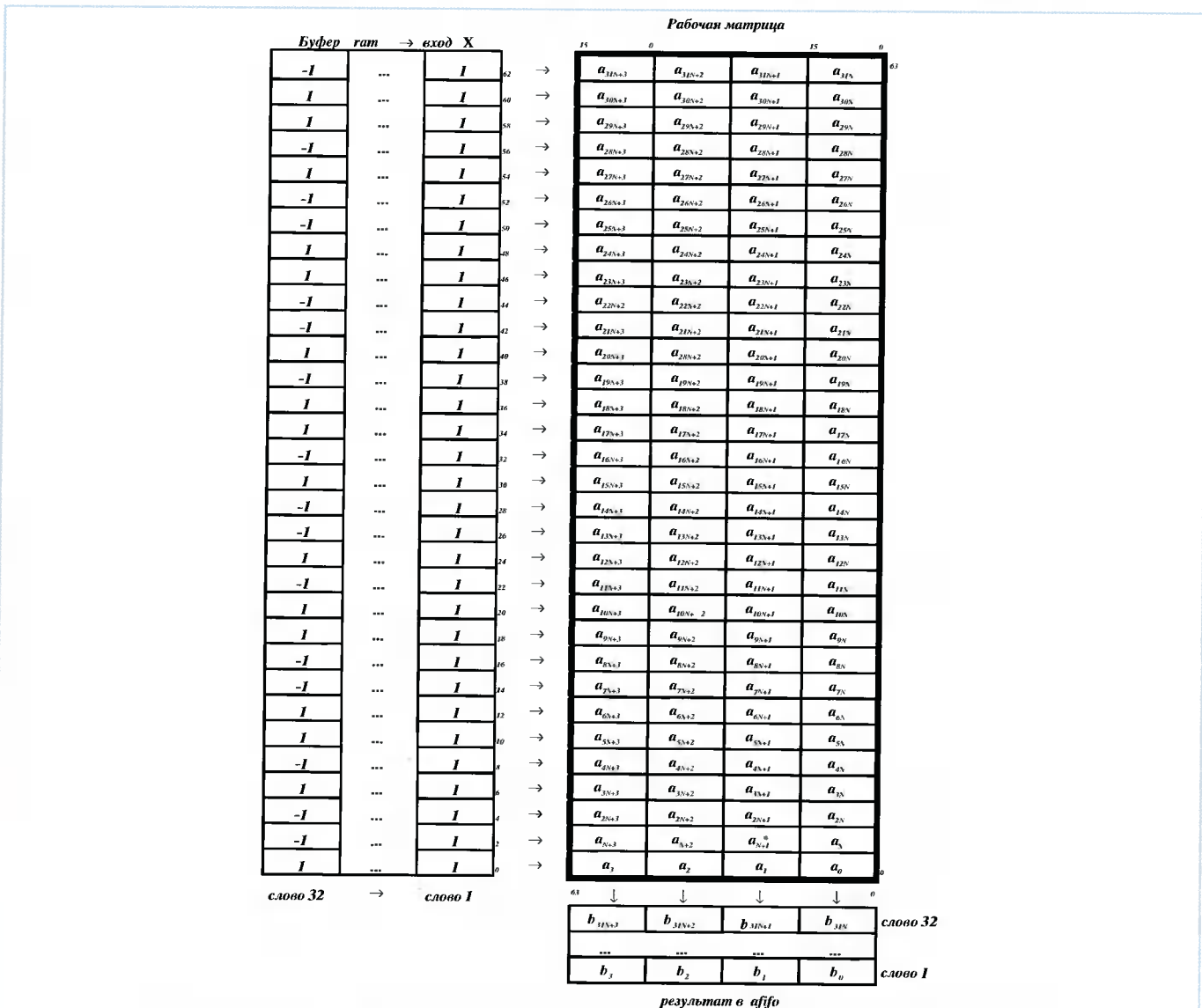


Рис. 5. Матричный узел процессора в режиме обработки 128 элементов

Таблица 2. Таблица связей между элементами пяти шагов преобразования Адамара

Шаг 3	Шаг 4	Шаг 5	Шаг 6	Шаг 7	Шаг 8
1	1 + 2	1 + 3	1 + 5	1 + 9	1 + 17
2	1 – 2	2 + 4	2 + 6	2 + 10	2 + 18
3	3 + 4	1 – 3	3 + 7	3 + 11	3 + 19
4	3 – 4	2 – 4	4 + 8	4 + 12	4 + 20
5	5 + 6	5 + 7	1 – 5	5 + 13	5 + 21
6	5 – 6	6 + 8	2 – 6	6 + 14	6 + 22
7	7 + 8	5 – 7	3 – 7	7 + 15	7 + 23
8	7 – 8	6 – 8	4 – 8	8 + 16	8 + 24
9	9 + 10	9 + 11	9 + 13	1 – 9	9 + 25
10	9 – 10	10 + 12	10 + 16	2 – 10	10 + 26
11	11 + 12	9 – 11	11 + 15	3 – 11	11 + 27
12	11 – 12	10 – 12	12 + 16	4 – 12	12 + 28
13	13 + 14	13 + 15	9 – 13	5 – 13	13 + 29
14	13 – 14	14 + 16	10 – 16	6 – 14	14 + 30
15	15 + 16	13 – 15	11 – 15	7 – 15	15 + 31
16	15 – 16	14 – 16	12 – 16	8 – 16	16 + 32
17	17 + 18	17 + 19	17 + 21	17 + 25	1 – 17
18	17 – 18	18 + 20	18 + 22	18 + 26	2 – 18
19	19 + 20	17 – 19	19 + 23	19 + 27	3 – 19
20	19 – 20	18 – 20	20 + 24	20 + 28	4 – 20
21	21 + 22	21 + 23	17 – 21	21 + 29	5 – 21
22	21 – 22	22 + 24	18 – 22	22 + 30	6 – 22
23	23 + 24	21 – 23	19 – 23	23 + 31	7 – 23
24	23 – 24	22 – 24	20 – 24	24 + 32	8 – 24
25	25 + 26	25 + 27	25 + 29	17 – 25	9 – 25
26	25 – 26	26 + 28	26 + 30	18 – 26	10 – 26
27	27 + 28	25 – 27	27 + 31	19 – 27	11 – 27
28	27 – 28	26 – 28	28 + 32	20 – 28	12 – 28
29	29 + 30	29 + 31	25 – 29	21 – 29	13 – 29
30	29 – 30	30 + 32	26 – 30	22 – 30	14 – 30
31	31 + 32	29 – 31	27 – 31	23 – 31	15 – 31
32	31 – 32	30 – 32	28 – 32	24 – 32	16 – 32

мого порядка требуется загрузка восьми слов, но их перекодировка во внутреннее представление независимо от этого длится 32 такта. Однако перекодировка ведется параллельно с закачкой весов и начинается с момента появления первого слова в блок wfifo.

После того как закачка весов в теневую матрицу завершена, выполняется команда wtw. Она за один такт переписывает содержимое теневой матрицы в рабочую. При этом значения регистров pb1 и sb1 копируются в pb2 и sb2. По окончании загрузки рабочей матрицы можно приступить непосредственно к вычислениям. Для этого необходимо задать адреса входного и выходного буферов. Желательно, чтобы эти буферы располагались в разных блоках памяти, находящихся на разных шинах данных. В этом случае

операции чтения входного буфера и запись выходного могут выполняться параллельно.

Каждая векторная инструкция содержит внутренний счетчик циклов (от 1 до 32). Эти элементарные циклы описывают выполнение одной и той же операции над потоком данных. Максимальное количество данных, обрабатываемых одной векторной командой, определяется глубиной внутренних буферов памяти процессора NM6403, которая равна тридцати двум 64-разрядным словам.

Реализация следующих пяти шагов преобразования Адамара

Как уже упоминалось, преобразование Адамара содержит только операции сложения и

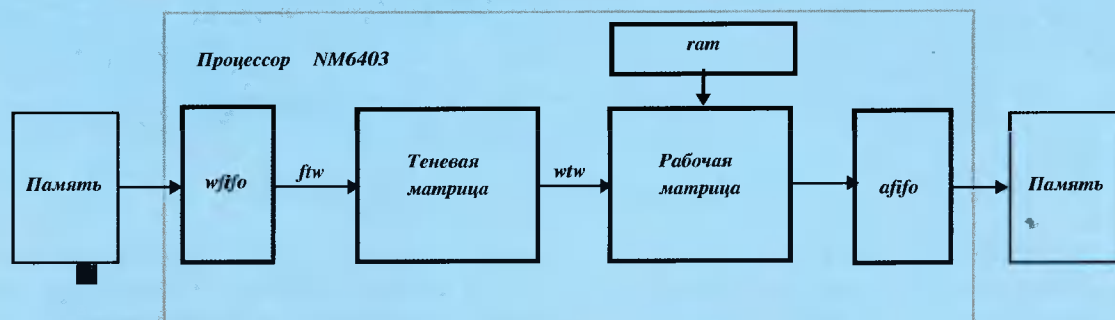


Рис. 5. Схема обработки данных в процессоре NM6403 при параллельном вычислении пяти шагов преобразования Адамара

вычитания. Таким образом, можно разбить рабочую матрицу процессора на 32 строки так, чтобы разрядность данных, поступающих на вход X , равнялась 2 битам. Этого достаточно для хранения чисел $-1, 0$ и 1 .

Разбиение матрицы на 32 строки позволяет выполнять в каждом столбце по 32 операции сложения/вычитания. При обработке 16-разрядных данных матрица разбивается на 4 столбца, поэтому можно за один процессорный такт выполнить 128 операций сложения/вычитания (рис. 5). Основным вопросом состоит в том, как обеспечить взаимодействие всех 32 строк матрицы. Его решение содержится в табл. 2.

Цифры в каждом столбце обозначают номера ячеек предыдущего шага. Из табл. 2 видно, что возможно выразить значения элементов, получаемых на 8-м шаге через элементы, вычисленные на 3-м шаге.

В качестве примера рассмотрим, каким образом значение 7-й ячейки на 8-м шаге выражается через значения ячеек 3-го шага (см. выделенные ячейки). Значение 7-ой ячейки 8-го шага записывается как сумма значений 7-й и 23-й ячеек шага 7, каждая из которых в свою очередь выражается через значения ячеек 6-го шага, и т.д. В результате значение 7-й ячейки 8-го шага выражается через значения ячеек 3-го шага следующим образом:

$$b_7 = a_1 + a_2 - a_3 - a_4 - a_5 - a_6 + a_7 + a_8 + a_9 + a_{10} - a_{11} - a_{12} - a_{13} - a_{14} + a_{15} + a_{16} + a_{17} + a_{18} - a_{19} - a_{20} - a_{21} - a_{22} + a_{23} + a_{24} + a_{25} + a_{26} - a_{27} - a_{28} - a_{29} - a_{30} + a_{31} + a_{32}$$

Все коэффициенты правой части выражения могут быть собраны в 64-разрядную константу: 05FF55FF55FF55FF5hI, где под каждый знак отведено 2 бита. Схема вычислений, используемая в рассматриваемой функции, отлична от той, что была выбрана при выполнении первых трех шагов. Основное отличие состоит в том, что в рабочую матрицу загружаются не весовые коэффициен-

ты, а данные, тогда как веса подаются на вход X .

Выполнение параллельного вычисления пяти шагов преобразования Адамара осуществляется по следующему сценарию:

- загрузка весовых коэффициентов из внешней памяти в блок ram;
- загрузка четных слов входных данных в блок wfifo → Тен. Матр. → Раб. Матр.;
- загрузка нечетных слов входных данных в блок wfifo → Тен. Матр. с одновременным выполнением вычислений над четными словами;
- сохранение четных слов результата в памяти;
- выполнение вычислений над нечетными словами и параллельное сохранение вычисленных слов результата в памяти.

Схема обработки данных, используемая в данной задаче, представлена на рис.5.

Заключение

Приведенные в данной статье подходы к программированию процессора NM6403 могут быть обобщены на решение других задач из области обработки сигналов и изображений. При этом с увеличением разрядности задачи производительность процессора возрастает, поскольку процентные затраты на начальную загрузку весовых коэффициентов снижаются на фоне общего количества вычислений.

Литература

1. НТЦ Модуль. "Процессор NM6403. Введение в архитектуру."
2. С. Кун. Матричные процессоры на СБИС. М.: Мир, 1991, с. 86 – 87.
3. НТЦ Модуль. "ПО процессора NM6403. Описание языка ассемблера".
4. Кашкаров В.А., Мушкаев С.В. Организация параллельных вычислений в алгоритмах БПФ на процессоре NM6403 // Цифровая обработка сигналов, 2001, № 1.

Согласование последовательных АЦП с высокопроизводительными сигнальными процессорами Texas Instruments

В современных приложениях цифровой обработки сигналов встречаются задачи, в которых необходимо обеспечить обработку входного сигнала, значения которого через последовательный АЦП поступают в сигнальный процессор [1]. Последовательные преобразователи, применяемые в таких приложениях, в десятки раз медленнее процессора, вследствие чего при организации циклов задержки во время ввода данных процессор будет простаивать, что является весьма неэффективным решением.

Выходом из этого положения является использование последовательного порта, содержащегося в сигнальных процессорах TI (McBSP) и прямого доступа в память (DMA). При этом ядро процессора не участвует в процессе ввода, за исключением стадии инициализации каналов контроллера прямого доступа. Данное решение является весьма эффективным в случае, если нам необходимо обрабатывать в реальном времени, допустим, аудиоданные. Входные значения сигнала посредством прямого доступа в память через последовательный порт заносятся в первую область памяти, в это время идет обработка второй области, а из третьей данные поступают на внешнее устройство (ЦАП). При заполнении буфера ввода генерируется прерывание, в ответ на которое управляющая программа изменяет назначение областей: область ввода становится областью обработки, область обработки – областью вывода, область вывода – областью ввода.

Приведем краткое описание процесса инициализации последовательного порта, контроллера прямого доступа в память процессора TMS320VC5402, а также аналого-цифрового преобразователя TLV2548.

McBSP

McBSP (multi-channel buffered serial port) является наследником последовательного порта более ранних моделей сигнальных процессоров Texas Instruments. McBSP имеет свойства, не только присутствующие прошлым реализациям порта, но также обеспечивает прямое согласование с T1/E1 кадрами, IOM-2, ST-BUS, AC97, IIS, SPI – совместимыми устройствами. Он позволяет выбирать размер данных из широкого диапазона, применять u-law и A-law компадирование, программировать полярность импульсов синхронизации кадров и данных.

Разберем устройство регистров McBSP. Данные регистры отображены на память с использованием схемы регистровой подадресации, как показано на рис. 1. Схема работает по принципу мультиплексиро-

вания управляющих регистров в одну ячейку памяти. Регистр данных (SPSDx) используется для чтения или записи нужного подадресуемого регистра. Для доступа к необходимому регистру в регистр подадреса (SPSAx) заносится адрес необходимого регистра. По-

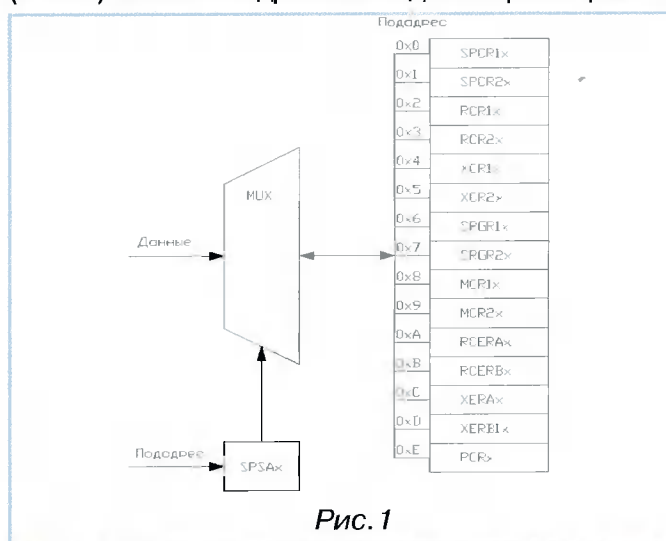


Рис. 1

сле чего мультиплексор переключается на необходимую ячейку. Во время цикла доступа данные из SPSDx заносятся по адресу, содержащемуся в SPSAx. Таким же образом происходит чтение, данные из регистра по адресу SPSAx заносятся в SPSDx.

В примере 1 приведен пример записи 0x0000 в управляющий регистр 1. Адрес регистра SPSD 0x039, SPSA – 0x038.

Пример 1. Алгоритм записи в управляющий регистр McBSP

```
SPSA0      .set  038h ;Подадресный регистр McBSP0
SPSD0      .set  039h ;Буферный регистр McBSP0
SPCR10_SUB .set  000h ;Подадрес управляющего регистра 1
mmr(#SPSA0) = #SPCR10-SUB
mmr(#SPSD0) = #0000h
```

Для управления последовательным портом служат 16 управляющих регистров. Для связи с TLV2548 необходимо использовать 9 из них. Последовательный порт конфигурируется посредством записи необходимых значений в следующие регистры:

1. Serial Port Control Register 1 (SPCR1). Содержит состояние приемника, а также разрешает или запрещает прием информации. Данный регистр также содержит биты управления синхронизацией и прерываниями от McBSP.
2. Serial Port Control Register 2 (SPCR2). Содержит биты состояния передатчика, биты разреше-

ния передачи. Здесь также содержатся биты сброса генератора синхронизации.

3. Pin Control Register (PCR). Содержит биты для управления режимом ввода/вывода во время работы порта. С помощью этих разрядов в случае запрета приемника или передатчика выводы порта можно переконфигурировать в режим работы ввода/вывода общего назначения. Биты данного регистра также управляют режимами синхронизации (CLKX/R, FSX/R работают в режиме ввода или вывода и управляют выбором полярности).
4. Receive Control Register 1 (RCR1). Содержит биты, управляющие различными режимами приемника. Битами данного регистра определяется размер передаваемых слов (8 – 32 бит), количество слов в кадре (1 – 128).
5. Receive Control Register 2 (RCR2). Битами данного регистра выбирается режим компадирования, количество бит задержки между кадрами. При неиспользовании компадирования битами данного регистра можно задать очередность передачи битов слова (первый – старший или младший).
6. Transmit Control Register 1 (XCR1). Определяет длину слова (8 – 32) и количество слов в кадре (1 – 128).
7. Transmit Control Register 2 (XCR2). Регистр задает компадирование, количество бит задержки, очередность бит передачи.
8. и 9. Sample-Rate Generator 1 (SRGR1) и Sample-Rate Generator 2 (SRGR2). Регистры задают режимы работы внутреннего генератора последовательного порта.

Полное описание регистров управления можно найти в [2,3].

Значения управляющих регистров McBSP не зависят от того, что управляет портом: ядро процес-

сора или котроллер прямого доступа в память. В табл. 1 приведены значения вышеописанных регистров для работы McBSP с TLV2548.

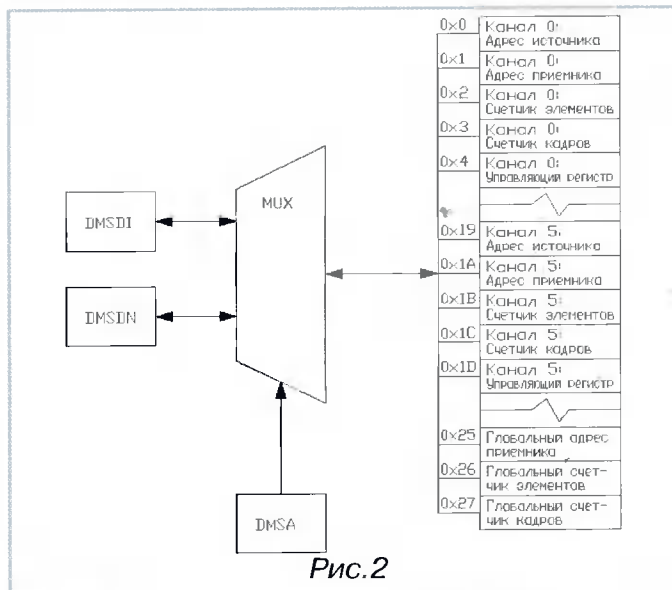


Рис.2

Прямой доступ в память (DMA)

В контроллере прямого доступа в память используется та же схема подадресации, что и в последовательном порте, и, следовательно, такая же процедура записи и чтения. В контроллер добавлен еще один регистр, позволяющий упростить программирование первого. На рис. 2 показаны два регистра на входе мультиплексора: DMSDI, DMSDN. Их отличие заключается в том, что при записи/чтении в DMSDI происходит автоматическое наращивание адреса в DMSA. Таким образом, пользователю нет необходимости «вручную» зада-

Таблица 1. Пример установки регистров McBSP

Адрес в общем адресном пространстве	Подадрес в пространстве адресов McBSP	Обозначение	Значение	Комментарии
0041		DRR11		Принимаемые данные
0043		DXR11		Передаваемые данные
0048		SPSA1		Регистр подадреса
0049	0x0000	SPCR11	0x0001	Запрет передачи на время конфигурации McBSP
	0x0001	SPCR21	0x02C1	Запрет приема на время конфигурации McBSP
	0x0002	RCR11	0x0040	Выбрать режим обмена 16-битными словами по одному на кадр
	0x0003	RCR21	0x0001	1 бит задержки на прием
	0x0004	XCR11	0x0040	Выбрать режим обмена 16-битовыми словами по одному на кадр
	0x0005	XCR21	0x0001	Передатчик «сдвигает» данные на выход сразу же за спадом на FSX
	0x0006	SRGR11	0x0009	Задаем частоту CLKX=10 МГц при частоте процессора, равной 100 МГц. $f_{CLKX} = f_{CPU} / (CLKGDV + 1)$, где CLKG = SRGR1(7,0)
	0x0007	SRGR21	0x2000	Тактовые импульсы получаем от внутреннего генератора
	0x000E	PCR1	0x0A00	CLKX управляется синхрогенератором. Передача данных происходит на спаде CLKR. При приеме и передаче в синхронизации кадра активный уровень – низкий

вать поадрес регистров контроллера прямого доступа после каждого обращения к ним. Например, при программировании нулевого канала необходимо вначале записать 0x00 в DMSA. Далее первое значение, записанное в DMSDI, попадает в DMSCRC0, второе – в DMDST0, третье – в DMCTR0 и т.д. Регистр DMSDN используется в том случае, если нам необходимо изменить только один регистр. Для реализации прямого доступа необходимо запрограммировать 5 регистров для каждого канала, а также регистр DMPREG и регистр приоритетов. Описания этих регистров даны ниже.

1. DMSRCn, DMDSTn. Данные регистры содержат адреса чтения и записи данных.
2. DMCTRn. Содержит значение, равное количеству оставшихся до завершения итераций.
3. DMSFCn. Регистр выполняет три функции: 1) определяет тип события, запускающего обмен данными, 2) определяет размер слова (16 или 32 бит), 3) определяет количество кадров для передачи.
4. DMMCR. Регистр управляет режимами обмена. Здесь задается инкрементирование/декрементирование адреса источника/приемника, события, по которым контроллер выставляет прерывания ядру процессора, адрес источника и приемника.
5. DMPREC. Регистр отвечает за приоритетность и разрешение каналов, приоритетность прерываний.

Значения регистров приведены в табл. 2.

TLV2548

TLV2548 – 12-разрядный АЦП, имеет 8 каналов ввода, 8-уровневый FIFO; в целях отладки его легко можно подключить к C5402 DSK (стартовый набор разработчика). Способ подключения микросхемы к McBSP процессора показан на рис.3.

TLV2548 может работать в четырех режимах преобразования. Остановимся на одном из них – это одноканальный режим: каждое следующее слово данных АЦП выдает в обмен на управляющее слово. В этом режиме полученные данные должны быть считаны до начала следующего цикла преобразования. Данный режим отличается также большим

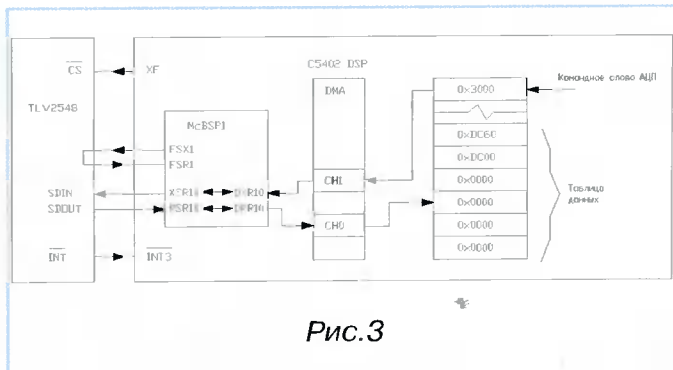


Рис.3

объемом информации, который должен получать АЦП (на одно слово данных должно быть получено одно командное слово). Для большинства приложений это недопустимо, т.к. с относительно медленным преобразователем приходится использовать достаточно быстрый процессор. Выходом из этой ситуации, как уже отмечалось выше, является использование внешнего устройства контроллера прямого доступа в память. Посредством первого канала контроллер получает данные от АЦП через последовательный порт, считывая значения из регистра-приемника McBSP и записывая по адресу, содержащемуся в регистре адреса приемника DMA, после чего инкрементирует адрес в последнем регистре. Таким образом, каждое полученное значение прописывается в отдельную ячейку памяти. Через второй канал контроллер выдает в последовательный порт управляющее слово, при этом, не изменяя адрес источника (управляющее слово одно – прочитать полученные данные и запустить следующий цикл преобразования). Процесс происходит до заполнения входного буфера, после чего ядро процессора получает прерывание, по которому инициируется обработка полученных данных, смена буферов и т.д.

Литература

1. Analog Applications Journal, Texas Instruments, 08.2000, slyt018.
2. TMS320C54X DSP CPU and Peripherals Reference Set, Vol.1, spru 131.
3. TMS320C54X DSP Enhanced Peripherals Reference Set, Vol. 5, spru 302.

Таблица 2. Значения управляющих регистров контроллера DMA

Регистр	Значение	Комментарии
DMSRC0	&DRR11	Канал 0: адрес источника – адрес регистра чтения McBSP
DMDST0	&DataTable	Канал 0: адрес приемника – таблица данных
DMCTR0	NSAMPLES-1	Канал 0: количество итераций считывания – 1
DMSFC0	0x5000	Канал 0: синхронизировать обмен через канал с приемом по McBSP1
DMMCR0	0xC004	Канал 0: адрес источника не изменяется, адрес приемника инкрементируется, генерируется прерывание после передачи блока
DMSRC1	&ADC Cmd	Канал 1: адрес командного слова, передаваемого в АЦП
DMDST1	&DXR11	Канал 1: адрес регистра передаваемых данных McBSP1
DMCTR1	NSAMPLES-1	Канал 1: число, равное количеству раз передачи командного слова АЦП – 1
DMSFC1	0xE000	Канал 1: синхронизировать передачу данных с INT3
DMMCR1	0x8000	Канал 1: адрес источника и приемника во время передачи не изменяется
DMPREC	0x0103	Разрешить каналы 0 и 1. Канал 0 имеет более высокий приоритет – канал 1 инициирует новый цикл преобразования данных в АЦП только после того, как канал 0 принял предыдущие данные

Analog Devices: новые разработки DSP

Настоящий краткий обзор продолжает серию публикаций, посвященных разработкам компании Analog Devices в области DSP-технологий. Обзор инструментальных средств проектирования систем ЦОС был представлен в [1]. Ниже рассматриваются новые архитектурные решения, состояние и перспективы развития цифровых сигнальных процессоров компании Analog Devices. Особое внимание акцентируется на совместной разработке компаний ADI и Intel – семействе сигнальных микроконтроллеров, появление которых открывает новые возможности по эффективному применению ЦСП в области современных телекоммуникаций, а также в других традиционных сферах применения.

В последующих выпусках журнала планируется более подробное описание наиболее оригинальных и перспективных архитектурных разработок компании Analog Devices, и в первую очередь семейства процессоров ADSP-21535 Blackfin.

Сигнальные процессоры

ADSP-21161 – новый недорогой SHARC-процессор

ADSP-21161 – это новейший программируемый 32-битный ЦСП с плавающей точкой семейства SHARC, способный выполнять 600 миллионов математических операций в секунду (MFLOPS), что более чем в три раза превышает производительность подобных моделей той же стоимости. Это новый предел вычислительной мощности недорогих SHARC-процессоров.

ADSP-21161 содержит (рис. 1) 1 Мбит внутренней кристалльной двухпортовой статической памяти (SRAM). Он построен по SIMD-архитектуре, оптимальной для ЦОС-алгоритмов. Подобно всем SHARC-процессорам ADSP-21161 яв-

ляется кодосовместимым со всеми другими членами семейства и поддерживает оба типа данных – как с фиксированной, так и с плавающей точкой. ADSP-21161 устанавливает новый уровень цен на процессоры SHARC с SIMD-архитектурой и является идеальным решением для многих “чувствительных” к стоимости применений.

Характерные особенности:

- 100 МГц (10 нс), SIMD архитектура ядра SHARC;
- 600MFLOPS (данные с плавающей точкой по 32 бита); 600MOPS (данные с фиксированной точкой по 32 бита);
- совместимость кодов с SHARC ADSP-21x6x;
- поддержка IEEE-совместимой 32-битной арифметики с плавающей точкой, 40-битной арифметики с плавающей точкой и 32-битной арифметики с фиксированной точкой;

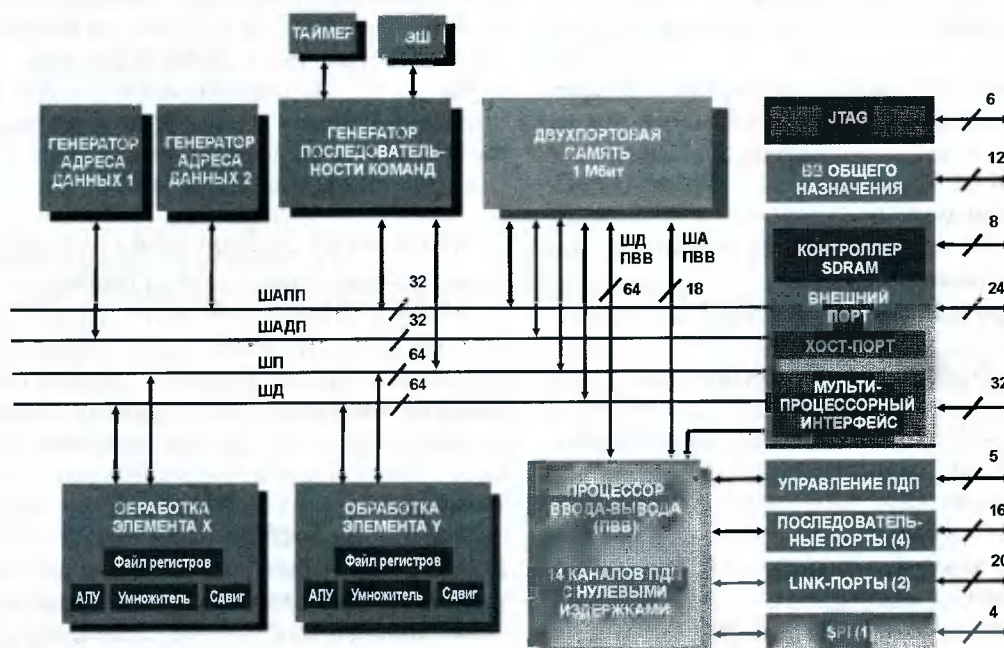


Рис. 1. Архитектура ADSP-21161

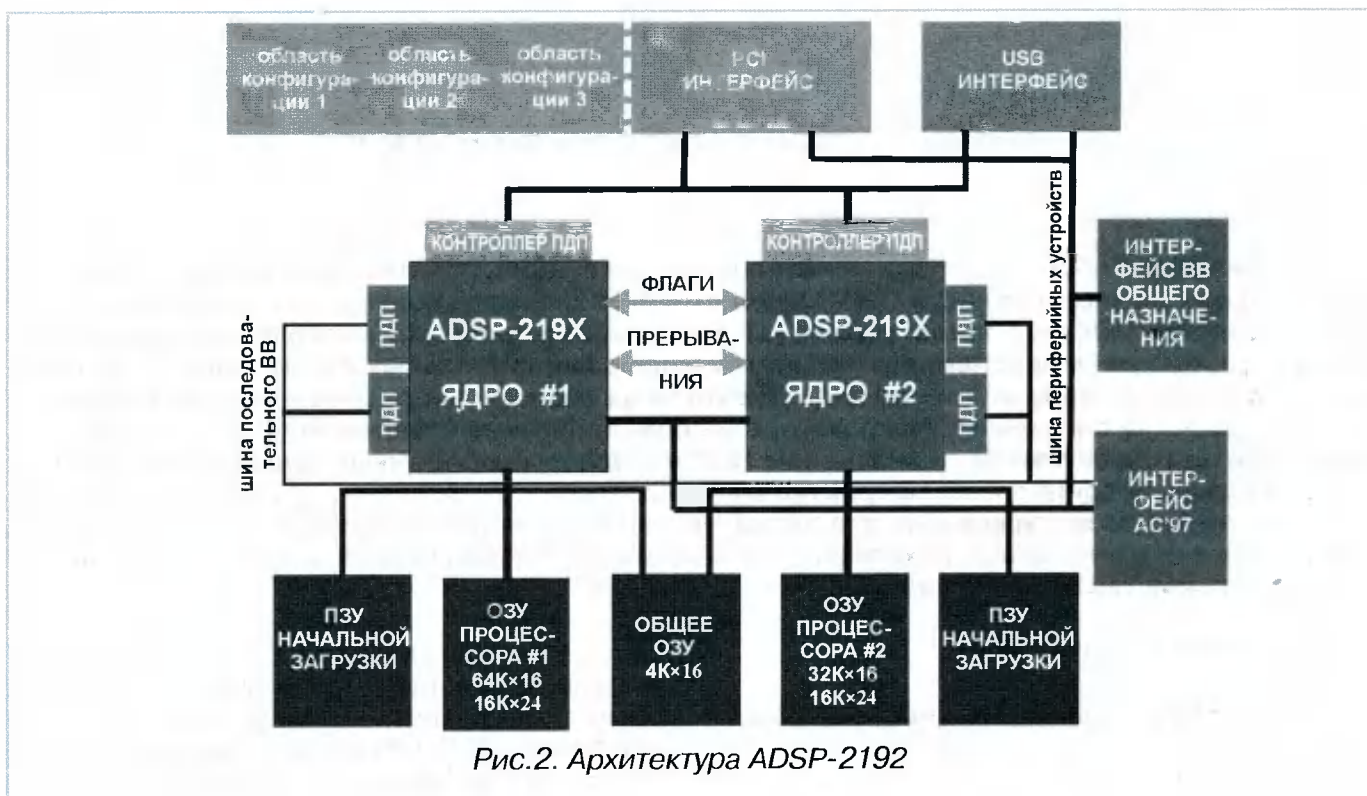


Рис.2. Архитектура ADSP-2192

- ❑ выполнение команд за один цикл с использованием SIMD в обоих вычислительных блоках;
- ❑ 1 Мбит внутренней SRAM с двумя портами;
- ❑ внутрикристалльная скорость передачи данных 2,4 Гбайта/с;
- ❑ 14 каналов DMA с нулевыми задержками;
- ❑ 4 синхронных последовательных порта;
- ❑ последовательные порты поддерживают 128 каналов с временным разделением (TDM);
- ❑ интегрированная поддержка внешней памяти SDRAM и SPSDRAM;
- ❑ логика "упаковки" обеспечивает доступ к данным и командам из 8-, 16-, 32-битной внешней памяти;
- ❑ поддержка двух методов интегрирования мультипроцессорных вычислений (внешняя логика не требуется);
- ❑ два link-порта на 100 Мбайт/с;
- ❑ SPI-совместимый интерфейс;
- ❑ алгебраический синтаксис ассемблера для удобства программирования;
- ❑ поддержка программирования Си/Си++ компилятором.

Семейство ADSP-21161 поддерживается полным набором программных и аппаратных средств отладки: VisualDSP, генераторами кодов, EZ-KIT-Lite, симуляторами, эмуляторами.

Области применения:

- ❑ распознавание речи;
- ❑ профессиональное аудио высокого качества;
- ❑ аудио в автомобилях;
- ❑ распознавание отпечатков пальцев;
- ❑ цифровое радиовещание;
- ❑ беспроводные телекоммуникации;
- ❑ управление двигателями;

- ❑ медицинское оборудование;
- ❑ телефония;
- ❑ тестовое оборудование.

Дальнейшее развитие семейства идет по двум направлениям: повышение производительности вычислительного процесса для мультипроцессорных систем и обеспечение оптимального соотношения цена/производительность. К первому направлению относится разработка процессора SHARC на 10 GFLOPS. Его внутрикристалльная память достигнет беспрецедентного уровня 64 Мбит. Дальнейшая оптимизация соотношения цена/производительность приведет к появлению SHARC-процессора с вычислительной производительностью до 1200MFLOPS при уменьшении цены до \$5. Это необходимо, чтобы поддержать внедрение новых технологий, которые требуют от вычислительных устройств высокой эффективности при приемлемой для потребителя цене.

ADSP-2192 – высокоэффективный ЦСП для применения в моноканальных системах

ADSP-2192 – это 16-битный ЦСП с фиксированной запятой, имеющий двойное операционное ядро, оптимизированное для многоканальных систем цифровой обработки речи и других применений, требующих высокой скорости работы. Выполненный с использованием КМОП-технологии и выполняющий команды (машинный цикл) за 6,25 нс (160 МГц/320MIPS), ADSP-2192 обеспечивает высокую эффективность вычислений при малой потребляемой мощности. Имея два операционных ядра (рис.2) и обладая полным набором команд, процессор выполняет сложные вычисления параллельно: любые две команды могут быть исполнены за один цикл

процессора. ADSP-2192 – это первый процессор серии ADSP-219x, работающий с производительностью 320 миллионов операций в секунду.

Полностью совместимый своими кодами с семейством ADSP-218x, новый процессор использует базовую архитектуру ADSP-2100, включающую три вычислительных блока, генераторы адресов данных и генератор последовательности команд. Процессор поддерживает широкий ряд стандартов интерфейса: межсоединения периферийных устройств (PCI) в режимах CardBus и Sub-ISA для технологий встраиваемых ЦСП, универсальную последовательную шину (USB) для ППЗУ, интерфейс AC'97 и прямой доступ к памяти DMA. Другие особенности архитектуры ADSP-2192: программируемый таймер для каждого ядра и возможности расширенного обслуживания прерываний с использованием внутрикристалльной памяти программ и данных.

Процессор P0 первого ядра ADSP-2192 имеет 132 К слов внутрикристалльной памяти, сконфигурированных, как (64 К по 16 бит)-памяти данных и (16 К по 24 бит)-памяти программ. Процессор P1 рассчитан на (32 К по 16 бит)-памяти данных и (16 К по 24 бит)-памяти программ. Каждый из процессоров работает со своим аппаратным загрузчиком. Дополнительно в состав ADSP-2192 входит блок памяти (4 К по 16 бит), используемый совместно двумя ядрами.

Характерные особенности:

- 16 бит; 160 МГц/320MIPS; двойное операционное ядро;
- межпроцессорная связь;
- кодовая совместимость с семейством ADSP-218x;

- 2,4 Мбит статической памяти (SRAM) на кристалле;
- внутрикристалльное ПЗУ начальной загрузки;
- интерфейс PCI 2.2 (33 МГц/32 бита);
- контроль по шине PCI в режиме “управляющий – управляемый”;
- режим интерфейса Sub-ISA для технологий встроенных ЦСП;
- поддержка интерфейса USB 1.1;
- интерфейс AC'97 rev 2.1 – удобный для подключения внешних аудиосистем, модемов и кодеков мобильных устройств;
- 14 каналов DMA (прямого доступа к памяти);
- 8 выводов “Вход/Выход” общего назначения;
- последовательный интерфейс для внешних ППЗУ/СПЗУ;
- алгебраический синтаксис ассемблера для удобства программирования;
- поддержка программирования Си-компилятором.

Специальная схемотехника, контролирующая энергопотребление, обеспечивает возможность использования этих процессоров в носимых устройствах с питанием от батарей. ADSP-2192 – выпускается в LQFP-корпусе с 144 выводами.

Области применения:

- устройства интегрального доступа;
- SOHO-телефония;
- системы сбора информации;
- мультирежимные модемы;
- IP-факс/телефония;
- голосовые ATM-технологии;
- системы голосовой почты;
- удлинительные учреждений АТС;
- эхо-компенсаторы.



Рис.3. Архитектура ADSP-2191

Семейство ADSP-2192 поддерживается полным набором программных и аппаратных средств отладки: VisualDSP, генераторами кодов, EZ-KIT-Lite, симуляторами, эмуляторами.

Новые продукты серии ADSP-219x разрабатываются по двум направлениям: первое – повышение вычислительной производительности до 1200MIPS; второе – понижение мощности потребления до уровня, не превышающего 0,1мВт/MIPS.

ADSP-218x серии N – процессоры с пониженным уровнем расхода энергии

Компания Analog Devices расширила уже заслужившее доверие семейство сигнальных процессоров ADSP-218x новой серией 'N'. Процессоры ADSP-218xN отличаются низким потреблением мощности и невысокой стоимостью. При этом обеспечена кодовая совместимость со всеми продуктами, базирующимися на платформе ADSP-21xx, и совместимость по выводам с семейством ADSP-218x, что защищает тысячи систем на базе ADSP-21xx от морального износа. Процессоры серии ADSP-218xN работают от источника питания напряжением 1,8 В и характеризуются расходом энергии 0,55 мВт/MIP. Внутреннее статическое ОЗУ имеет объем от 320 Кбит до 2 Мбит. Распределение объемов памяти программ/память данных: от 4 К/4 К до 48 К/56 К. Цена варьируется от \$4 до 20. Серия ADSP-218xN является идеальным решением для обычных сфер применения DSP, оборудования телекоммуникаций, проводных и беспроводных систем и в особенности там, где требуется пониженный расход энергии питающих батарей, как, например, для PDA (персональных секретарей) и других портативных устройств.

ADSP-2191 – сигнальный процессор удвоенной производительности, предназначенный для телекоммуникационного оборудования

ADSP-2191 – это 16-разрядный цифровой сигнальный процессор с рабочей частотой 160 МГц, совместимый по своим кодам с семейством ADSP-218x и обладающий вычислительной производительностью, вдвое превышающей показатели моделей этого семейства. Сконструированный для применения в телекоммуникационном оборудовании, процессор ADSP-2191 имеет встроенные (внутрикристалльные) интерфейсы взаимодействия с T1, E1 и H.100-системами с высокой плотностью передачи информации, в которых ЦСП служит для сжатия и повышения качества речи. Построенный на базе ядра ADSP-219x, (рис.3) процессор имеет свойственные для этого семейства средства поддержки алгоритмов ЦОС, алгебраический синтаксис ассемблера и возможности исполнения всех команд за один цикл, характерные для всех продуктов ADSP-21xx в целом. Кодовая совместимость позво-

лит программистам при работе с ADSP-2191 использовать библиотеки программ, разработанные ранее для предыдущих моделей. Данный процессор может быть использован службами доступа к сетям (провайдерами) для создания средств повышения качества передачи голоса, что оказывается необходимым на протяженных линиях, в сетях, имеющих шлюзы или в частных сетях. Кроме того, с полным набором программных модулей, разработанных для ADSP-219x в соответствии с рекомендациями Международного союза электросвязи, процессор ADSP-2191 становится оптимальным для изготовителей оборудования, позволяя им обеспечивать службы провайдеров качественными и эффективными в стоимостном отношении платформами обработки и передачи речевой информации следующего поколения. Применительно к телефонии данный процессор может быть использован при построении эхокомпенсаторов, учрежденческих АТС с межсетевым интерфейсом, ADPCM-транскодеров (транскодеров адаптивной дифференциальной импульсно-кодовой модуляции); шлюзов VoIP; серверов удаленного доступа и DSL-оборудования центральных телефонных станций.

ADSP-2191 содержит 64 К слов внутрикристалльной SRAM, сконфигурированной на 32 К слов памяти данных и 32 К слов памяти программ, и имеет возможность доступа к внешней памяти объемом до 16 М слов. Процессор включает три последовательных порта, поддерживающих временное разделение каналов (до 128 каналов на порт), два порта последовательного интерфейса с периферией (SPI-порты), контроллер прямого доступа к памяти, один порт универсального асинхронного приемопередатчика (UART), программируемые выходы флагов общего назначения и 8- или 16-разрядный порт интерфейса с хост-системой. ADSP-2191 имеет четыре режима работы на малой мощности, которые значительно сокращают рассеивание энергии, в моменты, когда устройством находится в состоянии ожидания. Это позволяет существенно уменьшить расход энергии в системах телекоммуникаций и в целом затраты на систему за счет устранения необходимости в дополнительных цепях питания.

Процессор ADSP-2191 поддерживается широким рядом программных и аппаратных инструментов разработки. Эти средства включают интегрированную программную среду проектирования, макетную плату и JTAG-эмуляторы. Программной средой проектирования является последняя версия VisualDSP++, которая в настоящий момент оценивается в \$2995. Макетная плата ADDS-2191-EZLITE имеет ориентировочную стоимость \$295 и должна быть доступна с июня 2001 г. Сам процессор будет стоить около \$15 за штуку при заказе от 10 000 шт. и может быть приобретен в конце 2001 – начале 2002 г. (модель



ADSP-2191MKST160). Инженерные образцы этих процессоров можно заказать прямо сейчас.

Инструментальные средства проектирования

VisualDSP++ версии 2.0

Analog Devices представила новую версию программной среды проектирования систем на базе DSP – VisualDSP++ 2.0. К нововведениям следует отнести:

- включение инструмента VisualDSP++ Kernel (VDK), позволяющего реализовывать коды процессоров в структурированном, удобном для внесения изменений виде, что обеспечивается наличием управляемого ядра, используемого приложениями;
- объединение среды проектирования и отладчика в рамках одного интерфейса;
- дополнение пакета средствами трехмерного отображения.

Инструмент VDK осуществляет автоматическое распределение и управление ресурсами с учетом временных ограничений и ограничений по памяти при программировании DSP. Такое средство позволяет разработчикам создавать коды с большей эффективностью, снимая необходимость программирования нового устройства с самого начала. VDK включает механизм организации потоков, выявления критических и нерегламентированных секций, а также семафоры и флаги событий и устройств. Кроме того, VDK поддерживает приоритетное коллективное обслуживание и распределение ресурсов с временным разделением. VisualDSP++ Kernel является средством, “масштабируемым” под требования конкретной задачи. Если разрабатываемая система не нуждается в определенной функции, автоматически предоставляемой VDK, то коды этой функции могут быть легко удалены. Средство VisualDSP++ Kernel является интегрированным в среду проектирования и может использоваться через стандартный интерфейс командной строки.

VDK представляет собой библиотеку услуг, и разработчик решает сам, использовать ее или нет. Если VDK используется, то среда проектирования помогает разработчику справиться со многими задачами, часто вызывающими ошибки, и с организацией управления системными ресурсами, автоматически создавая различные базовые объекты VDK и отображая состояние системы при ее отладке.

Среда проектирования и отладчик системы объединены в рамках одного интерфейса. Это позволяет выполнять все этапы разработки в едином интерфейсе, не требуя дополнительных переключений. Средства управления рабочей областью, предоставляемые VisualDSP++, позволяют разработчику создавать столько рабочих областей, сколько ему нужно, и быстро переключаться между ними.

Средства отображения среды VisualDSP++ предоставляют возможность визуально оценить

эффективность разрабатываемой программы. Чтобы сделать эти средства максимально удобными, в пакет был добавлен ряд возможностей отображения в трехмерном пространстве, включая индикаторные диаграммы, отображение созвездий и другие.

VisualDSP++ 2.0 будет применима к процессорам семейства ADSP-219x. Она включает интегрированную среду проектирования, отладчик, VDK, Си/Си++ компилятор, ассемблер, линкер, загрузчик, сплиттер, а также средства поддержки эмуляции и симуляции. Данная версия будет поддерживать процессоры семейства ADSP-218x, предоставляя для них такую же проектировочную среду и компилятор языка Си. Различные справки по пакету, документацию и демонстрационные программы можно получить по каналам Internet.

Революционная архитектура обработки сигналов века Internet

Совместная разработка ADI и Intel – сигнальный микроконтроллер

Вычислительная техника и средства телекоммуникаций в своем развитии достигли переломного момента, когда грань между ними стирается и появляется необходимость обеспечивать широкие массы потребителей недорогими, малогабаритными, мобильными и надежными цифровыми устройствами, способными принимать, обрабатывать и передавать огромные объемы информации в реальном времени. Обязательным стандартом, требуемым пользователем, становится обеспечение его полной мультимедийной информацией: высококачественными речью, звуком, видеоизображением. Вместе с тем промышленность начинает ориентироваться уже не на проводные Internet-технологии, а на выпуск интерактивных мобильных устройств следующего поколения, таких, как 3G-телефоны, web-терминалы, двусторонние пейджеры, персональные секретари (PDAs), цифровые камеры, цифровые аудиоплееры, электронные игры. Эти устройства должны будут работать с качественной аудио- и видеoinформацией и использовать преобразования “речь – текст”, “текст – речь”, в том числе распознавание рукописных текстов, что требует высочайшей вычислительной производительности.

Таким образом, возникает необходимость в недорогих, мощных процессорах с малой потребляемой мощностью, архитектура которых оптимизирована для беспроводной передачи и приема данных в широкой полосе частот. Традиционные микропроцессоры, способные работать с необходимым потоком информации, не отвечают этим условиям: они дороги, потребляют большую мощность, не могут работать на аккумуляторных батареях долгое время, излучают большое количество тепловой энергии. Реше-

нием, казалось бы, является применение цифровых сигнальных процессоров (ЦСП), оптимальных для обработки сигналов в реальном времени, способных работать с огромными объемами информации, но они также требуют большой мощности потребления.

Уже долгое время развитие технологий ЦСП идет по пути простого усовершенствования базовых архитектур, построенных более 10 лет назад. Требуются новые технологии, новые архитектурные решения. В 1999 г. компании ADI и Intel совместно организовали исследовательскую работу многих независимых разработчиков программного и аппаратного обеспечения. Задача была разработать надежную ЦСП-архитектуру, совмещающую в себе высокие вычислительные способности сигнальных процессоров и простоту программирования, характерную для обычных микропроцессоров и микроконтроллеров.

5 декабря 2000 г. компании Intel и Analog Devices представили результат своей совместной деятельности – новую архитектуру – сигнальный микроконтроллер (Micro signal Architecture). Новая архитектура совмещает в себе высокую вычислительную производительность классического ЦСП и свойства, более характерные для микроконтроллеров, например оптимизацию программ, написанных на языках высокого уровня, защиту памяти, байтовую адресацию. В результате одно операционное ядро способно обеспечить решение сложных задач цифровой обработки сигналов и простое решение задач управления, что исключает необходимость в дополнительных процессорах.

Обладая вычислительной производительностью 2000MIPS и системой динамического управления потреблением энергии, которая увеличивает срок службы питающих батарей в 10 раз, разработанная архитектура идеально подходит для использования в портативных устройствах доступа в

Internet, цифровых камерах, видеотелефонах, сотовой 3G-связи. Она также применима в промышленных контроллерах, системах электросвязи, автомобилестроении и в бытовой электронике, где требуется высокоинтенсивная обработка сигналов при ограничении потребления мощности.

Ядро сигнального микроконтроллера построено на основе базовой архитектуры, которая объединяется с рядом блоков: счетчиком событий, таймером, блоком управления памятью, блоком мониторинга вычислений, статической памятью (SRAM), кэш-памятью и JTAG-интерфейсом. При необходимости можно использовать также дополнительную периферию: последовательные порты, хост-порты, расширенную память, микшеры сигналов, специальные схемы, необходимые в том или ином конкретном случае.

Прикладные программы пишутся на языках Си/Си++, при этом специальные средства указывают области программы, где для большей эффективности алгоритма может потребоваться введение кодов ассемблера. Это ускоряет и делает более удобным процесс написания программ. Высокая вычислительная производительность обеспечивается наличием двух умножителей-накопителей и двух АЛУ, данные в которые могут поступать в каждом цикле, что существенно усовершенствует архитектуру классического ЦСП, а также набором RISC-подобных команд, позволяющих увеличить тактовую частоту. Малое потребление энергии достигается за счет использования динамической системы управления питанием, которая осуществляет непрерывное регулирование напряжения и частоты работы процессора, оптимизируя расход мощности и вычислительную производительность в системах реального времени.

Все архитектуры данного класса будут содержать в ядре процессора умножители-накопители, АЛУ и сдвигатели, однако количество и состав

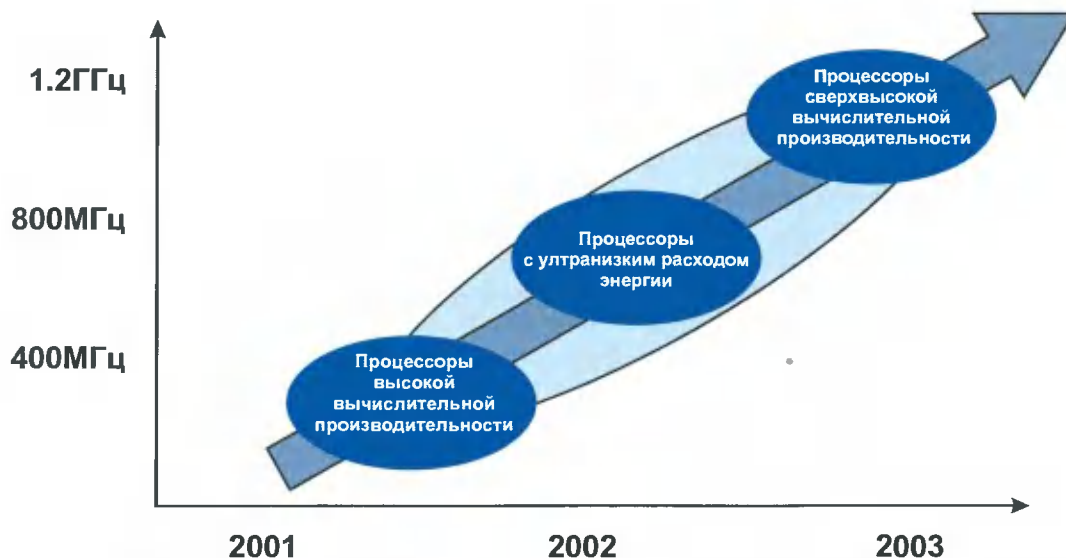


Рис. 4. Развитие продуктов на базе сигнального микроконтроллера



этих блоков будут меняться от одного процессорного ядра к другому. Первые процессоры семейства содержат два множителя-накопителя, два АЛУ и один сдвигатель. Множители-накопители выполняют умножение 16 на 16 бит в каждом цикле с накоплением результата размером 40 бит. АЛУ оперируют с 8-, 16-, 32- или 40-битными данными и за один цикл способны выполнить:

- две 32-битных операции или
- четыре 16-битных операции или
- четыре 8-битных операции.

40-битный сдвигатель осуществляет сдвиг, циклический сдвиг, нормализацию, выборку и запись данных. Данные для вычислительных блоков хранятся в многопортовом файле регистров на 16 входов по 16 бит или на 8 входов по 32 бита.

Команды процессора оптимизированы так, что 16-битный формат употребляется наиболее часто, что обеспечивает компактность кодов после компиляции, а 32- и 64-битные многофункциональные команды позволяют программисту задействовать сразу ряд ресурсов процессора. Автоматическая сортировка команд обеспечивает непрерывное заполнение памяти программ в любой последовательности без каких-либо пропусков. Команды процессора имеют алгебраический синтаксис и легко читаемы программистом даже на уровне ассемблера.

Сигнальный микроконтроллер использует Гарвардскую архитектуру ядра в сочетании с иерархической структурой памяти:

- уровень 1 (L1) – области памяти самого ядра;
- уровень 2 (L2) – другие области памяти кристалла и внешняя память.

Уровень L1 включает блок памяти программ, содержащий только команды, два блока памяти данных и блок сверхоперативной памяти, используемой как стек и память локальных переменных. Уровень L2 включает блок памяти программ и данных емкостью 4 Гбайта. Память команд и память данных уровня L1 могут быть сконфигурированы как SRAM или кэш-память. Блок управления памятью обеспечивает защиту хранящихся данных для индивидуального пользования и предотвращает непреднамеренное искажение информации.

Сигнальный микроконтроллер предоставляет два режима работы: пользовательский и администраторский. Первый имеет ограниченный доступ к некоторым системным ресурсам, обеспечивая защиту программного окружения. Второй отличается неограниченным доступом к ресурсам ядра.

Для оценки производительности служит специальный блок, содержащий пункты просмотра адресов команд и адресов данных, которые разбиваются по парам и образуют массив просмотра адресов. Пункты просмотра объединяются с различными счетчиками и дают возможность провести оценку производительности процессора. Кроме того, буфер трассировки сохраняет "непоследовательные" состояния счетчика ко-

манд так, чтобы вся предыдущая последовательность команд процессора могла быть реконструирована.

В настоящий момент компаниями ADI и Intel ведется освоение анонсированной архитектуры и создание на ее основе новых продуктов. Analog Devices представила план развития процессоров на базе сигнального микроконтроллера (рис. 4). Разработки обеих компаний будут характеризоваться совместимостью исходных программных и объектных кодов, чтобы дать возможность пользоваться ими клиентам и той и другой стороны.

Analog Devices представила первый DSP, демонстрирующий возможности новой архитектуры, разработанной совместно с Intel.

Компания Analog Devices открывает новое семейство 16-разрядных цифровых сигнальных процессоров Blackfin на основе архитектуры сигнального микроконтроллера, разработанного совместно с Intel. Первый процессор семейства Blackfin, разработанный специально для применения в области телекоммуникаций и в ряде Internet-устройств, сокращает время программного и аппаратного проектирования благодаря удобной для программиста архитектуре и встроенным элементам интерфейса. Чтобы максимизировать эффективность динамического управления мощностью, в архитектуру может быть добавлен сопутствующий кристалл управления энергией, осуществляющий точное регулирование напряжения на процессоре. Такие меры позволяют уменьшить энергетические затраты более чем на 60%.

Новый процессор работает на частоте 300 МГц с производительностью 600MMACs (миллионов операций умножения с накоплением в секунду) при мощности потребления 42 мВт для напряжения 0,9 В.

Процессор Blackfin ADSP-21535 – это высокоинтегрированный, мощный процессор, предназначенный для видеосистем и устройств, ориентированных на использование в Internet. К его применениям можно отнести видеотелефоны, электронные игры, web-терминалы, сетевые телевизоры (Net TVs) и переносные интеллектуальные устройства. Процессор имеет интерфейсы PCI и USB. USB – это стандартный интерфейс, служащий для связи мобильного устройства и стационарной системы. Интерфейс PCI обычно служит для ПК-контроля системы в оборудовании PBX-телекоммуникаций и заводских автоматических систем, в робототехнике и медицинском оборудовании, в компьютерных устройствах печати и многочисленных тестирующих и измеряющих системах.

Процессор содержит 2,4 Мбит памяти SRAM и уровень памяти, конфигурируемой как SRAM или кэш. Наличие обширной внутрикристалльной памяти минимизирует количество обращений к внеш-

ней памяти, замедляющих работу и увеличивающих расход энергии. Ввод данных осуществляется через АЦП, а вывод через ЦАП и четыре последовательных порта, расположенных на кристалле.

Команды архитектуры позволяют обрабатывать поток данных полной мультимедийной информации, демонстрируя производительность, в 10 раз превышающую возможности других процессоров по обработке таких потоков данных. Процессор содержит блоки видеоарифметики и логики (видео-АЛУ), которые выполняют до четырех 8-разрядных математических действий за один тактовый цикл. Это позволяет создавать программы, эффективно реализующие сжатие изображений и алгоритмы оценки перемещения и кодирования Хаффмана, которые широко используются стандартами обработки изображений и видео MPEG2, MPEG4 и JPEG.

Процессор ADSP-21535 Blackfin – это первый процессор, имеющий систему динамического управления энергопотреблением, которая осуществляет независимое регулирование напряжения и частоты, минимизируя затраты мощности для каждой конкретной решаемой процессором задачи. Специальный вспомогательный кристалл управления, разработанный Analog Devices, регулирует напряжение и управляет зарядом аккумуляторных батарей. Организация регулирования мощности основана на подобных методах, используемых в ПК типа ноутбук, использующих динамическое управление напряжением и технологию ADOPT. Система динамического управления энергопотреблением уменьшает затраты мощности, осуществляя расширенный тактовый контроль – пять режи-

мов пониженной мощности ядра процессора и возможность независимого перевода любого функционального блока в составе ядра процессора, блоков памяти и периферии в состояние пониженного питания.

Две модели процессоров Blackfin должны стать доступны в сентябре 2001 г. – первом квартале 2002 г. Процессор ADSP-21535PKCA-200, имеющий частоту 200 МГц, имеет ориентировочную стоимость \$27 при заказе от 10 000 шт. ADSP-21535PKCA-300 с частотой 300 МГц имеет цену \$34 при таком же объеме заказа.

Литература

1. *Витязев С.В.* Стратегия компании Analog Devices в области DSP: обзор новых средств проектирования систем цифровой обработки сигналов. Цифровая обработка сигналов, 2001, №1.

Примечание. Все цены на упоминаемые в данной статье изделия приведены на условиях FOB-США и не включают налоги и расходы на транспортировку.

Всю необходимую справочную информацию по процессорам обработки сигналов фирмы Analog Devices можно получить у официального дистрибьютера Analog Devices на территории России, ЗАО “АРГУССОФТ Компани” Тел.:(095) 217-2505, 217-2519, факс (095) 216-6642.

E-mail : components@argussoft.ru
<http://www.argussoft.ru>

ПРИЛОЖЕНИЕ

НОВЫЕ КНИГИ

Шелухин О.И., Лукьянцев Н.Ф. Цифровая обработка и передача речи \ Под ред. О.И. Шелухина. М.: Радио и связь, 2000. 456 с.

Рассматриваются проблемы цифровой обработки и передачи речи в системах со сжатием, статистическим уплотнением, пакетной коммутацией, IP-телефонии, сетях ATM и Frame Relay. Анализируются принципы построения, характеристики и особенности функционирования кодеров формы, вокодеров, гибридных кодеров, реализующих алгоритмы CELP, LD-CELP, ACELP, MBE, MBEV, протоколы и кодеры IP-телефонии, особенности IP-мультикастинга; методы построения и характеристики кодеров мобильных радиосистем стандартов GSM, TETRA, APCO-25 INMARSAT-M; построение корпоративных сетей спутниковой связи VSAT в режимах DAMA и PAMA; системы множественного доступа: TDMA, FDMA, ALOXA, S-ALOXA; технологии пакетной передачи информации в корпоративных сетях и IP-телефонии; построение систем асимметричного доступа в Internet.

Для широкого круга специалистов в области радиосвязи, информатики и цифровой обработки речевых сигналов.

Прокис Джон. Цифровая связь. Пер. с англ. \ Под ред. Д.Д. Кловского М.: – Радио и связь, 2000. – 800 стр.

В систематизированном виде излагаются основные сведения по цифровым системам связи: элементам теории стохастических процессов и кодирования, вопросам определения и оценки параметров систем связи и их сигналов, блочным и сверточным канальным кодам, каналам с ограниченной полосой пропускания и линейным фильтрам, адаптивной коррекции, системам с мультинесущей, многоканальным пользовательским системам.

Для специалистов в области связи и цифровой обработки сигналов.

В.Б. Стешенко

Система проектирования ПЛИС фирмы ALTERA

В предыдущей статье были представлены основные архитектуры ПЛИС, дана информация о фирмах-производителях и приведены характеристики свободно распространяемой через Internet версии MAX+PLUS II Baseline. В данной статье рассматриваются работа над проектом в среде системы MAX+PLUS II фирмы ALTERA и основные черты САПР 4-го поколения фирмы ALTERA – системы Quartus.

Название системы MAX+PLUS II является аббревиатурой от Multiple Array Matrix Programmable Logic User System. Система MAX+PLUS II имеет средства удобного ввода проекта, компиляции и отладки, а также непосредственного программирования устройств (рис. 1).

Процедуру разработки нового проекта от концепции до завершения можно упрощенно представить следующим образом:

- создание нового файла (design file) проекта или иерархической структуры нескольких файлов проекта с использованием различных редакторов разработки проекта в системе MAX+PLUS II, т.е. графического, текстового и сигнального редакторов;
- задание имени файла проекта верхнего уровня (Top of hierarchy) в качестве имени проекта (Project name);
- назначение семейства ПЛИС для реализации проекта (пользователь может сам назначить конкретное устройство или пред-

ставить это компилятору для того, чтобы оценить требуемые ресурсы);

- открытие окна компилятора (**Compiler**) и его запуск нажатием кнопки **Start** для начала компиляции проекта (по желанию пользователя можно подключить модуль извлечения временных задержек Timing SNF Extractor для создания файла разводки, используемого при тестировании и анализе временных параметров);

- в случае успешной компиляции возможно тестирование и временной анализ, для проведения которого необходимо выполнить следующие действия:

- для проведения временного анализа открыть окно **Timing Analyzer**, выбрать режим анализа и нажать кнопку **Start**;
- для проведения тестирования нужно сначала создать тестовый вектор в файле канала тестирования (.scf), пользуясь сигнальным редактором, или в файле вектора (.vec), пользуясь текстовым редакто-

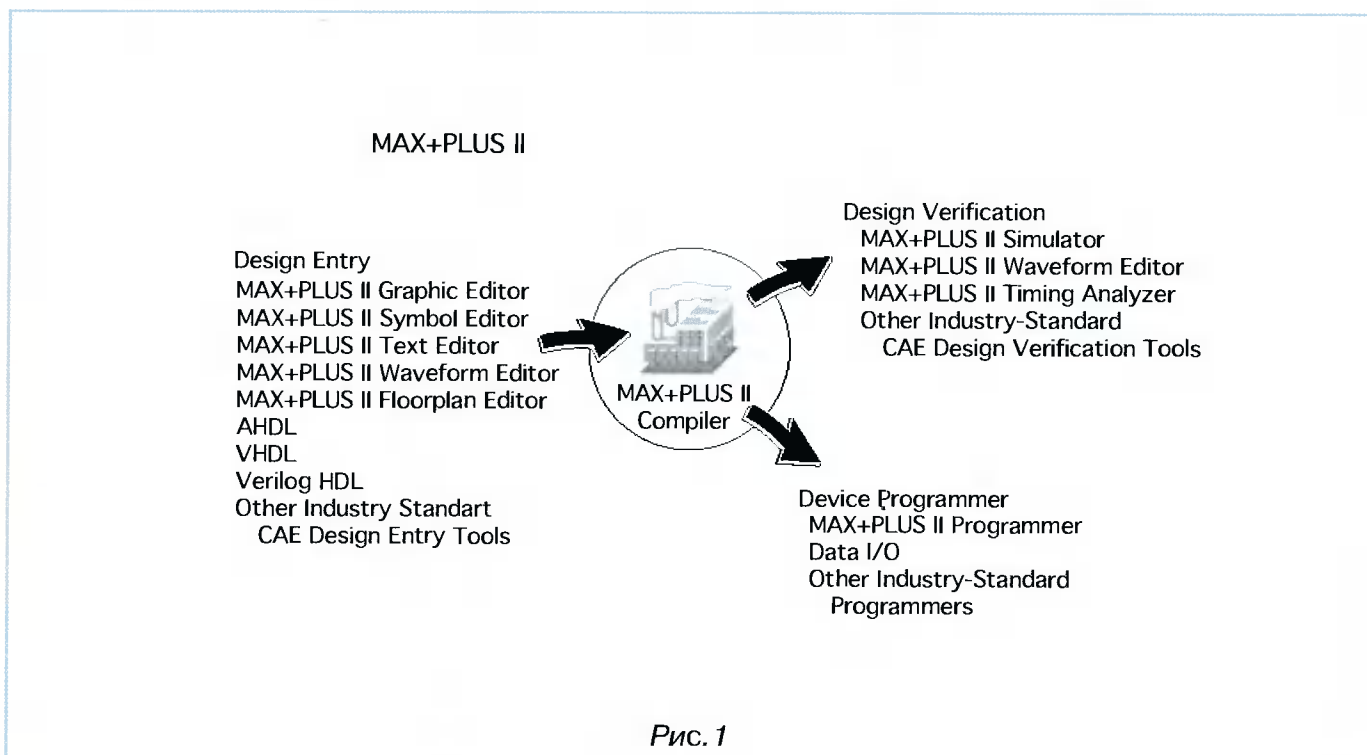


Рис. 1

Таблица 1

Приложение	Выполняемая функция
Hierarchy Display	Обзор иерархии – отображает текущую иерархическую структуру файлов в виде дерева с ветвями, представляющими собой подпроекты.
Graphic Editor	Графический редактор – позволяет разрабатывать схемный логический проект в формате реального отображения на экране WYSIWYG.
Symbol Editor	Символьный редактор – позволяет редактировать существующие символы и создавать новые.
Text Editor	Текстовый редактор – позволяет создавать и редактировать текстовые файлы логического дизайна, написанные на языках AHDL, VHDL и Verilog HDL.
Waveform Editor	Сигнальный редактор – выполняет двойную функцию: инструмент для разработки дизайна и инструмент для ввода тестовых векторов и наблюдения результатов тестирования.
Floorplan Editor	Поуровневый планировщик – позволяет графическими средствами устанавливать назначения контактов устройства и ресурсов логических элементов.
Compiler	Компилятор – обрабатывает логические проекты.
Simulator	Симулятор – позволяет тестировать логические операции и внутреннюю синхронизацию проектируемой логической цепи.
Timing Analyzer	Временной анализатор – анализирует работу проектируемой логической цепи после того, как она была синтезирована и оптимизирована компилятором.
Programmer	Программатор – позволяет программировать, конфигурировать, проводить верификацию и тестировать ПЛИС фирмы Altera.
Message Processor	Генератор сообщений – выдает на экран сообщения об ошибках, предупреждающие и информационные сообщения.

ром, а. затем открыть окно отладчика – симулятора (**Simulator**) и нажать кнопку **Start**;

- программирование или загрузка конфигурации выполняются путем запуска модуля программатора (**Programmer**) с последующей установкой устройства MPU (Master Programming Unit) или подключением устройств MasterBlaster, BitBlaster, ByteBlaster и кабеля загрузки FLEX (FLEX Download Cable) к устройству, программируемому в системе;
- выбор кнопки **Program** для программирования устройств с памятью типа EPROM или EEPROM (MAX, EPC) либо выбор кнопки **Configure** для загрузки конфигурации устройства с памятью типа SRAM (FLEX).

На рис.2 представлено окно пакета MAX+PLUS II с загруженной схемой.

Программное обеспечение (ПО) системы MAX+PLUS II содержит 11 приложений и главную управляющую программу. Различные приложения, обеспечивающие создание проекта, могут быть активизированы мгновенно, что

позволяет пользователю переключаться между ними щелчком мыши или с помощью команд меню. В это же время может работать одно из фоновых приложений, например компилятор, симулятор, анализатор синхронизации или программатор. Одни и те же команды разных приложений работают одинаково, что облегчает задачу разработки логического дизайна.

В табл. 1 приведено описание приложений.

Перед тем как начать работать в системе MAX+PLUS II, следует понять разницу между файлами проекта, вспомогательными файлами и проектами.

Файл проекта – это графический, текстовый или сигнальный файл, созданный с помощью графического или сигнального редакторов системы MAX+PLUS II или в любом другом используемом промышленные стандарты схемном или текстовом редакторе, либо при помощи программы **netlist writer**, имеющейся в пакетах, поддерживающих EDIF, VHDL и Verilog HDL. Этот файл содержит логику для проекта

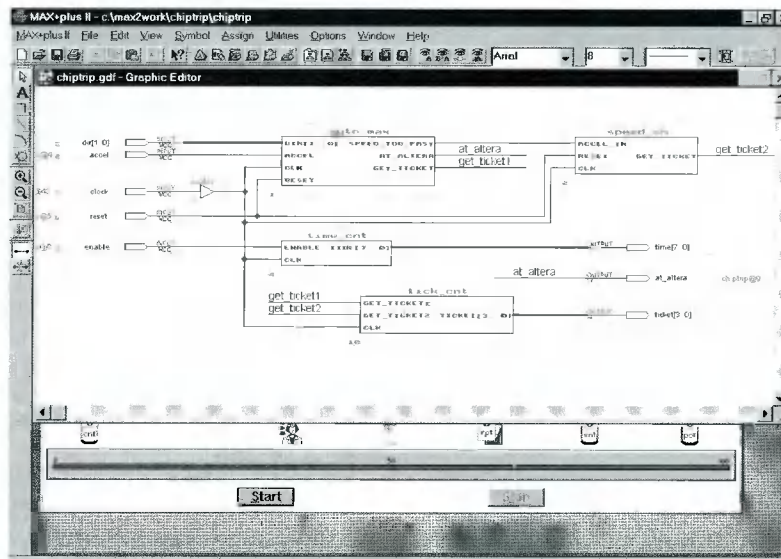


Рис. 2. Главное окно системы MAX+PLUS II

MAX+PLUS II и транслируется компилятором. Компилятор может автоматически обрабатывать следующие файлы проекта: графические файлы проекта (**.gdf**); текстовые файлы проекта на языке AHDL (**.tdf**); сигнальные файлы проекта (**.wdf**); файлы проекта на языке VHDL (**.vhd**); файлы проекта на языке Verilog (**.v**); схемные файлы OrCAD (**.sch**); входные файлы EDIF (**.edf**); файлы формата Xilinx Netlist (**.xnf**); файлы проекта Altera (**.adf**); файлы цифрового автомата (**.smf**).

Вспомогательные файлы – это файлы, связанные с проектом MAX+PLUS II, но не являющиеся частью иерархического дерева проекта. Большинство таких файлов не содержит логики проекта. Некоторые из них создаются автоматически приложением системы MAX+PLUS II, другие – пользователем. Примерами вспомогательных файлов являются файлы назначений и конфигурации (**.acf**), символьные файлы (**.sym**), файлы отчета (**.rpt**) и файлы тестовых векторов (**.vec**).

Проект состоит из всех файлов иерархической структуры проекта, в том числе вспомогательных и выходных файлов. Именем проекта является имя файла проекта верхнего уровня без расширения. Система MAX+PLUS II выполняет компиляцию, тестирование, анализ синхронизации и программирование сразу целого проекта, хотя пользователь может в это время редактировать файлы этого проекта в рамках другого проекта. Для каждого проекта желательно создавать отдельный подкаталог в рабочем каталоге системы MAX+PLUS II.

В системе MAX+PLUS II легко доступны все инструменты для создания проекта. Разработка проекта ускоряется за счет имеющихся стандартных функций, в том числе примитивов, мегафункций, библиотеки параметризованных модулей (LPM) и макрофункций устаревшего типа микросхем 74-й серии. В системе MAX+PLUS II есть три редактора для разработки проекта: графический, текстовый и сигнальный, а также два вспомогательных редактора: поуровневый планировщик и символьный редактор.

Графические символы, представляющие любой тип файла проекта, могут быть автоматически созданы в любом из редакторов MAX+PLUS II, предназначенных для разработки проектов с помощью команды File/Create Default Symbol Command. С помощью символьного редактора MAX+PLUS II можно редактировать символы или создавать собственные, а затем использовать их в любом схемном файле проекта.

В иерархической структуре проекта на любом уровне допускается смешанное использование файлов расширениями **.gdf .tdf .vhd .v .edf .sch**. Однако файлы с расширением **.wdf .xnf .adf .smf** должны быть либо на самом нижнем иерархическом уровне проекта, либо быть единственным файлом.

Способы задания файлов проекта показаны на рис. 3.

Во всех приложениях MAX+PLUS II есть возможность с помощью команд из меню **Assign** (Назначить) вводить, редактировать и удалять типы назначений ресурсов, устройств и параметров, которые управляют компиляцией

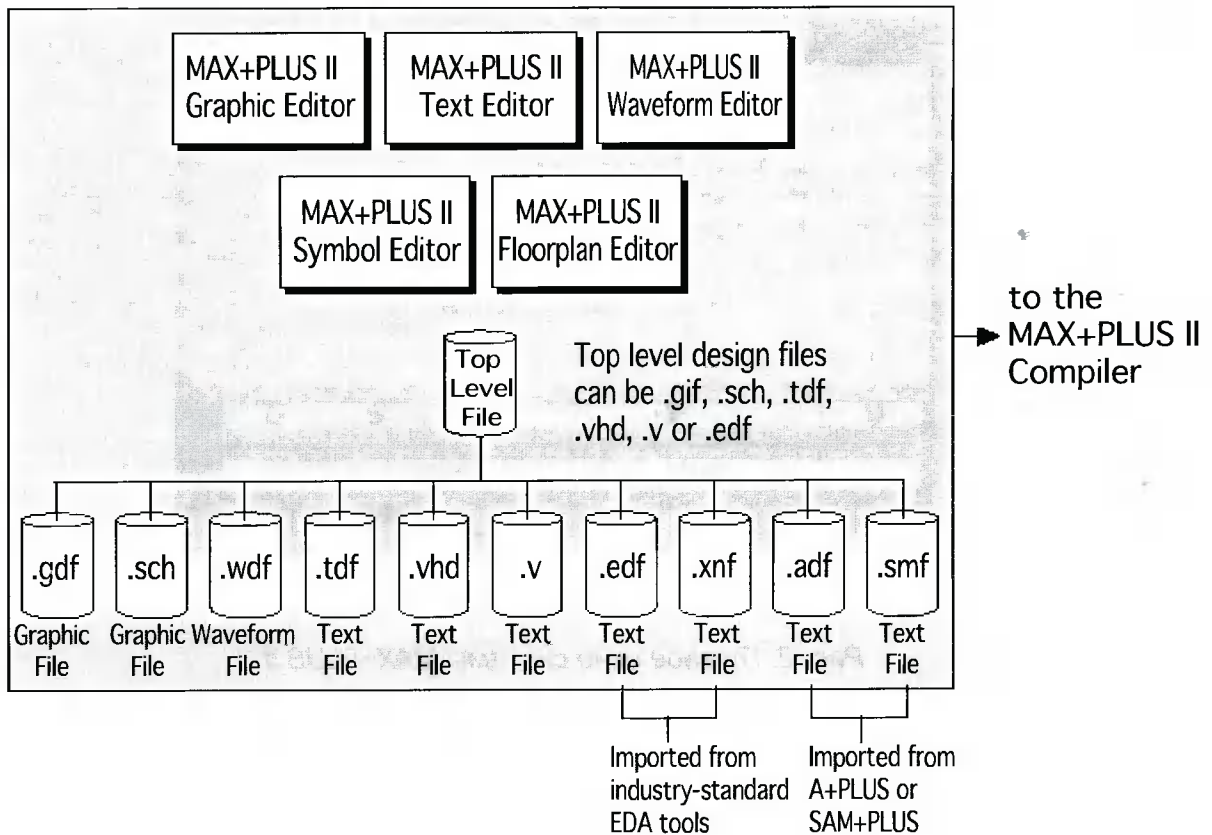


Рис. 3. Способы описания файлов проекта

проекта, в том числе логическим синтезом, разделением на части и подгонкой. Пользователь может делать назначения для текущего проекта независимо от того, открыт ли какой-нибудь файл проекта или окно приложений. Система MAX+PLUS II сохраняет информацию для проекта в файле с расширением **.acf**. Изменения назначений, сделанные в окне поровневого планировщика, также сохраняются в файле ACF. Кроме того, можно редактировать файл ACF для проекта в текстовом редакторе.

Доступны следующие типы назначений.

Clique assignment (Назначение клики) задает, какие логические функции должны оставаться вместе в одном и том же блоке логической структуры LAB, блоке ячеек памяти EAB, в одном ряду или устройстве.

Chip assignment (Назначение кристалла) задает, какие логические функции должны быть реализованы в одном и том же устройстве в случае разделения проекта на несколько устройств.

Pin assignment (Назначение контакта) назначает вход или выход одной логической функции конкретному контакту или нескольким контактам чипа.

Location assignment (Назначение ячейки) назначает единственную логическую функцию конкретной ячейке чипа.

Probe assignment (Назначение зонда) присваивает уникальное имя входу или выходу логической функции.

Connected pin assignment (Назначение соединительных контактов) задает внешнее соединение двух или более контактов на схеме пользователя.

Local routing assignment (Назначение местной разводки) присваивает коэффициент разветвления по выходу узла логическому элементу, находящемуся в том же блоке LAB, что и узел, или в соседнем LAB, прилежащем к узлу, с использованием общих местных разводок.

Device assignment (Назначение устройства) назначает тип ПЛИС, на которой реализуется проект.

Logic option assignment (Назначение логической опции) управляет логическим синтезом отдельных логических функций во время компиляции с применением стиля логического синтеза.

Timing assignment (Назначение временных параметров) управляет логическим



ким синтезом и подгонкой отдельных логических функций для получения требуемых характеристик времени задержки: t_{PD} (вход – неподрегистренный выход), t_{CO} (синхросигнал – выход), t_{SU} (синхросигнал – время установки), f_{MAX} (частота синхросигнала).

Можно определить глобальные опции устройства для компилятора с тем, чтобы он их использовал для всех устройств при обработке проекта. В целях резервирования дополнительных возможностей на будущее можно задать процентное соотношение контактов и логических элементов, которые должны оставаться неиспользованными во время текущей компиляции.

С использованием команды **Global Project Parameters** можно задать имена и глобальные установки, которые будут использованы компилятором для параметров во всех параметризованных функций проекта. При помощи команды **Global Project Timing Requirements** можно ввести глобальные требования по синхронизации для проекта, задавая общие характеристики времени задержки. Команда **Global Project Logic Synthesis** позволяет сделать глобальные установки для компилятора в части логического синтеза проекта.

Все пять редакторов MAX+PLUS II и три редактора создания дизайна (графический, текстовый и сигнальный) имеют общие функции, такие, как, например, создание, сохранение и открытие файла. Кроме того, приложения редактора MAX+PLUS II имеют следующие общие функции: создание файлов символов и файлов с прототипами функций; поиск узлов; просмотр иерархического дерева; всплывающие окна меню, зависящего от контекста; временной анализ; поиск и замена фрагментов текста; отмена последнего шага редактирования, его возвращения; вырезка; копирование; вставка и удаление выбранных фрагментов; обмен фрагментами между приложениями MAX PLUS II или приложениями Windows; печать.

Графический редактор (Graphic Editor) обеспечивает проектирование в реальном формате изображения (WYSIWIG). Графические файлы проекта (**.gdf**) или схемные файлы OrCAD (**.sch**), созданные в данном графическом редакторе, могут включать любую комбинацию символов примитивов, мегафункций и макрофункций. Символы могут представлять собой любой тип файлов проекта (**.gdf .sch .tdf .vhd .v .wdf .edf .xnf .adf .smf**).

Инструмент выбора (“стрелка”) позволяет двигать и копировать объекты, а также вводить новые символы. Когда вы помещаете его на контакт или конец линии, он автоматически преобразуется в инструмент рисования ортогональных линий. Если им щелкнуть на тексте,

он автоматически преобразуется в инструмент редактирования текста.

Графический редактор обеспечивает ряд других возможностей. Например, можно увеличить или уменьшить масштаб отображения на экране, выбирать гарнитуру и размер шрифта, задавать стили линий, устанавливать и отображать направляющие, получать зеркальное отображение, поворачивать выделенные фрагменты на 90, 180 или 270 градусов; задавать размер, ориентацию текущего листа схемы.

Символьный редактор (Symbol Editor) позволяет просматривать, создавать и редактировать символьные файлы. Символьный файл имеет то же имя, что и проект, и расширение **.sym**. Команда **Creat Default Symbol** меню **File**, которая есть в графическом, текстовом и сигнальном редакторах, создает символ для любого файла проекта. Символьный редактор обладает следующими характеристиками: можно переопределить символ, представляющий файл проекта, создавать и редактировать выводы и их имена, используя входные, выходные и двунаправленные выводы, а также задавать варианты ввода символа в файл графического редактора, задать значения параметров и их значения по умолчанию. При это сетка и направляющие помогают выполнить точное выравнивание объектов, а в символьные файлы можно вводить комментарии.

Текстовый редактор (**Text Editor**) является инструментом для создания текстовых файлов проекта на языках описания аппаратуры: AHDL (**.tdf**), VHDL (**.vhd**), Verilog HDL (**.v**). В этом текстовом редакторе можно работать также с произвольным файлом формата ASCII. Все перечисленные файлы проекта можно создавать в любом текстовом редакторе, однако данный редактор имеет встроенные возможности ввода файлов проекта, их компиляции и отладки с выдачей сообщений об ошибках и их локализацией в исходном тексте или в тексте вспомогательных файлов. Кроме того, существуют шаблоны языковых конструкций для AHDL, VHDL и Verilog HDL, и выполняется окрашивание синтаксических конструкций. В данном редакторе можно вручную редактировать файлы назначений и конфигурации (**.acf**), а также делать установки конфигурации для компилятора, симулятора и временного анализатора. Пользуясь текстовым редактором, можно создавать тестовые векторы (**.vec**), используемые для тестирования, отладки функций и при вводе сигнального проекта, а также командные файлы (**.cmd** – для симулятора и **.edc** – для EDIF), и макробιβотеки (**.lmf**). В текстовом редакторе MAX +PLUS II обеспечивается контекстная справка.

Сигнальный редактор (Waveform Editor) служит инструментом описания проекта и инструментом ввода тестовых векторов и просмотра результатов тестирования. Пользователь может создавать сигнальные файлы проекта (**.wdf**), которые содержат временные диаграммы, описывающие логику работы проекта, а также файлы каналов тестирования (**.scf**), которые содержат входные вектора для тестирования и функциональной отладки. Разработка описания проекта в сигнальном редакторе является альтернативой его созданию в графическом или текстовом редакторах. Здесь есть возможность графическим способом задавать комбинации входных логических уровней и требуемых выходов. Созданный таким образом файл WDF содержит как логические входы, так и входы цифрового автомата, а также выходы комбинаторной логики, счетчиков и цифровых автоматов. Способ разработки дизайна в сигнальном редакторе лучше подходит для цепей с четко определенными последовательными входами и выходами, т.е. для цифровых автоматов, счетчиков и регистров.

С помощью сигнального редактора легко преобразовывать временные диаграммы сигналов целиком или частично, создавая и редактируя узлы и группы. Простыми командами можно создавать файл таблицы ASCII-символов (**.tbl**) или импортировать файл тестовых векторов в формате ASCII (**.vec**) для создания файлов тестируемых каналов SCF и сигнального дизайна WDF. Можно также сохранить файл WDF как SCF для проведения тестирования или преобразовать SCF в WDF для использования его в качестве файла проекта.

Сигнальный редактор имеет следующие отличительные черты:

- возможность создания и редактирования узлов, задав его тип (при разработке WDF можно задать тип логики узла и его значения по умолчанию, а также имя состояния по умолчанию в узле типа цифрового автомата);
- возможность создания тестового вектора путем простого добавления в файл тестируемых каналов SCF нескольких узлов или всего информационного файла симулятора (**.snf**), существующего для полностью откомпилированного проекта;
- предоставление процедуры объединения от 2 до 256 узлов для создания новой группы узлов или их разгруппирования, а также объединения групп друг с другом; при этом значения групп могут быть отображены в двоичной, десятичной, шестнадцатичной или восьмиричной системе счисления с преобразованием или без в код Грэя;

- предоставление возможности копировать, вставлять, перемещать или удалять выбранный часть ("интервал") сигнала, а также весь узел или группу узлов; инвертировать, вставлять, переписывать, повторять, расширять или сжимать интервал сигнала любой длины с любым логическим уровнем, тактовым сигналом, последовательностью счета или именем состояния; задавать сетку для выравнивания переходов между логическими уровнями; в любом месте файла вводить комментарии между сигналами и менять масштаб отображения.

Для облегчения тестирования можно сделать наложение любых выходов в текущем файле или наложить второй файл сигнального редактора для сравнения сигналов его узлов и групп с соответствующими сигналами текущего файла.

Поуровневый планировщик (Floorplan Editor) предназначен для назначения ресурсов физических устройств и просмотра результатов разводки, сделанных компилятором. В окне поуровневого планировщика могут быть представлены два типа изображения:

Device View (Вид устройства) показывает все контакты устройства и их функции;

LAB View (Вид логического структурного блока) показывает внутреннюю часть устройства, в том числе все логические структурные блоки (LAB) и отдельные логические элементы

После выполнения всех назначений и задания проекта приступают к его компиляции. Сначала компилятор извлекает информацию об иерархических связях между файлами проекта и проверяет проект на простые ошибки ввода описания проекта. Компилятор применяет разнообразные способы увеличения эффективности проекта и минимизации использования ресурсов устройства. Если проект слишком большой, чтобы быть реализованным в одном устройстве, компилятор может автоматически разбить его на части для реализации в нескольких устройствах того же самого семейства устройств, при этом минимизируется число соединений между устройствами. В файле отчета (**.rpt**) будет отражено, как проект будет реализован: в одном или нескольких устройствах.

Кроме того, компилятор создает программируемые файлы, которые программатор будет использовать для программирования одного или нескольких устройств. У разработчика также есть возможность настроить обработку проекта. Например, возможно задать стиль логического синтеза проекта по умолчанию и другие параметры логического синтеза в рамках всего проекта, что позволит провести логический синтез в соответствии с вашими потребностями. Кроме того,



вы можете ввести требования по синхронизации в рамках всего проекта, точно задать разбиение большого проекта на части для реализации в нескольких устройствах и выбрать варианты параметров устройств, которые будут применены для всего проекта в целом.

Загрузку готового проекта в ПЛИС или конфигурационное ПЗУ выполняют с помощью программатора (**Programmer**).

Увеличение логической емкости ПЛИС и появление новой идеологии проектирования систем на кристалле (System on Chip) привели к тому, что ведущие производители ПЛИС вместе с выпуском на рынок собственно кристаллов с эквивалентной емкостью более 500 000 вентилей существенно обновили программное обеспечение, предоставив разработчику возможность использовать все преимущества новых БИС.

В середине 1999 года на рынок вышел САПР 4-го поколения фирмы Altera – система Quartus.

Отличительные свойства данного пакета:

- интеграция с программным обеспечением третьих фирм (Advanced Tools Integration). В рамках программы Native Link обеспечена совместимость с САПР ведущих производителей ПО. Поддерживаются стандарты EDIF, SDF, Vital 95, VHDL 1987 и 1993, Verilog HDL;
- возможность коллективной работы над проектом (Workgroup Computing);
- возможность анализа сигналов «внутри» ПЛИС с использованием функции Signal Tap;
- итерационная компиляция проекта, позволяющая не изменять уже отлаженные участки проекта (nSTEP Compiler);
- улучшенные средства синтеза в архитектуре APEX (CoreSyn);
- многоплатформенность (Win NT, Sun, HP);
- полная интеграция системы;
- разнообразие средств описания проекта;
- поддержка языков описания аппаратуры;

□ Internet-поддержка;

□ поддержка мегафункций MegaCore.

В дополнение к редакторам, используемым в пакете MAX+PLUS II, введен редактор блоков (Block Editor), позволяющий упростить графическое описание проекта, используя механизм параметризуемых блоков. Поуровневый планировщик (Floorplan Editor) имеет возможность распределять ресурсы как внутри одного, так и по мегаблокам. Новым средством, облегчающим работу над иерархическим проектом, является навигатор проекта (Project Navigator), позволяющий легко ориентироваться во всех файлах проекта. Улучшены возможности синтеза с заданными временными параметрами (Time driven Compilation).

Возрастающее внимание уделяется функциональному и поведенческому моделированию с использованием языков описания аппаратуры, в том числе тестирование проектов из нескольких ПЛИС. Наличие встроенного логического анализатора Signal TAP позволяет проводить контроль сигналов внутри ПЛИС.

Механизм подсказок соориентирован на использование Internet-технологий.

Для нормальной работы Quartus на базе PC-совместимого компьютера требуется следующая аппаратная поддержка:

- процессор с тактовой частотой не менее 400 МГц;
- от 256 Мб до 1 Гб оперативной памяти;
- 4 Гб свободного пространства на жестком диске;
- Windows NT 4.0 и более поздние версии;
- привод CD-ROM;
- желательно монитор с диагональю не менее 17 дюймов;
- установленный Microsoft Internet Explorer 4.0 и более поздние версии.

В следующих статьях цикла мы продолжим рассмотрение приемов проектирования аппаратуры на ПЛИС.